



Vlaanderen
is wetenschap



Afbakenen van potentiële leefgebiedenkaarten voor Europese en Vlaamse prioritaire soorten in het kader van de voortoets

Versie 2.0

Dirk Maes, Dries Adriaens, Maarten van der Meulen, Lien Poelmans, Wouter Van Landuyt, Anny Anselin, Jim Casaer, Geert De Knijf, Koen Devos, Jo Packet, Jeroen Speybroeck, Eric Stienen, Jan Stuyck, Arno Thomaes, Filiep Tjollyn, Toon Van Daele, Koen Van Den Berge, Bernard Van Elegem, Glenn Vermeersch, Carine Wils & Marc Pollet

INSTITUUT
NATUUR- EN BOSONDERZOEK

Auteurs:

Dirk Maes, Dries Adriaens, Maarten van der Meulen, Lien Poelmans, Wouter Van Landuyt, Anny Anselin, Jim Casaer, Geert De Knijf, Koen Devos, Jo Packet, Jeroen Speybroeck, Eric Stienen, Jan Stuyck, Arno Thomaes, Filiep T'jollyn, Toon Van Daele, Koen Van Den Berge, Bernard Van Elegem, Glenn Vermeersch, Carine Wils & Marc Pollet
Instituut voor Natuur- en Bosonderzoek

Het Instituut voor Natuur- en Bosonderzoek (INBO) is het Vlaams onderzoeks- en kenniscentrum voor natuur en het duurzame beheer en gebruik ervan. Het INBO verricht onderzoek en levert kennis aan al wie het beleid voorbereidt, uitvoert of erin geïnteresseerd is.

Vestiging:

INBO Brussel
Kliniekstraat 25, B-1070 Brussel
www.inbo.be

e-mail:

dirk.maes@inbo.be

Wijze van citeren:

Maes D., Adriaens D., van der Meulen M., Poelmans L., Van Landuyt W., Anselin A., Casaer J., De Knijf G., Devos K., Packet J., Speybroeck J., Stienen E., Stuyck J., Thomaes A., T'jollyn F., Van Daele T., Van Den Berge K., Van Elegem B., Vermeersch G., Wils C. & Pollet M. (2015). Afbakenen van potentiële leefgebiedenkaarten voor Europese en Vlaamse prioritaire soorten in het kader van de voortoets. Versie 2.0. Rapporten van het Instituut voor Natuur- en Bosonderzoek 2015 (INBO.R.2015.10201559). Instituut voor Natuur- en Bosonderzoek, Brussel.

D/2015/3241/272

INBO.R.2015.10201559

ISSN: 1782-9054

Verantwoordelijke uitgever:

Maurice Hoffmann

Druk:

Managementondersteunende Diensten van de Vlaamse overheid

Foto cover:

ljsvogel (Glenn Vermeersch)

Dit onderzoek werd uitgevoerd in opdracht van:

Agentschap voor Natuur en Bos en in samenwerking met het Vlaams Instituut voor Technologisch Onderzoek (VITO)





Afbakenen van potentiële leefgebiedenkaarten voor Europese en Vlaamse prioritaire soorten in het kader van de voortoets

Versie 2.0

Dirk Maes, Dries Adriaens, Maarten van der Meulen, Lien Poelmans, Wouter Van Landuyt, Anny Anselin, Jim Casaer, Geert De Knijf, Koen Devos, Jo Packet, Jeroen Speybroeck, Eric Stienen, Jan Stuyck, Arno Thomaes, Filiep T'jollyn, Toon Van Daele, Koen Van Den Berge, Bernard Van Elegem, Glenn Vermeersch, Carine Wils & Marc Pollet

INBO.R.2015.10201559

Vraaggestuurd ANB-project 2015

Dankwoord

In de eerste plaats bedanken we de vele vrijwilligers die de data verzameld hebben waarmee we de opgestelde potentiële leefgebiedenkaarten hebben kunnen valideren. Deze data werden eveneens gebruikt voor het maken van de correlatieve modellen. De meerderheid van deze data werd ingegeven in de dataportaal van Natuurpunt (www.waarnemingen.be) en we bedanken Natuurpunt Studie voor het ter beschikking stellen van deze waarnemingen.

Met Thomas Defoort, Carl De Schepper, Katja Nagels, Floris Verhaeghe en Maurits Vandegehuchte (ANB) werden constructieve discussies gevoerd over de uitvoering van dit project, waarvoor dank.

Guy Engelen (VITO) bedanken we voor de toestemming om de GeoDynamix toolbox te mogen gebruiken.

Voorwoord

In het kader van de voortoets, werden op vraag van het Agentschap voor Natuur en Bos (ANB) de twee volgende projecten uitgewerkt die moeten helpen bij het inschatten van mogelijke betekenisvolle effecten van ingrepen op Europese (Habitatrichtlijn/Vogelrichtlijn) en Vlaamse prioritaire soorten:

- 1) de gevoeligheid van soorten voor de verschillende effectgroepen
- 2) een leefgebiedenbenadering die moet toelaten om op een wetenschappelijk onderbouwde manier potentiële leefgebieden voor soorten af te bakenen.

Het eerste luik werd in een vorig rapport opgeleverd als een Access-databank (Maes et al. 2015). Voor het tweede deel werd in een eerste versie een vrij ruwe BWK-aanpak gebruikt, die een zeer ruime leefgebiedenafbakening opleverde (Maes et al. 2015).

In dit rapport lichten we de werkwijze en de resultaten toe voor het verfijnen van de eerste potentiële leefgebiedenkaarten. Dit project draagt bij tot de operationalisering van een systeem dat het uitvoeren van een passende beoordeling ondersteunt (Louette et al. 2011) en past in het toepassen van een evidence-based beleidsondersteuning (Louette et al. 2015).

Samenvatting

In 2014 werden in een eerste fase potentiële leefgebiedenkaarten aangemaakt voor de Vlaamse en Europese prioritaire soorten op basis van de Biologische Waarderingskaart (BWK). In een tweede fase werden vier methoden gebruikt om deze kaarten verder te verfijnen: 1) mechanistische modellen, 2) correlatieve modellen, 3) het downscalen van correlatieve modellen naar een hogere resolutie en 4) het gebruik van begeleidende soorten. Dit rapport presenteert de resultaten van deze analyses.

Met behulp van de GeoDynamix toolbox, ontwikkeld door het Vlaams Instituut voor Technologisch Onderzoek (VITO), werden potentiële leefgebieden van 81 Vlaamse en Europese prioritaire soorten afgebakend. Door gebruik te maken van de soortspecifieke ecologische kennis en de voorhanden zijnde kaartlagen die deze kennis vertalen in mechanistische modellen, maakten we potentiële leefgebiedenkaarten met een resolutie van 20 x 20 m². In dit rapport lichten we kort toe hoe de GeoDynamix toolbox werkt en geven we voor alle soorten de volledige gdx-scripts en de resulterende potentiële leefgebiedenkaarten.

Correlatieve modellen maken gebruik van de aan- of afwezigheid (met een resolutie van 1 x 1 km²) van een soort en proberen die te relateren aan biotische en/of abiotische omgevingsvariabelen. Met behulp van het package “biomod2” in R, gaan we na of het gebruik van correlatieve modellen een bijdrage kan leveren aan het verfijnen van de potentiële leefgebiedenkaarten. We toetsen dit met een beperkte set van soorten, in dit geval 17 broedvogels van Bijlage I van de Vogelrichtlijn. Deze modellen leveren een kaart op met per kilometerhok de kans dat de soort er aanwezig is, gegeven de omgevingsvariabelen.

De hierboven vermelde correlatieve modellen maken gebruik van verspreidingsgegevens met een resolutie van 1 x 1 km². Door de data voor de voorspelling in het model aan te bieden op een hogere resolutie (bv. 100 x 100 m²), zouden zulke modellen een veel fijner beeld kunnen opleveren (*downscaling*). Omdat deze techniek nog maar zelden getest werd op deze schaalniveaus, gebruiken we hier het Vliegend hert (*Lucanus cervus*) om te toetsen of een dergelijke downscaling haalbaar en nuttig is voor deze toepassing.

Naast de drie vorige methoden, kunnen ook zogenaamde begeleidende soorten gebruikt worden om potentieel geschikte gebieden van doelsoorten op te sporen: plaatsen waar veel begeleidende soorten samen voorkomen, zijn vermoedelijk geschikter dan gebieden waar slechts een of enkele van deze soorten voorkomen. In dit rapport maken we gebruik van de Groenknolorchis en de broedvogels van Bijlage I van de Europese Vogelrichtlijn om deze methode te testen.

Tenslotte vergelijken we de uitkomsten van de 4 methoden en geven we een voorbeeld van hoe het gezamenlijk gebruik van de verschillende methoden kan leiden tot verfijnde potentiële leefgebiedenkaarten. Ten slotte, bespreken we de voor- en nadelen van elk van de gebruikte technieken en geven we aanbevelingen over welk type onderzoek nodig is om de potentiële leefgebiedenkaarten nog verder te kunnen verfijnen en/of aan te maken voor meer soorten.

Aanbevelingen voor het beleid

In het kader van de leefgebiedenbenadering voor de voortoets werden in 2014 potentiële leefgebiedenkaarten aangemaakt met behulp van een vrij ruwe BWK-benadering. De hiermee afgebakende leefgebieden resulteerden in een zeer ruime weergave van de potentiële leefgebieden voor de verschillende soorten (Maes et al. 2015). In dit rapport gebruiken we 4 manieren om deze ruime potentiële leefgebiedenkaarten te verfijnen:

- 1) een mechanistische benadering op basis van ecologische soortprofielen (resolutie 20 x 20 m²);
- 2) een correlatieve benadering op basis van de gekende verspreiding en de relatie van soorten met biotische en abiotische variabelen (resolutie 1 x 1 km²);
- 3) het projecteren van de correlatieve modellen naar een kleinere schaal (resolutie 100 x 100 m²);
- 4) het gebruik van begeleidende soorten.

Uit dit rapport blijkt duidelijk dat de mechanistische benadering op basis van het ecologische soortprofiel een goede vertrekbasis is om potentiële leefgebiedenkaarten voor Europese en Vlaamse prioritaire soorten af te bakenen. De voornaamste voordelen van deze benadering zijn de hoge resolutie (20 x 20 m²), het maximaal gebruik van ecologische soortkennis en de herhaalbaarheid door het gebruik van kwantitatieve scripts. In Hoofdstuk 3.5 vergelijken en evalueren we, als testcase en waar mogelijk, de verschillende methoden voor de broedvogels van Bijlage I van de Europese Vogelrichtlijn. Daarnaast tonen we, aan de hand van een voorbeeld hoe, vertrekkende van de mechanistische benadering en met behulp van het gezamenlijk gebruik van de andere methoden, de potentiële leefgebiedenkaarten verder verfijnd (lees verkleind) kunnen worden. In Hoofdstuk 4, ten slotte, bespreken we de sterkten, zwakten en verbeterpunten van elk van de methoden en verduidelijken we enkele termen die nodig zijn voor een goed begrip van de hier toegepaste leefgebiedenbenadering.



Grauwe klauwier (foto Yves Adams – Vilda)

English abstract

In 2014, we compiled the first potentially suitable habitat maps for species of European concern and of Flemish priority species based solely on the Flemish Biological Valuation Map (BWK). In a second phase, we used four methods to refine these maps: 1) mechanistic modelling, 2) correlative modelling, 3) downscaling correlative models to a higher resolution and 4) the use of accompanying species. The results of these analyses are presented in this report.

Using the GeoDynamix (gdx) toolbox, developed by the Flemish Institute for Technological Research (VITO), we delineated potentially suitable habitat maps for 81 species of European concern and/or species of Flemish priority. By compiling all available ecological knowledge on the species and maps with biotic and abiotic variables and subsequently feeding them into mechanistic models, we produced potentially suitable habitat maps with a resolution of 20 x 20 meter. In the present report, we give a brief description of how the GeoDynamix toolbox works, the full gdx scripts and the resulting maps.

Correlative models make use of presence/absence data (resolution of 1 x 1 km²) of a species and relate this to biotic and/or abiotic environmental variables. Using the package “biomod2” in R, we test if correlative models can contribute to refine the potentially suitable habitat maps. To test this, we use a limited set of species (i.e., 17 breeding birds of the Annex I of the EU Birds Directive). These models produce maps with the probability of a species' presence in the grid cells of 1 x 1 km² in Flanders.

The correlative models mentioned above use a resolution of 1 x 1 km² to calibrate the model. By predicting the outcome of these models to a higher resolution (e.g. 100 x 100 meter), we could refine these models considerably (*downscaling*). Because this technique has hitherto not been tested extensively at this resolution, we use the Stag beetle (*Lucanus cervus*) to test whether downscaling is a useful and feasible approach for refining potentially suitable habitat maps.

A fourth approach is the use of accompanying species to detect habitat suitability for focal species: sites with a large number of accompanying species are believed to have a higher probability of harbouring the focal species – or offering better conditions – than sites with less accompanying species. Here, we test this method using the Fen orchid (*Liparis loeselii*) and the breeding birds of Annex I of the European Birds Directive as an example.

Finally, we compare the different methods and give an example of how the joint use of the different methods can lead to realistic potentially suitable habitat maps. We discuss the pros and cons of the different techniques and give recommendations for further research to improve the existing potentially suitable habitat maps and what layers are necessary to produce such maps for more species.



Slechtvalk (foto Rollin Verlinde – Vilda)

Inhoudstafel

Voorwoord	2
Samenvatting.....	3
Aanbevelingen voor het beleid	4
English abstract.....	5
Lijst van figuren	7
Lijst van tabellen.....	8
1 Inleiding.....	9
2 Materiaal en methode	10
2.1 Mechanistische modellen	10
2.1.1 GeoDynamIX toolbox	13
2.1.2 Verzamelen van gedetailleerde ecologische informatie	13
2.1.3 Uitkomst mechanistische modellen	14
2.2 Correlatieve modellen.....	15
2.2.1 Broedvogels.....	15
2.2.2 Vliegend hert.....	17
2.2.3 Uitkomst correlatieve modellen.....	18
2.3 Downscaling	18
2.4 Begeleidende soorten	18
2.4.1 Groenknolorchis.....	18
2.4.2 Broedvogels van de Europese Vogelrichtlijn	20
3 Resultaten.....	22
3.1 Mechanistische modellen	22
3.2 Correlatieve modellen.....	23
3.2.1 Broedvogels van de Europese Vogelrichtlijn	23
3.2.2 Vliegend hert.....	24
3.3 Downscaling	25
3.4 Begeleidende soorten	26
3.4.1 Groenknolorchis.....	26
3.4.2 Broedvogels van de Europese Vogelrichtlijn	29
3.5 Vergelijking tussen de verschillende technieken	30
3.6 Gezamenlijk gebruik van de verschillende technieken	31
4 Conclusies en aanbevelingen.....	33
4.1 Mechanistische modellen	33
4.2 Correlatieve modellen.....	34
4.3 Downscaling	35
4.4 Begeleidende soorten	36
4.5 Terminologie	36
4.5.1 Potentieel leefgebied	36
4.5.2 Actueel gebruikt leefgebied	36
4.5.3 Actueel gekende verspreiding.....	37
4.5.4 Werkelijke verspreiding	37
Referenties	37
Bijlage 1 Ecologische profielen, gdx-scripts en de daarmee gegenereerde leefgebiedenkaarten en voor een selectie van soorten de resulterende kaarten van de correlatieve modellen.....	40

Lijst van figuren

Figuur 1	Clusteranalyse (Twinspan – Hill 1979) van 109 vegetatieopnamen met Groenknolorchis in Nederland, België en Noord-Frankrijk met aanduiding van de differentiërende soorten per groep. Opnamen van laagveen onderscheiden zich door de aanwezigheid van de differentiërende soorten Wateraardbei, Draadzegge en Waterdrieblad van de opnamen op minerale bodem met als differentiërende soorten Parnassia, Moeraswespenorchis, Kruiwilg en Fioringras.	19
Figuur 2	Potentiële leefgebiedenkaart versie 2.0 van het Vliegend hert zoals aangemaakt met de GeoDynamix toolbox (gdx) met een resolutie van 20 x 20 m ²	22
Figuur 3	Detail van de potentiële leefgebiedenkaart van het Vliegend hert. Donkergroen: potentiële leefgebied van het Vliegend hert (resolutie 20 x 20 m ²), Rood: recente waarnemingen van het Vliegend hert in de regio Huizingen.	22
Figuur 4	Kans (probabiliteit) op aanwezigheid van het Vliegend hert in de verschillende kilometerhokken in Vlaanderen op basis van correlatieve modellen (boven; geel = <25%, oranje = 25-50%, rood = 50-75% en bruin = 75-100%) en de omzetting van deze probabiliteit naar aanwezigheid (rood) op basis van een door de ensemble modellering bepaalde grenswaarde (onder).	24
Figuur 5	Gedownscalede potentiële verspreidingskaarten met de kans (probabiliteit) op aanwezigheid van het Vliegend hert in de verschillende hectarehokken in Vlaanderen op basis van correlatieve modellen (boven; geel = <25%, oranje = 25-50%, rood = 50-75% en bruin = 75-100%) en de omzetting van deze probabiliteit naar aanwezigheid (rood) op basis van een door de ensemble modellering bepaalde grenswaarde (onder).	25
Figuur 6	Detail van de gedownscalede potentiële leefgebiedenkaart van het Vliegend hert in de regio Huizingen op basis van het correlatieve model. Per hectarehok wordt de kans op aanwezigheid gegeven (geel = <25%, oranje = 25-50%, rood = 50-75% en bruin = 75-100%). In het donkergroen wordt het potentiële leefgebied van het Vliegend hert weergegeven zoals afgebakend met behulp van het gdx-script (resolutie 20 x 20 m ²).	26
Figuur 7	Zoekkaart voor potentieel geschikte locaties voor Groenknolorchis in Vlaanderen aan de hand van een aantal indicatorsoorten (13 vaatplanten, 2 bladmossen en 1 levermos – Tabel 6).	27
Figuur 8	Kaart met het aantal typische begeleidende vaatplantsoorten voor duinpannen met Groenknolorchis (zie Tabel 6) zoals aanwezig in het Westhoekreservaat (De Panne) per raster van 50 x 50 meter. Gele cirkels: 1 kenmerkende soort, oranje: 2 kenmerkende soorten, rood: 3 kenmerkende soorten, paars: 4 kenmerkende soorten, lichtblauw: 5 kenmerkende soorten, donkerblauw: 6 kenmerkende soorten.	28
Figuur 9	Kaart met het aantal typische begeleidende vaatplantsoorten voor duinpannen met Groenknolorchis (zie Tabel 6) in het duinengebied Ter Yde (Oostduinkerke) per raster van 50 x 50 meter. Gele cirkels: 1 kenmerkende soort, oranje: 2 kenmerkende soorten, rood: 3 kenmerkende soorten, paars: 4 kenmerkende soorten.	28
Figuur 10	Potentiële leefgebiedenkaart voor Boomleeuwerik op basis van de mechanistische modellen (groen) met een overlay van de resultaten van de correlatieve modellen (rode bollen) en de begeleidende soorten (zwart-grijze bollen).	31
Figuur 11	Verfijnde potentiële leefgebiedenkaart voor Boomleeuwerik (donkerrood). Van de potentiële leefgebiedenkaart bekomen met het mechanistisch model (lichtgroen) werden enkel die clusters overgehouden waarvoor het correlatieve model de soort als aanwezig voorspelde en de waar de indicatorwaardesom van de begeleidende soorten minstens 33% van de maximale indicatorsomwaarde in het kilometerhok bedroeg.	32

Lijst van tabellen

Tabel 1	Lijst van de Vlaamse en Europese prioritaire soorten met vermelding van de geraadpleegde INBO-soortenspecialist(en) per taxonomische groep. Indien de specialist nog een andere affiliatie heeft dan het INBO, wordt die tussen haakjes vermeld. v1.0: + = er werd een potentiële leefgebiedenkaart aangemaakt met behulp van BWK-codes; v2.0: + = er werd een verfijnde potentiële leefgebiedenkaart aangemaakt met behulp van de gdx-toolbox. Cm: + = er werden correlatieve modellen gemaakt. Ds: + = het correlatieve model werd gedownscaled naar een hogere resolutie. Bs: + = er werden begeleidende soorten gebruikt om de mogelijke verspreiding te bepalen.10
Tabel 2	Kaartlagen gebruikt in de GeoDynamix toolbox voor de ontwikkeling van mechanistische modellen. Tussen haakjes wordt het aantal mogelijke klassen voor elke kaartlaag aangegeven.14
Tabel 3	Broedvogels van de Europese Vogelrichtlijn waarvoor correlatieve modellen gemaakt werden met vermelding van het aantal kilometerhokken waarin ze broedend waargenomen werden. † momenteel worden voor deze soorten geen broedgevallen vastgesteld.16
Tabel 4	Gemiddelde waarden en standaardfout (se) van de landgebruiksvariabelen per kilometerhok (in m ²) die werden gebruikt in de correlatieve modellen voor de broedvogels van de Europese Vogelrichtlijn in Vlaanderen. Enkel kilometerhokken die voor minstens 50% in Vlaanderen liggen, werden gebruikt. De Biotopdiversiteitsindex werd berekend als een Shannon-Wiener index met de voor vogels relevante biotopen.17
Tabel 5	Gemiddelde waarden en standaardfout (se) van de landgebruiksvariabelen per kilometerhok (in m ²) die werden gebruikt in het correlatieve model voor het Vliegend hert in Vlaanderen. Enkel kilometerhokken die voor minstens 50% in Vlaanderen liggen werden gebruikt.17
Tabel 6	Lijst met kenmerkende begeleidende soorten van vegetaties met Groenknolorchis.20
Tabel 7	Begeleidende soorten en, tussen haakjes, hun indicatorwaarde voor de broedvogels van Bijlage I van de Europese Vogelrichtlijn.21
Tabel 8	Gemiddelde Area Under the Curve (AUC) van de Receiver Operating Characteristic (ROC) voor de 10 model runs per algoritme (GAM: Generalized Additive Models, GBM: Generalized Boosted Regression Modeling, GLM: Generalized Linear Models, MaxEnt: Maximum Entropy, RF: Random Forest) en het algemeen gemiddelde over alle algoritmen heen (Gem) van de 17 broedvogels van de Europese Vogelrichtlijn voor de verschillende algoritmen. De waarden in het vet zijn AUC's die kleiner zijn dan 0,7 en werden niet gebruikt in de ensemble modelling. N = aantal runs dat gebruikt werd voor de ensemble modelling (max = 50).23
Tabel 9	Gemiddelde Area Under the Curve (AUC) van de Receiver Operating Characteristic (ROC) voor de verschillende algoritmen (GAM: Generalized Additive Models, GBM: Generalized Boosted Regression Modeling, GLM: Generalized Linear Models, MaxEnt: Maximum Entropy, RF: Random Forest) en het algemeen gemiddelde over alle algoritmen heen (Gem) die gebruikt werden om het potentiële leefgebied van Vliegend hert in kaart te brengen.24
Tabel 10	De som van de indicatorwaarden van de begeleidende soorten in kilometerhokken met (Aanwezig) en zonder (Afwezig) de doelsoort. Tussen haakjes vermelden we het aantal kilometerhokken met en zonder de doelsoort. De significantie geeft aan of het verschil tussen de som in de kilometerhokken met en zonder de doelsoort beduidend is of niet (p-waarde van een ANOVA-test).29
Tabel 11	Aantal kilometerhokken met waarnemingen (N hokken) van de soort op basis van Anselin et al. (2014). MM = aantal kilometerhokken met potentieel geschikt leefgebied groter dan de minimaal noodzakelijke oppervlakte volgens de mechanistische modellen, Overlap WM = overlap tussen kilometerhokken met waarnemingen en de geschikte kilometerhokken volgens het mechanistisch model, CM = aantal kilometerhokken met potentieel geschikt leefgebied volgens correlatieve modellen, Overlap WC = overlap tussen hokken met waarnemingen en de geschikte kilometerhokken volgens het correlatieve model. BS = aantal kilometerhokken met minstens 1/3 van de maximale gesommeerde indicatorwaarde van de begeleidende soorten, Overlap BS = overlap tussen hokken met waarnemingen en de geschikte hokken volgens de begeleidende soorten. Het het best scorende model is telkens in rood weergegeven.30

1 Inleiding

Met het online-instrument voortoets (<https://milieuinfo.be/voortoets>) wil het Agentschap voor Natuur en Bos (ANB) aan initiatiefnemers de mogelijkheid bieden om op voorhand in te schatten wat de mogelijke implicaties zijn van een voorgenomen vergunningsplichtige activiteit op een Speciale Beschermingszone (SBZ).

Aan de hand van deze screening kan de initiatiefnemer nagaan of het risico bestaat op een betekenisvolle aantasting van de actuele en potentieel te realiseren habitats. In een verdere fase van de uitwerking kan dit ook voor de soorten die voorkomen in Speciale Beschermingszones en/of leefgebieden van Habitatrichtlijn- en Vogelrichtlijnsoorten. Als de voortoets aangeeft dat er geen risico op een betekenisvolle aantasting is, dan moet de initiatiefnemer geen passende beoordeling opmaken. Is er wel een waarschijnlijkheid of een risico, dan is verder onderzoek door de initiatiefnemer in overleg met het ANB aangewezen om te bepalen of er effectief een betekenisvolle aantasting kan zijn of welke milderende maatregelen aangewezen zijn om deze aantasting te vermijden of alvast te verminderen. Dit moet uitwijzen of een passende beoordeling alsnog vereist is. Het eindrapport van het voortoets instrument geeft in geval van een mogelijk risico ook aan wat de focus moet zijn van een eventuele passende beoordeling.

Om na te gaan welke soorten gevoelig zijn voor de verschillende mogelijke typen van ingrepen, werd eerder al door het INBO een databank aangelegd die de gevoeligheid van soorten voor de verschillende effectgroepen aangeeft (Maes et al. 2015). In een eerste oefening, die enkel gebaseerd was op BWK-eenheden en ecoregio's werden eerder kaarten aangemaakt die een ruw beeld gaven van de potentiële verspreiding van Vlaamse en Europese prioritaire soorten (Maes et al. 2015).

De ruwe leefgebieden kaarten die in 2014 opgeleverd werden, kunnen op 4 manieren verfijnd worden:

1. Met behulp van de GeoDynamix toolbox kunnen **mechanistische modellen** gemaakt worden met een zeer hoge resolutie (20 x 20 m²) op basis van de ecologie en de kwantitatieve en kwalitatieve noden van elke soort. Hierin kunnen, naast de BWK, ook andere kaartlagen gebruikt worden (bv. bodemtextuur, vochtigheid, bosleeftijd, ...). Deze modellen leveren een **potentiële leefgebiedenkaart** op per soort.
2. **Correlatieve modellen** kunnen, op basis van de gekende verspreiding en de variabelen op de plekken waar de soort aanwezig is, voor niet-onderzochte kilometerhokken een kans op aanwezigheid berekenen voor de soort. De hogervermelde potentiële leefgebiedenkaarten kunnen vervolgens vergeleken worden met de uitkomst van deze correlatieve modellen en waar nodig aangepast worden tot kilometerhokken met een grote kans op aanwezigheid van de soort.
3. Correlatieve modellen worden in Vlaanderen meestal gemaakt op kilometerhokniveau. Het **downscalen** van deze modellen naar een kleinere schaal (bv. 100 x 100 m²) kan een veel fijnere kaart opleveren en zou eveneens gebruikt kunnen worden om de potentiële leefgebiedenkaarten te verfijnen. Downscalen van modellen vergt echter bijkomend onderzoek naar de methodologische en ecologische relevantie ervan.
4. Door **begeleidende en/of habitattypische soorten** (soorten die vaak samen voorkomen met de soort in kwestie) te gebruiken kan de potentie voor de doelsoort in bepaalde kilometerhokken/gebieden verfijnd worden (cf. Groenknolorchis – Van Landuyt et al. 2014).

In de volgende hoofdstukken bespreken we hoe en voor welke soorten we deze verschillende methoden toegepast hebben (Hoofdstuk 2), wat de resultaten van deze analyses zijn (Hoofdstuk 3) en welke aanbevelingen en conclusies we bij elk van de toegepaste methoden kunnen geven (Hoofdstuk 4).

2 Materiaal en methode

In de projectomschrijving worden 4 manieren voorgesteld om de eerste, lage-resolutie potentiële leefgebiedenkaarten (Maes et al. 2015) te verfijnen (mechanistische modellen, correlatieve modellen, downscaling van correlatieve modellen en het gebruik van begeleidende soorten). Hier bespreken we hoe en voor welke soorten we deze vier manieren toegepast hebben.

2.1 Mechanistische modellen

Voor alle 113 Vlaamse en Europese prioritaire soorten (Tabel 1) werd nagegaan of er voldoende ecologische kennis enerzijds en de noodzakelijke kaartlagen anderzijds beschikbaar waren om verfijnde potentiële leefgebiedenkaarten te maken met behulp van GeoDynamix (gdx), een tool van het Vlaams Instituut voor Technologisch Onderzoek (VITO – zie 2.1.1). Op vraag van ANB werden bijkomend ook de potentiële leefgebieden van zeven “veldvogels” gemaakt. Het gaat hier om broedvogels die karakteristiek zijn voor grote, open en natuurkwaliteitsvolle weilandcomplexen, met name Graspieper, Grutto, Paapje, Roodborsttapuit, Tureluur, Watersnip en Wulp (Cfr. art 3§2b van N2000 besluit).

Tabel 1 Lijst van de Vlaamse en Europese prioritaire soorten met vermelding van de geraadpleegde INBO-soortenspecialist(en) per taxonomische groep. Indien de specialist nog een andere affiliatie heeft dan het INBO, wordt die tussen haakjes vermeld. v1.0: + = er werd een potentiële leefgebiedenkaart aangemaakt met behulp van BWK-codes; v2.0: + = er werd een verfijnde potentiële leefgebiedenkaart aangemaakt met behulp van de gdx-toolbox. Cm: + = er werden correlatieve modellen gemaakt. Ds: + = het correlatieve model werd gedownscaled naar een hogere resolutie. Bs: + = er werden begeleidende soorten gebruikt om de mogelijke verspreiding te bepalen.

Nederlandse naam (<i>Wetenschappelijke naam</i>)	v1.0	v2.0	Cm	Ds	Bs
Amfibieën (J Speybroeck – Hyla)					
1. Boomkikker (<i>Hyla arborea</i>)	+	+	-	-	-
2. Heikikker (<i>Rana arvalis</i>)	+	+	-	-	-
3. Kamsalamander (<i>Triturus cristatus</i>)	+	+	-	-	-
4. Knoflookpad (<i>Pelobates fuscus</i>)	+	+	-	-	-
5. Poelkikker (<i>Pelophylax lessonae</i>)	+	+	-	-	-
6. Rugstreeppad (<i>Epidalea calamita</i>)	+	+	-	-	-
7. Vroedmeesterpad (<i>Alytes obstetricans</i>)	+	+	-	-	-
Broedvogels (F T’jollyn, G Vermeersch & E Stienen)					
8. Blauwborst (<i>Luscinia svecica</i>)	+	+	+	-	-
9. Boomleeuwerik (<i>Lullula arborea</i>)	+	+	+	-	-
10. Bruine kiekendief (<i>Circus aeruginosus</i>)	+	+	+	-	-
11. Duinpieper (<i>Anthus campestris</i>)	+	+	-	-	-
12. Dwergstern (<i>Sterna albifrons</i>)	+	+	-	-	-
13. Grauwe kiekendief (<i>Circus pygargus</i>)	+	+	-	-	-
14. Grauwe klauwier (<i>Lanius collurio</i>)	+	+	+	-	-
15. Grote stern (<i>Sterna sandvicensis</i>)	+	+	-	-	-
16. IJsvogel (<i>Alcedo atthis</i>)	+	+	+	-	-
17. Kleine mantelmeeuw (<i>Larus fuscus</i>)	+	+	-	-	-
18. Kleine zilverreiger (<i>Egretta garzetta</i>)	+	+	-	-	-
19. Kluut (<i>Recurvirostra avosetta</i>)	+	+	+	-	-
20. Korhoen (<i>Tetrao tetrix</i>)	+	+	-	-	-
21. Kwak (<i>Nycticorax nycticorax</i>)	+	+	-	-	-
22. Kwartelkoning (<i>Crex crex</i>)	+	+	+	-	-
23. Lepelaar (<i>Platalea leucorodia</i>)	+	+	-	-	-
24. Middelste bonte specht (<i>Dendrocopos medius</i>)	+	+	+	-	-
25. Nachtzwaluw (<i>Caprimulgus europaeus</i>)	+	+	+	-	-
26. Ooievaar (<i>Ciconia ciconia</i>)	+	+	-	-	-

27.	Ortolaan (<i>Emberiza hortulana</i>)	+	+	-	-	-
28.	Porseleinhoen (<i>Porzana porzana</i>)	+	+	+	-	-
29.	Purperreiger (<i>Ardea purpurea</i>)	+	+	-	-	-
30.	Roerdomp (<i>Botaurus stellaris</i>)	+	+	+	-	-
31.	Slechtvalk (<i>Falco peregrinus</i>)	+	-	+	-	-
32.	Steltkluut (<i>Himantopus himantopus</i>)	+	+	-	-	-
33.	Strandplevier (<i>Charadrius alexandrinus</i>)	+	+	-	-	-
34.	Visdief (<i>Sterna hirundo</i>)	+	+	+	-	-
35.	Wespendief (<i>Pernis apivorus</i>)	+	+	+	-	-
36.	Woudaap (<i>Ixobrychus minutus</i>)	+	+	+	-	-
37.	Zwarte specht (<i>Dryocopus martius</i>)	+	+	+	-	-
38.	Zwarte stern (<i>Chlidonias niger</i>)	+	+	-	-	-
39.	Zwartkopmeeuw (<i>Larus melanocephalus</i>)	+	+	+	-	-
Kevers (A Thomaes)						
40.	Vliegend hert (<i>Lucanus cervus</i>)	+	+	+	+	-
Libellen (G De Knijf – Libellenvereniging Vlaanderen)						
41.	Gevlekte witsnuitlibel (<i>Leucorrhinia pectoralis</i>)	+	+	-	-	-
42.	Rivierrombout (<i>Gomphus flavipes</i>)	+	+	-	-	-
Mollusken (J Packet)						
43.	Nauwe korfslak (<i>Vertigo angustior</i>)	+	+	-	-	-
44.	Platte schijfhoren (<i>Anisus (Disculifer) vorticulus</i>)	+	+	-	-	-
45.	Zeggekorfslak (<i>Vertigo moulinsiana</i>)	+	+	-	-	-
Mossen (W Van Landuyt)						
46.	Geel schorpioenmos (<i>Hamatocaulis vernicosus</i>)	+	+	-	-	-
Nachtvinders (D Maes)						
47.	Spaanse vlag (<i>Euplagia quadripunctaria</i>)	+	+	-	-	-
Overwinterende watervogels (K Devos, F T'jollyn, A Anselin & E Stienen)						
48.	Bergeend (<i>Tadorna tadorna</i>)	+	+	-	-	-
49.	Blauwe kiekendief (<i>Circus cyaneus</i>)	+	-	-	-	-
50.	Goudplevier (<i>Pluvialis apricaria</i>)	+	+	-	-	-
51.	Grauwe gans (<i>Anser anser</i>)	+	+	-	-	-
52.	Grote zilverreiger (<i>Casmerodius albus</i>)	+	+	-	-	-
53.	Kemphaan (<i>Philomachus pugnax</i>)	+	+	-	-	-
54.	Kleine rietgans (<i>Anser brachyrhynchus</i>)	+	+	-	-	-
55.	Kleine zwaan (<i>Cygnus bewickii</i>)	+	+	-	-	-
56.	Kluut (<i>Recurvirostra avosetta</i>)	+	+	-	-	-
57.	Kokmeeuw (<i>Larus ridibundus</i>)	+	-	-	-	-
58.	Kolgans (<i>Anser albifrons</i>)	+	+	-	-	-
59.	Krakeend (<i>Anas strepera</i>)	+	+	-	-	-
60.	Kuifeend (<i>Aythya fuligula</i>)	+	+	-	-	-
61.	Pijlstaart (<i>Anas acuta</i>)	+	+	-	-	-
62.	Regenwulp (<i>Numenius phaeopus</i>)	+	+	-	-	-
63.	Rietgans (<i>Anser fabalis</i>)	+	+	-	-	-
64.	Slobeend (<i>Anas clypeata</i>)	+	+	-	-	-
65.	Smient (<i>Anas penelope</i>)	+	+	-	-	-
66.	Steenloper (<i>Arenaria interpres</i>)	+	+	-	-	-
67.	Stormmeeuw (<i>Larus canus</i>)	+	-	-	-	-
68.	Tafeleend (<i>Aythya ferina</i>)	+	+	-	-	-
69.	Waterrietzanger (<i>Acrocephalus paludicola</i>)	+	+	-	-	-
70.	Wintertaling (<i>Anas crecca</i>)	+	+	-	-	-
71.	Wulp (<i>Numenius arquata</i>)	+	+	-	-	-

Planten (W Van Landuyt)

72.	Drijvende waterweegbree (<i>Luronium natans</i>)	+	+	-	-	-
73.	Groenknolorchis (<i>Liparis loeselii</i>)	+	+	-	-	+
74.	Kruipend moerasscherm (<i>Apium repens</i>)	+	+	-	-	-

Reptielen (J Speybroeck – Hyla)

75.	Gladde slang (<i>Coronella austriaca</i>)	+	+	-	-	-
-----	---	---	---	---	---	---

Veldvogels (F T'jollyn, G Vermeersch)

76.	Graspieper	+	+	-	-	-
77.	Grutto	+	+	-	-	-
78.	Paapje	+	+	-	-	-
79.	Roodborsttapuit	+	+	-	-	-
80.	Tureluur	+	+	-	-	-
81.	Watersnip	+	+	-	-	-
82.	Wulp	+	+	-	-	-

Vissen (C Belpaire, G Van Thuyne & H Verreycken)

83.	Atlantische zalm (<i>Salmo salar</i>)	+	-	-	-	-
84.	Beekdonderpad (<i>Cottus rhenanus</i>)	+	-	-	-	-
85.	Beekprik (<i>Lampetra planeri</i>)	+	-	-	-	-
86.	Bittervoorn (<i>Rhodeus sericeus</i>)	+	-	-	-	-
87.	Fint (<i>Alosa fallax</i>)	+	-	-	-	-
88.	Grote modderkruiper (<i>Misgurnus fossilis</i>)	+	-	-	-	-
89.	Kleine modderkruiper (<i>Cobitis taenia</i>)	+	-	-	-	-
90.	Rivierdonderpad (<i>Cottus perifretum</i>)	+	-	-	-	-
91.	Rivierprik (<i>Lampetra fluviatilis</i>)	+	-	-	-	-

Vleermuizen (A Thomaes, R Gyselings, T Onkelinx)

92.	Baardvleermuis (<i>Myotis mystacinus</i>)	+	-	-	-	-
93.	Bechstein's vleermuis (<i>Myotis bechsteinii</i>)	+	-	-	-	-
94.	Bosvleermuis (<i>Nyctalus leisleri</i>)	+	-	-	-	-
95.	Brandt's vleermuis (<i>Myotis brandtii</i>)	+	-	-	-	-
96.	Franjestaart (<i>Myotis nattereri</i>)	+	-	-	-	-
97.	Gewone dwergvleermuis (<i>Pipistrellus pipistrellus</i>)	+	-	-	-	-
98.	Gewone grootoorvleermuis (<i>Plecotus auritus</i>)	+	-	-	-	-
99.	Grijze grootoorvleermuis (<i>Plecotus austriacus</i>)	+	-	-	-	-
100.	Grote hoefijzerneus (<i>Rhinolophus ferrumequinum</i>)	+	-	-	-	-
101.	Ingekorven vleermuis (<i>Myotis emarginatus</i>)	+	-	-	-	-
102.	Kleine dwergvleermuis (<i>Pipistrellus pygmaeus</i>)	+	-	-	-	-
103.	Laatvlieger (<i>Eptesicus serotinus</i>)	+	-	-	-	-
104.	Meervleermuis (<i>Myotis dasycneme</i>)	+	-	-	-	-
105.	Mopsvleermuis (<i>Barbastella barbastellus</i>)	+	-	-	-	-
106.	Rosse vleermuis (<i>Nyctalus noctula</i>)	+	-	-	-	-
107.	Ruige dwergvleermuis (<i>Pipistrellus nathusii</i>)	+	-	-	-	-
108.	Vale vleermuis (<i>Myotis myotis</i>)	+	-	-	-	-
109.	Watervleermuis (<i>Myotis daubentonii</i>)	+	-	-	-	-

Zoogdieren (K Van Den Berge, J Stuyck & J Casaer)

110.	Europese bever (<i>Castor fiber</i>)	+	+	-	-	-
111.	Europese hamster (<i>Cricetus cricetus</i>)	+	+	-	-	-
112.	Europese otter (<i>Lutra lutra</i>)	+	+	-	-	-
113.	Hazelmuis (<i>Muscardinus avellanarius</i>)	+	+	-	-	-

Zoals uit Tabel 1 blijkt, was het voor de meeste soorten mogelijk om een verfijnde potentiële leefgebiedenkaart te maken met behulp van mechanistische modellen. Enkel voor de vissen, de vleermuizen en voor enkele broedvogels bleek dat niet haalbaar.

Bij de broedvogels was het voor de Slechtvalk niet mogelijk om een potentiële leefgebiedenkaart te maken omdat er geen kaartlaag beschikbaar is van geschikte, hoge gebouwen waar de soort meestal broedt (koeltorens, kerken, ...). Voor de Ortolaan werd wel een script geschreven, waaruit bleek dat er voor deze soort momenteel geen geschikt leefgebied aanwezig is in Vlaanderen.

Bij de overwinterende watervogels werd vertrokken van de gebieden die in het kader van de watervogeltellingen afgebakend werden (Devos & Onkelinx 2013). Om de belangrijkste gebieden af te bakenen werden de jaarlijkse wintermaxima van 2010-2014 gesommeerd en de gebieden werden deze som werd vervolgens in dalende wintermaxima gerangschikt. De gebieden waarin 90% van deze gesommeerde maxima voorkwam werden behouden en onderworpen aan een verdere verfijning met behulp van mechanistische modellen. Het gebruik van de som van de wintermaxima zorgt ervoor dat zowel gebieden die jaarlijks ongeveer dezelfde aantallen wintervogels herbergen als gebieden die in maar in een winter een heel groot aantal individuen telden, opgenomen worden in de lijst.

Om voor vissen potentiële leefgebiedenkaarten af te bakenen is een kaartlaag van alle waterlopen in Vlaanderen nodig met daarin informatie over de oeverstructuur, de stroomsnelheid, het substraat, de waterkwaliteit, enzovoort. Momenteel wordt, in een proefproject van de VMM, geprobeerd om dergelijke kaarten aan te maken. De huidige kaart is echter nog te fragmentarisch om bruikbaar te zijn voor deze oefening.

Ook bij de vleermuizen ontbreken er voldoende gedetailleerde kaartlagen of kunnen enkel proxies gebruikt worden (holten in bomen, kleine landschapselementen in het landschap voor het foerageren, enz.) om de leefgebieden van de soorten voldoende nauwkeurig met te kunnen beschrijven enerzijds, maar is er ook onvoldoende kennis over de verspreiding van de soorten tijdens de zomermaanden anderzijds.

2.1.1 GeoDynamix toolbox

De GeoDynamix toolbox van het Vlaams Instituut voor Technologisch Onderzoek (VITO – <https://vito.be/nl/landgebruik/landgebruik/geodynamix-op-weg-naar-een-beter-landgebruik>) maakt het mogelijk om op een zeer hoge resolutie (hier: 20 x 20 m²) gebieden af te bakenen die voldoen aan bepaalde voorwaarden. Als basis worden hiervoor een of meerdere rasterkaarten gebruikt. De basis voor deze toepassing van gdx is de Biologische Waarderingskaart (BWK). Speciaal voor deze toepassing is de gdx scripting taal uitgebreid met een raster-versie van de Biologische Waarderingskaart (BWK). Deze BWK rasterkaart is bijzonder omdat hij meerdere BWK-eenheden per rastercel kan bevatten. Daarnaast kunnen ook andere relevante verrasterde kaarten worden gebruikt (zie verder). Door het gebruik van kwantitatieve criteria voor de benodigde oppervlakte van verschillende biotooptypen en de afstanden ertussen wordt door de toolbox berekend waar aan deze criteria wordt voldaan. De toetsing van de berekende kaarten gebeurt, indien mogelijk, aan de hand van de gekende verspreiding van de soort en in een iteratief proces worden de criteria in het script (biotooptype, oppervlakten, afstanden, ...) aangepast met behulp van actuele waarnemingen om zo de potentiële verspreiding zo goed mogelijk te kunnen inschatten. Ook in de correlatieve modellen worden verspreidingsgegevens van soorten gebruikt, maar dan als responsvariabele. Hierbij wordt er echter ook naar gestreefd om zo goed mogelijk de regio's of gebieden af te bakenen waar de soort niet voorkomt of de potentie bijzonder klein is dat de soort er kan voorkomen.

2.1.2 Verzamelen van gedetailleerde ecologische informatie

Op basis van de criteria voor het bepalen van de lokale staat van instandhouding (habitatrichtlijnsoorten: Adriaens & Lommaert 2015; vogelrichtlijnsoorten: Adriaens & Vermeersch in druk) werd voor elke soort een ecologisch profiel opgesteld dat de leefgebiedvereisten van de soort beschrijft op basis van kwalitatieve (bv. welke biotopen) en kwantitatieve informatie (bv. minimaal noodzakelijke oppervlakte per biotooptype, maximale afstanden tussen de verschillende onderdelen van het leefgebied). Dit gebeurt aan de hand van twee criteria: 1) de toestand van de populatie en 2) kwaliteit van het leefgebied. Indien het leefgebied bestaat uit verschillende onderdelen (bv. voor voortplanting, foerageren, rusten, overwinteren), wordt elk van de onderdelen apart beschreven. Indien de verschillende onderdelen van het leefgebied in een voor de soort geschikte configuratie voorkomen (i.e., afstand

tussen de onderdelen is overbrugbaar), wordt het geheel van de verschillende onderdelen uiteindelijk beschouwd als potentiële leefgebied van een soort.

Voor de beschrijving van het biotoopgebruik van de verschillende soorten werden de BWK-eenheden gebruikt uit Vriens et al. (2011). De selectie van BWK-eenheden zoals gebruikt bij de afbakening van de eerste versie van de potentiële leefgebiedenkaarten (Maes et al. 2015) werd voor deze oefening verder verfijnd en waar nodig bijgestuurd. Andere gebruikte kaartlagen zijn terug te vinden in Tabel 2.

Tabel 2 Kaartlagen gebruikt in de GeoDynamix toolbox voor de ontwikkeling van mechanistische modellen. Tussen haakjes wordt het aantal mogelijke klassen voor elke kaartlaag aangegeven.

Naam	Beschrijving
Basislaag	
BWK	Biologische Waarderingskaart (uitgave 2014 – De Saeger et al. 2014)
Bijkomende kaartlagen	
bodem_bodemtype	Gedetailleerde beschrijving van het bodemtype (4288)
bodem_drainage	Drainageklasse van de bodem (15)
bodem_profiel	Profielontwikkeling van de bodem (16)
bodem_textuur	Textuur van de bodem (32)
bosinventaris2001	Leeftijdsopbouw (9) en sluitingsgraad (3) van het bostype
bosleeftijd	Leeftijd van het bos (4)
boswijzer2013	Bos – geen bos (2)
ecodistrict	Ecodistricten van Vlaanderen (36)
epr2013	Gewassenkaart landbouw (246)
GRB_wegbaan	Weg – geen weg - kruispunt (3)
GRB_watergang	Water – geen water (2)
GRB_wegverbinding	Type weg (15)
groenkaart2013	Type groen (5)
hpg2013	Poldergraslandtype (3)
huetzonerings2009	Type waterloop (19)
Overstromingsgevoeligheid2014	Gevoeligheid voor overstroming (3)
ZuidhellingenVliegendHert	Zuidhelling – geen zuidhelling (2)

2.1.3 Uitkomst mechanistische modellen

De GeoDynamix toolbox wordt hier gebruikt om potentiële leefgebieden aan te duiden op een kwalitatieve manier. Dit betekent dat de afgebakende gebieden ofwel worden beschouwd als geschikt (waarde 1) of ongeschikt (waarde 0). Er wordt dus geen indicatie gegeven over kwalitatieve verschillen tussen de verschillende afgebakende gebieden. Alle geschikte cellen van 20 x 20 m² die binnen een voor de soort overbrugbare afstand liggen, vormen een zogenaamde *cluster* en elk van deze clusters krijgt een uniek nummer toegewezen. Binnen deze clusters zijn alle onderdelen van het leefgebied van de soort aanwezig. Tussen deze clusters wordt verwacht dat de uitwisseling van individuen eerder beperkt is. De afbakening van dergelijke clusters kan helpen bij een visualisatie van de ruimtelijke onderverdeling en aldus bijdragen tot het uitwerken van maatregelen die een duurzaam netwerk van leefgebieden beogen.

2.2 Correlatieve modellen

De mogelijkheid om met correlatieve modellen de potentiële leefgebiedenkaarten te verfijnen, werd getest met de 17 vogels die in Bijlage I van de Europese Vogelrichtlijn vermeld staan en die in minstens 25 kilometerhokken voorkomen (Tabel 3). Daarnaast werd ook een correlatief model opgemaakt voor het Vliegend hert omdat deze soort zich leent tot het downscalen van dergelijke modellen naar een hogere resolutie (100 x 100 m; zie verder).

2.2.1 Broedvogels

Voor het samenstellen van de dataset werd gebruikt gemaakt van de puntwaarnemingen van de Bijlage I soorten voor de Natura 2000 rapportage, op basis van het Bijzondere Broedvogelsproject (BBV – Anselin et al. 2014), aangevuld met gegevens uit het dataportaal van Natuurpunt (waarnemingen.be). Per kilometerhok (1 x 1 km²) werd de aan- of afwezigheid van de soort in de periode 2007-2014 vermeld.

Als databron voor de omgevingsvariabelen werd de landgebruikskaart van Vlaanderen gebruikt (Stevens et al. 2014). Per 1 x 1 km² werd de oppervlakte van verschillende landgebruikstypen berekend (Tabel 4). Voor gebruik in de modellen werden deze variabelen genormaliseerd. Aangezien biotoopdiversiteit voor heel wat vogelsoorten een belangrijk aspect van het leefgebied zou kunnen zijn, werd naast de landgebruiksvariabelen per kilometerhok ook een biotoopdiversiteitsindex berekend (in dit geval de Shannon-Wiener index – Spellerberg & Fedor 2003) van de natuurlijke biotopen (boomgaard, duin, halfnatuurlijke graslanden, heide, hooggroen, laaggroen, loofbos, moeras, naaldbos, slikken & schorren, strand, struweel en water – Tabel 4).

In het biomod2-package (Thuiller et al. 2012) in R version 3.1.1 (R Core Team 2015) werden 5 algoritmen gebruikt: Generalized Additive Models (GAM – Hastie & Tibshirani 1987), Generalized Boosted Regression Modeling (GBM – Friedman et al. 2000), Generalized Linear Models (GLM – McCullagh & Nelder 1989), Maximum Entropy (MaxEnt – Phillips et al. 2006) en Random Forest (RF – Breiman 2001). Per soort werd de dataset verdeeld in een calibratieset waarmee het model gemaakt werd (70% van de data) en een evaluatieset (30%) waarmee het gemaakte model werd geëvalueerd. Per soort werden 10 random splits gedaan. In totaal, werden er dus voor elke soort 50 modellen gemaakt: 5 algoritmen x 10 random splits. De modellen werden geëvalueerd door middel van de Area Under the Curve (AUC) van de Receiver Operating Characteristic (ROC – Swets 1988). Modellen waarvoor de AUC groter is dan 0,7 worden als aanvaardbaar beschouwd. Door een gewogen gemiddelde te maken van de aanvaardbare modellen werd *ensemble model* bekomen (Araújo & New 2007) met de potentiële verspreiding van de soort in Vlaanderen. Dergelijke modellen kunnen enkel gemaakt worden voor soorten die in minstens 25 kilometerhokken voorkomen of afwezig zijn (n = 18). Dit minimum aantal kilometerhokken is noodzakelijk om voldoende aan- en/of afwezigheden te hebben om bruikbare modellen mee te calibreren. Bovendien wordt de set van kilometerhokken nog onderverdeeld in een calibratie- en een evaluatieset waardoor het aantal aan- en/of afwezigheden waarmee de modellen gemaakt en getest worden nog lager ligt dan 25.

Tabel 3 Broedvogels van de Europese Vogelrichtlijn waarvoor correlatieve modellen gemaakt werden met vermelding van het aantal kilometerhokken waarin ze broedend waargenomen werden. † momenteel worden voor deze soorten geen broedgevallen vastgesteld.

Soort	#kilometerhokken
Blauwborst (<i>Luscinia svecica</i>)	662
Boomleeuwerik (<i>Lullula arborea</i>)	422
Bruine kiekendief (<i>Circus aeruginosus</i>)	243
Duinpieper (<i>Anthus campestris</i>)†	-
Dwergstern (<i>Sterna albifrons</i>)	1
Grauwe kiekendief (<i>Circus pygargus</i>)	1
Grauwe klauwier (<i>Lanius collurio</i>)	27
Grote stern (<i>Sterna sandvicensis</i>)	1
Ijsvogel (<i>Alcedo atthis</i>)	686
Kleine zilverreiger (<i>Egretta garzetta</i>)	3
Kluut (<i>Recurvirostra avosetta</i>)	87
Korhoen (<i>Tetrao tetrix</i>)†	-
Kwak (<i>Nycticorax nycticorax</i>)	6
Kwartelkoning (<i>Crex crex</i>)	25
Lepelaar (<i>Platalea leucorodia</i>)	4
Middelste bonte specht (<i>Dendrocopos medius</i>)	246
Nachtzwaluw (<i>Caprimulgus europaeus</i>)	170
Ooievaar (<i>Ciconia ciconia</i>)	7
Ortolaan (<i>Emberiza hortulana</i>)†	-
Porseleinhoen (<i>Porzana porzana</i>)	58
Purperreiger (<i>Ardea purpurea</i>)†	-
Roerdomp (<i>Botaurus stellaris</i>)	28
Slechtvalk (<i>Falco peregrinus</i>)	42
Steltkluut (<i>Himantopus himantopus</i>)	8
Strandplevier (<i>Charadrius alexandrinus</i>)	9
Visdief (<i>Sterna hirundo</i>)	40
Wespendief (<i>Pernis apivorus</i>)	1585
Woudaap (<i>Ixobrychus minutus</i>)	25
Zwarte specht (<i>Dryocopus martius</i>)	1299
Zwarte stern (<i>Chlidonias niger</i>)†	-
Zwartkopmeeuw (<i>Larus melanocephalus</i>)	33



Bruine kiekendief (foto Glenn Vermeersch)

Tabel 4 Gemiddelde waarden en standaardfout (se) van de landgebruiksvariabelen per kilometerhok (in m²) die werden gebruikt in de correlatieve modellen voor de broedvogels van de Europese Vogelrichtlijn in Vlaanderen. Enkel kilometerhokken die voor minstens 50% in Vlaanderen liggen, werden gebruikt. De Biotoopdiversiteitsindex werd berekend als een Shannon-Wiener index met de voor vogels relevante biotopen.

Landgebruik	Gemiddelde	se
Akker	200.519	1.416
Cultuurgrasland	138.340	990
Laaggroen	88.525	652
Loofbos	60.209	850
Hooggroen	45.495	471
Naaldbos	41.382	982
Water	23.137	558
Halfnatuurlijke grasland	13.254	211
Boomgaard	12.458	450
Heide	5.996	445
Struweel	2.804	76
Duin	1.422	200
Moeras	1.265	64
Slikken & Schorren	1.089	133
Strand	330	64
Biotoopdiversiteitsindex	4,848	0,014

2.2.2 Vliegend hert

Voor de modellering van het Vliegend hert werden gegevens gebruikt die verzameld werden voor het opmaken van de Rode Lijst (Thomaes & Maes 2014). De landgebruiksvariabelen die in het model gebruikt werden staan vermeld in Tabel 5 (Thomaes et al. 2008).

Tabel 5 Gemiddelde waarden en standaardfout (se) van de landgebruiksvariabelen per kilometerhok (in m²) die werden gebruikt in het correlatieve model voor het Vliegend hert in Vlaanderen. Enkel kilometerhokken die voor minstens 50% in Vlaanderen liggen werden gebruikt.

Landgebruik	Gemiddelde	se
Loofbos	60.700	863
Infrastructuur	60.095	428
Oud bos	22.999	674
Halfnatuurlijke grasland	13.268	212
Boomgaard	12.536	452
Heide	6.006	446
Struweel	2.792	76
Moeras	1.261	64
Andere omgevingsvariabelen		
Vochtklasse (mediaan per hok; varieert van 1-8)	3,63	0,01
Bodemtextuur (mediaan per hok; varieert van 1-8,5)	4,85	0,02

2.2.3 Uitkomst correlatieve modellen

De uitkomst van een correlatief model is een kans (*probabiliteit*) dat een soort in een bepaald kilometerhok voorkomt op basis van de daar aanwezige biotische en abiotische variabelen. Deze kans zou gebruikt kunnen worden als een kwaliteitsmaat: hoe hoger de probabiliteit, hoe beter geschikt de locatie voor de soort. Op basis van een grenswaarde kunnen deze probabiliteiten omgezet worden naar aan-/afwezigheden (1/0-waarden). Deze grenswaarde wordt door het model berekend en probeert zowel het aantal juist voorspelde aanwezigheden als de juist voorspelde afwezigheden te maximaliseren (Swets 1988).

2.3 Downscaling

Met het downscalen van modellen bedoelen we dat een model dat op een bepaalde schaal gecalibreerd wordt (bv. $1 \times 1 \text{ km}^2$) maar dat de projectie naar een hogere resolutie gebeurt (Araújo et al. 2005). We testen deze werkwijze uit op het Vliegend hert, waarbij we het hogervermelde correlatieve model met schaalniveau $1 \times 1 \text{ km}^2$, projecteren naar hectareniveau (i.e., $100 \times 100 \text{ m}^2$). Omwille van het schaalverschil maken we gebruik van procentuele verhoudingen in plaats van absolute getallen.

2.4 Begeleidende soorten

Onder begeleidende soorten verstaan we soorten gelijkaardige ecologische eisen stellen aan hun habitat en daardoor vaak samen voorkomen met de doelsoort. Deze soorten kunnen als indicatoren gebruikt worden voor de mogelijke verspreiding van de doelsoort (Maes & Van Dyck 2005b). In veel gevallen worden begeleidende soorten binnen dezelfde taxonomische groep gezocht, maar ook over de grenzen van taxonomische groepen heen kunnen begeleidende soorten gevonden worden (Maes & Van Dyck 2005a), bv. de verspreiding van bepaalde (waard)planten kan gebruikt worden om de mogelijke verspreiding van bepaalde dagvlinders te achterhalen. Voor dagvlinders werd een dergelijke oefening gedaan op basis van een Indicator Value (IndVal) analyse (Maes et al. 2007). Om een beeld te krijgen van de potentiële verspreiding van bijzonder zeldzame soorten, kunnen mechanistische modellen van begeleidende, meer algemene soorten gemaakt en gesommeerd worden waarbij ervan uit gegaan wordt dat plaatsen met een groot aantal begeleidende soorten het meest geschikt zijn voor de zeldzamere soort(en).

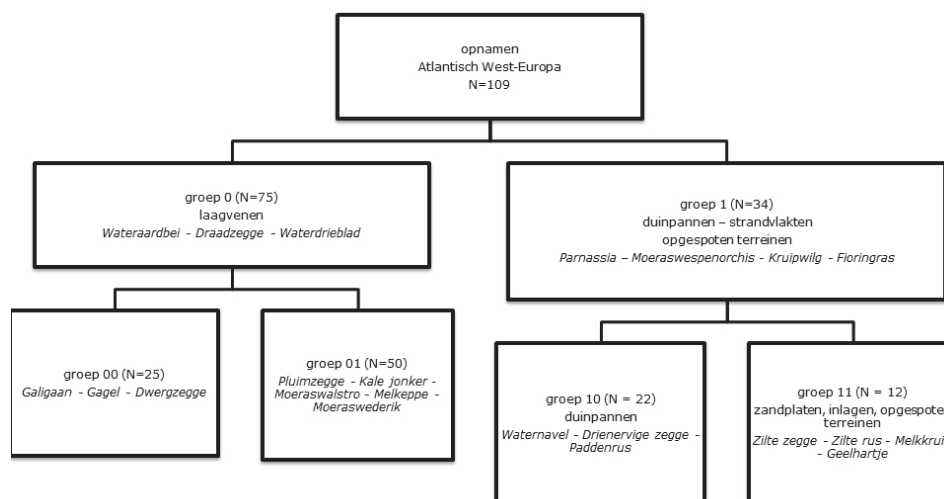
Het gebruiken van begeleidende soorten is een van de manieren om de mogelijke verspreiding van een doelsoort te achterhalen. Hier gebruiken we de 2 voorbeelden: Groenknolorchis (*Liparis loeselii*) en de broedvogels van Bijlage I van de Europese Vogelrichtlijn.

2.4.1 Groenknolorchis

Op niveau Vlaanderen kan gebruik gemaakt worden van de verspreidingsgegevens van vaatplanten en mossen in de databank Florabank (Van Landuyt et al. 2012). Wat betreft de vaatplanten zijn er over heel Vlaanderen gegevens voorhanden, de mossengegevens zijn veel beperkter (Van Landuyt 2011) alhoewel in kansrijke gebieden zoals duinpannen en alkalische laagveen wel beter gezocht is en dus in deze gebieden hierover doorgaans meer gegevens beschikbaar zijn.

Om de werkwijze van het gebruik van begeleidende soorten te verduidelijken, hernemen we een eerdere oefening die deze methode hanteerde voor de Groenknolorchis in Vlaanderen (Van Landuyt et al. 2014). Groenknolorchis is een soort die op vrij diverse terreinentypen kan voorkomen. In de Atlantische delen van West-Europa komt de soort voor in alkalische laagveenmoerassen, vochtige duinvalleien en natte overgangsmilieus tussen brakke en zoete pioniersvegetaties. Soms komt ze ook voor op natte plaatsen op kalkrijke opgespoten terreinen of in groeves (Andeweg 2011; Spanoghe et al. 2008; Valentin et al. 2010; Vanden Berghen 1943). Het betreft over het algemeen vrij open kruidenvegetaties alhoewel Groenknolorchis tijdelijk kan standhouden in open struweel.

Op basis van een opnamedataset van 109 vegetatieopnamen op de groeiplaatsen van Groenknolorchis in Nederland, België en Noord-Frankrijk werd gepoogd een typologie van de standplaatsen op te stellen en kenmerkende soorten voor elke vegetatietype te selecteren. Op basis van een clusteranalyse (Twinspan – Hill 1979) op deze dataset werden twee hoofdtypen onderscheiden, namelijk vegetaties op organische laagveenbodems en vegetaties op minerale bodems (duinpannen, strandvlakten, platen Grevelingen en opgespoten terreinen – Figuur 1).



Figuur 1 Clusteranalyse (Twinspan – Hill 1979) van 109 vegetatieopnamen met Groenknolorchis in Nederland, België en Noord-Frankrijk met aanduiding van de differentiërende soorten per groep. Opnamen van laagveen onderscheiden zich door de aanwezigheid van de differentiërende soorten Watersaardbei, Draadzegge en Waterdrieblad van de opnamen op minerale bodem met als differentiërende soorten Parnassia, Moeraswespenorchis, Kruiwilg en Fioringras.

Vegetatiekundig behoren de meeste opnamen tot het Knopbies-verbond (Schaminée et al. 1995). Deze omvatten natuurlijke of half-natuurlijke begroeiingen op min of meer basenrijk substraat. De standplaatsen worden gevoed door basenrijk, eventueel zwak brak grond- of oppervlaktewater. De grondwaterstand kan ofwel zeer constant zijn (bv. bij trilvenen en alkalische laagvenen) of juist sterk schommelen (bv. in duinpannen of op sommige opgespoten terreinen). Toch zijn de standplaatsen tijdens een groot deel van het jaar erg nat. De vegetaties ontwikkelen zich het best waar mineraalrijk grondwater aangevoerd wordt of in de contactzone tussen regenwater en grondwater. Nutriënten kunnen aanwezig zijn, maar zijn over het algemeen weinig beschikbaar doordat ze aan calcium of ijzer gebonden zijn.

Schaminée et al. (1995) vermelden als kenmerkende vaatplanten voor deze plantengemeenschap Parnassia, Dwergzegge, Moeraswespenorchis, Groenknolorchis, Armbloemige waterbies, Zwarte knopbies, Bonte paardenstaart en als kenmerkende mossen Sterrengoudmos, Goudsikkelmos, Groot vedermos, Gewoon moerasvorkje en lokaal ook Vierkantsmos.

De vegetaties op organische laagveenbodem (twinspangroep 0) worden gekenmerkt door de aanwezigheid van Watersaardbei, Draadzegge en Waterdrieblad. Het betreft trilvenen en andere laagvenen waar vaak enige invloed is van basenrijk grondwater. Paddenrus, Moerasvaren, Melkeppe en Waternavel zijn frequente begeleiders. In dit soort vegetaties staat Groenknolorchis vaak op de rand van zeggebulten of in kussens (veen)mos. Bij verdere opdeling van deze groep bekomt men een groep met Galigaan, Gagel en Rood schorpioenmos als indicatieve soorten (groep 00) en een groep vegetaties van trilvenen bestaande uit grote bultenvormende zeggen zoals Pluimzegge en soorten als Moeraswederik en Melkeppe (groep 01). Het betreft permanent natte standplaatsen aangezien de vegetatiemat in vele gevallen drijvend is en op en neer gaat met de waterstand.

Op minerale bodems (twinspangroep 1) zijn Fioringras, Moeraswespenorchis, Kruipwilg en Parnassia frequente begeleiders. Bij een verdere opdeling van deze groep in de twinspananalyse worden de vegetaties in duinpannen (twinspangroep 10) afgescheiden van vegetaties die vaak iets meer op de overgang tussen brak en zoet liggen zoals op de platen in het Grevelingenmeer, de groeiplaats langs het Oostvoornse meer en ook een deel van de opgespoten terreinen (twinspangroep 11) met als kenmerkende soort Zilte zegge. In duinpannen (twinspangroep 10) zijn Waternavel, Kruipwilg, Watermunt en Dwergzegge de meest frequente soorten naast een hele groep zeldzame maar meer kenmerkende soorten zoals Parnassia, Teer guichelheil, Moeraswespenorchis, Vleeskleurige orchis, Drienvervige zegge, Armbloemige waterbies en Bonte paardenstaart.

De lijst met kenmerkende begeleidende soorten voor Groenknolorchis is te vinden in Tabel 6.

Tabel 6 Lijst met kenmerkende begeleidende soorten van vegetaties met Groenknolorchis.

Nederlandse naam (Wetenschappelijke naam)

Bladmossen

Sterrengoudmos (*Campylium stellatum* (Hedw.) Lange & C.E.O.Jensen)

Tenger goudmos (*Campyliadelphus elodes* (Lindb.) Kanda)

Levermos

Vierkantmos (*Preissia quadrata* (Scop.) Nees)

Vaatplanten

Armbloemige waterbies (*Eleocharis quinqueflora* (F.X. Hartm.) O. Schwartz)

Bonte paardenstaart (*Equisetum variegatum* Schleich.)

Draadzegge (*Carex lasiocarpa* Ehrh.)

Dwergzegge (*Carex viridula* Michaux)

Galigaan (*Cladium mariscus* (L.) Pohl)

Knopbies (*Schoenus nigricans* L.)

Moeraswespenorchis (*Epipactis palustris* (L.) Crantz)

Parnassia (*Parnassia palustris* L.)

Ronde zegge (*Carex diandra* Schrank)

Slank wollengras (*Eriophorum gracile* Koch ex Roth)

Teer guichelheil (*Anagallis tenella* (L.) L.)

Tweehuizige zegge (*Carex dioica* L.)

Vleeskleurige orchis (*Dactylorhiza incarnata* (L.) Soó)

2.4.2 Broedvogels van de Europese Vogelrichtlijn

Als bijkomende en verkennende test voor het toepassen van begeleidende soorten voor het verfijnen van potentiële leefgebiedenkaarten, gebruiken we hier de 17 soorten van Bijlage I van de Europese Vogelrichtlijn. Voor elk van deze 17 soorten gaan we 1) op zoek naar begeleidende soorten op basis van verspreidingsgegevens van alle broedvogels afkomstig van het dataportaal van Natuurpunt (www.waarnemingen.be) en 2) berekenen we een zogenaamde *indicatorwaarde* (IndVal – Dufrêne & Legendre 1997). Hiervoor gebruikten we het package *indicspecies* (De Cáceres & Legendre 2009) in R (R Core Team 2015). Om als begeleidende soort in aanmerking te komen moet zowel de *specificiteit* (de kans dat een begeleidende soort samen voorkomt met de doelsoort) als de *trouwheid* (de kans op het vinden van de soort in kilometerhokken waarin de doelsoort voorkomt) groter zijn dan 0,7. in (Tabel 7) geven we voor de 17 broedvogelsoorten het aantal begeleidende soorten en hun indicatorwaarde. Om vervolgens te komen tot een potentiekaart wordt de indicatorwaarde van de aanwezige begeleidende soorten per kilometerhok gesommeerd.

Tabel 7 Begeleidende soorten en, tussen haakjes, hun indicatorwaarde voor de broedvogels van Bijlage I van de Europese Vogelrichtlijn.

Soort	Begeleidende soorten
Blauwborst (17)	Rietgors (0.887), Kleine karekiet (0.858), Meerkoet (0.816), Scholekster (0.808), Roodborsttapuit (0.803), Bergeend (0.803), Krakeend (0.801), Slechtvalk (0.793), Sprinkhaanzanger (0.789), Aalscholver (0.787), Kuifeend (0.783), Bruine kiekendief (0.765), Fuut (0.762), Dodaars (0.751), Ooievaar (0.747), Kleine mantelmeeuw (0.740), Kokmeeuw (0.721)
Boomleeuwerik (10)	Boompieper (0.892), Bonte vliegenvanger (0.828), Kruisbek (0.815), Gekraagde roodstaart (0.812), Kuifmees (0.796), Havik (0.794), Roodborsttapuit (0.794), Zwarte specht (0.782), Zwarte mees (0.763), Wulp (0.732)
Bruine kiekendief (5)	Blauwborst (0.814), Rietgors (0.814), Bergeend (0.762), Kwartel (0.753), Kuifeend (0.741)
Grauwe klauwier (6)	Boompieper (0.825), Geelgors (0.805), Havik (0.803), Appelvink (0.800), Goudvink (0.765), Kruisbek (0.748)
Ijsvogel (4)	Aalscholver (0.780), Fuut (0.775), Kuifeend (0.747), Krakeend (0.730)
Kluut (30)	Tureluur (0.920), Bergeend (0.888), Slobeend (0.879), Rietzanger (0.878), Blauwborst (0.869), Rietgors (0.862), Bruine kiekendief (0.861), Scholekster (0.845), Kleine karekiet (0.843), Zomertaling (0.842), Grutto (0.837), Meerkoet (0.834), Aalscholver (0.832), Slechtvalk (0.831), Kuifeend (0.827), Kleine plevier (0.824), Wulp (0.823), Kokmeeuw (0.823), Krakeend (0.821), Knobbelzwaan (0.815), Zilvermeeuw (0.805), Putter (0.801), Fuut (0.794), Cetti's zanger (0.792), Dodaars (0.791), Visdief (0.789), Wintertaling (0.785), Oeverzwaluw (0.782), Waterral (0.751), Zwartkopmeeuw (0.737)
Kwartelkoning (3)	Knobbelzwaan (0.724), Waterral (0.715), Oeverzwaluw (0.714)
Middelste bonte specht (12)	Boomklever (0.813), Kleine bonte specht (0.798), Bosuil (0.791), Wespendif (0.788), Havik (0.780), Boomkruiper (0.778), Goudhaan (0.770), Matkop (0.769), Kuifmees (0.738), Zwarte specht (0.726), Zwarte mees (0.721), Appelvink (0.683)
Nachtzwaluw (10)	Boompieper (0.889), Boomleeuwerik (0.874), Bonte vliegenvanger (0.848), Zwarte specht (0.835), Gekraagde roodstaart (0.832), Kruisbek (0.832), Kuifmees (0.828), Havik (0.807), Zwarte mees (0.805), Houtsnip (0.775)
Porseleinhoen (4)	Krakeend (0.802), Knobbelzwaan (0.788), Waterral (0.762), Slobeend (0.762)
Roerdomp (13)	Waterral (0.838), Knobbelzwaan (0.825), Krakeend (0.816), Slobeend (0.816), Dodaars (0.816), Kokmeeuw (0.812), Kuifeend (0.808), Rietgors (0.806), Fuut (0.797), Zomertaling (0.794), Tafelend (0.785), Wintertaling (0.758), Oeverzwaluw (0.741)
Slechtvalk (5)	Scholekster (0.768), Blauwborst (0.760), Bruine kiekendief (0.751), Rietgors (0.749), Kuifeend (0.727)
Visdief (20)	Bergeend (0.867), Meerkoet (0.847), Zilvermeeuw (0.840), Fuut (0.839), Kleine karekiet (0.833), Scholekster (0.832), Rietzanger (0.829), Aalscholver (0.821), Kuifeend (0.817), Blauwborst (0.806), Krakeend (0.796), Kokmeeuw (0.789), Slobeend (0.789), Kleine mantelmeeuw (0.787), Knobbelzwaan (0.787), Rietgors (0.785), Zwartkopmeeuw (0.781), Putter (0.751), Bruine kiekendief (0.751), Ijsvogel (0.732)
Wespendif (6)	Boomklever (0.832), Bosuil (0.806), Kleine bonte specht (0.802), Havik (0.798), Matkop (0.776), Goudhaan (0.771)
Woudaap (27)	Tafelend (0.876), Krakeend (0.875), Slobeend (0.872), Knobbelzwaan (0.864), Dodaars (0.852), Kuifeend (0.852), Aalscholver (0.846), Zomertaling (0.846), Rietzanger (0.834), Bergeend (0.833), Oeverzwaluw (0.833), Sprinkhaanzanger (0.831), Rietgors (0.827), Fuut (0.826), Blauwborst (0.826), Cetti's zanger (0.824), Kokmeeuw (0.816), Waterral (0.813), Ijsvogel (0.810), Visdief (0.805), Kleine zilverreiger (0.791), Tureluur (0.789), Zilvermeeuw (0.782), Kleine plevier (0.774), Wintertaling (0.768), Nachtegaal (0.727)
Zwarte specht (9)	Kuifmees (0.866), Boomklever (0.848), Havik (0.842), Matkop (0.823), Zwarte mees (0.815), Kleine bonte specht (0.809), Goudhaan (0.798), Boompieper (0.763), Gekraagde roodstaart (0.754)
Zwartkopmeeuw (11)	Scholekster (0.846), Kokmeeuw (0.783), Blauwborst (0.782), Bergeend (0.780), Aalscholver (0.779), Kleine mantelmeeuw (0.767), Krakeend (0.759), Kuifeend (0.756), Wulp (0.743), Bruine kiekendief (0.736), Rietgors (0.725)

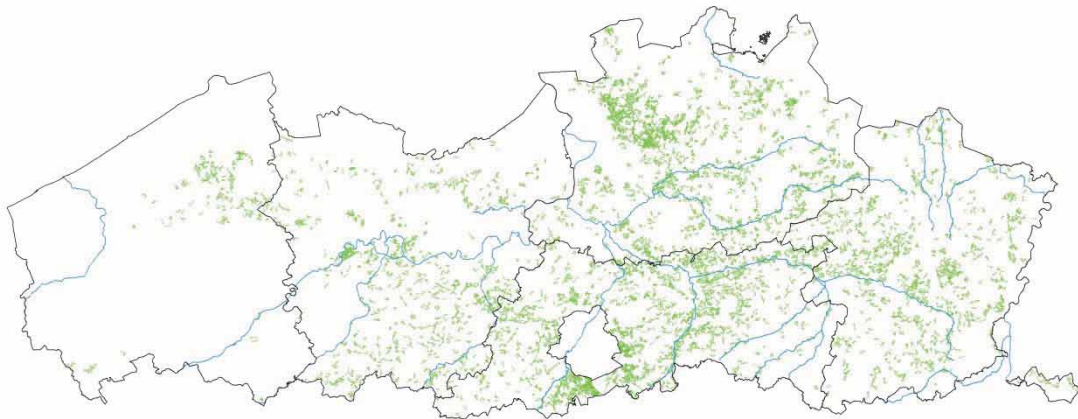
3 Resultaten

Hieronder bespreken we de resultaten van de 4 verschillende manieren om potentiële leefgebiedenkaarten te verfijnen.

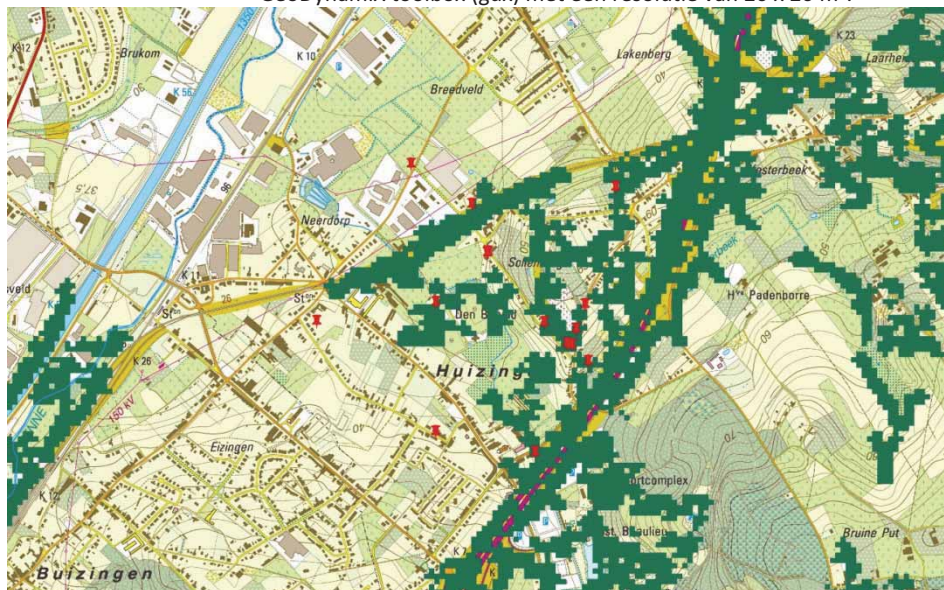
3.1 Mechanistische modellen

Voor 81 soorten was het mogelijk om kaarten met verfijnde potentiële leefgebieden te maken (Tabel 1). Om de gdx-scripts voor deze soorten en de daaruit voortvloeiende potentiële leefgebiedenkaarten te documenteren, geven we beide als dusdanig weer in Bijlage 1. De shape-files van deze kaarten kunnen opgevraagd worden bij het INBO.

Hier geven we als voorbeeld de potentiële verspreiding van het Vliegend hert zoals afgebakend met behulp van een gdx-script voor heel Vlaanderen (Figuur 2) en in meer detail voor de regio rond Huizingen (Figuur 3).



Figuur 2 Potentiële leefgebiedenkaart versie 2.0 van het Vliegend hert zoals aangemaakt met de GeoDynamix toolbox (gdx) met een resolutie van 20 x 20 m².



Figuur 3 Detail van de potentiële leefgebiedenkaart van het Vliegend hert. Donkergroen: potentiële leefgebied van het Vliegend hert (resolutie 20 x 20 m²), Rood: recente waarnemingen van het Vliegend hert in de regio Huizingen.

3.2 Correlatieve modellen

3.2.1 Broedvogels van de Europese Vogelrichtlijn

Voor 16 van de 17 broedvogels van de Europese Vogelrichtlijn konden correlatieve modellen worden gemaakt waarvoor de gemiddelde AUC hoger was dan 0,7 (Tabel 8). Enkel voor Kwartelkoning bleek dit niet het geval te zijn. De potentiële verspreidingskaarten voor deze 16 soorten op basis van de correlatieve modellen zijn te vinden in Bijlage 1.

Tabel 8 Gemiddelde Area Under the Curve (AUC) van de Receiver Operating Characteristic (ROC) voor de 10 model runs per algoritme (GAM: Generalized Additive Models, GBM: Generalized Boosted Regression Modeling, GLM: Generalized Linear Models, MaxEnt: Maximum Entropy, RF: Random Forest) en het algemeen gemiddelde over alle algoritmen heen (Gem) van de 17 broedvogels van de Europese Vogelrichtlijn voor de verschillende algoritmen. De waarden in het vet zijn AUC's die kleiner zijn dan 0,7 en werden niet gebruikt in de ensemble modelling. N = aantal runs dat gebruikt werd voor de ensemble modelling (max = 50).

Soort	GAM	GBM	GLM	MaxEnt	RF	Gem	N
Blauwborst	0.815	0.797	0.819	0.774	0.812	0.803	49
Boomleeuwerik	0.942	0.938	0.945	0.924	0.946	0.939	50
Bruine kiekendief	0.918	0.868	0.919	0.766	0.911	0.876	46
Grauwe klauwier	0.692	0.694	0.722	0.725	0.723	0.711	32
Ijsvogel	0.871	0.861	0.871	0.863	0.868	0.867	50
Kluut	0.911	0.852	0.920	0.922	0.923	0.905	50
Kwartelkoning	0.650	0.588	0.644	0.700	0.665	0.649	14
Middelste bonte specht	0.914	0.894	0.914	0.817	0.904	0.889	48
Nachtzwaluw	0.939	0.932	0.938	0.905	0.949	0.932	50
Porseleinhoen	0.828	0.790	0.851	0.802	0.808	0.816	49
Roerdomp	0.745	0.677	0.762	0.773	0.748	0.741	34
Slechtvalk	0.858	0.710	0.855	0.895	0.738	0.811	42
Visdief	0.852	0.774	0.854	0.920	0.876	0.855	46
Wespendief	0.746	0.746	0.743	0.734	0.749	0.744	50
Woudaap	0.773	0.732	0.853	0.847	0.793	0.800	45
Zwarte specht	0.899	0.889	0.900	0.888	0.897	0.894	50
Zwartkopmeeuw	0.719	0.691	0.802	0.747	0.785	0.749	36
Gemiddelde	0.828	0.790	0.842	0.824	0.829	0.823	44

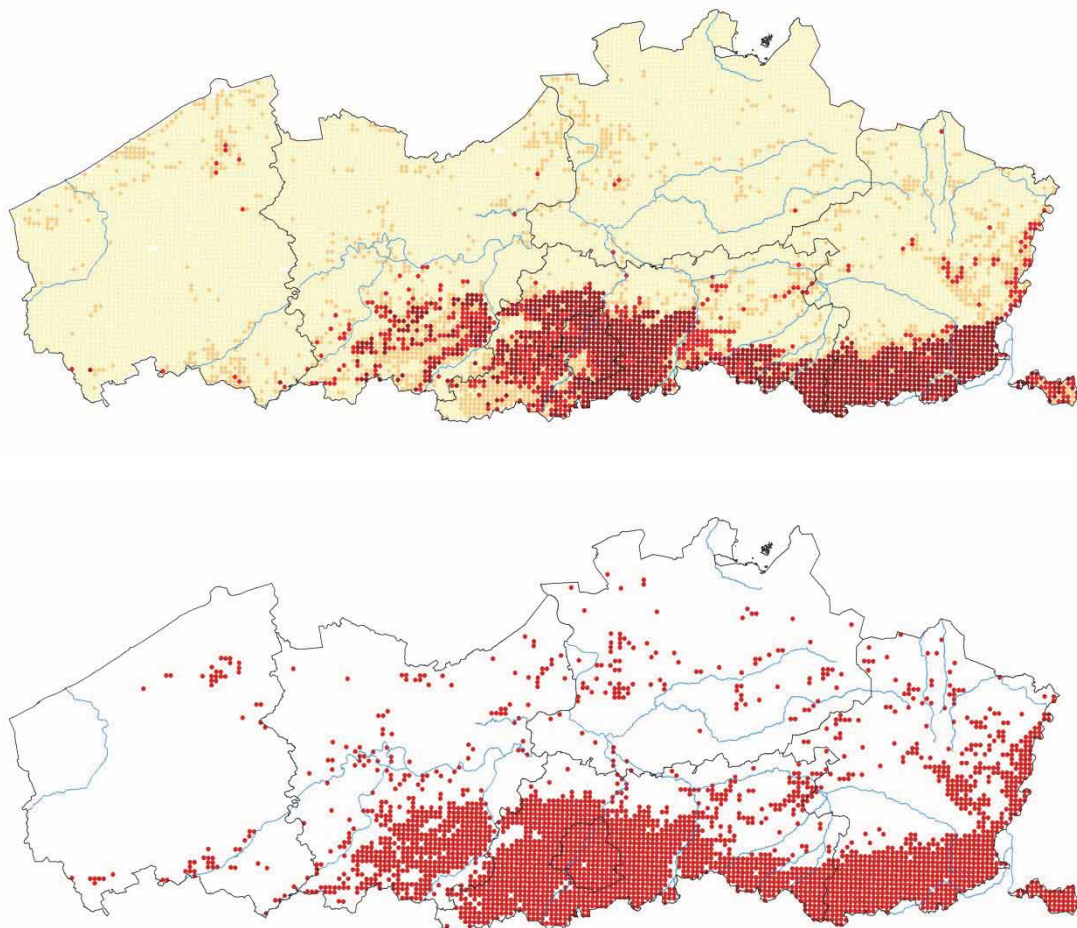
3.2.2 Vliegend hert

Voor het Vliegend hert lag de AUC voor alle 50 model runs boven de 0,7 (Tabel 9). Zo goed als alle model runs konden gebruikt worden voor het maken van de finale potentiële verspreidingskaart.

Tabel 9 Gemiddelde Area Under the Curve (AUC) van de Receiver Operating Characteristic (ROC) voor de verschillende algoritmen (GAM: Generalized Additive Models, GBM: Generalized Boosted Regression Modeling, GLM: Generalized Linear Models, MaxEnt: Maximum Entropy, RF: Random Forest) en het algemeen gemiddelde over alle algoritmen heen (Gem) die gebruikt werden om het potentiële leefgebied van Vliegend hert in kaart te brengen.

Soort	GAM	GBM	GLM	MaxEnt	RF	Gem
Vliegend hert	0.836	0.881	0.804	0.791	0.883	0.839

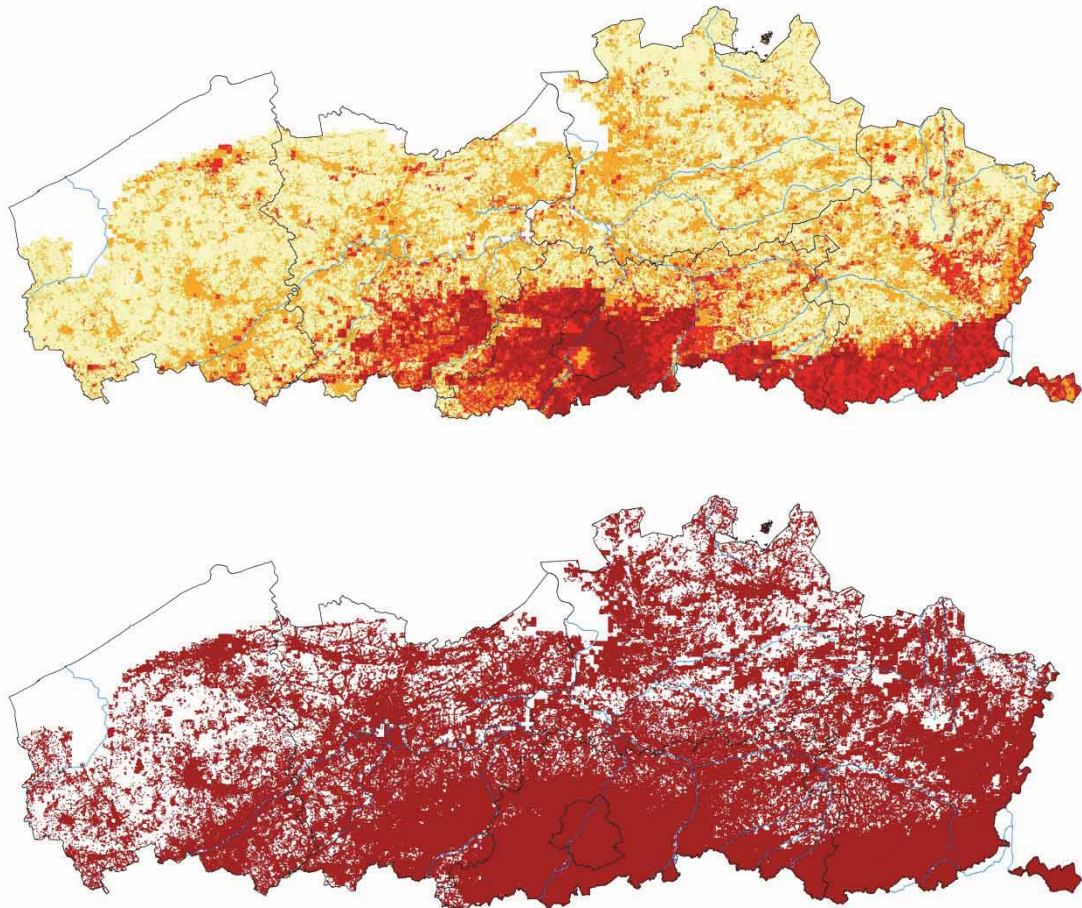
De potentiële verspreidingskaart met de kans op de aanwezigheid van het Vliegend hert per kilometerhok wordt weergegeven in Figuur 4.



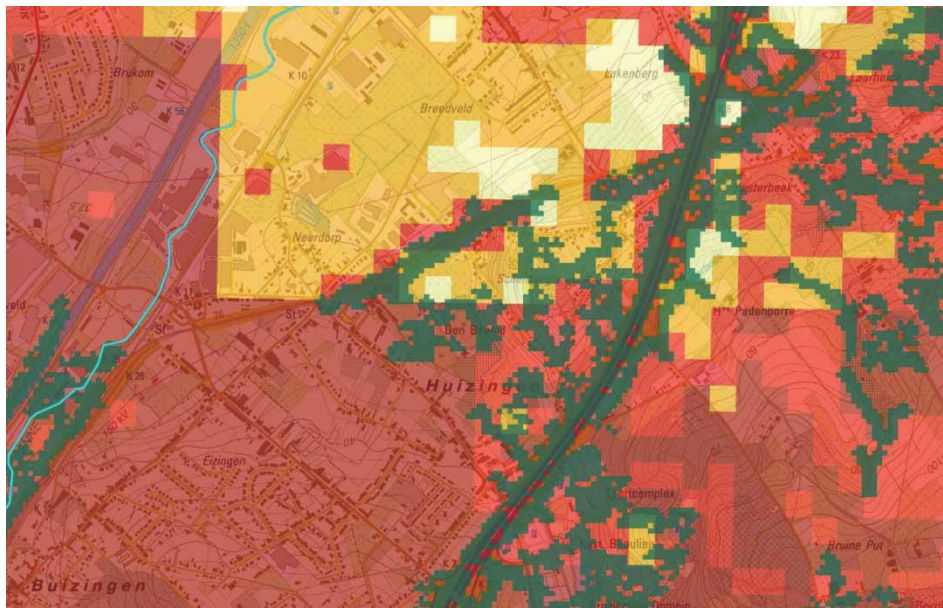
Figuur 4 Kans (probabiliteit) op aanwezigheid van het Vliegend hert in de verschillende kilometerhokken in Vlaanderen op basis van correlatieve modellen (boven; geel = <25%, oranje = 25-50%, rood = 50-75% en bruin = 75-100%) en de omzetting van deze probabiliteit naar aanwezigheid (rood) op basis van een door de ensemble modellering bepaalde grenswaarde (onder).

3.3 Downscaling

Met behulp van het model dat gemaakt werd op kilometerhokniveau werd een predictie gedaan op hectareniveau. Figuur 5 geeft de resultaten weer met per hectarehok de probabiliteiten en de omzetting ervan naar potentiële aan-/afwezigheid. Op Figuur 6 geven we het detail van deze downscaling weer voor Huizingen (dezelfde regio als op Figuur 3). Het hoekige patroon wordt veroorzaakt door de bodemtextuur- en bodemvochtigheidsvariabelen die door het gebrek aan voldoende gedetailleerde kaarten voor het hele kilometerhok dezelfde waarde hebben.



Figuur 5 Gedownscalede potentiële verspreidingskaarten met de kans (probabiliteit) op aanwezigheid van het Vliegend hert in de verschillende hectarehokken in Vlaanderen op basis van correlatieve modellen (boven; geel = <25%, oranje = 25-50%, rood = 50-75% en bruin = 75-100%) en de omzetting van deze probabiliteit naar aanwezigheid (rood) op basis van een door de ensemble modelling bepaalde grenswaarde (onder).



Figuur 6 Detail van de gedownscalede potentiële leefgebiedenkaart van het Vliegend hert in de regio Huizingen op basis van het correlatieve model. Per hectarehok wordt de kans op aanwezigheid gegeven (geel = <25%, oranje = 25-50%, rood = 50-75% en bruin = 75-100%). In het donkergroen wordt het potentiële leefgebied van het Vliegend hert weergegeven zoals afgebakend met behulp van het gdx-script (resolutie 20 x 20 m²).

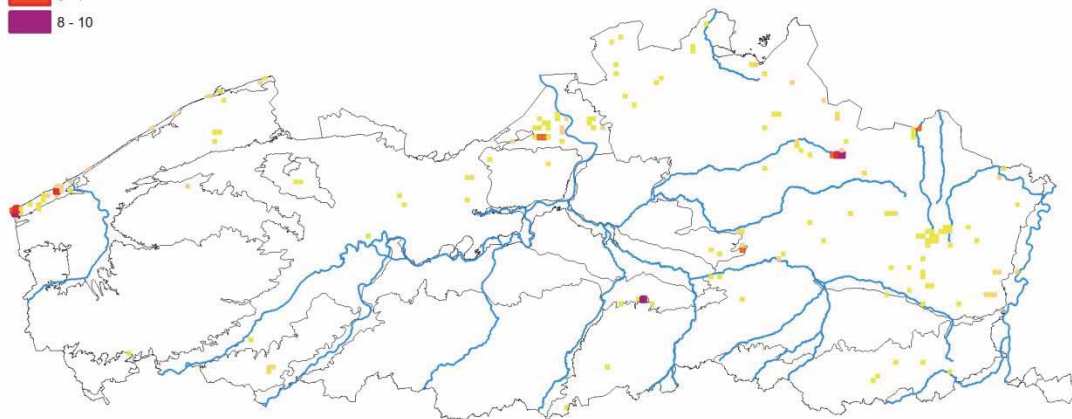
3.4 Begeleidende soorten

3.4.1 Groenknolorchis

Figuur 7 geeft een overzicht van de locaties waar kenmerkende begeleidende soorten (zie Tabel 6) die samen met Groenknolorchis voorkomen, gevonden zijn in de periode 1990-2013. Het hoogste aantal kenmerkende begeleidende soorten wordt gevonden in het Westhoekreservaat (tot 10 kenmerkende begeleiders per km²), het Torfbroek te Berg – Kampenhout (9 begeleidende soorten per km²) en het Buitengoor en omgeving te Mol (tot 9 begeleidende soorten per km² - Figuur 7). Dit zijn locaties waar Groenknolorchis ooit is gevonden, of in het geval van het Buitengoor, waar Groenknolorchis nog aanwezig. In het geval van het Buitengoor en omgeving werd in 2014 in het natuurreservaat zelf een nieuwe groeiplaats van Groenknolorchis ontdekt naast de oude locatie (Sluismeer) buiten het reservaat aan de andere kant van het Kanaal Dessel-Kwaadmechelen. Andere locaties met een weliswaar iets lager aantal kenmerkende begeleiders zijn het duinenreservaat Ter Yde (Oostduinkerke, 6 soorten) en de groeiplaats van Groenknolorchis op de Haasop (Beveren, Kallo, 4 soorten per km²). Dat deze laatste groeiplaats slechts een gering aantal kenmerkende begeleidende soorten telt, heeft mogelijk te maken met de recente ontstaansgeschiedenis van de Haasop (opgespoten gronden die pas in 1985 begonnen te stabiliseren – Spanoghe et al. 2008).

Zoekkaart potentieel geschikte groeiplaatsen groenknolorchis

Aantal indicatieve soorten



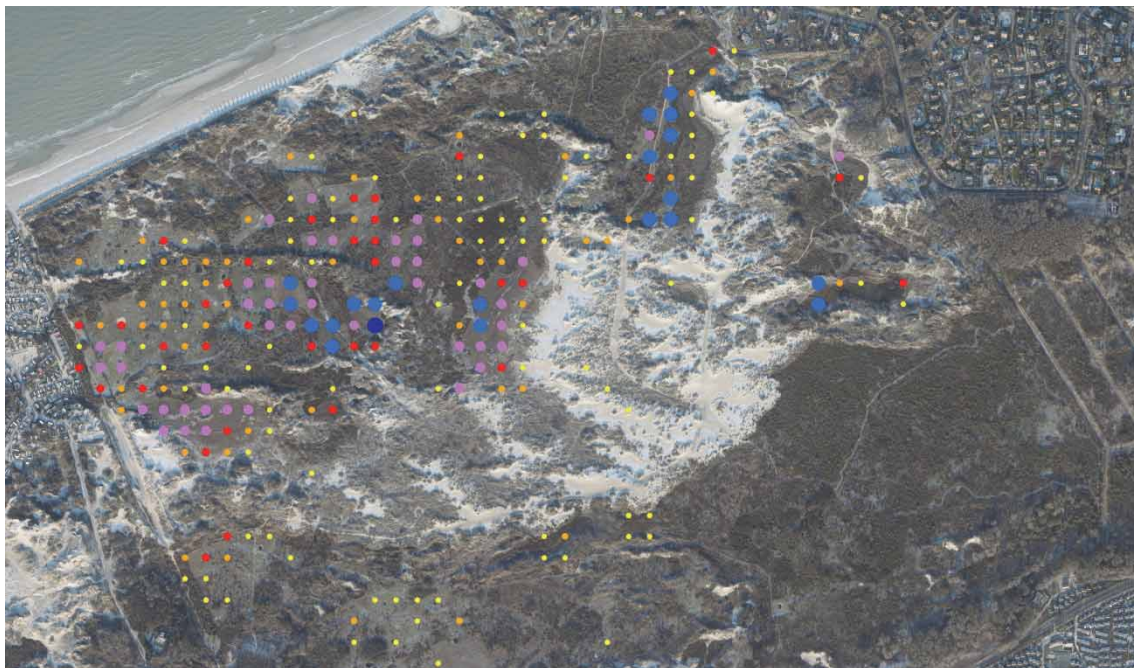
Figuur 7 Zoekkaart voor potentieel geschikte locaties voor Groenknolorchis in Vlaanderen aan de hand van een aantal indicatorsoorten (13 vaatplanten, 2 bladmossen en 1 levermos – Tabel 6).

Voor de kustduinen, waar de meest kansrijke gebieden voor Groenknolorchis in duinvalleien liggen, zijn voor alle gebieden in beheer bij het Agentschap Natuur en Bos nog veel gedetailleerde en recentere verspreidingsgegevens van vaatplanten beschikbaar (Provoost et al. 2010; Provoost et al. 2015). Dit is ook het geval voor het merendeel van de overige duingebieden. Dit laat ons toe veel gedetailleerder in te zoomen op potentiële groeiplaatsen voor Groenknolorchis en op deze locaties ook peilbuisgegevens te gaan vergelijken met de grondwaterfluctuaties op actuele groeiplaatsen.

Het grootste aantal kenmerkende soorten aan de kust is te vinden in de noordelijke pannengordel van het Westhoekreservaat in De Panne (Figuur 8), maar ook in het natuurreservaat Ter Yde in Koksijde zijn grote aantallen kenmerkende soorten gevonden (Figuur 9).



Groenknolorchis (foto Wouter Van Landuyt)



Figuur 8 Kaart met het aantal typische begeleidende vaatplantsoorten voor duinpannen met Groenknolorchis (zie Tabel 6) zoals aanwezig in het Westhoekreservaat (De Panne) per raster van 50 x 50 meter. Gele cirkels: 1 kenmerkende soort, oranje: 2 kenmerkende soorten, rood: 3 kenmerkende soorten, paars: 4 kenmerkende soorten, lichtblauw: 5 kenmerkende soorten, donkerblauw: 6 kenmerkende soorten.



Figuur 9 Kaart met het aantal typische begeleidende vaatplantsoorten voor duinpannen met Groenknolorchis (zie Tabel 6) in het duinengebied Ter Yde (Oostduinkerke) per raster van 50 x 50 meter. Gele cirkels: 1 kenmerkende soort, oranje: 2 kenmerkende soorten, rood: 3 kenmerkende soorten, paars: 4 kenmerkende soorten.

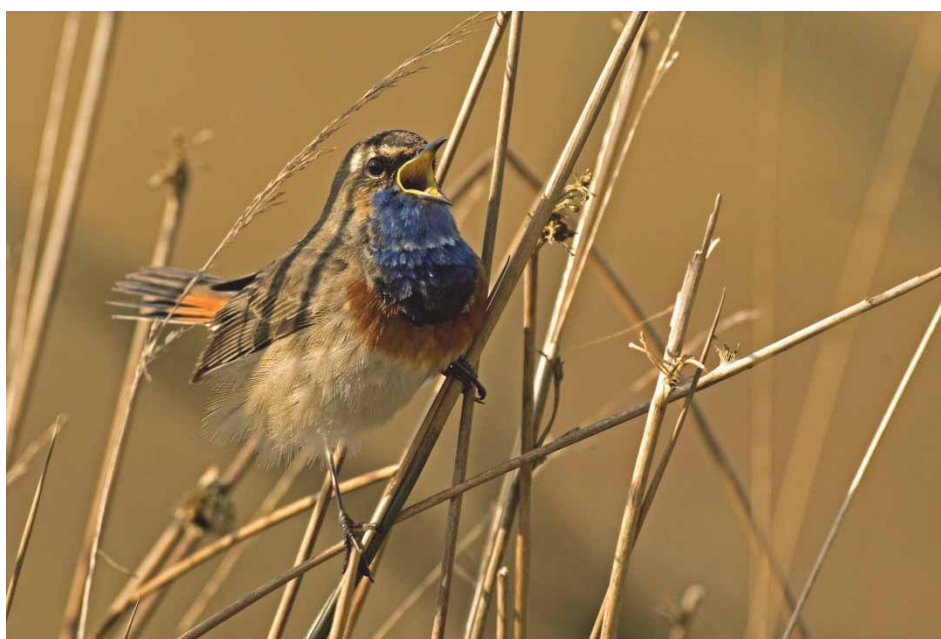
3.4.2 Broedvogels van de Europese Vogelrichtlijn

Om te testen of de set van begeleidende soorten inderdaad wijst op een hogere kans op het aantreffen van de doelsoort, gaan we na of er een verschil is in de som van de indicatorwaarde tussen kilometerhokken met en zonder de doelsoort (Tabel 10). Hiervoor gebruiken we de dataset waarmee de correlatieve modellen gekalibreerd werden. We moeten hier opmerken dat een begeleidende soort niet noodzakelijkerwijze in dezelfde biotoop moet voorkomen als de doelsoort: de indicatorwaarde geeft enkel aan in hoeverre de verspreiding van de doelsoort en de begeleidende soorten met elkaar overlappen.

Tabel 10 De som van de indicatorwaarden van de begeleidende soorten in kilometerhokken met (Aanwezig) en zonder (Afwezig) de doelsoort. Tussen haakjes vermelden we het aantal kilometerhokken met en zonder de doelsoort. De significantie geeft aan of het verschil tussen de som in de kilometerhokken met en zonder de doelsoort beduidend is of niet (p-waarde van een ANOVA-test).

Soort	Aanwezig	Afwezig	Significantie
Blauwborst	5.65 ± 0.10 (554)	2.21 ± 0.05 (1570)	<0.001
Boomleeuwerik	4.72 ± 0.05 (319)	1.74 ± 0.04 (1509)	<0.001
Bruine kiekendief	2.05 ± 0.03 (91)	1.43 ± 0.02 (1068)	<0.001
Grauwe klauwier	2.37 ± 0.03 (26)	1.31 ± 0.02 (1215)	<0.001
Ijsvogel	1.98 ± 0.02 (362)	1.35 ± 0.02 (549)	<0.001
Kluut	12.57 ± 0.17 (58)	3.67 ± 0.11 (1969)	<0.001
Kwartelkoning	1.28 ± 0.01 (9)	1.02 ± 0.02 (399)	0.125
Middelste bonte specht	4.96 ± 0.16 (232)	2.45 ± 0.04 (1943)	<0.001
Nachtzwaluw	5.53 ± 0.05 (155)	2.07 ± 0.04 (1187)	<0.001
Porseleinhoen	2.35 ± 0.02 (44)	1.36 ± 0.02 (507)	<0.001
Roerdomp	6.16 ± 0.08 (22)	2.49 ± 0.06 (955)	<0.001
Slechtvalk	1.35 ± 0.02 (19)	1.47 ± 0.02 (1152)	0.604
Visdief	10.22 ± 0.13 (22)	3.19 ± 0.08 (1771)	<0.001
Wespendief	2.42 ± 0.03 (1162)	1.51 ± 0.02 (1199)	<0.001
Woudaap	12.74 ± 0.17 (22)	3.52 ± 0.10 (1413)	<0.001
Zwarte specht	3.69 ± 0.05 (980)	1.70 ± 0.03 (1261)	<0.001
Zwartkopmeeuw	4.89 ± 0.07 (26)	2.14 ± 0.05 (1521)	<0.001

Voor 15 van de 17 soorten broedvogels is de som van de indicatorwaarden van de begeleidende soorten beduidend hoger in kilometerhokken met dan zonder de doelsoort (Tabel 10). Enkel voor Kwartelkoning en Slechtvalk blijkt de set van begeleidende soorten niet goed overeen te stemmen met de actuele verspreiding van de doelsoort.



Blauwborst (foto Glenn Vermeersch)

3.5 Vergelijking tussen de verschillende technieken

Omdat de 4 technieken om te komen tot potentiële leefgebiedenkaarten nog niet op alle soorten toegepast konden worden, kunnen we nog geen uitspraken doen over welke van gebruikte technieken nu de beste resultaten oplevert. Enkel voor de broedvogels werden zowel mechanistische als correlatieve modellen gemaakt en kan de vergelijking tussen beide gemaakt worden.

In dit rapport geven we bovendien telkens het resultaat van de verschillende technieken afzonderlijk, maar in een vervolgproject zal nagegaan moeten worden of het combineren van de toegepaste technieken kan leiden tot meer verfijnde potentiële leefgebiedenkaarten.

Voor de broedvogels werden zowel mechanistische als correlatieve modellen opgemaakt en werden begeleidende soorten gebruikt om de potentiële verspreiding van deze groep te visualiseren. We gebruiken de broedvogels dan ook voor een eerste analyse over welke van de drie technieken het beste scoort. Dit doen we door na te gaan in hoeverre de uitkomst van de mechanistische en de correlatieve modellen en de verspreiding van de begeleidende soorten overlapt met de waarnemingen van de respectievelijke soorten op basis van Anselin et al. (2014). Voor de mechanistische modellen beschouwen we een kilometerhok als geschikt wanneer de minimale oppervlakte voortplantingsgebied in het hok voorkomt volgens het gdx-script (zie Bijlage 1 voor de minimaal noodzakelijke oppervlakte voor de respectievelijke soorten). In Tabel 11 geven we het aantal kilometerhokken waarin een soort waargenomen werd en hoeveel kilometerhokken geschikt zijn volgens de mechanistische en correlatieve modellen en wat de overlap is met de gesommeerde indicatorwaarde van de begeleidende soorten. Vervolgens gaan we na hoeveel van de hokken met waarnemingen van de soort overlappen met de uitkomsten van de mechanistische en correlatieve modellen en met de meest geschikte kilometerhokken volgens de begeleidende soorten.

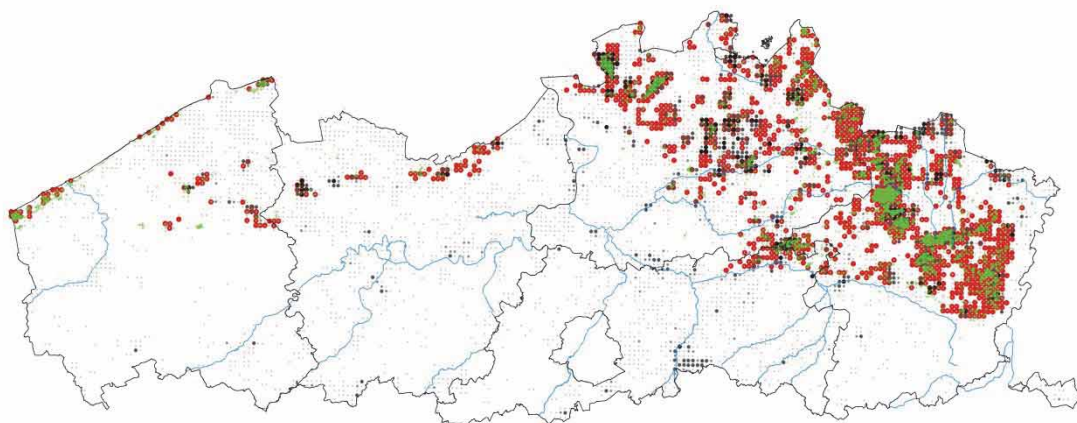
Tabel 11 Aantal kilometerhokken met waarnemingen (N hokken) van de soort op basis van Anselin et al. (2014). MM = aantal kilometerhokken met potentieel geschikt leefgebied groter dan de minimaal noodzakelijke oppervlakte volgens de mechanistische modellen, Overlap WM = overlap tussen kilometerhokken met waarnemingen en de geschikte kilometerhokken volgens het mechanistisch model, CM = aantal kilometerhokken met potentieel geschikt leefgebied volgens correlatieve modellen, Overlap WC = overlap tussen hokken met waarnemingen en de geschikte kilometerhokken volgens het correlatieve model. BS = aantal kilometerhokken met minstens 1/3 van de maximale gesommeerde indicatorwaarde van de begeleidende soorten, Overlap BS = overlap tussen hokken met waarnemingen en de geschikte hokken volgens de begeleidende soorten. Het het best scorende model is telkens in rood weergegeven.

	N hokken	MM	Overlap WM	CM	Overlap WC	BS	Overlap BS
Blauwborst	649	3162	408 (13.9%)	3296	619 (18.8%)	1068	294 (27.5%)
Boomleeuwerik	338	1346	277 (20.6%)	1518	337 (22.2%)	952	246 (25.8%)
Bruine kiekendief	138	1395	110 (7.9%)	1236	138 (11.2%)	1421	61 (4.3%)
Grauwe klauwier	27	261	17 (6.5%)	132	26 (19.7%)	1023	22 (2.2%)
Ijsvogel	525	2073	290 (14.0%)	3262	525 (16.1%)	1020	277 (27.2%)
Kluut	58	367	49 (13.4%)	568	57 (10.0%)	741	40 (5.4%)
Middelste bonte specht	239	492	109 (25.4%)	1275	236 (18.5%)	1846	171 (9.3%)
Nachtzwaluw	163	300	110 (36.7%)	587	161 (27.4%)	826	131 (15.9%)
Porseleinhoen	55	164	32 (19.5%)	369	55 (14.9%)	665	39 (5.9%)
Roerdomp	24	71	5 (7.0%)	142	24 (16.9%)	577	16 (2.8%)
Visdief	22	5	0 (0%)	210	7 (3.3%)	1002	16 (1.6%)
Wespendief	1541	353	104 (29.5%)	5301	1540 (29.1%)	2227	791 (35.5%)
Woudaap	23	61	18 (29.5%)	92	23 (25.0%)	564	20 (3.5%)
Zwarte specht	1102	880	368 (41.8%)	3850	1101 (28.6%)	1566	689 (44.0%)
Zwartkopmeeuw	32	41	12 (29.3%)	524	32 (6.1%)	1139	18 (1.6%)
Gemiddelde		731	19.7%	1491	17.9%	1109	14.2%

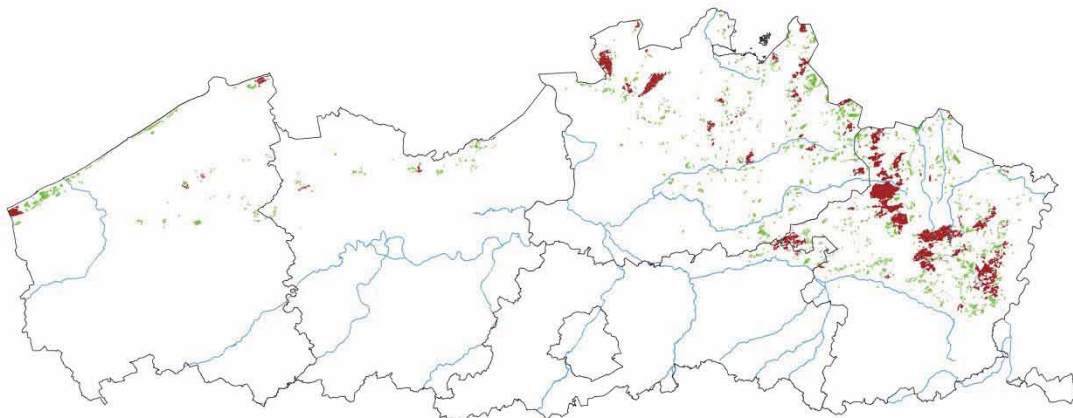
Uit Tabel 11 blijkt dat gemiddeld genomen de mechanistische modellen beter overlappen met de waarnemingen dan de correlatieve modellen en dan de verspreiding van de begeleidende soorten. Wanneer we kijken naar het aantal soorten dat beter scoort met de ene of de ander techniek dan zien we dat 6 soorten het best scoren met de mechanistische (Kluut, Middelste bonte specht, Nachtzwaluw, Porseleinhoen, Woudaap en Zwartkopmeeuw), 4 met de correlatieve modellen (Bruine kiekendief, Grauwe klauwier, Roerdomp en Visdief) en 5 met de begeleidende soorten (Blauwborst, Boomleeuwerik, IJsvogel, Wespandief en Zwarte specht). We moeten hier echter wel vermelden dat het totaal aantal kilometerhokken dat overlapt met de potentiële leefgebieden uit de mechanistische modellen veel hoger ligt wanneer het criterium dat er een minimale oppervlakte van het voortplantingsbiotoop in moet voorkomen, achterwege gelaten wordt. Voor Boomleeuwerik bijvoorbeeld overlappen er in totaal 1954 kilometerhokken met de potentieel geschikte leefgebieden volgens de mechanistische modellen. De oppervlakte potentieel geschikt leefgebied in deze kilometerhokken varieert van 100 ha (dus het hele kilometerhok is geschikt) tot minder dan 1 m². Wanneer we als voorwaarde stellen dat er minstens 5 ha potentieel geschikt leefgebied aanwezig moet zijn in het kilometerhok (cf. minimale oppervlakte in het gdx-script voor de mechanistische modellen) dan blijven er nog 1346 “geschikte” kilometerhokken over volgens de mechanistische modellen.

3.6 Gezamenlijk gebruik van de verschillende technieken

Als we de resultaten van de 3 methoden over elkaar plotten, krijgen we een kaart zoals in Figuur 10 (bij wijze van voorbeeld gebruiken we hier Boomleeuwerik). Vertrekkende van de kaart die we verkregen met behulp van de meest gedetailleerde kaart (mechanistische model), kunnen we op zoek gaan naar potentiële leefgebieden die ook met de correlatieve modellen (de wordt als aanwezig voorspeld) en de begeleidende soorten voldoende indicatief zijn voor de soort (de indicatorwaardesom van de begeleidende soorten bedraagt minstens 33% van de maximale indicatorsomwaarde in het kilometerhok). Hiermee kan de oorspronkelijke potentiële leefgebiedenkaart voor Boomleeuwerik verfijnd worden tot de gebieden waarvoor meerdere methoden aangeven dat ze geschikt leefgebied voor deze soort kunnen herbergen (Figuur 11). Op deze manier wordt het potentieel leefgebied van Boomleeuwerik verkleind van 33.573 hectare tot 18.748 hectare, een reductie van 45%.



Figuur 10 Potentiële leefgebiedenkaart voor Boomleeuwerik op basis van de mechanistische modellen (groen) met een overlay van de resultaten van de correlatieve modellen (rode bollen) en de begeleidende soorten (zwart-grijze bollen).



Figuur 11 Verfijnde potentiële leefgebiedenkaart voor Boomleeuwerik (donkerrood). Van de potentiële leefgebiedenkaart bekomen met het mechanistisch model (lichtgroen) werden enkel die clusters overgehouden waarvoor het correlatieve model de soort als aanwezig voorspelde en de waar de indicatorwaardesom van de begeleidende soorten minstens 33% van de maximale indicatorsomwaarde in het kilometerhok bedroeg.



Boomleeuwerik (foto Glenn Vermeersch)

4 Conclusies en aanbevelingen

De hier voorgestelde technieken voor het verfijnen van de potentiële leefgebiedenkaarten zijn bruikbare tools in het wetenschappelijk onderbouwen van verschillende toepassingen in het Vlaamse natuurbehoud.

De oorspronkelijke toepassing van de leefgebiedenbenadering was om ze te gebruiken in het on-line instrument **voortoets** van het Agentschap voor Natuur en Bos. Wanneer een ingreep of activiteit potentieel negatieve effecten zou kunnen hebben op het (potentiële) leefgebied van een Vlaamse of Europese prioritaire soort, kan aan de initiatiefnemer gevraagd worden om een passende beoordeling te maken om enerzijds aan te tonen dat er geen negatieve effecten zijn of anderzijds aan te geven hoe ze maximaal gemilderd kunnen worden.

Een tweede toepassing van deze benadering was het zogenaamde **zoekzonemodel**. Voor soorten waarvoor Vlaanderen de regionale instandhoudingsdoelstellingen momenteel niet haalt en waarvoor er bijkomende doelen gesteld werden – en dus bijkomende gebieden afgebakend moeten worden – kunnen de potentiële leefgebiedenkaarten gebruikt worden om de meest optimale locaties voor deze extra af te bakenen gebieden aan te duiden.

Naast beide oorspronkelijke toepassingsmogelijkheden zijn er echter nog bijkomende mogelijke gebruiken van de leefgebiedenbenadering: het opmaken van zoekkaarten voor soorten waarvan de verspreiding momenteel slecht gekend is, het afbakenen van gebieden waarin soortbeschermingsprogramma's voor soorten uitgevoerd kunnen worden en het opsporen van locaties voor een eventuele herintroductie van in Vlaanderen uitgestorven soorten.

Met behulp van de leefgebiedenbenadering kunnen **zoekkaarten** aangemaakt worden voor soorten waarvan de verspreiding momenteel slecht gekend is. Deze zoekkaarten zullen gebruikt worden bij de inhaalslagen van de te monitoren Vlaamse en Europese prioritaire soorten (bv. Gerande oeverspin – Van Keer et al. 2015; Europese otter – De Bruyn et al. 2015).

Elk van de technieken heeft echter zijn beperkingen en zou, mits verder onderzoek, nog verbeterd kunnen worden. Hieronder sommen we de voor- en nadelen van elk van de technieken op en wat er bijkomend nodig is om de techniek te verbeteren. Naast Vlaamse en Europese prioritaire soorten, kunnen deze benaderingen uiteraard ook gebruikt worden voor andere relevante soorten in het natuurbehoud in Vlaanderen (bv. habitattypische soorten – De Knijf et al. 2013; habitatrichtlijnsoorten die niet in de lijst van soorten in dit rapport opgenomen werden bv. Juchtleerkever – Thomaes et al. 2015).

4.1 Mechanistische modellen

Sterkten

1. De resolutie van de aangemaakte kaarten is zeer hoog (20 x 20 meter).
2. Er zijn, strikt genomen, geen verspreidingsgegevens nodig om mechanistische modellen te maken, al kan in de praktijk de gekende verspreiding uiteraard wel gebruikt worden om na te gaan of de aangemaakte kaarten realistisch zijn en kan de potentiële leefgebiedkaart a.h.v. deze gegevens verder verfijnd worden.
3. Door het feit dat er geen verspreidingsgegevens nodig zijn, kan deze techniek toegepast worden op (zeer) zeldzame soorten.

Beperkingen

1. Er is veel (kwantificeerbare) soortenkennis nodig om het ecologische profiel van elke soort op te stellen.
2. Er zijn vaak veel en goede basiskaarten nodig om de ecologie van de soort te vertalen naar potentiële leefgebieden op het terrein. De basislaag die gebruikt wordt voor de mechanistische modellen (BWK) is niet overal even actueel waardoor er soms met gedateerde kaarten gemodelleerd moet worden. Voor sommige soorten(groepen) zijn de kaartlagen met kwantitatieve gegevens onvoldoende gedetailleerd (bv. kleine landschapselementen), onvolledig (bv. bodem) of zelfs helemaal niet beschikbaar (bv. structuurkaarten van waterlopen).

3. Van sommige soorten is de ecologie onvoldoende gekend om een bruikbaar ecologisch profiel op te stellen (bv. vleermuizen).
4. Doordat er in deze benadering geen rekening wordt gehouden met de gekende verspreiding van de soort, geven de aangemaakte kaarten vaak een overschatting van de werkelijke verspreiding.
5. Er kan geen kwaliteitsinschatting gemaakt worden van de leefgebieden omdat de uitkomst enkel zegt of een configuratie van 20 x 20 meter hokken geschikt of ongeschikt is voor de soort.
6. Hoe langer en complexer het script wordt, hoe moeilijker het wordt om de oorzaak van ongewenste uitkomsten te vinden: het maken van scripts kan daarom zeer arbeidsintensief zijn.

Verbeterpunten

1. Het aanmaken, aanvullen en/of verfijnen van ontbrekende kaartlagen (waterlopen, kleine landschapselementen, bodem, ...).
2. Verwerven van betere ecologische kennis over sommige soortengroepen (bv. vleermuizen, insecten, ...).
3. Nagaan of gdx toolbox ook kan gebruikt worden voor een kwaliteitsinschatting (zeer geschikt, matig geschikt, weinig geschikt, ongeschikt).

4.2 Correlatieve modellen

Sterkten

1. Strikt genomen is er 'geen' ecologische kennis nodig om correlatieve modellen te maken. In de praktijk zal dat echter wel nodig zijn om te bepalen welke variabelen in het model opgenomen zullen worden.
2. Aangezien de uitkomst van een correlatief model een waarde is tussen 0 en 1 (hoe dichter bij 1, hoe hoger de kans dat de soort er voorkomt), kan deze waarde gebruikt worden als een soort kwaliteitsmaat voor het betrokken kilometerhok.

Beperkingen

1. Ten opzichte van de mechanistische modellen is de resolutie lager (1 x 1 km²)
2. Er zijn voldoende verspreidingsgegevens nodig om correlatieve modellen te calibreren en de modellen zijn, uit statistische overwegingen, enkel zinvol voor soorten die in minstens 25 kilometerhokken voorkomen.
3. Correlatieve modellen werken met zowel aan- als afwezigheden. Een aanwezigheid is duidelijk vast te stellen, maar het is ook noodzakelijk om zo zeker mogelijk te zijn over de afwezigheden. Dit houdt in dat de groep als geheel vrij goed geïnventariseerd moet zijn, zodat ervan uit gegaan kan worden dat wanneer de soort niet werd gezien in een hok, ze ook echt afwezig is ("true absence").
4. Voor sommige soorten ontbreken kaarten met relevante omgevingsvariabelen (bv. dood hout, waterlopen, ...).

Verbeterpunten

1. Ecologische kennis verbeteren en/of verspreidingsgegevens aanvullen voor sommige soortengroepen (bv. kleine zoogdieren, vleermuizen, insecten en andere ongewervelden).
2. Kaartlagen van relevante omgevingsvariabelen maken en/of verbeteren.

4.3 Downscaling

Sterkten

1. Aangezien een kilometerhok een vrij ruwe eenheid is in het natuurbehoud, zou het downscalen van modellen kunnen resulteren in een realistischere (kleinere) spatiale weergave van een potentieel leefgebied.

Beperkingen

1. Het downscalen van de correlatieve modellen vraagt bijzonder veel computertijd en de resulterende shape-files kaarten nemen veel schijfruimte in beslag. Met een oppervlakte van 13.682 km², moet er immers voor meer 1,3 miljoen hectarehokken in Vlaanderen een kans op de aanwezigheid van elke soort worden berekend en dat gaat boven de limiet van het software package R. Hierdoor moet de predictie in verschillende modelruns gebeuren.
2. Bij het downscalen van modellen stelt zich vooral een probleem van schaalverschillen. Indien een kilometerhok bijvoorbeeld volledig bestaat uit bos, dan beslaat dat een oppervlakte van 100 hectare; wanneer dergelijke modellen geprojecteerd worden naar hectareniveau, kan dat getal nooit voorkomen aangezien dat maximaal 1 hectare kan zijn. In de downscaling-oefening die we hier gedaan hebben, maakten we gebruik van proporties: als een kilometerhok voor 10% uit bos bestaat, dan werd bij de predictie naar hectareniveau ook 10% gebruikt. Strikt genomen is deze benadering niet correct, aangezien soorten niet rekenen in percentages maar in absolute getallen: 10% van een hok van een kilometerhok = 10 hectare, terwijl 10% van een hectarehok slechts 0,1 hectare is.
3. Het gebruik van kilometerhokken weerspiegelt niet alleen welke variabelen voor de verschillende soorten belangrijk zijn, maar zegt ook iets over hoe meerdere van deze variabelen zich tot elkaar verhouden (landschapschaal). Op hectareniveau zal een hok al sneller bestaan uit slechts een of enkele variabelen en krijgt men geen “beeld” van het belang van het omliggende landschap.
4. Voor sommige omgevingsvariabelen zijn er geen voldoende gedetailleerde kaarten beschikbaar om ze te verrasteren op een resolutie van 100 x 100 m² (bv. bodem). Anderzijds zijn sommige variabelen weinig onderscheidend op hectareniveau: bv. klimaat- of bodemvariabelen in nabijgelegen hectarehokken zullen heel vaak dezelfde waarden hebben, aangezien deze variabelen vooral variatie vertonen op grotere schaal.

Verbeterpunten

1. Nagaan voor welke soorten downscaling haalbaar en relevant is.
2. Vooral het schaalverschil tussen de hier gebruikte resoluties zorgt bij het downscalen voor een ecologisch probleem. Er zal dus onderzocht moeten worden bij andere soorten in hoeverre het gebruik van proporties toelaatbaar is en niet tot incorrecte afbakeningen leidt.
3. Er moet bekeken worden of het calibreren van modellen op hectareniveau computerteknisch haalbaar is (zeer groot aantal hectarehokken, zeer veel schijfruimte nodig). Dit zou getoetst kunnen worden voor een soort waarvoor voldoende en zeer gedetailleerde verspreidingsgegevens nodig.

4.4 Begeleidende soorten

Sterkten

1. Soorten uit andere en/of betere onderzochte taxonomische groepen kunnen gebruikt worden om potentieel geschikte gebieden voor een doelsoort aan te duiden (Maes et al. 2009).
2. Als er voldoende verspreidingsgegevens beschikbaar zijn, kunnen de begeleidende soorten correlatieve gemodelleerd worden, die op hun beurt gesommeerd kunnen worden en zo een 'kans' berekenen op de aanwezigheid van de doelsoort.
3. Door meerdere soorten te gebruiken, geeft deze benadering een maat voor de kwaliteit van een leefgebied.

Beperkingen

1. Vergt zeer veel tijd en analysewerk om te bepalen wanneer een soort als begeleidend kan worden beschouwd.
2. Enkel haalbaar indien er voldoende gedetailleerde verspreidingsgegevens voorhanden zijn van zowel de doel- als de begeleidende soorten.

Verbeterpunten

1. Nagaan voor welke soorten het haalbaar en relevant is om begeleidende soorten aan te duiden.

4.5 Terminologie

Tijdens overlegmomenten of discussies over potentiële leefgebiedenkaarten en de mogelijke toepassingen ervan is het bijzonder belangrijk om de verschillende termen die gebruikt werden in dit rapport volledig te begrijpen en correct te gebruiken. We zetten de betekenissen van enkele termen hier nog even op een rijtje:

4.5.1 Potentieel leefgebied

Het potentieel leefgebied bestaat uit de biotopen of de cluster van biotopen waarin een soort alle ecologische hulpbronnen aantreft om zijn levenscyclus te volbrengen. Hier wordt geen rekening gehouden met de actuele verspreiding van de soort, enkel met de potentie van biotopen om de soort te herbergen. Door allerlei andere factoren (bv. versnippering, jaarlijkse populatieschommelingen ...) kan een soort niet voorkomen of (nog) niet waargenomen zijn in het afgebakende potentieel leefgebied op basis van de leefgebiedenbenadering. Het ontbreken van waarnemingen van een bepaalde soort in een potentieel leefgebied, betekent echter niet dat de soort er ook effectief ontbreekt. Ze kan immers moeilijk te detecteren zijn en dus over het hoofd gezien zijn of de locatie is niet onderzocht op de aanwezigheid van de soort. Een potentieel leefgebied geeft enkel aan dat, op basis van het ecologisch profiel van een soort, in de afgebakende biotopen of cluster van biotopen alle hulpbronnen aanwezig zijn voor de soort. Door het werken met potenties van biotopen wordt met behulp van de leefgebiedenbenadering doorgaans een grotere oppervlakte afgebakend dan de werkelijke verspreiding van de soort.

4.5.2 Actueel gebruikt leefgebied

Het actueel gebruikt leefgebied van een soort bestaat uit de biotopen of een cluster van biotopen waarvan geweten is dat de soort er zich momenteel voortplant of er zich recent voortgeplant heeft. Doordat niet alle gekende populaties omschreven kunnen worden in een ecologisch profiel (uitzonderlijke broedgevallen in bepaalde jaren, occasionele voortplanting in niet-typische biotopen) of niet alle kaartlagen voldoende gedetailleerd of voldoende recent zijn, kan het gebeuren dat niet alle actueel voorkomende populaties aangeduid worden als potentieel leefgebied.

4.5.3 Actueel gekende verspreiding

De actueel gekende verspreiding van een soort zijn de locaties waar waarnemingen van de soort gemeld werden die duiden op voortplanting, bijvoorbeeld in www.waarnemingen.be of andere dataportalen. De actueel gekende verspreiding komt niet noodzakelijkerwijze overeen met de werkelijke verspreiding van de soort (zie verder).

4.5.4 Werkelijke verspreiding

De werkelijke verspreiding van soorten is slechts in uitzonderlijke gevallen volledig gekend. Soorten kunnen zich namelijk ongemerkt voortplanten in moeilijk toegankelijke gebieden (militaire domeinen, grote natuurgebieden ...), het detecteren/determineren van sommige soorten kan moeilijk zijn of niet alle waarnemingen zijn toegankelijk in de dataportalen (ze kunnen onder embargo van de waarnemer staan, vertrouwelijke data in militaire domeinen ...) . Hierdoor is het niet altijd mogelijk om een volledig beeld van de verspreiding te bekomen. Waarnemingen van soorten geven daarom meestal slecht een deel van de werkelijke verspreiding weer.

Referenties

- Adriaens D. & Lommaert L. (eds.) 2015. Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0, Brussel: Instituut voor Natuur- en Bosonderzoek.
- Adriaens P. & Vermeersch G. in druk. Criteria voor de beoordeling van de lokale staat van instandhouding van de Vogelrichtlijnsoorten, versie 2.0. Rapporten van het Instituut voor Natuur- en Bosonderzoek. Instituut voor Natuur- en Bosonderzoek, Brussel.
- Andeweg R.W.G. 2011. Monitoring Groenknolorchis 2011. Havengebied Rotterdam. Rotterdam.
- Anselin A., Devos K., Vermeersch G., Stienen E.W.M. & Onkelinx T. 2014. Toelichting bij het opstellen van de rapportage in het kader van artikel 12 van de Vogelrichtlijn en status van vogelsoorten met instandhoudings-populatiedoelen en van typische soorten van de Natura 2000 habitattypes. Rapporten van het Instituut voor Natuur- en Bosonderzoek INBO.R.2014.1567208. Instituut voor Natuur- en Bosonderzoek, Brussel.
- Araújo M.B. & New M. 2007. Ensemble forecasting of species distributions. *Trends in Ecology & Evolution* 22: 42-47.
- Araújo M.B., Thuiller W., Williams P.H. & Reginster I. 2005. Downscaling European species atlas distributions to a finer resolution: implications for conservation planning. *Global Ecology and Biogeography* 14: 17-30.
- Breiman L. 2001. Random forests. *Machine Learning* 45: 5-32.
- De Bruyn L., Verbeylen G., Scheppers T., Van Den Berge K., Maes D., De Knijf G., Onkelinx T., Piesschaert F., Pollet M., Van Calster H., Westra T. & Quataert P. 2015. Monitoringsprotocollen zoogdieren: hamster, hazelmuus, das, otter. Rapporten van het Instituut voor Natuur- en Bosonderzoek INBO.R.2015.11336560. Instituut voor Natuur- en Bosonderzoek, Brussel.
- De Cáceres M. & Legendre P. 2009. Associations between species and groups of sites: indices and statistical inference. *Ecology* 90: 3566-3574.
- De Knijf G., Adriaens D., Van Elegem B. & Paelinckx D. 2013. Natura 2000 habitattypen - Meer dan flora! Selectiecriteria en gebruik van typische faunasoorten bij de gewestelijke beoordeling van de staat van instandhouding. *Natuur.focus* 12: 109-120.
- De Knijf G. & Adriaens T. 2015. Geleedpotigen - Libellen. In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 68-81
- De Saeger S., Guelinckx R., Van Dam G., Oosterlynck P., Van Hove M., Wils C. & Paelinckx D. 2014. Biologische Waarderingskaart en Natura 2000 Habitatkaart, uitgave 2014. Rapporten van het Instituut voor Natuur- en Bosonderzoek INBO.R.2014.1698392. Instituut voor Natuur- en Bosonderzoek, Brussel.
- Denys L., Packet J., Leyssen A. & Van Landuyt W. 2015. Drijvende waterweegbree (*Luronium natans*). In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 31-35
- Devos K. & Onkelinx T. 2013. Overwinterende watervogels in Vlaanderen. Populatieschattingen en trends (1992 tot 2013). *Natuur.oriolus* 79: 113-130.
- Dufrêne M. & Legendre P. 1997. Species assemblages and indicator species: the need for a flexible asymmetrical approach. *Ecological Monographs* 67: 345-366.
- Friedman J., Hastie T. & Tibshirani R. 2000. Additive logistic regression: A statistical view of boosting. *Annals of Statistics* 28: 337-374.

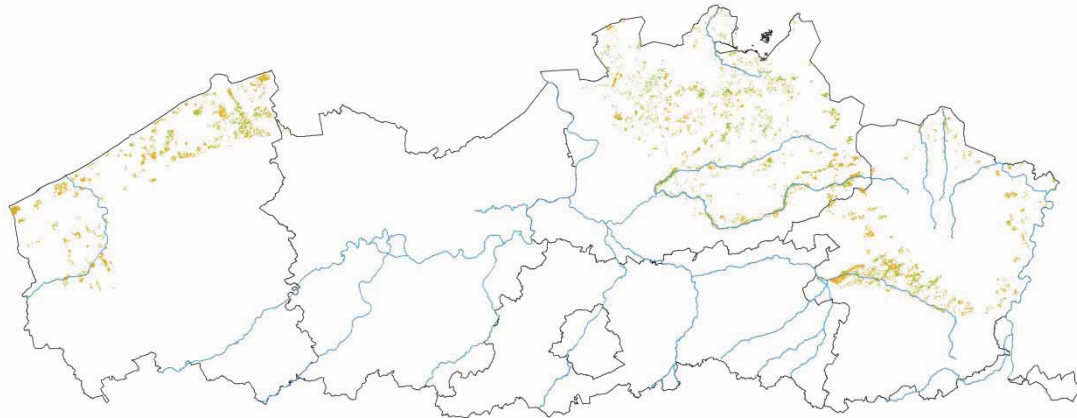
- Hastie T. & Tibshirani R. 1987. Generalized additive models: some applications. *Journal of the American Statistical Association* 82: 371-386.
- Hill M. 1979. TWINSpan-FORTRAN Program for arranging multivariate data in an ordered two-way table by classification of individuals and attributes. Cornell University Ithaca, New York.
- Louette G., Adriaens D., Adriaens P., Anselin A., Devos K., Sannen K., Van Landuyt W., Paelinckx D. & Hoffmann M. 2011. Bridging the gap between the Natura 2000 regional conservation status and local conservation objectives. *Journal for Nature Conservation* 19: 224-235.
- Louette G., Adriaens D., Paelinckx D. & Hoffmann M. 2015. Implementing the Habitats Directive: How science can support decision making. *Journal for Nature Conservation* 23: 27-34.
- Maes D. 2015. Spaanse vlag (*Euplagia quadripunctaria*). In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 87-91
- Maes D., Anselin A., van Swaay C.A.M., Sierdsema H., De Bruyn L., Vermeersch G. & Van Dyck H. 2009. Can we predict the distribution of heathland butterflies with heathland bird data? *Animal Biology* 59: 335-349.
- Maes D. & Van Dyck H. 2005a. Doorbreek de taxonomische hokjesmentaliteit! Pleidooi voor een multisoortenaanpak in het Vlaamse natuurbehoud. *Natuur.focus* 4: 24-26.
- Maes D. & Van Dyck H. 2005b. Habitat quality and biodiversity indicator performances of a threatened butterfly versus a multispecies group for wet heathlands in Belgium. *Biological Conservation* 123: 177-187.
- Maes D., Vanreusel W. & Van Dyck H. 2007. Vlinderindicatoren. Een handige hulp bij het inventariseren. *Natuur.focus* 6: 60-64.
- Maes D., Wils C., Anselin A., Belpaire C., Casaer J., De Knijf G., Devos K., Dhondt P.-J., Gyselings R., Packet J., Speybroeck J., Stienen E.W.M., Stuyck J., Thomaes A., T'jollyn F., Van Den Berge K., Van Landuyt W., Van Thuyne G., Vermeersch G., Verreycken H. & Pollet M. 2015. Afbakenen van potentiële leefgebieden voor Europese en Vlaamse prioritaire soorten in het kader van de voortoets: versie 1.0. Rapporten van het Instituut voor Natuur- en Bosonderzoek INBO.R.2015.6960681. Instituut voor Natuur- en Bosonderzoek, Brussel.
- McCullagh P. & Nelder J.A. 1989. Generalized linear models. 2nd edition. Chapman & Hall, London.
- Packet J. 2015a. Platte schijfhoren (*Anisus vorticulus*). In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 124-127
- Packet J. 2015b. Zeggekorfslak (*Vertigo moulinsiana*). In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 128-131
- Packet J. & Provoost S. 2015. Nauwe korfslak (*Vertigo angustior*). In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 132-135
- Phillips S.J., Anderson R.P. & Schapire R.E. 2006. Maximum entropy modeling of species geographic distributions. *Ecological Modelling* 190: 231-259.
- Provoost S., Van Gompel W., Feys S., Vercruysse W., Packet J., Van Lierop F., Adams Y. & Denys L. 2010. Permanente Inventarisatie van de Natuurreservaten aan de Kust. Eindrapport periode 2007-2010. Rapporten van het Instituut voor Natuur- en Bosonderzoek INBO.R.2010.19. Instituut voor Natuur- en Bosonderzoek, Brussel.
- Provoost S., Van Gompel W., Vercruysse W., Packet J. & Denys L. 2015. Permanente Inventarisatie van de Natuurreservaten aan de Kust, PINK II: eindrapport periode 2012-2014. Rapporten van het Instituut voor Natuur- en Bosonderzoek INBO.R.2015.8890955. Instituut voor Natuur- en Bosonderzoek, Brussel.
- R Core Team 2015. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.
- Schaminée J.H.J., Weeda E.J. & Westhoff V. 1995. De vegetatie van Nederland. Deel 2. Plantengemeenschappen van wateren, moerassen en natte heiden. Opulus Press, Uppsala/Leiden.
- Spanoghe G., Van Landuyt W. & Gyselings R. 2008. Nieuwe vindplaats van Groenknolorchis (*Liparis loeselii*) in de Waaslandhaven. *Dumortiera* 95: 1-3.
- Spellerberg I.F. & Fedor P.J. 2003. A tribute to Claude Shannon (1916-2001) and a plea for more rigorous use of species richness, species diversity and the 'Shannon-Wiener' Index. *Global Ecology and Biogeography* 12: 177-179.
- Speybroeck J. 2015. Amfibieën en reptilen. In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 36-67
- Stevens M., Demolder H., Jacobs S., Michels H., Schneiders A., Simoens I., Spanhove T., Van Gossum P., Van Reeth W. & Peymen J. 2014. Natuurrapport - Toestand en trend van ecosystemen en ecosysteemdiensten in Vlaanderen. Syntheserapport. Mededelingen van het Instituut voor Natuur- en Bosonderzoek INBO.M. 2014.1988666. Instituut voor Natuur- en Bosonderzoek, Brussel.
- Swets J.A. 1988. Measuring the accuracy of diagnostic systems. *Science* 204: 1285-1293.

- Thomaes A. 2015. Vliegend hert (*Lucanus cervus*). In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 82-86
- Thomaes A., Drumont A., Crevecoeur L. & Maes D. 2015. Rode lijst van de saproxyle bladsprietkevers (Lucanidae, Cetoniidae en Dynastidae) in Vlaanderen. Rapporten van het Instituut voor Natuur- en Bosonderzoek, INBO.R.2015.7843021. Instituut voor Natuur- en Bosonderzoek, Brussel.
- Thomaes A., Kervyn T. & Maes D. 2008. Applying species distribution modelling for the conservation of the threatened saproxylic Stag Beetle (*Lucanus cervus*). Biological Conservation 141: 1400-1410.
- Thomaes A. & Maes D. 2014. Rode-Lijststatus van het Vliegend hert (*Lucanus cervus*). Rapporten van het Instituut voor Natuur- en Bosonderzoek INBO.R.2014.1549345. Instituut voor Natuur- en Bosonderzoek, Geraardsbergen.
- Thuiller W., Georges D. & Engler R. 2012. biomod2: Ensemble platform for species distribution modeling. R package version 1.3.7/r529.
- Valentin B., Toussaint B., Duhamel F. & Valet J.-M. 2010. Plan national d'actions en faveur du Liparis de Loesel (*Liparis loeselii*) 2010-2014. Conservatoire botanique national de Bailleul - Ministère de l'Ecologie, de l'Energie, du Développement durable et de la Mer, Bailleul.
- Van Den Berge K., Adriaens D., Hens M., Verbist V. & Holsbeek L. 2015. Europese hamster (*Cricetus cricetus*). In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 270-274
- Van Den Berge K. & Gouwy J. 2015. Otter (*Lutra lutra*). In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 263-269
- Van Keer K., De Knijf G., Lambeets K., Maes D., De Bruyn L., Onkelinx T., Piesschaert F., Pollet M., Westra T. & Quataert P. 2015. Monitoringsprotocol spinnen. Rapporten van het Instituut voor Natuur- en Bosonderzoek INBO.R.2015.10069665. Instituut voor Natuur- en Bosonderzoek, Brussel.
- Van Landuyt W. 2011. Een gecentraliseerde databank voor de bryologie in Vlaanderen: na 30 jaar eindelijk uit de startblokken? Muscillanea 30: 4-16.
- Van Landuyt W. 2015a. Geel schorpioenmos (*Hamatocaulis/Drepanocladus vernicosus*). In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 22-24
- Van Landuyt W. 2015b. Groenknolorchis (*Liparis loeselii*). In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 28-30
- Van Landuyt W. 2015c. Kruipend moerasscherm (*Apium repens*). In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 25-27
- Van Landuyt W., Gyselings R., T'jollyn F. & Vanden Broeck A. 2014. Groenknolorchis (*Liparis loeselii*) in Vlaanderen: ecologie, populatiedynamica en potenties. Rapporten van het Instituut voor Natuur- en Bosonderzoek INBO.R.2014.2942320. Instituut voor Natuur- en Bosonderzoek, Brussel.
- Van Landuyt W., Vanhecke L. & Brosens D. 2012. Florabank1: a grid-based database on vascular plant distribution in the northern part of Belgium (Flanders and the Brussels Capital region). PhytoKeys 12: 59-67.
- Vanden Berghen C. 1943. La végétation de sables d'Anvers-rive gauche. Les Naturalistes Belges 24: 135-139.
- Verbeylen G., Nijs G., Verbelen D. & Van Den Berge K. 2015. Hazelmuis (*Musccardinus avellanarius*). In: Adriaens D. & Lommaert L. (eds.) Criteria voor de beoordeling van de lokale staat van instandhouding van de Habitatrichtlijnsoorten in Vlaanderen, versie 2.0. Instituut voor Natuur- en Bosonderzoek, Brussel. pp. 275-279
- Vriens L., Bosch H., De Knijf G., De Saeger S., Oosterlynck P., Guelinckx R., T'jollyn F., Van Hove M. & Paelinckx D. 2011. De Biologische Waarderingskaart. Biotopen en hun verspreiding in Vlaanderen en het Brussels Hoofdstedelijk Gewest. Mededelingen van het Instituut voor Natuur- en Bosonderzoek INBO.M.2011.1. Instituut voor Natuur- en Bosonderzoek, Brussel.

Bijlage 1 Voor de verschillende soorten vermelden we de bronnen waarop de ecologische profielen gebaseerd werden, we geven de uitgeschreven gdx-scripts en de daarmee gegenereerde potentiële leefgebiedenkaarten. Voor de soorten waarvoor we ook correlatieve modellen gemaakt hebben, geven we eveneens de resulterende kaart. Voor de broedvogels van de Europese Vogelrichtlijn en het Vliegend hert geven we eveneens de actuele waarnemingen. De potentiële leefgebiedenkaarten kunnen als webservice geraadpleegd worden op <https://data.inbo.be/potleefgebieden/>.

AMFIBIEËN

Boomkikker (*Hyla arborea*) – Speybroeck (2015)



Potentiële leefgebiedenkaart voor Boomkikker op basis van het onderstaande gdx-script.

Groen = waterbiotoop, oranje = landbiotoop.

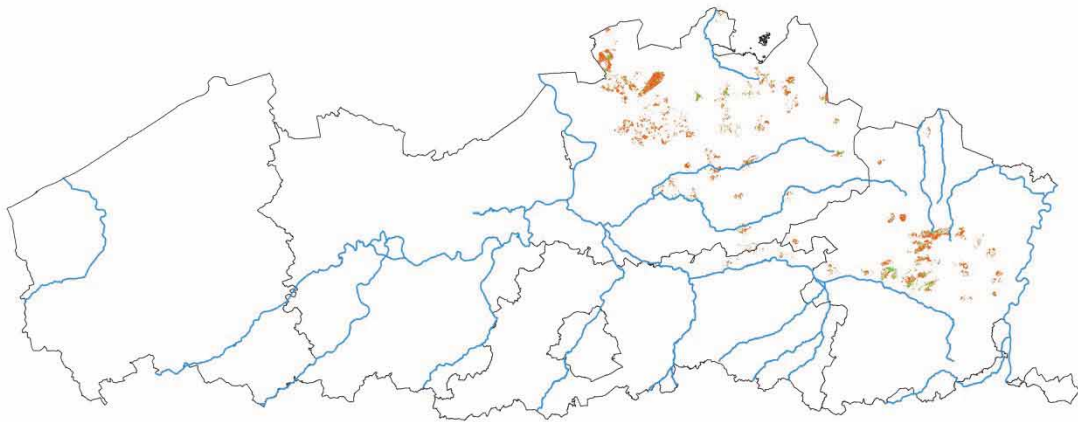
Gdx-script

```
vlaanderen_20m.asc
// baken geschikte waterbiotopen af
assign(boomkikker_waterbiotoop_bwk, bwk(ae%, ao%, kn%, mp%, 3130%, 3150%, 2190%))
// uit GRB-watergang, kleine waters proberen te filteren
assign(GRB_water, Eq('GRB_watergang_at.asc', 1))
assign(GRB_water_cluster, fuzzyClusterIdU(GRB_water, 10, meter))
assign(boomkikker_kleinwater, cFilterLeArea(GRB_water_cluster, 0.4, hectare))
deleteVariable(GRB_water)
deleteVariable(GRB_water_cluster)
assign(boomkikker_waterbiotoop1, or(boomkikker_waterbiotoop_bwk, boomkikker_kleinwater))
deleteVariable(boomkikker_kleinwater)
deleteVariable(boomkikker_waterbiotoop_bwk)
// binnen een straal van 500m moeten minstens 5 (6-1 dwz 6 - de eigen poel) andere geschikte
// voortplantingsgebieden liggen
assign(boomkikker_waterbiotoop1_cluster, clusterId(boomkikker_waterbiotoop1))
deleteVariable(boomkikker_waterbiotoop1)
assign(boomkikker_waterbiotoop2, GeUniqueCountInBuffer(boomkikker_waterbiotoop1_cluster, 500, meter,
boomkikker_waterbiotoop1_cluster, 6))
deleteVariable(boomkikker_waterbiotoop1_cluster)
// landbiotoop (inclusief overwinteringsbiotoop) moet aanwezig zijn binnen de 500m
// baken landbiotopen af
assign(boomkikker_landbiotoop_hms, bwk(hc%, hd%, hf%, hj%, mp%, mr%, sd%, sf%, sp%, sz%, rbbhc%, 2130%,
6430%, rbbhf%, 2190%, rbbmr%, 2160%, 2170%, rbbasp%))
assign(boomkikker_landbiotoop_k, bwk(k(ae%, k(ao%, k(hc%, k(hd%, k(hf%, k(hj%, k(ku%, k(mr%, kt(hc%, kt(hd%,
kt(hf%, kt(hj%, kt(ku%, kt(mr%, kb%, kh%, khw%, kn%, mp%, mr%, kb%, kh%, khw%, kn%, ku%, 3150%, 3130%,
2190%, rbbmr%, 6430%))
// landbiotoop moet gevarieerd zijn en verschillende biotooptypes bevatten
assign(boomkikker_landbiotoop_bwk1, and(boomkikker_landbiotoop_hms,
cellDistanceLe(boomkikker_landbiotoop_k, 500, meter)))
assign(boomkikker_landbiotoop_bwk2, and(boomkikker_landbiotoop_k,
cellDistanceLe(boomkikker_landbiotoop_hms, 500, meter)))
assign(boomkikker_landbiotoop_bwk3, or(boomkikker_landbiotoop_bwk1, boomkikker_landbiotoop_bwk2))
```

```

deleteVariable(boomkikker_landbiotoop_bwk1)
deleteVariable(boomkikker_landbiotoop_bwk2)
deleteVariable(boomkikker_landbiotoop_k)
deleteVariable(boomkikker_landbiotoop_hms)
// landbiotoop mag geen akker zijn
assign(boomkikker_akkers, bwk(b%))
assign(boomkikker_landbiotoop_na, and(boomkikker_landbiotoop_bwk3, not(boomkikker_akkers)))
deleteVariable(boomkikker_akkers)
deleteVariable(boomkikker_landbiotoop_bwk3)
// landbiotoop moet aanwezig zijn binnen de 500m van waterbiotoop
assign(boomkikker_landbiotoop1, and(boomkikker_landbiotoop_na, cellDistanceLe(boomkikker_waterbiotoop2,
500, meter)))
// waterbiotoop moet aanwezig zijn binnen de 500m van landbiotoop
assign(boomkikker_waterbiotoop3, and(boomkikker_waterbiotoop2, cellDistanceLe(boomkikker_landbiotoop_na,
500, meter)))
deleteVariable(boomkikker_landbiotoop_na)
deleteVariable(boomkikker_waterbiotoop2)
assign(boomkikker_waterbiotoop3_cluster, FuzzyClusterIdU(boomkikker_waterbiotoop3, 500, meter))
deleteVariable(boomkikker_waterbiotoop3_cluster)
assign(boomkikker_leefgebied1, or(boomkikker_waterbiotoop3, boomkikker_landbiotoop1))
// clustering? Alle leefgebied op een onderlinge afstand van maximaal 500m en oppervlakte moet minstens 5ha zijn
assign(boomkikker_leefgebied1_cluster, FuzzyClusterIdU(boomkikker_leefgebied1, 500, meter))
assign(boomkikker_leefgebied, cFilterGeArea(boomkikker_leefgebied1_cluster, 5, hectare))
deleteVariable(boomkikker_leefgebied1_cluster)
// baken ecoregio van ecodistrictgebieden af
assign(boomkikker_ecodistrict, OrEq('ecoregio.asc', 1, 2, 12, 13, 16, 17, 36))
assign(boomkikker_waterbiotoop_ecodistrict1, and(boomkikker_waterbiotoop3, boomkikker_leefgebied,
boomkikker_ecodistrict))
assign(boomkikker_landbiotoop_ecodistrict1, and(boomkikker_landbiotoop1, boomkikker_leefgebied,
boomkikker_ecodistrict))
deleteVariable(boomkikker_landbiotoop1)
deleteVariable(boomkikker_waterbiotoop3)
assign(boomkikker_leefgebied_ecodistrict1, and(boomkikker_leefgebied, boomkikker_ecodistrict))
deleteVariable(boomkikker_leefgebied)
deleteVariable(boomkikker_leefgebied1)
deleteVariable(boomkikker_ecodistrict)
assign(boomkikker_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(boomkikker_leefgebied_ecodistrict1, 500,
meter))
assign(boomkikker_leefgebied_ecodistrict, cFilterGeArea(boomkikker_leefgebied_ecodistrict1_cluster, 5, hectare))
//write('boomkikker_leefgebied_ecodistrict.asc', if(boomkikker_leefgebied_ecodistrict,
boomkikker_leefgebied_ecodistrict1_cluster, 0))
deleteVariable(boomkikker_leefgebied_ecodistrict1_cluster)
deleteVariable(boomkikker_leefgebied_ecodistrict1)
assign(boomkikker_waterbiotoop_ecodistrict, and(boomkikker_waterbiotoop_ecodistrict1,
boomkikker_leefgebied_ecodistrict))
write('boomkikker_waterbiotoop_ecodistrict.asc', IsNonZero(boomkikker_waterbiotoop_ecodistrict))
assign(boomkikker_landbiotoop_ecodistrict, and(boomkikker_landbiotoop_ecodistrict1,
boomkikker_leefgebied_ecodistrict))
write('boomkikker_landbiotoop_ecodistrict.asc', IsNonZero(boomkikker_landbiotoop_ecodistrict))
deleteVariable(boomkikker_landbiotoop_ecodistrict1)
deleteVariable(boomkikker_landbiotoop_ecodistrict)
deleteVariable(boomkikker_leefgebied_ecodistrict)
deleteVariable(boomkikker_waterbiotoop_ecodistrict)
deleteVariable(boomkikker_waterbiotoop_ecodistrict1)

```



Potentiële leefgebiedenkaart voor Heikikker op basis van het onderstaande.gdx-script.
Groen = waterbiotoop, oranje = landbiotoop.

Gdx-script

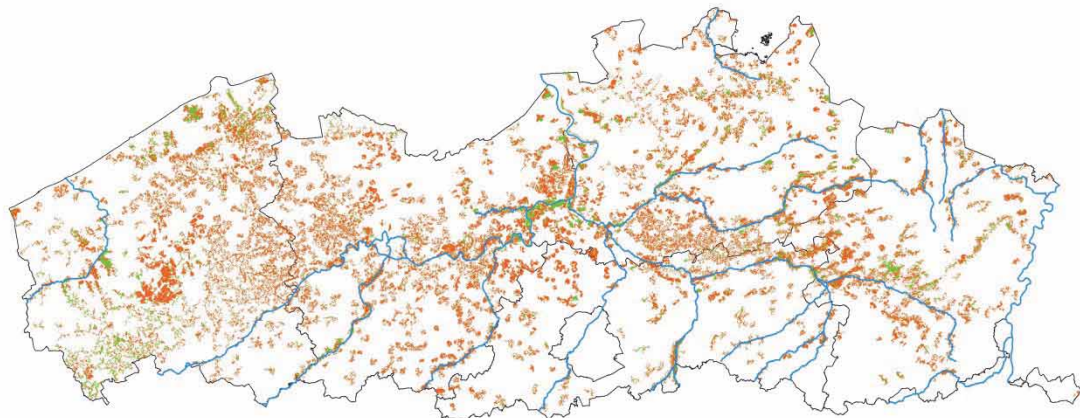
```
vlaanderen_20m.asc
// baken geschikte waterbiotopen af
assign(heikikker_waterbiotoop0, bwk(ao%, 3110%, 3130%, 3160%))
// uit GRB-watgang, kleine waters proberen te filteren
assign(GRB_water, OrEq('GRB_watgang_at.asc', 1))
assign(GRB_water_cluster, fuzzyClusterIdU(GRB_water, 10, meter))
deleteVariable(GRB_water)
assign(heikikker_kleinwater, cFilterLeArea(GRB_water_cluster, 0.4, hectare))
assign(heikikker_waterbiotoop1, or(heikikker_waterbiotoop0, heikikker_kleinwater))
deleteVariable(heikikker_waterbiotoop0)
deleteVariable(GRB_water_cluster)
deleteVariable(heikikker_kleinwater)
// binnen een straal van 500m moeten minstens 5 (6-1 dwz 6 - de eigen poel) andere geschikte
// voortplantingsgebieden liggen
assign(heikikker_waterbiotoop1_cluster, clusterId(heikikker_waterbiotoop1))
deleteVariable(heikikker_waterbiotoop1)
assign(heikikker_waterbiotoop2, GeUniqueCountInBuffer(heikikker_waterbiotoop1_cluster, 500, meter,
  heikikker_waterbiotoop1_cluster, 6))
deleteVariable(heikikker_waterbiotoop1_cluster)
// baken landbiotopen af
assign(heikikker_landbiotoop1, bwk(ce%, cm, cm+, ct, ct+, hj%, hm%, hmo%, mc%, mcb%, md%, mk%, ms%, pms,
  pms+, ppms, ppms+, qb, qb+, qs, qs+, sm%, so%, t%, vo%, vt%, 4010%, 7150%, 2310%, 4030%, 6230%, 6410%,
  rbbms%, 7110%, 7140%, 7230%, rbbso%, rbbso%, rbbso%))
// waterbiotoop moet binnen een straal van 500 m van landbiotoop liggen
assign(heikikker_waterbiotoop3, and(heikikker_waterbiotoop2, cellDistanceLe(heikikker_landbiotoop1, 500,
  meter)))
// landbiotoop moet binnen een straal van 500 m van waterbiotoop liggen
assign(heikikker_landbiotoop2, and(heikikker_landbiotoop1, cellDistanceLe(heikikker_waterbiotoop2, 500, meter)))
deleteVariable(heikikker_landbiotoop1)
deleteVariable(heikikker_waterbiotoop2)
// leefgebied = waterbiotoop + landbiotoop
assign(heikikker_leefgebied1, or(heikikker_landbiotoop2, heikikker_waterbiotoop3))
// leefgebiedbiotoop moet minstens 50 ha groot zijn
// clustering? Alle landbiotoop op een onderlinge afstand van maximaal 500m
assign(heikikker_leefgebied1_cluster, fuzzyClusterIdU(heikikker_leefgebied1, 500, meter))
deleteVariable(heikikker_leefgebied1)
// oppervlakte landbiotoop moet minstens 50ha zijn
assign(heikikker_leefgebied, cFilterGeArea(heikikker_leefgebied1_cluster, 50, hectare))
deleteVariable(heikikker_leefgebied1_cluster)
```

```
// baken ecoregio van ecodistrictgebieden af
assign(heikikker_ecodistrict, OrEq('ecoregio.asc', 12, 13, 14, 15, 16, 17))
assign(heikikker_landbiotoop_ecodistrict1, and(heikikker_landbiotoop2, heikikker_ecodistrict))
deleteVariable(heikikker_landbiotoop2)
assign(heikikker_waterbiotoop_ecodistrict1, and(heikikker_waterbiotoop3, heikikker_ecodistrict))
deleteVariable(heikikker_waterbiotoop3)
assign(heikikker_leefgebied_ecodistrict1, and(heikikker_leefgebied, heikikker_ecodistrict))
assign(heikikker_leefgebied_ecodistrict1_cluster, fuzzyClusterIdU(heikikker_leefgebied_ecodistrict1, 500, meter))
deleteVariable(heikikker_leefgebied)
deleteVariable(heikikker_ecodistrict)
deleteVariable(heikikker_leefgebied_ecodistrict1)
assign(heikikker_leefgebied_ecodistrict, cFilterGeArea(heikikker_leefgebied_ecodistrict1_cluster, 50, hectare))
//write('heikikker_leefgebied_ecodistrict.asc', if(heikikker_leefgebied_ecodistrict,
//    heikikker_leefgebied_ecodistrict1_cluster, 0))
deleteVariable(heikikker_leefgebied_ecodistrict1_cluster)
assign(heikikker_landbiotoop_ecodistrict, and(heikikker_landbiotoop_ecodistrict1,
//    heikikker_leefgebied_ecodistrict))
write('heikikker_landbiotoop_ecodistrict.asc', heikikker_landbiotoop_ecodistrict)
deleteVariable(heikikker_landbiotoop_ecodistrict1)
deleteVariable(heikikker_landbiotoop_ecodistrict)
assign(heikikker_waterbiotoop_ecodistrict, and(heikikker_waterbiotoop_ecodistrict1,
//    heikikker_leefgebied_ecodistrict))
write('heikikker_waterbiotoop_ecodistrict.asc', IsNonZero(heikikker_waterbiotoop_ecodistrict))
deleteVariable(heikikker_waterbiotoop_ecodistrict1)
deleteVariable(heikikker_waterbiotoop_ecodistrict)
deleteVariable(heikikker_leefgebied_ecodistrict)
```



Parende heikikkers (foto Lars Soerink – Vilda)

Kamsalamander (*Triturus cristatus*) – Speybroeck (2015)



Potentiële leefgebiedenkaart voor Kamsalamander op basis van het onderstaande gdx-script.
Groen = waterbiotoop, oranje = landbiotoop.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(kamsalamander_waterbiotoop_bwk, bwk(ae%, aer%, aev%, aom%, kn%, mp%, mr%, 3130%, 3150%, 2190%,
rbbmr%))
// uit GRB-watgang, kleine waters proberen te filteren
assign(GRB_water, eq('GRB_watgang_at.asc', 1))
assign(GRB_water_cluster, fuzzyClusterIdU(GRB_water, 10, meter))
deleteVariable(GRB_water)
assign(kamsalamander_kleinwater, cFilterLeArea(GRB_water_cluster, 0.4, hectare))
deleteVariable(GRB_water_cluster)
assign(kamsalamander_waterbiotoop1, or(kamsalamander_waterbiotoop_bwk, kamsalamander_kleinwater))
deleteVariable(kamsalamander_waterbiotoop_bwk)
deleteVariable(kamsalamander_kleinwater)
// binnen een straal van 500m moeten minstens 5 (6-1 dwz 6 - de eigen poel) andere geschikte
voortplantingsgebieden liggen
assign(kamsalamander_waterbiotoop1_cluster, clusterId(kamsalamander_waterbiotoop1))
deleteVariable(kamsalamander_waterbiotoop1)
assign(kamsalamander_waterbiotoop2, GeUniqueCountInBuffer(kamsalamander_waterbiotoop1_cluster, 500,
meter, kamsalamander_waterbiotoop1_cluster, 6))
deleteVariable(kamsalamander_waterbiotoop1_cluster)
// baken landbiotoop af
assign(kamsalamander_landbiotoop_bwk, bwk(hc%, hf%, hj%, hp%, hp+%, hpr%, hpr+%, hr%, kb%, kh%, khw%, kt%,
ku%, kw%, lhb%, lhi%, sd%, sf%, sp%, sz%, va%, vn%, rbbhc%, rbbhf%, 6430%, 2160%, 2170%, rbbfs%, rbbsp%,
91E0%))
// geschikt waterbiotoop moet binnen de 500m geschikt landbiotoop liggen
assign(kamsalamander_waterbiotoop3, and(kamsalamander_waterbiotoop2,
cellDistanceLe(kamsalamander_landbiotoop_bwk, 500, meter)))
// geschikt landbiotoop moet binnen de 500m geschikt waterbiotoop liggen
assign(kamsalamander_landbiotoop1, and(kamsalamander_landbiotoop_bwk,
cellDistanceLe(kamsalamander_waterbiotoop2, 500, meter)))
deleteVariable(kamsalamander_landbiotoop_bwk)
deleteVariable(kamsalamander_waterbiotoop2)
assign(kamsalamander_leefgebied1, or(kamsalamander_waterbiotoop3, kamsalamander_landbiotoop1))
// clustering: indien geschikte cel binnen 500 meter, dan hoort die cel bij het leefgebiedcluster
assign(kamsalamander_leefgebied1_cluster, fuzzyClusterIdU(kamsalamander_leefgebied1, 500, meter))
deleteVariable(kamsalamander_leefgebied1)
// leefgebiedclusters moeten minimaal 14ha groot zijn
assign(kamsalamander_leefgebied, cFilterGeArea(kamsalamander_leefgebied1_cluster, 14, hectare))
// write('kamsalamander_leefgebied.asc', if(kamsalamander_leefgebied, kamsalamander_leefgebied1_cluster, 0))
```

```

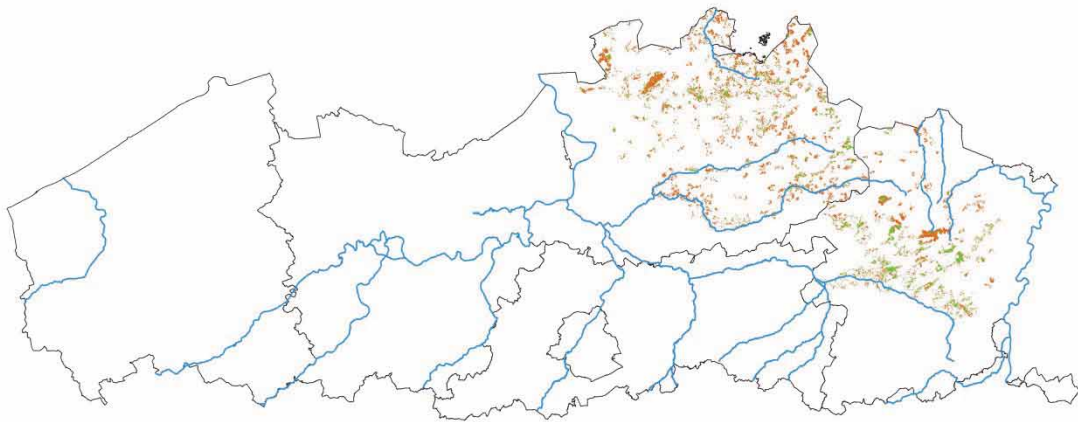
deleteVariable(kamsalamander_leefgebied1_cluster)
assign(kamsalamander_landbiotoop2, and(kamsalamander_landbiotoop1, kamsalamander_leefgebied))
deleteVariable(kamsalamander_landbiotoop1)
assign(kamsalamander_waterbiotoop4, and(kamsalamander_waterbiotoop3, kamsalamander_leefgebied))
deleteVariable(kamsalamander_waterbiotoop3)
// baken ecoregio van ecodistrictgebieden af
assign(kamsalamander_ecodistrict, ge('ecoregio.asc', 1))
assign(kamsalamander_landbiotoop_ecodistrict1, and(kamsalamander_landbiotoop2, kamsalamander_ecodistrict))
deleteVariable(kamsalamander_landbiotoop2)
assign(kamsalamander_waterbiotoop_ecodistrict1, and(kamsalamander_waterbiotoop4,
  kamsalamander_ecodistrict))
deleteVariable(kamsalamander_waterbiotoop4)
assign(kamsalamander_leefgebied_ecodistrict1, and(kamsalamander_leefgebied, kamsalamander_ecodistrict))
deleteVariable(kamsalamander_leefgebied)
deleteVariable(kamsalamander_ecodistrict)
assign(kamsalamander_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(kamsalamander_leefgebied_ecodistrict1,
  100, meter))
deleteVariable(kamsalamander_leefgebied_ecodistrict1)
assign(kamsalamander_leefgebied_ecodistrict, cFilterGeArea(kamsalamander_leefgebied_ecodistrict1_cluster, 50,
  hectare))
//write('kamsalamander_leefgebied_ecodistrict.asc', if(kamsalamander_leefgebied_ecodistrict,
  kamsalamander_leefgebied_ecodistrict1_cluster, 0))
deleteVariable(kamsalamander_leefgebied_ecodistrict1_cluster)
assign(kamsalamander_landbiotoop_ecodistrict, and(kamsalamander_landbiotoop_ecodistrict1,
  kamsalamander_leefgebied_ecodistrict))
write('kamsalamander_landbiotoop_ecodistrict.asc', kamsalamander_landbiotoop_ecodistrict)
deleteVariable(kamsalamander_landbiotoop_ecodistrict)
deleteVariable(kamsalamander_landbiotoop_ecodistrict1)
assign(kamsalamander_waterbiotoop_ecodistrict, and(kamsalamander_waterbiotoop_ecodistrict1,
  kamsalamander_leefgebied_ecodistrict))
write('kamsalamander_waterbiotoop_ecodistrict.asc', IsNonZero(kamsalamander_waterbiotoop_ecodistrict))
deleteVariable(kamsalamander_waterbiotoop_ecodistrict)
deleteVariable(kamsalamander_waterbiotoop_ecodistrict1)
deleteVariable(kamsalamander_leefgebied_ecodistrict)

```



Kamsalamander (foto Yves Adams – Vilda)

Knoflookpad (*Pelobates fuscus*) – Speybroeck (2015)



Potentiële leefgebiedenkaart voor Knoflookpad op basis van het onderstaande gdx-script.
Groen = waterbiotoop, oranje = landbiotoop.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(knoflookpad_waterbiotoop_bwk, bwk(ae, ae+, aer, aer+, ao, ao+, aom, aom+, kn, ku%))
// uit GRB-watgang, kleine waters proberen te filteren
assign(GRB_water, Eq('GRB_watgang_at.asc', 1))
assign(GRB_water_cluster, fuzzyClusterIdU(GRB_water, 10, meter))
assign(knoflookpad_kleinwater, cFilterLeArea(GRB_water_cluster, 0.4, hectare))
deleteVariable(GRB_water)
deleteVariable(GRB_water_cluster)
assign(knoflookpad_waterbiotoop1, or(knoflookpad_waterbiotoop_bwk, knoflookpad_kleinwater))
deleteVariable(knoflookpad_waterbiotoop_bwk)
deleteVariable(knoflookpad_kleinwater)
// binnen een straal van 500m moeten minstens 3 (4-1 dwz 4 - de eigen poel) andere geschikte
// voortplantingsgebieden liggen
assign(knoflookpad_waterbiotoop1_cluster, clusterId(knoflookpad_waterbiotoop1))
deleteVariable(knoflookpad_waterbiotoop1)
assign(knoflookpad_waterbiotoop2, GeUniqueCountInBuffer(knoflookpad_waterbiotoop1_cluster, 500, meter,
knoflookpad_waterbiotoop1_cluster, 4))
deleteVariable(knoflookpad_waterbiotoop1_cluster)
// baken landbiotoop af
assign(knoflookpad_landbiotoop_bwk, bwk(bs%, cd%, cg%, cgb%, cm%, cp%, cv%, cvb%, dm%, ha%, hab%, hm%,
hmo%, k(ha%, k(hm%, k(cg%, ku%, ko%, mr%, sf%, sg%, sz%))
// landbiotoop moet binnen de 500 m van waterbiotoop liggen
assign(knoflookpad_waterbiotoop3, and(knoflookpad_waterbiotoop2,
cellDistanceLe(knoflookpad_landbiotoop_bwk, 250, meter)))
// binnen landbiotoop van 1 ha moet binnen de 50m geschikt water liggen
assign(knoflookpad_landbiotoop1, and(knoflookpad_landbiotoop_bwk,
cellDistanceLe(knoflookpad_waterbiotoop2, 250, meter)))
deleteVariable(knoflookpad_landbiotoop_bwk)
deleteVariable(knoflookpad_waterbiotoop2)
assign(knoflookpad_leefgebied1, or(knoflookpad_waterbiotoop3, knoflookpad_landbiotoop1))
// clustering: indien geschikte cel binnen 500 meter, dan hoort die cel bij het landbiotoopcluster
assign(knoflookpad_leefgebied1_cluster, fuzzyClusterIdU(knoflookpad_leefgebied1, 250, meter))
deleteVariable(knoflookpad_leefgebied1)
// leefgebiedclusters moeten minimaal 20ha groot zijn
assign(knoflookpad_leefgebied, cFilterGeArea(knoflookpad_leefgebied1_cluster, 20, hectare))
deleteVariable(knoflookpad_leefgebied1_cluster)
// write('knoflookpad_leefgebied.asc', IsNonZero(knoflookpad_leefgebied))
// baken ecoregio van de Kempen af af
```

```

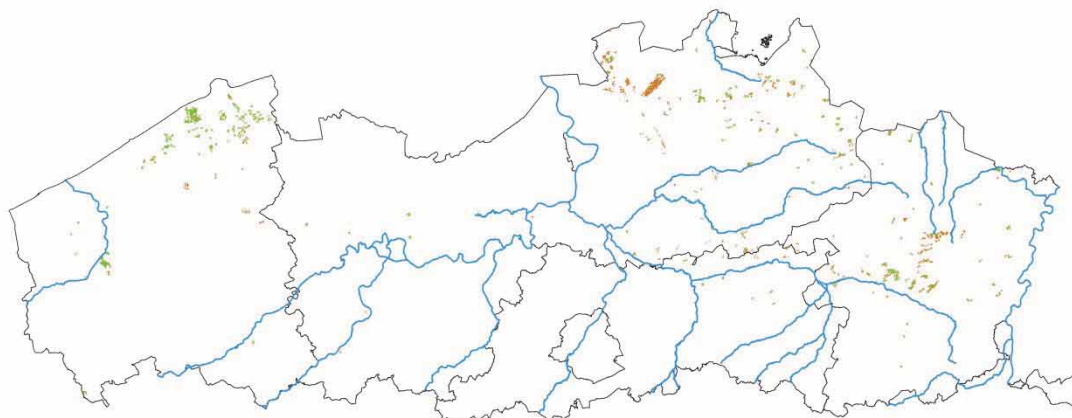
assign(knoflookpad_ecodistrict, OrEq('ecoregio.asc', 12, 13, 15))
assign(knoflookpad_waterbiotoop_ecodistrict1, and(knoflookpad_waterbiotoop3, knoflookpad_ecodistrict))
deleteVariable(knoflookpad_waterbiotoop3)
assign(knoflookpad_landbiotoop_ecodistrict1, and(knoflookpad_landbiotoop1, knoflookpad_ecodistrict))
deleteVariable(knoflookpad_landbiotoop1)
assign(knoflookpad_leefgebied_ecodistrict1, and(knoflookpad_leefgebied, knoflookpad_ecodistrict))
deleteVariable(knoflookpad_ecodistrict)
deleteVariable(knoflookpad_leefgebied)
assign(knoflookpad_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(knoflookpad_leefgebied_ecodistrict1, 250,
meter))
deleteVariable(knoflookpad_leefgebied_ecodistrict1)
assign(knoflookpad_leefgebied_ecodistrict, cFilterGeArea(knoflookpad_leefgebied_ecodistrict1_cluster, 20,
hectare))
//write('knoflookpad_leefgebied_ecodistrict.asc', if(knoflookpad_leefgebied_ecodistrict,
knoflookpad_leefgebied_ecodistrict1_cluster, 0))
deleteVariable(knoflookpad_leefgebied_ecodistrict1_cluster)
assign(knoflookpad_waterbiotoop_ecodistrict, and(knoflookpad_waterbiotoop_ecodistrict1,
knoflookpad_leefgebied_ecodistrict))
deleteVariable(knoflookpad_waterbiotoop_ecodistrict1)
write('knoflookpad_waterbiotoop_ecodistrict.asc', IsNonZero(knoflookpad_waterbiotoop_ecodistrict))
deleteVariable(knoflookpad_waterbiotoop_ecodistrict)
assign(knoflookpad_landbiotoop_ecodistrict, and(knoflookpad_landbiotoop_ecodistrict1,
knoflookpad_leefgebied_ecodistrict))
deleteVariable(knoflookpad_landbiotoop_ecodistrict1)
write('knoflookpad_landbiotoop_ecodistrict.asc', IsNonZero(knoflookpad_landbiotoop_ecodistrict))
deleteVariable(knoflookpad_landbiotoop_ecodistrict)
deleteVariable(knoflookpad_leefgebied_ecodistrict)

```



Knoflookpad (foto Yves Adams – Vilda)

Poelkikker (*Pelophylax lessonae*) – Speybroeck (2015)



Potentiële leefgebiedenkaart voor Poelkikker op basis van het onderstaande gdx-script.
Groen = waterbiotoop, oranje = landbiotoop.

Gdx-script

```
vlaanderen_20m.asc
// baken geschikte waterbiotopen af
assign(poelkikker_waterbiotoop0, bwk(ao%, aom%, aoo%, ap, ap+, apo, apo+, kn%, 3110%, 3130%, 3150%, 3160%))
// uit GRB-watgang, kleine waters proberen te filteren
assign(GRB_water, OrEq('GRB_watgang_at.asc', 1))
assign(GRB_water_cluster, fuzzyClusterIdU(GRB_water, 10, meter))
deleteVariable(GRB_water)
assign(poelkikker_kleinwater, cFilterLearea(GRB_water_cluster, 0.4, hectare))
assign(poelkikker_waterbiotoop1, or(poelkikker_waterbiotoop0, poelkikker_kleinwater))
deleteVariable(poelkikker_waterbiotoop0)
deleteVariable(GRB_water_cluster)
deleteVariable(poelkikker_kleinwater)
// binnen een straal van 500m moeten minstens 3 (4-1) andere geschikte voortplantingsgebieden liggen
assign(poelkikker_waterbiotoop_cluster, clusterId(poelkikker_waterbiotoop1))
deleteVariable(poelkikker_waterbiotoop1)
assign(poelkikker_waterbiotoop2, GeUniqueCountInBuffer(poelkikker_waterbiotoop_cluster, 500, meter,
  poelkikker_waterbiotoop_cluster, 4))
deleteVariable(poelkikker_waterbiotoop_cluster)
// baken landbiotopen af
assign(poelkikker_landbiotoop1, bwk(ce%, cg, cg+, cm, cm+, ct, ct+, hf%, hj%, hm%, hmo%, hp+, hpr+, mc%, mcb%,
  md%, mk%, mr%, ms%, pms, pms+, ppmb, ppmb+, ppms, ppms+, qb, qb+, qs, qs+, sm%, so%, sz%, t%, vo%, vt%,
  2310%, 4010%, 4030%, 6230%, 6410%, 6430%, 7110%, 7140%, 7150%, 7230%, 9120%, 9190%, 91E0%, rbbmr%,
  rbbms%, rbbppm%, rbbso%, rbbso%))
// waterbiotoop moet in de buurt (<100m) van landbiotoop liggen
assign(poelkikker_waterbiotoop3, and(poelkikker_waterbiotoop2, cellDistanceLe(poelkikker_landbiotoop1, 100,
  meter)))
// landbiotoop moet binnen een straal van 100 m liggen
assign(poelkikker_landbiotoop2, and(poelkikker_landbiotoop1, cellDistanceLe(poelkikker_waterbiotoop2, 100,
  meter)))
deleteVariable(poelkikker_landbiotoop1)
deleteVariable(poelkikker_waterbiotoop2)
assign(poelkikker_leefgebied1, or(poelkikker_waterbiotoop3, poelkikker_landbiotoop2))
// oppervlakte moet minstens 10ha zijn
assign(poelkikker_leefgebied1_cluster, fuzzyClusterIdU(poelkikker_leefgebied1, 100, meter))
deleteVariable(poelkikker_leefgebied1)
assign(poelkikker_leefgebied, cFilterGeArea(poelkikker_leefgebied1_cluster, 10, hectare))
deleteVariable(poelkikker_leefgebied1_cluster)
// eventueel nog ecodistricten afbaken en gebruiken om verder te verfijnen
assign(poelkikker_ecodistrict, OrEq('ecoregio.asc', 2, 3, 4, 7, 11, 12, 13, 14, 15, 16, 17, 29, 31, 32))
```

```

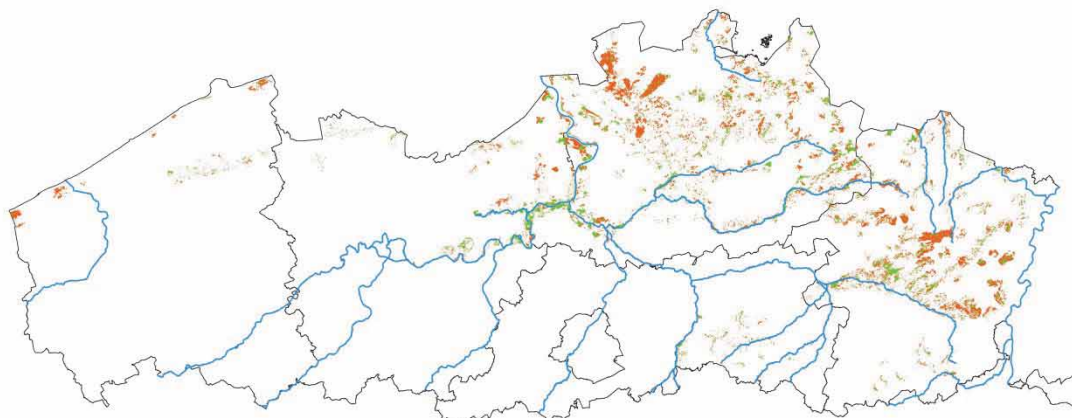
assign(poelkikker_waterbiotoop_ecodistrict1, and(poelkikker_waterbiotoop3, poelkikker_ecodistrict))
deleteVariable(poelkikker_waterbiotoop3)
assign(poelkikker_landbiotoop_ecodistrict1, and(poelkikker_landbiotoop2, poelkikker_ecodistrict))
deleteVariable(poelkikker_landbiotoop2)
assign(poelkikker_leefgebied_ecodistrict1, and(poelkikker_leefgebied, poelkikker_ecodistrict))
deleteVariable(poelkikker_leefgebied)
deleteVariable(poelkikker_ecodistrict)
assign(poelkikker_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(poelkikker_leefgebied_ecodistrict1, 100,
    meter))
deleteVariable(poelkikker_leefgebied_ecodistrict1)
assign(poelkikker_leefgebied_ecodistrict, cFilterGeArea(poelkikker_leefgebied_ecodistrict1_cluster, 10, hectare))
//write('poelkikker_leefgebied_ecodistrict.asc', if(poelkikker_leefgebied_ecodistrict,
    poelkikker_leefgebied_ecodistrict1_cluster, 0))
deleteVariable(poelkikker_leefgebied_ecodistrict1_cluster)
assign(poelkikker_waterbiotoop_ecodistrict, and(poelkikker_waterbiotoop_ecodistrict1,
    poelkikker_leefgebied_ecodistrict))
deleteVariable(poelkikker_waterbiotoop_ecodistrict1)
write('poelkikker_waterbiotoop_ecodistrict.asc', IsNonZero(poelkikker_waterbiotoop_ecodistrict))
deleteVariable(poelkikker_waterbiotoop_ecodistrict)
assign(poelkikker_landbiotoop_ecodistrict, and(poelkikker_landbiotoop_ecodistrict1,
    poelkikker_leefgebied_ecodistrict))
deleteVariable(poelkikker_leefgebied_ecodistrict)
deleteVariable(poelkikker_landbiotoop_ecodistrict1)
write('poelkikker_landbiotoop_ecodistrict.asc', poelkikker_landbiotoop_ecodistrict)
deleteVariable(poelkikker_landbiotoop_ecodistrict)

```



Poelkikker (foto Yves Adams – Vilda)

Rugstreeppad (*Bufo calamita*) – Speybroeck (2015)



Potentiële leefgebiedenkaart voor Rugstreeppad op basis van het onderstaande gdx-script.
Groen = waterbiotoop, oranje = landbiotoop.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(rugstreeppad_waterbiotoop_bwk, bwk(ae%, ao%, ap, ap+, apo, apo+, mp%, mr%, 3110%, 3130%, 2190%,
rbbmr%))
// uit GRB-watgang, kleine waters proberen te filteren
assign(GRB_water, OrEq('GRB_watgang_at.asc',1))
assign(GRB_water_cluster, fuzzyClusterIdU(GRB_water, 10, meter))
assign(rugstreeppad_kleinwater, cFilterLeArea(GRB_water_cluster, 0.4, hectare))
deleteVariable(GRB_water)
deleteVariable(GRB_water_cluster)
assign(rugstreeppad_waterbiotoop1, or(rugstreeppad_waterbiotoop_bwk, rugstreeppad_kleinwater))
deleteVariable(rugstreeppad_kleinwater)
deleteVariable(rugstreeppad_waterbiotoop_bwk)
// binnen een straal van 500m moeten minstens 3 (4-1 dwz 4 - de eigen poel) andere geschikte
voortplantingsgebieden liggen
assign(rugstreeppad_waterbiotoop1_cluster, clusterId(rugstreeppad_waterbiotoop1))
assign(rugstreeppad_waterbiotoop2, GeUniqueCountInBuffer(rugstreeppad_waterbiotoop1_cluster, 500, meter,
rugstreeppad_waterbiotoop1_cluster, 4))
deleteVariable(rugstreeppad_waterbiotoop1)
deleteVariable(rugstreeppad_waterbiotoop1_cluster)
// cluster waterbiotopen
assign(rugstreeppad_waterbiotoop2_cluster, fuzzyClusterIdU(rugstreeppad_waterbiotoop2, 500, meter))
deleteVariable(rugstreeppad_waterbiotoop2)
// waterbiotoopcluster moet 5ha groot zijn
assign(rugstreeppad_waterbiotoop_5ha, cFilterGeArea(rugstreeppad_waterbiotoop2_cluster, 5, hectare))
deleteVariable(rugstreeppad_waterbiotoop2_cluster)
// baken landbiotoop af
assign(rugstreeppad_land, bwk(cd%, ce%, cg%, cm%, cp%, cv%, dd%, dm%, ha%, hc%, hd%, hf%, hmo%, hn%, hr%,
kc%, kg%, kn%, kt%, ku%, kz%, mp%, mr%, pmb%, ppmb%, sd%, sf%, sg%, sm%, so%, sz%, 2130%, 2150%, 2160%,
2170%, 2190%, 2310%, 2330%, 4010%, 4030%, 6230%, 6410%, 6430%, 7150%, 7230%, rbbsg%, rbbism%, rbbhc%,
rbbhf%, rbbmr%, 2190%, rbbppm%, rbbso%))
// clustering: indien geschikte cel binnen 500 meter, dan hoort die cel bij het landbiotoopcluster
assign(rugstreeppad_land_cluster, fuzzyClusterIdU(rugstreeppad_land, 500, meter))
deleteVariable(rugstreeppad_land)
// landclusters moeten minimaal 20ha groot zijn
assign(rugstreeppad_land_cluster_20ha, cFilterGeArea(rugstreeppad_land_cluster, 20, hectare))
deleteVariable(rugstreeppad_land_cluster)
// zoek voortplantingsclusters die in de buurt van landclusters liggen (<500m)
```

```

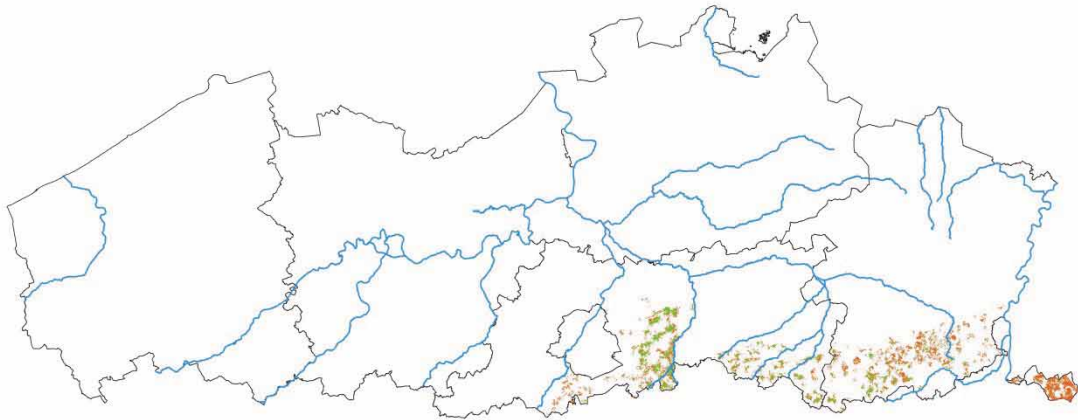
assign(rugstreepad_waterbiotoop3, and(rugstreepad_waterbiotoop_5ha,
  cellDistanceLe(rugstreepad_land_cluster_20ha, 500, meter)))
assign(rugstreepad_landbiotoop1, and(rugstreepad_land_cluster_20ha,
  cellDistanceLe(rugstreepad_waterbiotoop_5ha, 500, meter)))
deleteVariable(rugstreepad_waterbiotoop_5ha)
deleteVariable(rugstreepad_land_cluster_20ha)
assign(rugstreepad_leefgebied1, or(rugstreepad_waterbiotoop3, rugstreepad_landbiotoop1))
// clustering? Alle leefgebied op een onderlinge afstand van maximaal 500m
assign(rugstreepad_leefgebied1_cluster, FuzzyClusterIdU(rugstreepad_leefgebied1, 500, meter))
deleteVariable(rugstreepad_leefgebied1)
// oppervlakte moet minstens 20ha zijn
assign(rugstreepad_leefgebied, cFilterGeArea(rugstreepad_leefgebied1_cluster, 20, hectare))
deleteVariable(rugstreepad_leefgebied1_cluster)
assign(rugstreepad_ecodistrict, OrEq('ecoregio.asc', 1, 3, 5, 9, 10, 12, 13, 15, 16, 17, 28, 30, 32, 33))
assign(rugstreepad_waterbiotoop_ecodistrict1, and(rugstreepad_waterbiotoop3, rugstreepad_ecodistrict))
deleteVariable(rugstreepad_waterbiotoop3)
assign(rugstreepad_landbiotoop_ecodistrict1, and(rugstreepad_landbiotoop1, rugstreepad_ecodistrict))
deleteVariable(rugstreepad_landbiotoop1)
assign(rugstreepad_leefgebied_ecodistrict1, and(rugstreepad_leefgebied, rugstreepad_ecodistrict))
deleteVariable(rugstreepad_leefgebied)
deleteVariable(rugstreepad_ecodistrict)
assign(rugstreepad_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(rugstreepad_leefgebied_ecodistrict1, 500,
  meter))
deleteVariable(rugstreepad_leefgebied_ecodistrict1)
assign(rugstreepad_leefgebied_ecodistrict, cFilterGeArea(rugstreepad_leefgebied_ecodistrict1_cluster, 20,
  hectare))
//write('rugstreepad_leefgebied_ecodistrict.asc', if(rugstreepad_leefgebied_ecodistrict,
  rugstreepad_leefgebied_ecodistrict1_cluster, 0))
deleteVariable(rugstreepad_leefgebied_ecodistrict1_cluster)
assign(rugstreepad_waterbiotoop_ecodistrict, and(rugstreepad_waterbiotoop_ecodistrict1,
  rugstreepad_leefgebied_ecodistrict))
write('rugstreepad_waterbiotoop_ecodistrict.asc', IsNonZero(rugstreepad_waterbiotoop_ecodistrict))
deleteVariable(rugstreepad_waterbiotoop_ecodistrict1)
deleteVariable(rugstreepad_waterbiotoop_ecodistrict)
assign(rugstreepad_landbiotoop_ecodistrict, and(rugstreepad_landbiotoop_ecodistrict1,
  rugstreepad_leefgebied_ecodistrict))
write('rugstreepad_landbiotoop_ecodistrict.asc', IsNonZero(rugstreepad_landbiotoop_ecodistrict))
deleteVariable(rugstreepad_landbiotoop_ecodistrict1)
deleteVariable(rugstreepad_landbiotoop_ecodistrict)
deleteVariable(rugstreepad_leefgebied_ecodistrict)

```



Rugstreepad (foto Lars Soerink – Vilda)

Vroedmeesterpad (*Alytes obstetricans*) – Speybroeck (2015)



Potentiële leefgebiedenkaart voor Vroedmeesterpad op basis van het onderstaande gdx-script.
Groen = waterbiotoop, oranje = landbiotoop.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(vroedmeesterpad_waterbiotoop_bwk, bwk(ae%, aer%, aev%, ao%, aom%, aoo%, kn%, ur%))
// uit GRB-watgang, kleine waters proberen te filteren
assign(GRB_water, Eq('GRB_watgang_at.asc', 1))
assign(GRB_water_cluster, fuzzyClusterIdU(GRB_water, 10, meter))
assign(vroedmeesterpad_kleinwater, cFilterLeArea(GRB_water_cluster, 0.4, hectare))
deleteVariable(GRB_water)
deleteVariable(GRB_water_cluster)
assign(vroedmeesterpad_waterbiotoop1, or(vroedmeesterpad_waterbiotoop_bwk, vroedmeesterpad_kleinwater))
deleteVariable(vroedmeesterpad_kleinwater)
// binnen een straal van 500m moeten minstens 3 (4-1 dwz 4 - de eigen poel) andere geschikte
// voortplantingsgebieden liggen
assign(vroedmeesterpad_waterbiotoop1_cluster, clusterId(vroedmeesterpad_waterbiotoop1))
deleteVariable(vroedmeesterpad_waterbiotoop_bwk)
deleteVariable(vroedmeesterpad_waterbiotoop1)
assign(vroedmeesterpad_waterbiotoop2, GeUniqueCountInBuffer(vroedmeesterpad_waterbiotoop1_cluster, 500,
meter, vroedmeesterpad_waterbiotoop1_cluster, 4))
deleteVariable(vroedmeesterpad_waterbiotoop1_cluster)
// baken landbiotoop af
assign(vroedmeesterpad_landbiotoop_bwk, bwk(ha%, hc%, hk%, hmo%, hn%, hp%, hp+%, hpr%, hu%, kb%, kc%,
kh%, khw%, kj%, kp%, kr%, kt%, kw%, sg%, sk%, sp%, sz%, ur%))
// landbiotoop moet binnen de 500 m van waterbiotoop liggen
assign(vroedmeesterpad_waterbiotoop3, and(vroedmeesterpad_waterbiotoop2,
cellDistanceLe(vroedmeesterpad_landbiotoop_bwk, 500, meter)))
// binnen landbiotoop van 1 ha moet binnen de 500m geschikt water liggen
assign(vroedmeesterpad_landbiotoop1, and(vroedmeesterpad_landbiotoop_bwk,
cellDistanceLe(vroedmeesterpad_waterbiotoop2, 500, meter)))
deleteVariable(vroedmeesterpad_landbiotoop_bwk)
deleteVariable(vroedmeesterpad_waterbiotoop2)
assign(vroedmeesterpad_leefgebied1, or(vroedmeesterpad_waterbiotoop3, vroedmeesterpad_landbiotoop1))
// clustering: indien geschikte cel binnen 500 meter, dan hoort die cel bij het landbiotoopcluster
assign(vroedmeesterpad_leefgebied1_cluster, fuzzyClusterIdU(vroedmeesterpad_leefgebied1, 500, meter))
deleteVariable(vroedmeesterpad_leefgebied1)
// landclusters moeten minimaal 5ha groot zijn
assign(vroedmeesterpad_leefgebied, cFilterGeArea(vroedmeesterpad_leefgebied1_cluster, 5, hectare))
deleteVariable(vroedmeesterpad_leefgebied1_cluster)
// eventueel nog ecodistricten afbaken en gebruiken om verder te verfijnen
assign(vroedmeesterpad_ecodistrict, OrEq('ecoregio.asc', 27, 32, 34, 35))
```

```

assign(vroedmeesterpad_waterbiotoop_ecodistrict1, and(vroedmeesterpad_waterbiotoop3,
  vroedmeesterpad_ecodistrict))
deleteVariable(vroedmeesterpad_waterbiotoop3)
assign(vroedmeesterpad_landbiotoop_ecodistrict1, and(vroedmeesterpad_landbiotoop1,
  vroedmeesterpad_ecodistrict))
deleteVariable(vroedmeesterpad_landbiotoop1)
assign(vroedmeesterpad_leefgebied_ecodistrict1, and(vroedmeesterpad_leefgebied,
  vroedmeesterpad_ecodistrict))
deleteVariable(vroedmeesterpad_leefgebied)
deleteVariable(vroedmeesterpad_ecodistrict)
assign(vroedmeesterpad_leefgebied_ecodistrict1_cluster,
  FuzzyClusterIdU(vroedmeesterpad_leefgebied_ecodistrict1, 500, meter))
assign(vroedmeesterpad_leefgebied_ecodistrict, cFilterGeArea(vroedmeesterpad_leefgebied_ecodistrict1_cluster,
  5, hectare))
write('vroedmeesterpad_leefgebied_ecodistrict.asc', if(vroedmeesterpad_leefgebied_ecodistrict,
  vroedmeesterpad_leefgebied_ecodistrict1_cluster, 0))
deleteVariable(vroedmeesterpad_leefgebied_ecodistrict1_cluster)
assign(vroedmeesterpad_waterbiotoop_ecodistrict, and(vroedmeesterpad_waterbiotoop_ecodistrict1,
  vroedmeesterpad_leefgebied_ecodistrict))
write('vroedmeesterpad_waterbiotoop_ecodistrict.asc', IsNonZero(vroedmeesterpad_waterbiotoop_ecodistrict))
deleteVariable(vroedmeesterpad_waterbiotoop_ecodistrict1)
deleteVariable(vroedmeesterpad_waterbiotoop_ecodistrict)
assign(vroedmeesterpad_landbiotoop_ecodistrict, and(vroedmeesterpad_landbiotoop_ecodistrict1,
  vroedmeesterpad_leefgebied_ecodistrict))
write('vroedmeesterpad_landbiotoop_ecodistrict.asc', IsNonZero(vroedmeesterpad_landbiotoop_ecodistrict))
deleteVariable(vroedmeesterpad_landbiotoop_ecodistrict1)
deleteVariable(vroedmeesterpad_landbiotoop_ecodistrict)

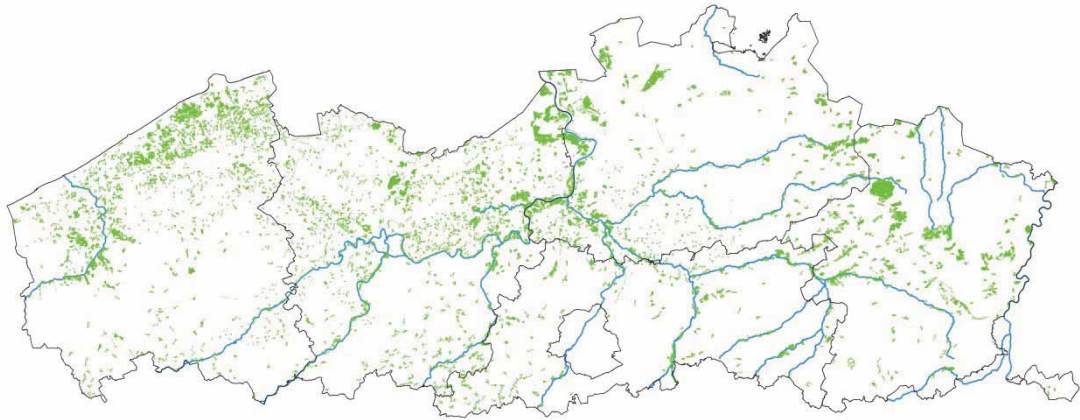
```



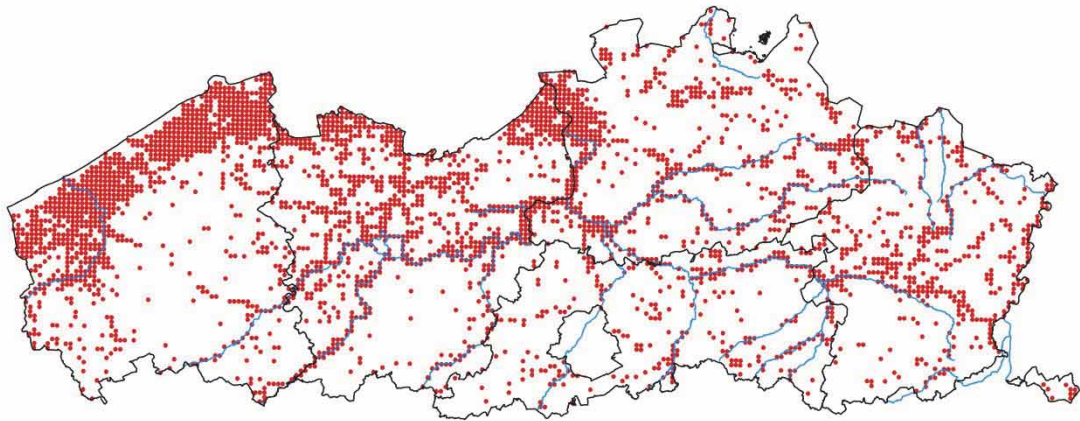
Vroedmeesterpad (foto Yves Adams – Vilda)

BROEDVOGELS

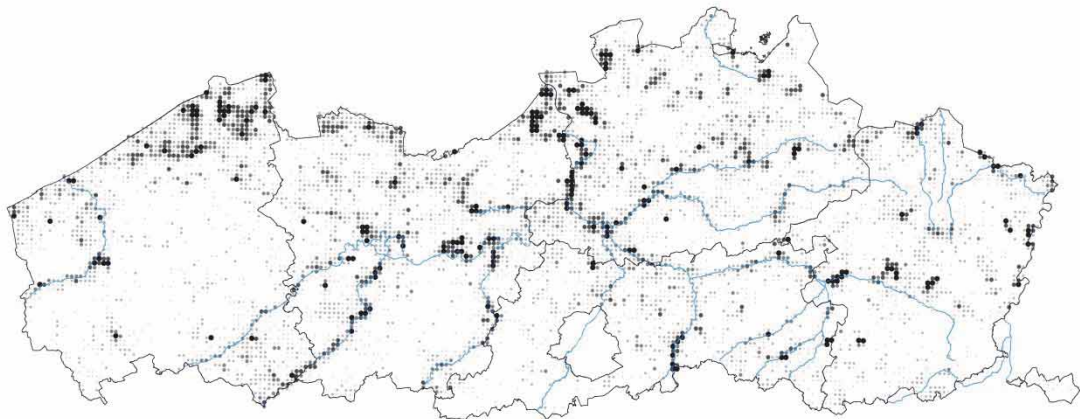
Blauwborst (*Luscinia svecica*) – Adriaens & Vermeersch (in druk)



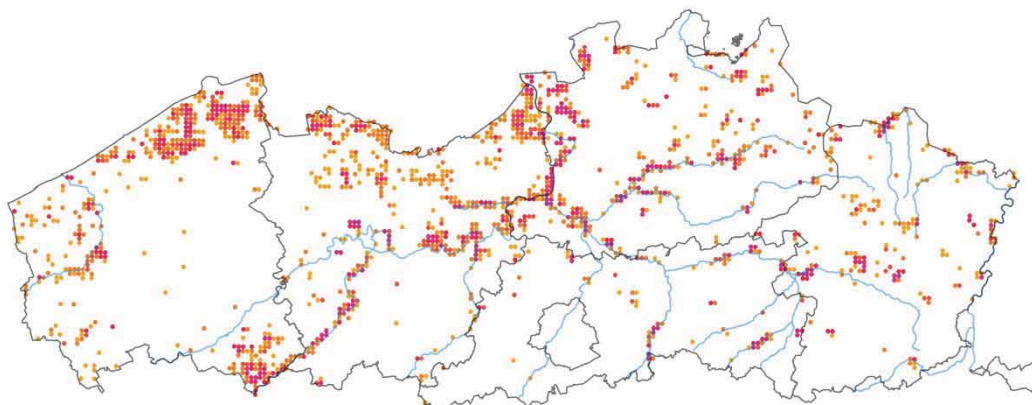
Potentiële leefgebiedenkaart voor Blauwborst op basis van het onderstaande gdx-script.



Potentiële verspreidingskaart voor Blauwborst op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Blauwborst per kilometerhok in Vlaanderen (Tabel 7). Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Blauwborst uit het Natuurpunt data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

Gdx-script

vlaanderen_20m.asc

```
// baken de voortplantingsgebieden af in 2 stappen: eerst voor polders en zandig vlaanderen en daarna voor de rest
// polders en zandig vlaanderen
```

```
assign(blauwborst_bwk, bwk(app, app+, ce%, ceb%, ces%, cm%, cmb%, hf%, hfb%, hfc%, hft%, hr%, hrb%, ka%,
kbs%, ku%, kub%, kz%, mc%, md%, mdb%, mk%, mr%, mru%, ms%, msb%, mz%, ru%, se%, sf%, sm%, sz%, k(ce%,
k(cm%, k(cmb%, k(hf%, k(hr%, k(mc%, k(md%, k(mk%, k(mr%, k(ms%, k(mz%, k(ru%, k(se%, k(sf%, k(sm%, k(sz%,
1330%, 2180%, 2190%, 2310%, 3130%, 3270%, 4010%, 4030%, 6430%, 7140%, 7150%, 7230%, rbbhf, rbbmr,
rbbms))
```

```
// clustering: indien geschikte cel binnen 20 meter, dan hoort die cel bij het voortplantingscluster
```

```
assign(blauwborst_voorplanting1_cluster, fuzzyClusterIdU(blauwborst_bwk, 20, meter))
```

```
deleteVariable(blauwborst_bwk)
```

```
// optimale voortplantingsclusters moeten minimaal 5 ha groot zijn
```

```
assign(blauwborst_leefgebied5, cFilterGeArea(blauwborst_voorplanting1_cluster, 5, hectare))
```

```
// eventueel nog ecodistricten afbakenen en gebruiken om verder te verfijnen
```

```
assign(blauwborst_polders_zandig, OrEq('ecoregio.asc', 2, 3, 4, 5, 6, 7, 8, 9, 10, 11))
```

```
assign(blauwborst_leefgebied_polders_zandig, and(blauwborst_leefgebied5, blauwborst_polders_zandig))
```

```
deleteVariable(blauwborst_leefgebied5)
```

```
// write('blauwborst_leefgebied_polders_zandig.asc', if(blauwborst_leefgebied_polders_zandig,
blauwborst_voorplanting1_cluster, 0))
```

```
// rest van Vlaanderen
```

```
// optimale voortplantingsclusters moeten minimaal 15 ha groot zijn
```

```
assign(blauwborst_leefgebied10, cFilterGeArea(blauwborst_voorplanting1_cluster, 15, hectare))
```

```
// eventueel nog ecodistricten afbakenen en gebruiken om verder te verfijnen
```

```
assign(blauwborst_leefgebied_rest, and(blauwborst_leefgebied10, not(blauwborst_polders_zandig)))
```

```
deleteVariable(blauwborst_leefgebied10)
```

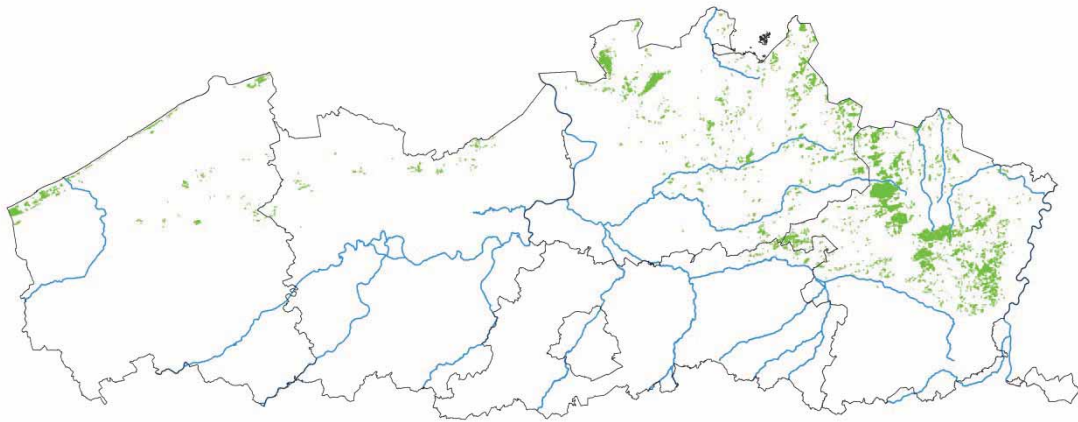
```
// write('blauwborst_leefgebied_rest.asc', if(blauwborst_leefgebied_rest, blauwborst_voorplanting1_cluster, 0))
```

```
deleteVariable(blauwborst_voorplanting1_cluster)
```

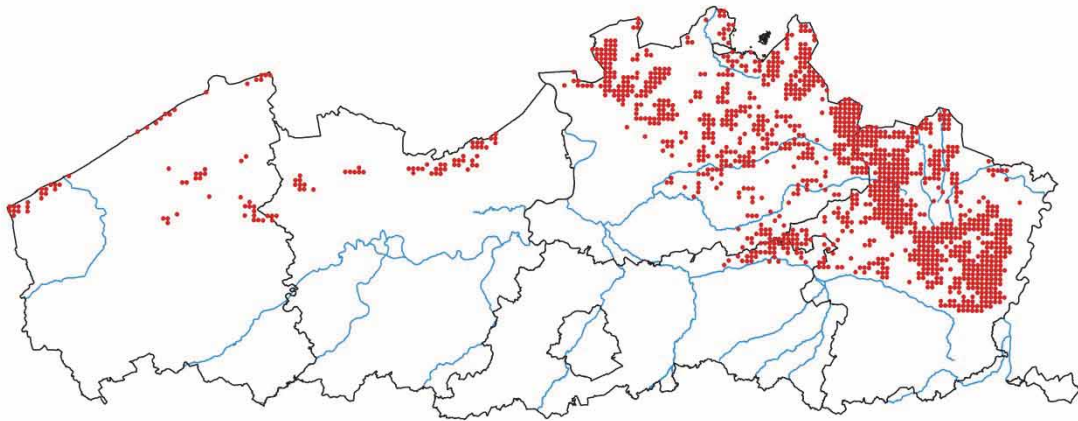
```
assign(blauwborst_leefgebied, or(blauwborst_leefgebied_polders_zandig, blauwborst_leefgebied_rest))
```

```
assign(blauwborst_leefgebied_cluster, fuzzyClusterIdU(blauwborst_leefgebied, 20, meter))
```

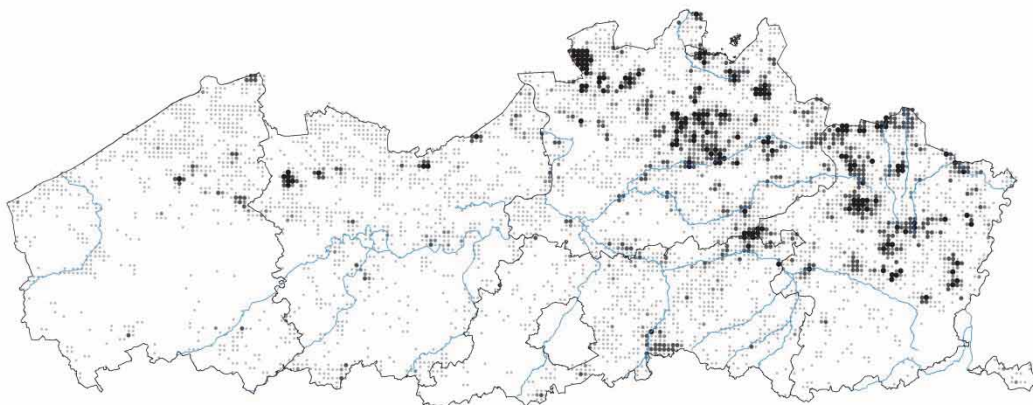
```
write('blauwborst_leefgebied.asc', if(blauwborst_leefgebied, blauwborst_leefgebied_cluster, 0))
```



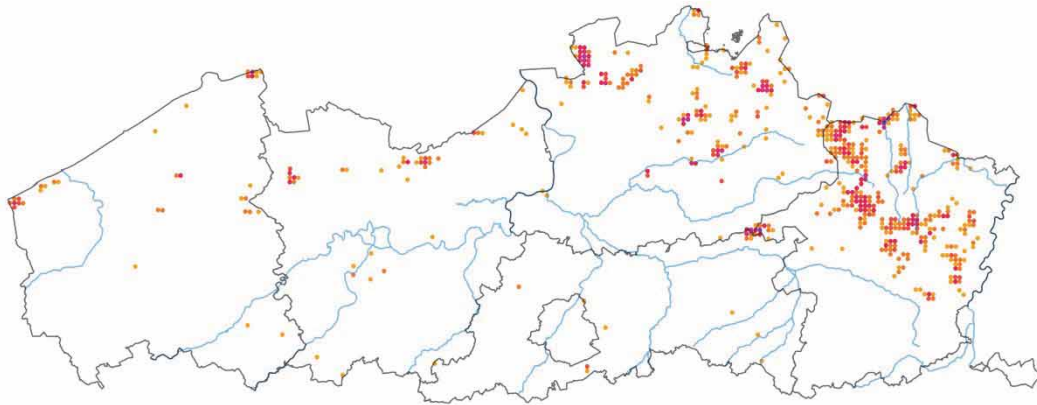
Potentiële leefgebiedenkaart voor Boomleeuwerik op basis van onderstaande het gdx-script.



Potentiële verspreidingskaart voor Boomleeuwerik op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Boomleeuwerik per kilometerhok in Vlaanderen (Tabel 7). Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Boomleeuwerik uit het Natuurpunt data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

Gdx-script

vlaanderen_20m.asc

// baken de voortplantingsgebieden af

```
assign(boomleeuwerik_voortplanting, bwk(cd%, ce%, cg%, cm%, cv%, dm%, ha%, hd%, hk%, hn%, pa%, pi%, pms%,
    ppa%, ppi%, sd%, se%, sg%, 2130%, 2150%, 2310%, 2330%, 3130%, 3270%, 4010%, 4030%, 5130%, 6210%,
    6230%, 6430%))
```

// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het voortplantingscluster

```
assign(boomleeuwerik_voortplanting_cluster, fuzzyClusterIdU(boomleeuwerik_voortplanting, 50, meter))
```

```
deleteVariable(boomleeuwerik_voortplanting)
```

// optimale voortplantingsclusters moeten minimaal 5ha groot zijn

```
assign(boomleeuwerik_leefgebied, cFilterGeArea(boomleeuwerik_voortplanting_cluster, 5, hectare))
```

// baken ecoregio van ecodistrictgebieden af

```
assign(boomleeuwerik_ecodistrict, OrEq('ecoregio.asc', 1, 5, 7, 8, 12, 13, 14, 15, 16, 17))
```

```
assign(boomleeuwerik_leefgebied_ecodistrict, and(boomleeuwerik_leefgebied, boomleeuwerik_ecodistrict))
```

```
deleteVariable(boomleeuwerik_leefgebied)
```

```
deleteVariable(boomleeuwerik_ecodistrict)
```

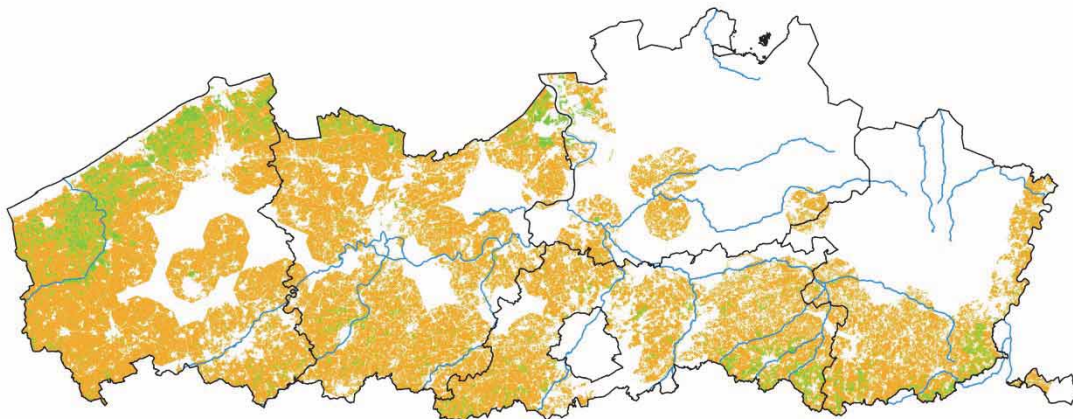
```
write('boomleeuwerik_leefgebied_ecodistrict.asc', if(boomleeuwerik_leefgebied_ecodistrict,
    boomleeuwerik_voortplanting_cluster, 0))
```

// write('boomleeuwerik_leefgebied_ecodistrict.asc', IsNonZero(boomleeuwerik_leefgebied_ecodistrict))

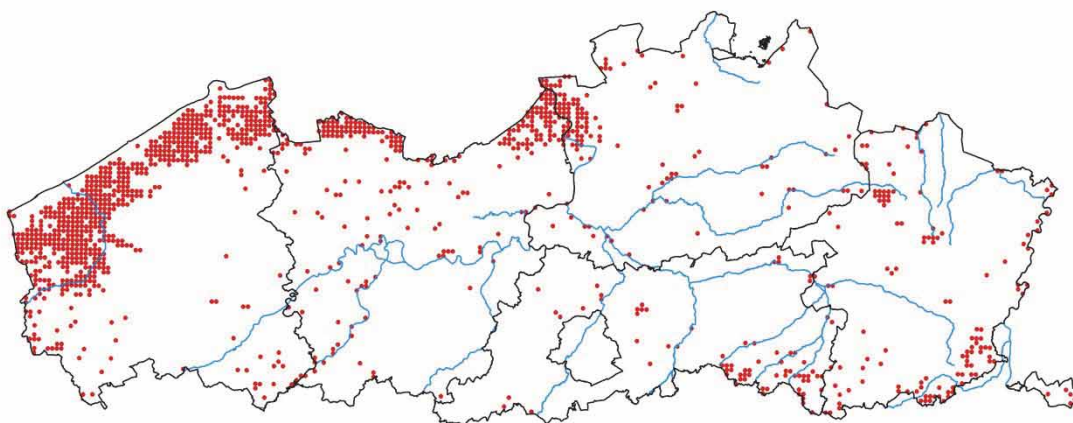
```
deleteVariable(boomleeuwerik_voortplanting_cluster)
```

```
deleteVariable(boomleeuwerik_leefgebied_ecodistrict)
```

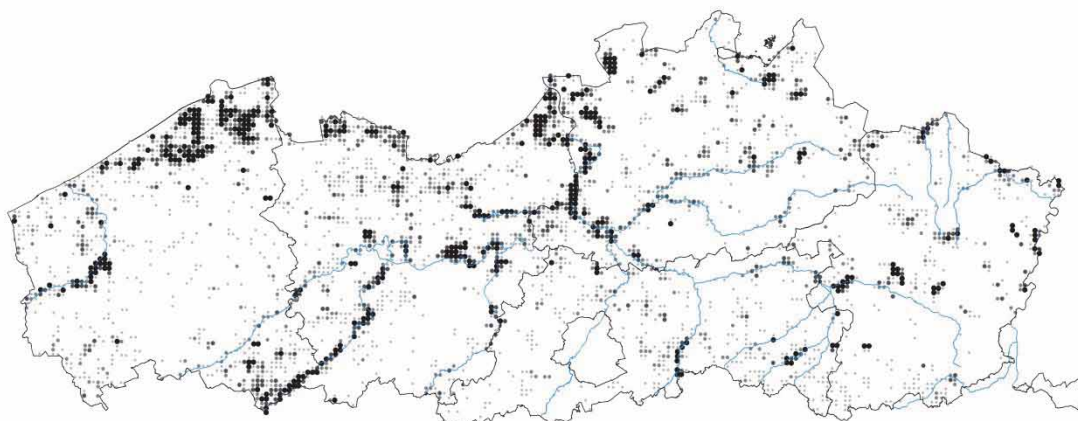
Bruine kiekendief (*Circus aeruginosus*) – Adriaens & Vermeersch (in druk)



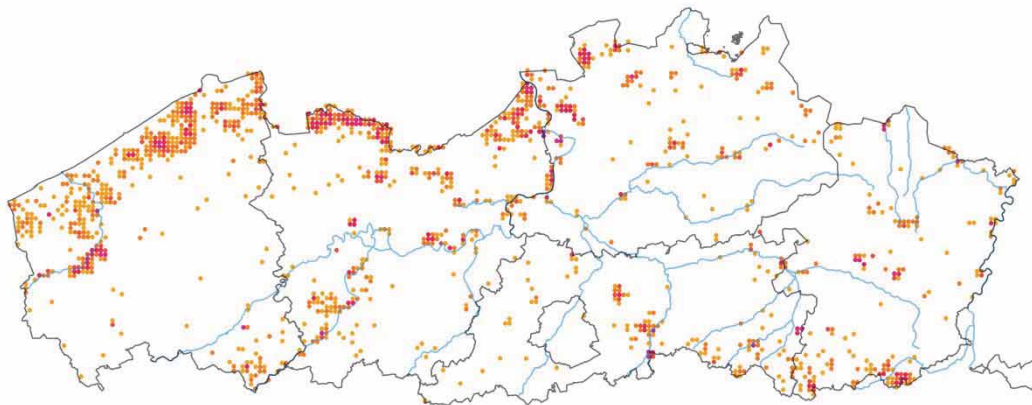
Potentiële leefgebiedenkaart voor Bruine kiekendief op basis van het onderstaande gdx-script. Groen = voortplantingsgebied, oranje = foerageergebied.



Potentiële verspreidingskaart voor Bruine kiekendief op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Bruine kiekendief per kilometerhok in Vlaanderen (Tabel 7). Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Bruine kiekendief uit het data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

Gdx-script

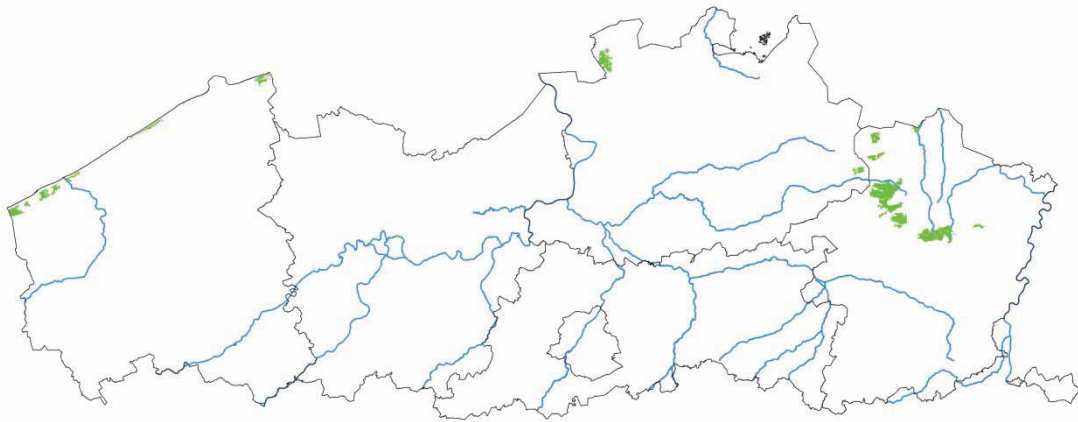
vlaanderen_20m.asc

```
// baken de voortplantingsgebieden af
assign(bruinekiekendief_epr, OrEq('epr2013.asc', 6, 38, 100))
assign(bruinekiekendief_voortplanting_bwk, bwk(ku%, k(mc%, k(mr%, k(mz%, mc%, mr%, mz%))
assign(bruinekiekendief_voortplanting, or(bruinekiekendief_epr, bruinekiekendief_voortplanting_bwk))
deleteVariable(bruinekiekendief_epr)
deleteVariable(bruinekiekendief_voortplanting_bwk)
// geen bos in de wijde omgeving van de voortplantingsgebieden
assign(bruinekiekendief_bos, bwk(ek%, es%, fa%, fe%, fk%, fl%, fm%, fs%, gml%, gmn%, lh%, lhb%, lhi%, ls%, lsb%,
    lsh%, lsi%, n%, pa%, pi%, pm%, pmb%, pmh%, pms%, pp%, ppa%, ppi%, ppm%, ppmb%, ppmh%, ppms%, qa%,
    qb%, qd%, qe%, qk%, ql%, qs%, qx%, ru%, rud%, va%, vc%, vf%, vm%, vn%, vo%, vt%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het boscluster
assign(bruinekiekendief_bos_cluster, fuzzyClusterIdU(bruinekiekendief_bos, 50, meter))
deleteVariable(bruinekiekendief_bos)
// geen bossen groter dan 5ha in de buurt van voortplantingsgebieden
assign(bruinekiekendief_bos_5ha, cFilterGeArea(bruinekiekendief_bos_cluster, 5, hectare))
deleteVariable(bruinekiekendief_bos_cluster)
// sluit voortplantingsgebieden uit die binnen de 1km meer dan 5 ha bos hebben
assign(bruinekiekendief_leefgebied_zonderbos, and(bruinekiekendief_voortplanting,
    cellDistanceGe(bruinekiekendief_bos_5ha, 1000, meter)))
deleteVariable(bruinekiekendief_bos_5ha)
// cluster voortplantingsgebied
assign(bruinekiekendief_leefgebied0_cluster, fuzzyClusterIdU(bruinekiekendief_leefgebied_zonderbos, 100,
    meter))
deleteVariable(bruinekiekendief_leefgebied_zonderbos)
// optimale voortplantingsclusters moeten minimaal 10ha groot zijn
assign(bruinekiekendief_voortplanting, cFilterGeArea(bruinekiekendief_leefgebied0_cluster, 10, hectare))
deleteVariable(bruinekiekendief_leefgebied0_cluster)
// baken foerageergebied af
assign(bruinekiekendief_foerageer, bwk(cd%, ce%, cg%, cm%, cp%, cv%, da%, ha%, hc%, hf%, hj%, hp%, hp+%,
    hpr%, hpr+%, hr%, hu%, hx%, ku%, mc%, mr%, mrb%, mz%, mru%, cdb%, ceb%, cgb%, cmb%, cpb%, cvb%, hab%,
    hcb%, hfb%, hjb%, hft%, hfc%, hrb%, hub%, kub%, mcb%, ka%, bl%, bs%, bu%))
// clustering: indien geschikte cel binnen 100 meter, dan hoort die cel bij het foerageercluster
assign(bruinekiekendief_foerageer_cluster, fuzzyClusterIdU(bruinekiekendief_foerageer, 100, meter))
// optimale foerageerclusters moeten minimaal 100ha groot zijn
assign(bruinekiekendief_foerageer, cFiltergeArea(bruinekiekendief_foerageer_cluster, 100, hectare))
deleteVariable(bruinekiekendief_foerageer_cluster)
// baken leefgebied af: voldoende groot voortplantingsgebied moet binnen de 5 km van voldoende groot
foerageergebied liggen
```

```

assign(bruinekiekendief_leefgebied_voortplanting, and(bruinekiekendief_voortplanting,
  cellDistanceLe(bruinekiekendief_foerageer, 5000, meter)))
write('bruinekiekendief_leefgebied_voortplanting.asc', IsNonZero(bruinekiekendief_leefgebied_voortplanting))
// baken leefgebied af: voldoende groot foerageergebied moet binnen de 5 km van voldoende groot
  voortplantingsgebied liggen
assign(bruinekiekendief_leefgebied_foerageer, and(bruinekiekendief_foerageer,
  cellDistanceLe(bruinekiekendief_voortplanting, 5000, meter)))
write('bruinekiekendief_leefgebied_foerageer.asc', IsNonZero(bruinekiekendief_leefgebied_foerageer))
//deleteVariable(bruinekiekendief_foerageer)
//deleteVariable(bruinekiekendief_voortplanting)
assign(bruinekiekendief_leefgebied1, or(bruinekiekendief_leefgebied_voortplanting,
  bruinekiekendief_leefgebied_foerageer))
// baken ecoregio's af
assign(bruinekiekendief_ecodistrict, OrEq('ecoregio.asc', 2, 3))
assign(bruinekiekendief_leefgebied_voortplanting_ecodistrict1, and(bruinekiekendief_leefgebied_voortplanting,
  bruinekiekendief_ecodistrict))
deleteVariable(bruinekiekendief_leefgebied_voortplanting)
assign(bruinekiekendief_leefgebied_foerageer_ecodistrict1, and(bruinekiekendief_leefgebied_foerageer,
  bruinekiekendief_ecodistrict))
deleteVariable(bruinekiekendief_leefgebied_foerageer)
assign(bruinekiekendief_leefgebied_ecodistrict1, and(bruinekiekendief_leefgebied1, bruinekiekendief_ecodistrict))
deleteVariable(bruinekiekendief_leefgebied1)
deleteVariable(bruinekiekendief_ecodistrict)
assign(bruinekiekendief_leefgebied_ecodistrict1_cluster,
  FuzzyClusterIdU(bruinekiekendief_leefgebied_ecodistrict1, 500, meter))
deleteVariable(bruinekiekendief_leefgebied_ecodistrict1)
assign(bruinekiekendief_leefgebied_ecodistrict, cFilterGeArea(bruinekiekendief_leefgebied_ecodistrict1_cluster,
  100, hectare))
//write('bruinekiekendief_leefgebied_ecodistrict.asc', if(bruinekiekendief_leefgebied_ecodistrict,
  bruinekiekendief_leefgebied_ecodistrict1_cluster, 0))
//write('bruinekiekendief_leefgebied_ecodistrict.asc', IsNonZero(bruinekiekendief_leefgebied_ecodistrict))
deleteVariable(bruinekiekendief_leefgebied_ecodistrict1_cluster)
assign(bruinekiekendief_leefgebied_voortplanting_ecodistrict,
  and(bruinekiekendief_leefgebied_voortplanting_ecodistrict1, bruinekiekendief_leefgebied_ecodistrict))
deleteVariable(bruinekiekendief_leefgebied_voortplanting_ecodistrict1)
write('bruinekiekendief_leefgebied_voortplanting_ecodistrict.asc',
  IsNonZero(bruinekiekendief_leefgebied_voortplanting_ecodistrict))
deleteVariable(bruinekiekendief_leefgebied_voortplanting_ecodistrict)
assign(bruinekiekendief_leefgebied_foerageer_ecodistrict,
  and(bruinekiekendief_leefgebied_foerageer_ecodistrict1, bruinekiekendief_leefgebied_ecodistrict))
deleteVariable(bruinekiekendief_leefgebied_foerageer_ecodistrict1)
deleteVariable(bruinekiekendief_leefgebied_ecodistrict)
write('bruinekiekendief_leefgebied_foerageer_ecodistrict.asc',
  IsNonZero(bruinekiekendief_leefgebied_foerageer_ecodistrict))

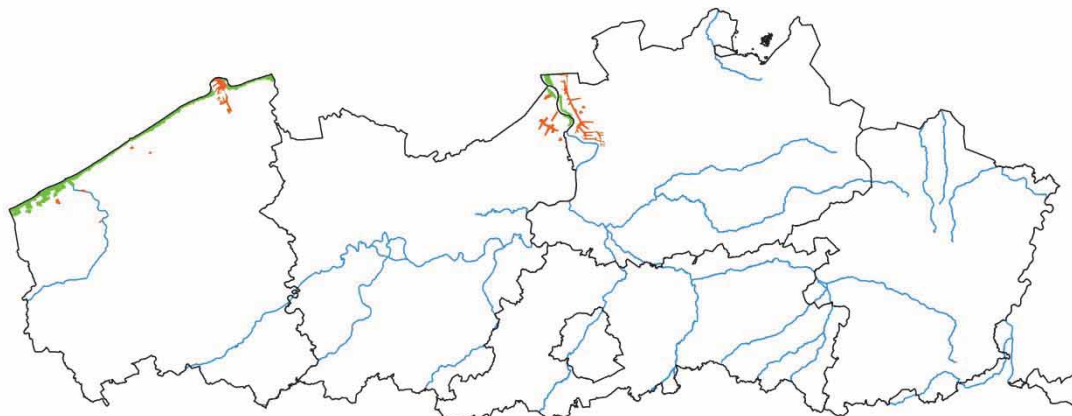
```



Potentiële leefgebiedenkaart voor Duinpieper op basis van het onderstaande gdx-script.

Gdx-script

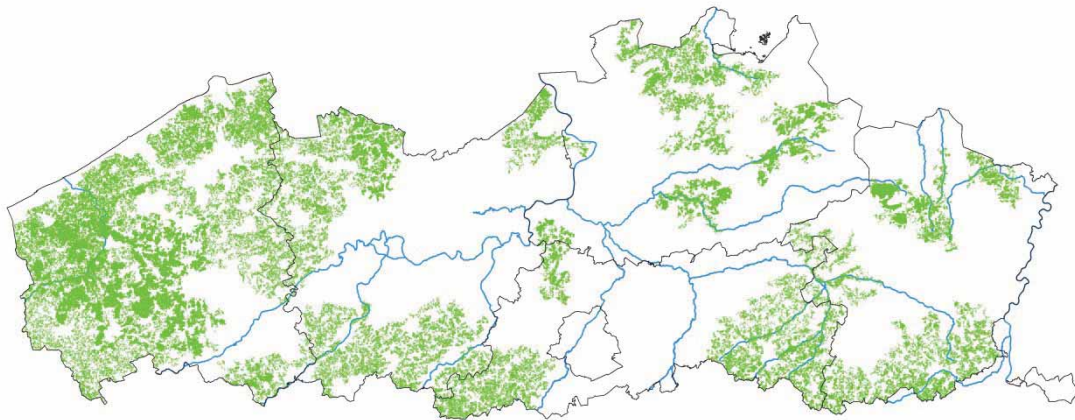
```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(duinpieper_voortplanting, bwk(cd%, cg%, cm%, dd%, dm%, ha%, had%, hd%, hn%, ku%, sd%, sg%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het voortplantingscluster
assign(duinpieper_voortplanting_cluster, fuzzyClusterIdU(duinpieper_voortplanting, 50, meter))
// leefgebied moet groot genoeg zijn >75 ha
assign(duinpieper_leefgebied1, cFilterGeArea(duinpieper_voortplanting_cluster, 75, hectare))
// van het leefgebied van minstens 75 ha moet minstens 5 ha stuifduin zijn
// baken stuifduin af
assign(duinpieper_stuifduin, bwk(dd%, dm%))
// baken stuifduinclusters af (binnen 20m is zelfde cluster)
assign(duinpieper_stuifduin_cluster, fuzzyClusterIdU(duinpieper_stuifduin, 20, meter))
// stuifduincluster moeten minimaal 5ha groot zijn
assign(duinpieper_stuifduin1, cFilterGeArea(duinpieper_stuifduin_cluster, 5, hectare))
// leefgebied is voldoende groot voortplantingsgebied waarvan minstens 5 ha stuifduin is
assign(duinpieper_leefgebied, cFilter(duinpieper_leefgebied1, and(duinpieper_leefgebied1,
    cellDistanceLe(duinpieper_stuifduin1, 10, meter))))
// write('duinpieper_leefgebied.asc', if(duinpieper_leefgebied, duinpieper_voortplanting_cluster, 0))
write('duinpieper_leefgebied.asc', IsNonZero(duinpieper_leefgebied))
```



Potentiële leefgebiedenkaart voor Dwergstern op basis van het onderstaande gdx-script.
Groen = voortplantingsgebied, oranje = foerageergebied.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(dwergstern_voortplanting1, bwk(dd%, dl%, ds%, dz%))
// , ku%, kz%
// leefgebied mag geen open water of ruigte of moeras zijn
assign(openwater, clusterId(bwk(ad%, ae%, aer%, aes%, aev%, ah%, am%, ao%, aom%, aoo%, ap%, apo%, app%,
mz%)))
assign(dwergstern_voortplanting2, and(dwergstern_voortplanting1, not(openwater)))
// leefgebied moeten minimaal 20ha groot zijn
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het cluster
assign(dwergstern_voortplanting2_cluster, fuzzyClusterIdU(dwergstern_voortplanting2, 50, meter))
assign(dwergstern_voortplanting3, cFilterGeArea(dwergstern_voortplanting2_cluster, 20, hectare))
// Zoek openwater in de kaarten van bwk, zee en waterlopen, die nog verrasterd moeten worden
assign(dwergstern_openwater, bwk(ad%, ae%, ah%, ap%, zee%))
// waterlopen
assign(dwergstern_waterlopen, OrEq('rastert_huetzon1.asc', 17, 19))
assign(dwergstern_foerageer1, or(dwergstern_openwater, dwergstern_waterlopen))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het openwaterscluster
assign(dwergstern_foerageer1cl, fuzzyClusterIdU(dwergstern_foerageer1, 50, meter))
// Open water moet minstens 1 ha groot zijn
assign(dwergstern_openwater, cFilterGeArea(dwergstern_foerageer1cl, 1, hectare))
// zoek voortplantingsgebieden die maximaal 5000 meter van groot open water liggen
assign(dwergstern_voortplanting4, cFilter(dwergstern_voortplanting3, cellDistanceLe(dwergstern_openwater,
5000, meter)))
assign(dwergstern_foerageer4, cFilter(dwergstern_openwater, cellDistanceLe(dwergstern_voortplanting3, 5000,
meter)))
assign(dwergstern_leefgebied1, or(dwergstern_voortplanting4, dwergstern_foerageer4))
assign(dwergstern_leefgebied1_cluster, FuzzyClusterIdU(dwergstern_leefgebied1, 50, meter))
assign(dwergstern_leefgebied, cFilterGeArea(dwergstern_leefgebied1_cluster, 20, hectare))
// write('dwergstern_leefgebied.asc', if(dwergstern_leefgebied, dwergstern_leefgebied1_cluster, 0))
//write('dwergstern_leefgebied.asc', IsNonZero(dwergstern_leefgebied))
assign(dwergstern_foerageer, and(dwergstern_foerageer4, dwergstern_leefgebied))
write('dwergstern_foerageer.asc', IsNonZero(dwergstern_foerageer))
assign(dwergstern_voortplanting, and(dwergstern_voortplanting4, dwergstern_leefgebied))
write('dwergstern_voortplanting.asc', dwergstern_voortplanting)
```



Potentiële leefgebiedenkaart voor Grauwe kiekendief op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
assign(bebouwing, and(bwk(u%), not(bwk(ulm))))
assign(bebouwing_cluster_10ha_100m, ge(cSum(FuzzyClusterId(bebouwing, 100/cellsize), 1), 10*10000/cellsize2))
assign(bebouwing_afstand, celldistance(bebouwing_cluster_10ha_100m, 0))
//write('grauwekiekendief_bebouwing_cluster_10ha_100m.asc', bebouwing_cluster_10ha_100m)
// enkel nestgebieden (selectie uit epr2013) overhouden die op minder dan 100 m gelegen zijn van
// bebouwingsclusters
assign(nestgebied_epr, and(OrEq('epr2013.asc', 6, 38, 100), ge(bebouwing_afstand, 100/cellsize)))
//write('grauwekiekendief_biotopen_nestgebied.asc', nestgebied_epr)
// enkel foerageergebieden overhouden die op minder dan 100 m gelegen zijn van bebouwingsclusters
assign(foerageergebied_niet_akker, bwk(cd-, cd+, ce-, ce+, cg-, cg+, cm-, cm+, cp-, cp+, cv-, cv+,
    dm-, dm+, ha-, ha+, hp-, hp+, hpr-, hpr+, hr-, hr+, hx-, hx+, ku-, ku+))
assign(foerageergebied, and(or(foerageergebied_niet_akker, nestgebied_epr), ge(bebouwing_afstand,
    100/cellsize)))
//write('grauwekiekendief_biotopen_foerageergebied.asc', foerageergebied)
// definitie randen: alle laaggroen geselecteerd uit de groenkaart van 2013 dat geen landbouw is en overlapt met
// foerageergebied (incl. nestgebied dus)
// assign(randen, bwk(k(h%, kt(h%, k(c%, kt(c%)))
assign(randen, and(eq('groenkaart2013_20m.asc', 2), foerageergebied))
//write('grauwekiekendief_biotopen_randen_foerageergebied.asc', randen)
// geschikte biotopen zijn dan de compilatie van de foerageergebieden (omvatten sowieso nestgebieden) en de
// randen (randen eigenlijk ook al vervat in foerageergebied want randen = laaggroen in overlap met
// foerageergebied)
assign(grauwekiekendief_geschikte_biotopen, or(foerageergebied, randen))
// densiteit aan nestbiotopen binnen buffer met radius 400 meter
assign(densiteit_nestbiotopen, suminbuffer(nestgebied_epr, 400, meter)/suminbuffer(1, 400, meter)*100)
//write('grauwekiekendief_densiteit_nestbiotoop_400m.asc', densiteit_nestbiotopen)
assign(densiteit_nestbiotopen_ha, suminbuffer(nestgebied_epr, 400, meter)*cellsize2/10000)
//write('grauwekiekendief_densiteit_nestbiotoop_400m_ha.asc', densiteit_nestbiotopen_ha)
// densiteit aan randen binnen buffer met radius 400 meter
assign(densiteit_rand_laaggroen_landbouw_400m, suminbuffer(randen, 400, meter)/suminbuffer(1, 400,
    meter)*100)
//write('grauwekiekendief_densiteit_rand_laaggroen_landbouw_400m.asc',
    densiteit_rand_laaggroen_landbouw_400m)
assign(densiteit_rand_laaggroen_landbouw_400m_ha, suminbuffer(randen, 400, meter)*cellsize2/10000)
//write('grauwekiekendief_densiteit_rand_laaggroen_landbouw_400m_ha.asc',
    densiteit_rand_laaggroen_landbouw_400m_ha)
// clustering
assign(id_400m, fuzzyClusterId(grauwekiekendief_geschikte_biotopen, 400/cellsize))
assign(id_300m, fuzzyClusterId(grauwekiekendief_geschikte_biotopen, 300/cellsize))
```

```

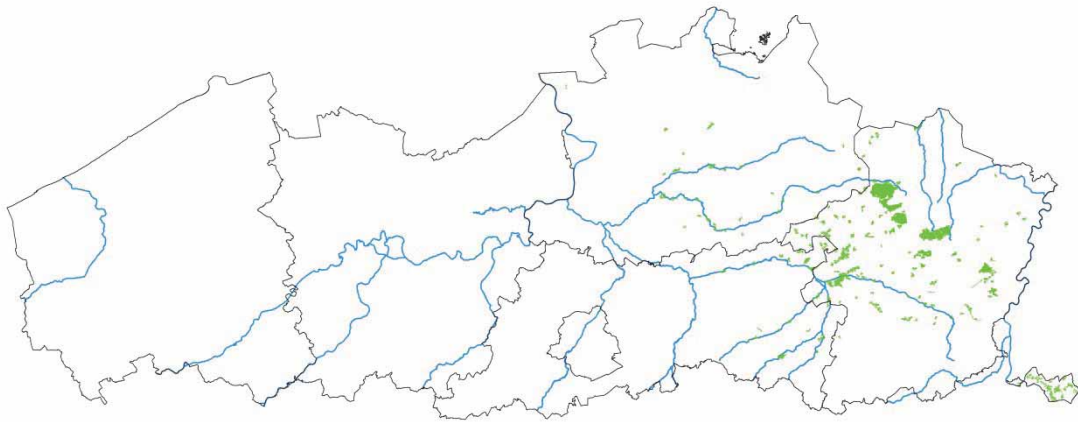
assign(id_200m, fuzzyClusterId(grauwekiekendief_geschikte_biotopen, 200/cellSize))
assign(id_100m, fuzzyClusterId(grauwekiekendief_geschikte_biotopen, 100/cellSize))
// cluster minimaal 3000 ha
assign(cluster_400m, cFilter(id_400m, ge(csum(id_400m, 1), 3000*10000/cellSize2)))
//write('gauwekiekendief_clusters_3000ha_400m.asc', cluster_400m)
deletevariable(id_400m)
assign(cluster_300m, cFilter(id_300m, ge(csum(id_300m, 1), 3000*10000/cellSize2)))
//write('gauwekiekendief_clusters_3000ha_300m.asc', cluster_300m)
deletevariable(id_300m)
assign(cluster_200m, cFilter(id_200m, ge(csum(id_200m, 1), 3000*10000/cellSize2)))
//write('gauwekiekendief_clusters_3000ha_200m.asc', cluster_200m)
deletevariable(id_200m)
assign(cluster_100m, cFilter(id_100m, ge(csum(id_100m, 1), 3000*10000/cellSize2)))
//write('gauwekiekendief_clusters_3000ha_100m.asc', cluster_100m)
deletevariable(id_100m)
// Oppervlakte en aandeel louter foerageerbiotoop
assign(cluster_400m_foerageer_opp_ha, csum(cluster_400m, foerageergebied_niet_akker)*cellsize2/10000)
//write('gauwekiekendief_clusters_3000ha_400m_foerageer_opp_ha.asc', cluster_400m_foerageer_opp_ha)
deletevariable(cluster_400m_foerageer_opp_ha)
assign(cluster_300m_foerageer_opp_ha, csum(cluster_300m, foerageergebied_niet_akker)*cellsize2/10000)
//write('gauwekiekendief_clusters_3000ha_300m_foerageer_opp_ha.asc', cluster_300m_foerageer_opp_ha)
deletevariable(cluster_300m_foerageer_opp_ha)
assign(cluster_200m_foerageer_opp_ha, csum(cluster_200m, foerageergebied_niet_akker)*cellsize2/10000)
//write('gauwekiekendief_clusters_3000ha_200m_foerageer_opp_ha.asc', cluster_200m_foerageer_opp_ha)
deletevariable(cluster_200m_foerageer_opp_ha)
assign(cluster_100m_foerageer_opp_ha, csum(cluster_100m, foerageergebied_niet_akker)*cellsize2/10000)
//write('gauwekiekendief_clusters_3000ha_100m_foerageer_opp_ha.asc', cluster_100m_foerageer_opp_ha)
deletevariable(cluster_100m_foerageer_opp_ha)
assign(cluster_400m_foerageer_fractie, csum(cluster_400m, foerageergebied_niet_akker)/csum(cluster_400m, 1))
//write('gauwekiekendief_clusters_3000ha_400m_foerageer_fractie.asc', cluster_400m_foerageer_fractie)
deletevariable(cluster_400m_foerageer_fractie)
assign(cluster_300m_foerageer_fractie, csum(cluster_300m, foerageergebied_niet_akker)/csum(cluster_300m, 1))
//write('gauwekiekendief_clusters_3000ha_300m_foerageer_fractie.asc', cluster_300m_foerageer_fractie)
deletevariable(cluster_300m_foerageer_fractie)
assign(cluster_200m_foerageer_fractie, csum(cluster_200m, foerageergebied_niet_akker)/csum(cluster_200m, 1))
//write('gauwekiekendief_clusters_3000ha_200m_foerageer_fractie.asc', cluster_200m_foerageer_fractie)
deletevariable(cluster_200m_foerageer_fractie)
assign(cluster_100m_foerageer_fractie, csum(cluster_100m, foerageergebied_niet_akker)/csum(cluster_100m, 1))
//write('gauwekiekendief_clusters_3000ha_100m_foerageer_fractie.asc', cluster_100m_foerageer_fractie)
deletevariable(cluster_100m_foerageer_fractie)
// Oppervlakte en aandeel nestbiotoop
assign(cluster_400m_nest_opp_ha, csum(cluster_400m, nestgebied_epr)*cellsize2/10000)
//write('gauwekiekendief_clusters_3000ha_400m_nest_opp_ha.asc', cluster_400m_nest_opp_ha)
deletevariable(cluster_400m_nest_opp_ha)
assign(cluster_300m_nest_opp_ha, csum(cluster_300m, nestgebied_epr)*cellsize2/10000)
//write('gauwekiekendief_clusters_3000ha_300m_nest_opp_ha.asc', cluster_300m_nest_opp_ha)
deletevariable(cluster_300m_nest_opp_ha)
assign(cluster_200m_nest_opp_ha, csum(cluster_200m, nestgebied_epr)*cellsize2/10000)
//write('gauwekiekendief_clusters_3000ha_200m_nest_opp_ha.asc', cluster_200m_nest_opp_ha)
deletevariable(cluster_200m_nest_opp_ha)
assign(cluster_100m_nest_opp_ha, csum(cluster_100m, nestgebied_epr)*cellsize2/10000)
//write('gauwekiekendief_clusters_3000ha_100m_nest_opp_ha.asc', cluster_100m_nest_opp_ha)
deletevariable(cluster_100m_nest_opp_ha)
assign(cluster_400m_nest_fractie, csum(cluster_400m, nestgebied_epr)/csum(cluster_400m, 1))
//write('gauwekiekendief_clusters_3000ha_400m_nest_fractie.asc', cluster_400m_nest_fractie)
deletevariable(cluster_400m_nest_fractie)
assign(cluster_300m_nest_fractie, csum(cluster_300m, nestgebied_epr)/csum(cluster_300m, 1))
//write('gauwekiekendief_clusters_3000ha_300m_nest_fractie.asc', cluster_300m_nest_fractie)
deletevariable(cluster_300m_nest_fractie)
assign(cluster_200m_nest_fractie, csum(cluster_200m, nestgebied_epr)/csum(cluster_200m, 1))
//write('gauwekiekendief_clusters_3000ha_200m_nest_fractie.asc', cluster_200m_nest_fractie)

```

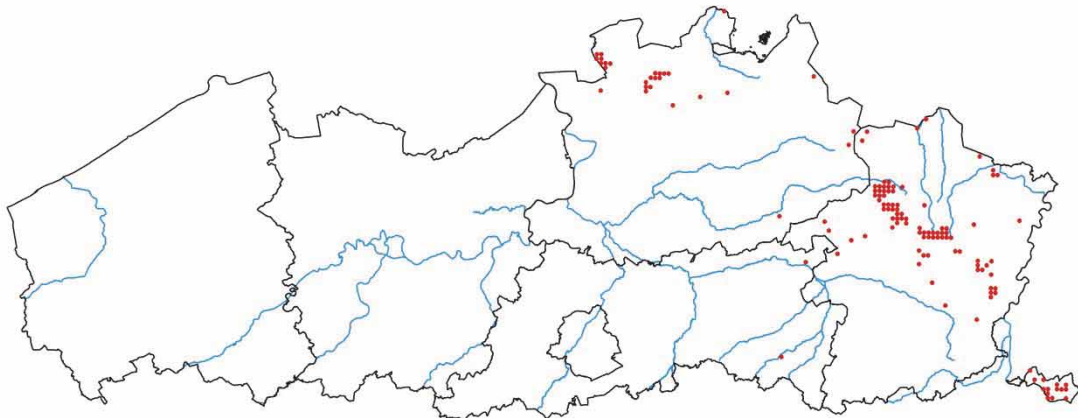
```

deletevariable(cluster_200m_nest_fractie)
assign(cluster_100m_nest_fractie, csum(cluster_100m, nestgebied_epr)/csum(cluster_100m, 1))
//write('grauwekiekendief_clusters_3000ha_100m_nest_fractie.asc', cluster_100m_nest_fractie)
deletevariable(cluster_100m_nest_fractie)
// Oppervlakte en aandeel randbiotoop
assign(cluster_400m_rand_opp_ha, csum(cluster_400m, randen)*cellsize2/10000)
//write('grauwekiekendief_clusters_3000ha_400m_rand_opp_ha.asc', cluster_400m_rand_opp_ha)
deletevariable(cluster_400m_rand_opp_ha)
assign(cluster_300m_rand_opp_ha, csum(cluster_300m, randen)*cellsize2/10000)
//write('grauwekiekendief_clusters_3000ha_300m_rand_opp_ha.asc', cluster_300m_rand_opp_ha)
deletevariable(cluster_300m_rand_opp_ha)
assign(cluster_200m_rand_opp_ha, csum(cluster_200m, randen)*cellsize2/10000)
//write('grauwekiekendief_clusters_3000ha_200m_rand_opp_ha.asc', cluster_200m_rand_opp_ha)
deletevariable(cluster_200m_rand_opp_ha)
assign(cluster_100m_rand_opp_ha, csum(cluster_100m, randen)*cellsize2/10000)
//write('grauwekiekendief_clusters_3000ha_100m_rand_opp_ha.asc', cluster_100m_rand_opp_ha)
deletevariable(cluster_100m_rand_opp_ha)
assign(cluster_400m_rand_fractie, csum(cluster_400m, randen)/csum(cluster_400m, 1))
//write('grauwekiekendief_clusters_3000ha_400m_rand_fractie.asc', cluster_400m_rand_fractie)
assign(cluster_300m_rand_fractie, csum(cluster_300m, randen)/csum(cluster_300m, 1))
//write('grauwekiekendief_clusters_3000ha_300m_rand_fractie.asc', cluster_300m_rand_fractie)
assign(cluster_200m_rand_fractie, csum(cluster_200m, randen)/csum(cluster_200m, 1))
//write('grauwekiekendief_clusters_3000ha_200m_rand_fractie.asc', cluster_200m_rand_fractie)
assign(cluster_100m_rand_fractie, csum(cluster_100m, randen)/csum(cluster_100m, 1))
//write('grauwekiekendief_clusters_3000ha_100m_rand_fractie.asc', cluster_100m_rand_fractie)
// wat met vereisten naar absolute of relatieve oppervlakte aan foerageer-, nest- en randbiotoop?
// clusters met minstens 5% voedselrijke randzones
assign(cluster_400m_rand_ge5, cfilter(cluster_400m, ge(cluster_400m_rand_fractie, 0.05)))
//write('grauwekiekendief_clusters_3000ha_400m_rand_ge5.asc', cluster_400m_rand_ge5)
deletevariable(cluster_400m_rand_ge5)
assign(cluster_300m_rand_ge5, cfilter(cluster_300m, ge(cluster_300m_rand_fractie, 0.05)))
//write('grauwekiekendief_clusters_3000ha_300m_rand_ge5.asc', cluster_300m_rand_ge5)
deletevariable(cluster_300m_rand_ge5)
assign(cluster_200m_rand_ge5, cfilter(cluster_200m, ge(cluster_200m_rand_fractie, 0.05)))
//write('grauwekiekendief_clusters_3000ha_200m_rand_ge5.asc', cluster_200m_rand_ge5)
deletevariable(cluster_200m_rand_ge5)
assign(cluster_100m_rand_ge5, cfilter(cluster_100m, ge(cluster_100m_rand_fractie, 0.05)))
//write('grauwekiekendief_clusters_3000ha_100m_rand_ge5.asc', cluster_100m_rand_ge5)
deletevariable(cluster_100m_rand_ge5)
write('grauwekiekendief_leefgebied.asc', cluster_200m)

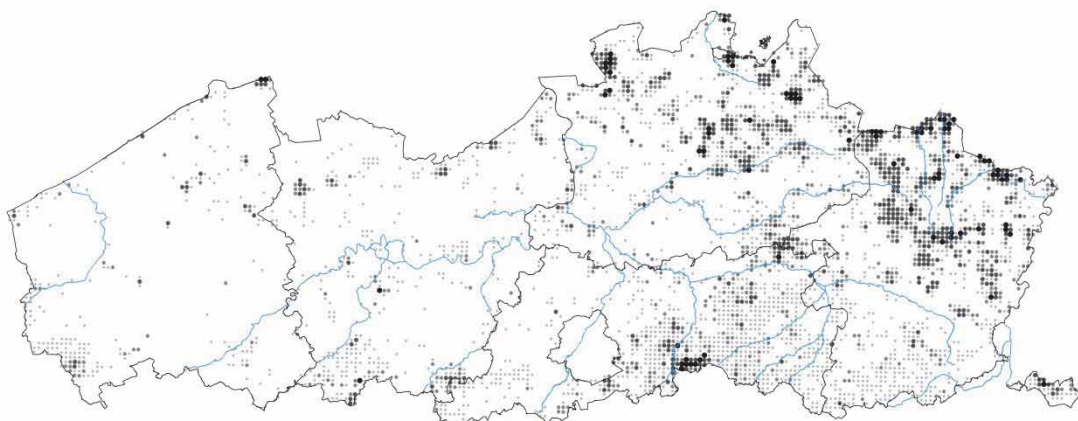
```



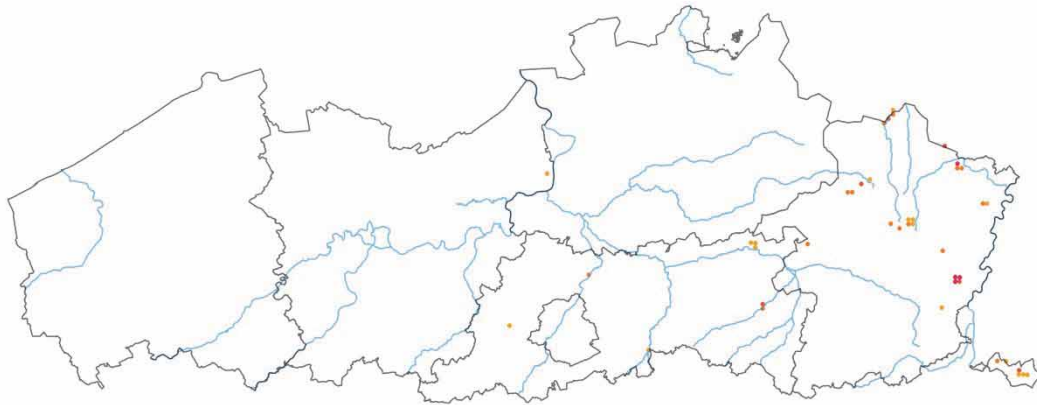
Potentiële leefgebiedenkaart voor Grauwe klauwier op basis van het onderstaande gdx-script.



Potentiële verspreidingskaart voor Grauwe klauwier op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Grauwe klauwier per kilometerhok in Vlaanderen (Tabel 7). Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Grauwe klauwier uit het data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

Gdx-script

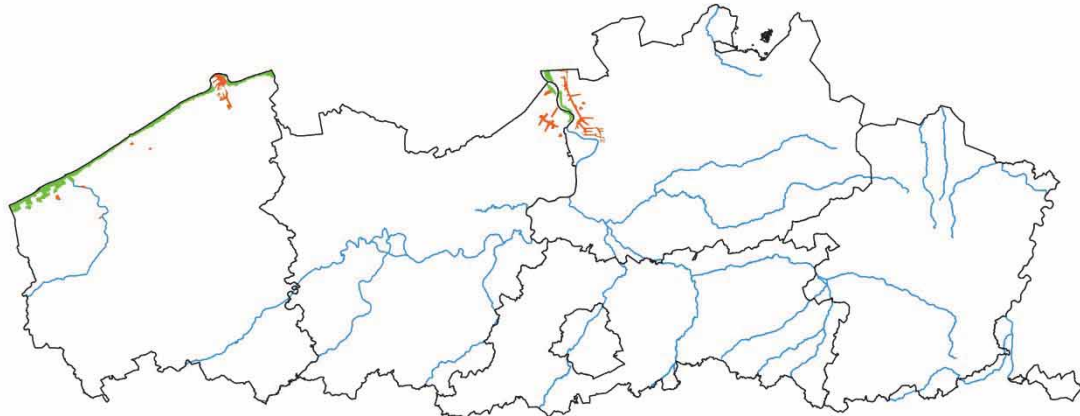
```
vlaanderen_20m.asc
// Externe lagen
assign(Groenkaart2013, 'groenkaart2013_20m.asc')
assign(wegen, 'GRB_Wegverbinding_at.asc')
// afstand tot bebouwing in rekening brengen, i.e. indien die clusters vormen van meer dan 10 ha (met
// tussenafstand van 50m)
assign(bebouwing, bwk(u, ua, ua-, uc, ud, ui, ui-, un, un-, ur, ur-, ur+, uv, uv-, uv+))
assign(bebouwing_id, FuzzyClusterId(bebouwing, 50/cellSize))
// assign(oppervlakte_bebouwing, cSum(bebouwing_id, 1))
assign(bebouwing_cluster_10ha_50m, cFilterGeArea(bebouwing_id, 10, hectare))
// write('grauweklauwier_bebouwing_cluster_10ha_50m.asc', bebouwing_cluster_10ha_50m)
assign(afstand_bebouwing, cellDistance(bebouwing_cluster_10ha_50m, 0))
deletevariable(bebouwing)
deletevariable(bebouwing_id)
deletevariable(bebouwing_cluster_10ha_50m)
// afstand tot grote wegen (voorstel ANB); enkel autosnelwegen en tweevaksbanen dan?
assign(afstand_wegen, cellDistance(or(eq(wegen, 101), eq(wegen, 102)), 0))
deletevariable(wegen)
// geschikte opgaande kleine landschapselementen (kle) uit Groenkaart2013
// lijnvormig hooggroen
assign(biotopen_kle_groenkaart_gr3m, edge(eq(Groenkaart2013, 1), gt(Groenkaart2013, 1)))
assign(biotopen_kle_groenkaart_gr3m, and(biotopen_kle_groenkaart_gr3m, ne(MajorityFilter(Groenkaart2013, 1,
2), 1)))
// write('grauweklauwier_biotopen_kle_groenkaart_gr3m.asc', biotopen_kle_groenkaart_gr3m)
// lijnvormig laaggroen dat geen landbouw is
assign(biotopen_kle_groenkaart_lt3m, edge(eq(Groenkaart2013, 2), and(ge(Groenkaart2013, 1),
ne(Groenkaart2013, 2))))
assign(biotopen_kle_groenkaart_lt3m, and(biotopen_kle_groenkaart_lt3m, ne(MajorityFilter(Groenkaart2013, 1,
2), 2)))
// write('grauweklauwier_biotopen_kle_groenkaart_lt3m.asc', biotopen_kle_groenkaart_lt3m)
// enkel lijnvormig hoog- en laaggroen meenemen indien overlap met bwk-codes bwk(kb%, kh%, kj%, kw%)
// niet verboste kernbiotopen, met verbossing afgeleid uit bwk; k() varianten nog meenemen? En enkel de
// kernbiotopen de afstand van minimaal 100m tot bebouwing en wegen in rekening brengen, of moet dit ook voor
// foerageerbiotoop?
// assign(kernbiotoop, and(bwk(ce, ce-, ce+, cg, cg+, cg-, ha%, hd%, hk%, hm%, hn%, hp+, hpr+, hu%, sd%, sg%, sk%,
// sp+, sp-, sz%, 2120%, 2130%, 2150%, 2160%, 2170%, 2310%, 2330%, 4010%, 4030%, 5130%, 6120%, 6210%,
// 6230%, 6410%, 6430%, 6510%, 7150%, rbsg%, rbsp%), not(bwk(91%, gml, gmn, f%, n, n+, n-, l, ls%, lh%, p%, q%,
// ru%))), gt(afstand_bebouwing, 100/cellSize), gt(afstand_wegen, 100/cellSize)))
```

```

// niet verboste kernbiotopen, met verbossing afgeleid uit groenkaart, waarbij bossen gedefinieerd als
// aaneengesloten oppervlakte aan hooggroen van meer dan 0.25 ha (excl. kle); En enkel de kernbiotopen op een
// afstand van minimaal 100m tot bebouwing en wegen in rekening brengen
assign(kernbiotoop, and(bwk(ce, ce-, ce+, cg, cg+, cg-, ha%, hd%, hk%, hm%, hn%, hp+, hpr+, hu%, sd%, sg%, sk%,
sp, sp+, sp-, sz%), not(ge(csum(clusterid(and(eq(Groenkaart2013, 1), not(biotopen_kle_groenkaart_gr3m))), 1),
0.25*10000/cellsize2))), not(bwk(91%, rbbppm%, rbbpf%, rbbso%))), gt(afstand_bebouwing, 100/cellsize),
gt(afstand_wegen, 100/cellsize)))
// write('grauweklauwier_kernbiotoop.asc', kernbiotoop)
deletevariable(afstand_bebouwing)
deletevariable(afstand_wegen)
// foerageerbiotopen omvatten de kernbiotopen naast nog een resem andere bwk-codes (niet verbost, met
// verbossing vanaf 0.25 ha aaneengesloten hooggroen volgens groenkaart 2013), én het lijnvormige hoog- en
// laaggroen uit de groenkaart 2013
assign(foerageerbiotoop, or(and(bwk(cd%, ce, ce-, ce+, ceb, ceb-, ceb+, cg%, cm%, cp%, cv%, dd%, dm%, ha%, hc%,
hd%, hf%, hj%, hk%, hm%, hn%, hp+, hpr+, hr%, hu%, ku%, kw%, mr%, sd%, sg%, sk%, sp, sp+, sp-, sz%, 2120%,
2130%, 2150%, 2160%, 2170%, 2190%, 2310%, 2330%, 3130%, 3270%, 4010%, 4030%, 5130%, 6120%, 6210%,
6230%, 6410%, 6430%, 6510%, 7140%, 7150%, rbbhc%, rbbhf%, rbbmr%, rbbsg%, rbbsp%),
not(ge(csum(clusterid(and(eq(Groenkaart2013, 1), not(biotopen_kle_groenkaart_gr3m))), 1),
0.25*10000/cellsize2))), not(bwk(91%, rbbppm%, rbbpf%, rbbso%))), biotopen_kle_groenkaart_gr3m,
biotopen_kle_groenkaart_lt3m))
// write('grauweklauwier_foerageerbiotoop.asc', foerageerbiotoop)
deletevariable(Groenkaart2013)
// clusters van 10 ha aaneengesloten kernbiotoop
assign(id_kernbiotoop, fuzzyclusterid(kernbiotoop, 20/cellsize))
assign(clusters_kernbiotoop, cfilterGeArea(id_kernbiotoop, 10, hectare))
// write('grauweklauwier_kernbiotoop_10ha_0m.asc', clusters_kernbiotoop)
// eis van minstens 500m aan lijnvormige kle in kernbiotoop
assign(clusters_kernbiotoop_kle, cfilter(clusters_kernbiotoop, ge(csum(clusters_kernbiotoop,
or(biotopen_kle_groenkaart_gr3m, biotopen_kle_groenkaart_lt3m)), 500/cellsize)))
// write('grauweklauwier_kernbiotoop_10ha_kle_gt500m.asc', clusters_kernbiotoop_kle)
deletevariable(biotopen_kle_groenkaart_gr3m)
deletevariable(biotopen_kle_groenkaart_lt3m)
// enkel foerageerbiotoop overhouden op een afstand van maximum 100m rond de 10 ha clusters van
// kernbiotopen, en het geheel van kern- en foerageerbiotoop moet minstens 15 ha groot zijn.
assign(id_clusters, fuzzyclusterid(and(le(celldistance(clusters_kernbiotoop_kle, 0), 100/cellsize), foerageerbiotoop),
100/cellsize))
// write('grauweklauwier_foerageerbiotoop_rond_kernbiotoop_100m.asc', id_clusters)
assign(clusters, cfilter(id_clusters, and(cfiltergearea(id_clusters, 15, hectare), ge(csum(id_clusters,
clusters_kernbiotoop_kle), 1))))
// write('grauweklauwier_clusters_15ha_100m.asc', clusters)
// Checken of de biotopen niet monotoon uit permanent cultuurgrasland bestaan (bwk-codes hp%). Eisen dat niet
// meer dan 90% van de oppervlakte mag bestaan uit permanent cultuurgrasland?
assign(clusters_hp_fractie, csum(clusters, and(foerageerbiotoop, bwk(hp%)))/csum(clusters, 1))
// write('grauweklauwier_clusters_15ha_100m_hp_fractie.asc', clusters_hp_fractie)
assign(clusters_hp, cfilter(clusters, lt(clusters_hp_fractie, 0.9)))
// write('grauweklauwier_clusters_15ha_100m_hp_lt90.asc', clusters_hp)
// leefgebieden nog beperken op basis van ecodistricten: Oost-Kempisch puinwaaierdistrict, Zuid-Kempisch
// heuveldistrict, Velpe-Getedistrict, Centraal-Kempisch rivier- en duinendistrict, Vochtig-Haspengouws leemdistrict,
// Roerdalslenkdistrict, Zandig Maasterrassendistrict, Voeren
assign(ecodistricten, or(and(ge('Ecoregio.asc', 13), le('Ecoregio.asc', 17)), oreq('Ecoregio.asc', 30, 31, 34, 35)))
assign(clusters_ecodistricten, cfilter(clusters_hp, ge(csum(clusters_hp, ecodistricten), 1)))
write('grauweklauwier_clusters_15ha_100m_hp_lt90_ecodistricten.asc', clusters_ecodistricten)

```

Grote stern (*Sterna sandvicensis*) – Adriaens & Vermeersch (in druk)



Potentiële leefgebiedenkaart voor Grote stern op basis van het onderstaande gdx-script.
Groen = voortplantingsgebied, oranje = foerageergebied.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(grotestern_voortplanting1, bwk(dd%, dl%, ds%, dz%))
// leefgebied mag geen open water of ruigte of moeras zijn
assign(openwater, bwk(ad%, ae%, aer%, aes%, aev%, ah%, am%, ao%, aom%, aoo%, ap%, apo%, app%, mz%))
assign(grotestern_voortplanting2, and(grotestern_voortplanting1, not(openwater)))
deleteVariable(grotestern_voortplanting1)
deleteVariable(openwater)
// leefgebied moeten minimaal 20ha groot zijn
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het cluster
assign(grotestern_voortplanting2_cluster, fuzzyClusterIdU(grotestern_voortplanting2, 50, meter))
deleteVariable(grotestern_voortplanting2)
assign(grotestern_voortplanting3, cFilterGeArea(grotestern_voortplanting2_cluster, 20, hectare))
deleteVariable(grotestern_voortplanting2_cluster)
// Zoek openwater in de kaarten van bwk, zee en waterlopen, die nog verrasterd moeten worden
assign(grotestern_openwater, bwk(ad%, ae%, ah%, ap%, zee%))
// waterlopen
assign(grotestern_waterlopen, OrEq('rastert_huetzon1.asc', 17, 19))
assign(grotestern_foerageer1, or(grotestern_openwater, grotestern_waterlopen))
deleteVariable(grotestern_waterlopen)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het openwaterscluster
assign(grotestern_foerageer1cl, fuzzyClusterIdU(grotestern_foerageer1, 50, meter))
deleteVariable(grotestern_foerageer1)
// Open water moet minstens 1 ha groot zijn
assign(grotestern_openwater, cFilterGeArea(grotestern_foerageer1cl, 1, hectare))
deleteVariable(grotestern_foerageer1cl)
// zoek voortplantingsgebieden die maximaal 5000 meter van groot open water liggen
assign(grotestern_voortplanting4, cFilter(grotestern_voortplanting3, cellDistanceLe(grotestern_openwater, 5000,
meter)))
assign(grotestern_foerageer4, cFilter(grotestern_openwater, cellDistanceLe(grotestern_voortplanting3, 5000,
meter)))
deleteVariable(grotestern_voortplanting3)
deleteVariable(grotestern_openwater)
assign(grotestern_leefgebied1, or(grotestern_voortplanting4, grotestern_foerageer4))
assign(grotestern_leefgebied1_cluster, FuzzyClusterIdU(grotestern_leefgebied1, 50, meter))
deleteVariable(grotestern_leefgebied1)
assign(grotestern_leefgebied, cFilterGeArea(grotestern_leefgebied1_cluster, 20, hectare))
//write('grotestern_leefgebied.asc', if(grotestern_leefgebied, grotestern_leefgebied1_cluster, 0))
//write('grotestern_leefgebied.asc', IsNonZero(grotestern_leefgebied))
```

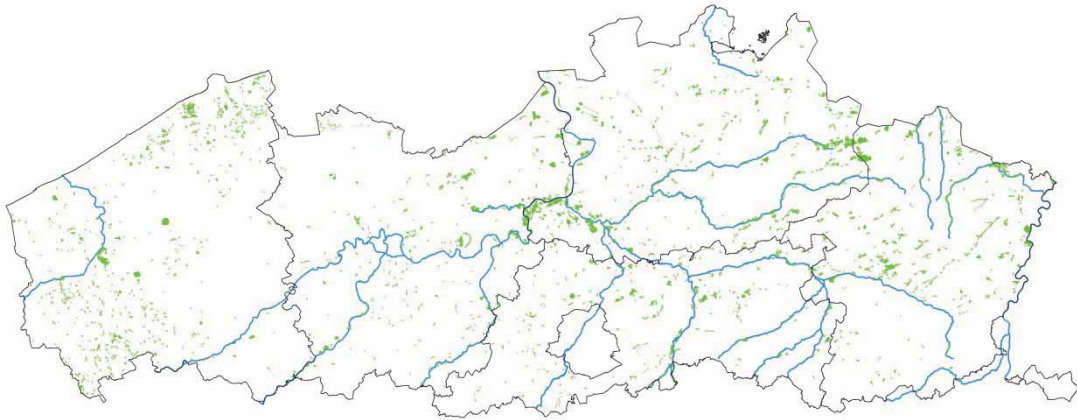
```
deleteVariable(grotestern_leefgebied1_cluster)
assign(grotestern_foerageer, and(grotestern_foerageer4, grotestern_leefgebied))
deleteVariable(grotestern_foerageer4)
write('grotestern_foerageer.asc', IsNonZero(grotestern_foerageer))
deleteVariable(grotestern_foerageer)
assign(grotestern_voortplanting, and(grotestern_voortplanting4, grotestern_leefgebied))
write('grotestern_voortplanting.asc', IsNonZero(grotestern_voortplanting))
deleteVariable(grotestern_voortplanting4)
deleteVariable(grotestern_voortplanting)
deleteVariable(grotestern_leefgebied)
```



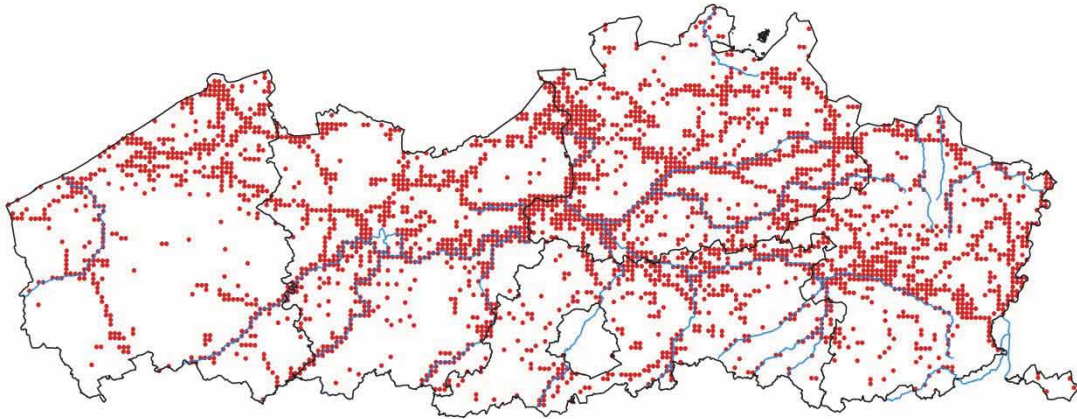
Grote stern (foto Yves Adams – Vilda)



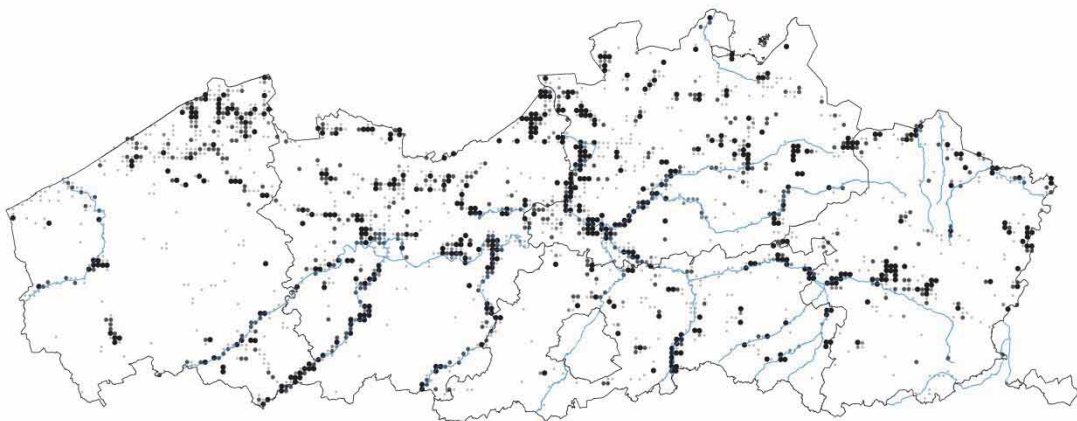
Zwarte specht (foto Glenn Vermeersch)



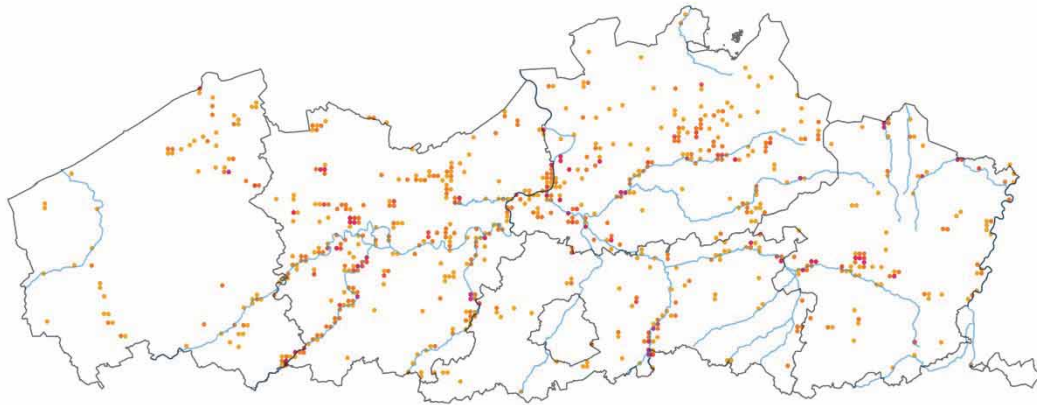
Potentiële leefgebiedenkaart voor Ijsvogel op basis van het onderstaande gdx-script.



Potentiële verspreidingskaart voor Ijsvogel op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Ijsvogel per kilometerhok in Vlaanderen (Tabel 7). Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.

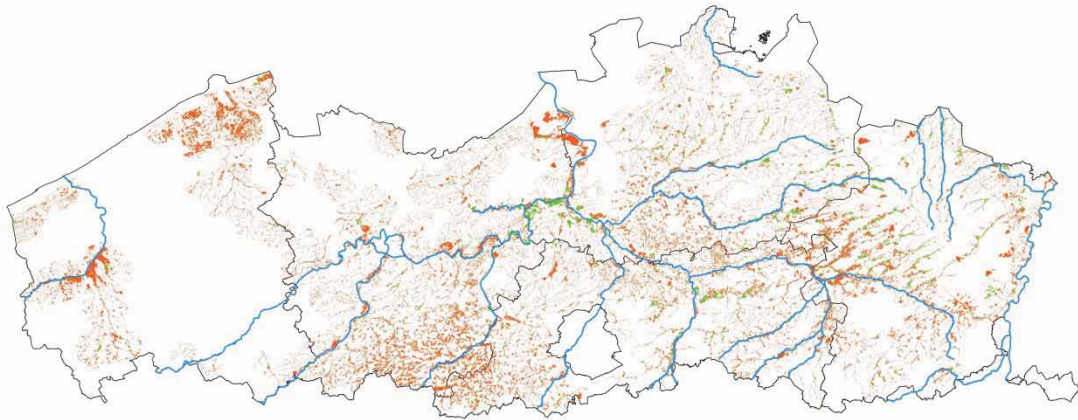


Waarnemingen van Ijsvogel uit het Natuurpunt data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

Gdx-script

```
vlaanderen_20m_met_maas.asc
// baken de leefgebieden af
assign(ijsvogel_bwk, bwk(ad, ad+, ae, ae+, aer, aer+, aev, aev+, ah, ah+, ap, ap+, apo, apo+, app, app+, ka%, kn%,
    mr%, mrb%, mru%, ru%, sf%, va%, vc%, vf%, vm%, vn%, vo%, vt%))
assign(ijsvogel_bwk_cluster, fuzzyClusterIdU(ijsvogel_bwk, 50, meter))
assign(ijsvogel_bwk1, cFilterGeArea(ijsvogel_bwk_cluster, 5, hectare))
// waterlopen
assign(ijsvogel_waterlopen, Ge('rastert_huetzon1.asc', 1))
assign(ijsvogel_maas, eq('maas.asc', 1))
assign(ijsvogel_leefgebied1, or(ijsvogel_bwk, ijsvogel_waterlopen, ijsvogel_maas))
deleteVariable(ijsvogel_bwk)
deleteVariable(ijsvogel_waterlopen)
deleteVariable(ijsvogel_maas)
// baken oevers af
assign(GRB_water, Eq('GRB_watergang_at.asc', 1))
assign(GRB_geenwater, Eq('GRB_watergang_at.asc', 0))
assign(ijsvogel_waterkant, edge(GRB_water, GRB_geenwater))
assign(ijsvogel_landkant, edge(GRB_geenwater, GRB_water))
deleteVariable(GRB_water)
deleteVariable(GRB_geenwater)
assign(ijsvogel_oever, or(ijsvogel_waterkant, ijsvogel_landkant))
deleteVariable(ijsvogel_waterkant)
deleteVariable(ijsvogel_landkant)
// oever moet 2 km lang zijn = 8 ha oever; bij 1 km is dat 4 ha
assign(ijsvogel_oever_cluster, fuzzyClusterIdU(ijsvogel_oever, 50, meter))
assign(ijsvogel_oever1, cFilterGeArea(ijsvogel_oever_cluster, 4, hectare))
assign(ijsvogel_leefgebied, cFilter(ijsvogel_bwk1, and(ijsvogel_bwk1, cellDistanceLe(ijsvogel_oever1, 100, meter))))
write('ijsvogel_leefgebied.asc', IsNonZero(ijsvogel_leefgebied))
```

Kleine zilverreiger (*Egretta garzetta*) – Adriaens & Vermeersch (in druk)



Potentiële leefgebiedenkaart voor Kleine zilverreiger op basis van het onderstaande gdx-script.
Groen = voortplantingsgebied, oranje = foerageergebied.

Gdx-script

```
vlaanderen_20m_met_maas.asc
// de eerste regel is een kaart t.b.v. de extend, resolutie en modelleringsgebied
// broedplaatsen: vochtige bossen
assign(kleinezilverreiger_voortplanting1, bwk(ru%, rud%, sf%, sm%, so%, va%, vc%, vf%, vm%, vn%, vo%, vt%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het cluster
assign(kleinezilverreiger_voortplanting1_cluster, fuzzyClusterIdU(kleinezilverreiger_voortplanting1, 50, meter))
deleteVariable(kleinezilverreiger_voortplanting1)
// voortplantingscluster moet minimaal 5 ha zijn
assign(kleinezilverreiger_voortplanting2, cFilterGeArea(kleinezilverreiger_voortplanting1_cluster, 5, hectare))
deleteVariable(kleinezilverreiger_voortplanting1_cluster)
assign(eikenbos, bwk(q%, f%))
assign(kleinezilverreiger_voortplanting3, and(kleinezilverreiger_voortplanting2, not(eikenbos)))
deleteVariable(kleinezilverreiger_voortplanting2)
deleteVariable(eikenbos)
// foerageergebieden:
// moerassen
assign(kleinezilverreiger_moerassen, bwk(da%, ds%, dz%, hc%, hcb%, hj%, hjb%, hp+%, hpr%, hpr+%, hu%, hub%,
    kc%, kn%, ku%, kub%, kz%, mc%, mcb%, md%, mdb%, mr%, mrb%, mru%, mz%))
// waterlopen
assign(kleinezilverreiger_waterlopen, OrEq('rastert_huetzon1.asc', 1, 2, 3, 4, 5, 6, 9, 11, 12, 13, 14, 15, 16, 17, 18,
    19))
assign(kleinezilverreiger_maas, OrEq('maas.asc', 1))
// open water
assign(kleinezilverreiger_openwater0, bwk(ae%, ah%, apo%, app, app+))
assign(kleinezilverreiger_openwater, or(kleinezilverreiger_openwater0, kleinezilverreiger_waterlopen,
    kleinezilverreiger_maas))
deleteVariable(kleinezilverreiger_openwater0)
assign(kleinezilverreiger_openwater_cluster, FuzzyClusterIdU(kleinezilverreiger_openwater, 50, meter))
deleteVariable(kleinezilverreiger_openwater)
assign(kleinezilverreiger_foerageer1, or(kleinezilverreiger_moerassen, kleinezilverreiger_waterlopen,
    kleinezilverreiger_maas))
deleteVariable(kleinezilverreiger_moerassen)
deleteVariable(kleinezilverreiger_maas)
deleteVariable(kleinezilverreiger_waterlopen)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het cluster
assign(kleinezilverreiger_foerageer1_cluster, fuzzyClusterIdU(kleinezilverreiger_foerageer1, 50, meter))
// clusters minimaal 200 ha
assign(kleinezilverreiger_foerageer2, cFilterGeArea(kleinezilverreiger_foerageer1_cluster, 200, hectare))
deleteVariable(kleinezilverreiger_foerageer1)
```

```

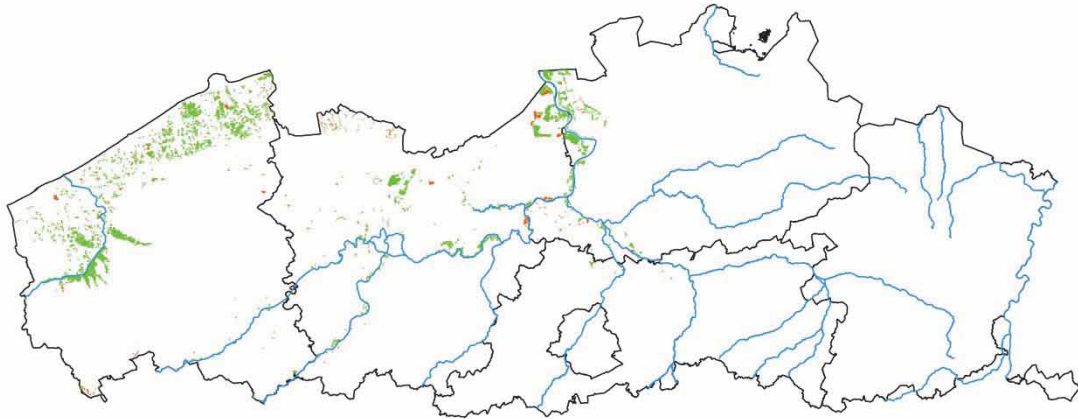
deleteVariable(kleinezilverreiger_foerageer1_cluster)
// foerageerclusters moet minimaal 40 ha open water bevatten, d.i. 20% van 200 ha
assign(kleinezilverreiger_foerageer3, cFilterGeArea(kleinezilverreiger_foerageer2, 40, hectare))
deleteVariable(kleinezilverreiger_openwater_cluster)
deleteVariable(kleinezilverreiger_foerageer2)
// geschikte foerageergebieden liggen minder dan 5 km van voortplantingsgebieden en omgekeerd
assign(kleinezilverreiger_foerageer4, and(kleinezilverreiger_foerageer3,
  cellDistanceLe(kleinezilverreiger_voortplanting3, 5000, meter)))
assign(kleinezilverreiger_voortplanting4, and(kleinezilverreiger_voortplanting3,
  cellDistanceLe(kleinezilverreiger_foerageer3, 5000, meter)))
deleteVariable(kleinezilverreiger_voortplanting3)
deleteVariable(kleinezilverreiger_foerageer3)
assign(kleinezilverreiger_leefgebied1, or(kleinezilverreiger_voortplanting4, kleinezilverreiger_foerageer4))
assign(kleinezilverreiger_leefgebied1_cluster, FuzzyClusterIdU(kleinezilverreiger_leefgebied1, 50, meter))
deleteVariable(kleinezilverreiger_leefgebied1)
assign(kleinezilverreiger_leefgebied, cFilterGeArea(kleinezilverreiger_leefgebied1_cluster, 200, hectare))
// write('kleinezilverreiger_leefgebied.asc', if(kleinezilverreiger_leefgebied, kleinezilverreiger_leefgebied1_cluster,
  0))
//write('kleinezilverreiger_leefgebied.asc', IsNonZero(kleinezilverreiger_leefgebied))
deleteVariable(kleinezilverreiger_leefgebied1_cluster)
assign(kleinezilverreiger_foerageer, and(kleinezilverreiger_foerageer4, kleinezilverreiger_leefgebied))
deleteVariable(kleinezilverreiger_foerageer4)
write('kleinezilverreiger_foerageer.asc', kleinezilverreiger_foerageer)
deleteVariable(kleinezilverreiger_foerageer)
assign(kleinezilverreiger_voortplanting, and(kleinezilverreiger_voortplanting4, kleinezilverreiger_leefgebied))
deleteVariable(kleinezilverreiger_voortplanting4)
deleteVariable(kleinezilverreiger_leefgebied)
write('kleinezilverreiger_voortplanting.asc', kleinezilverreiger_voortplanting)
deleteVariable(kleinezilverreiger_voortplanting)

```

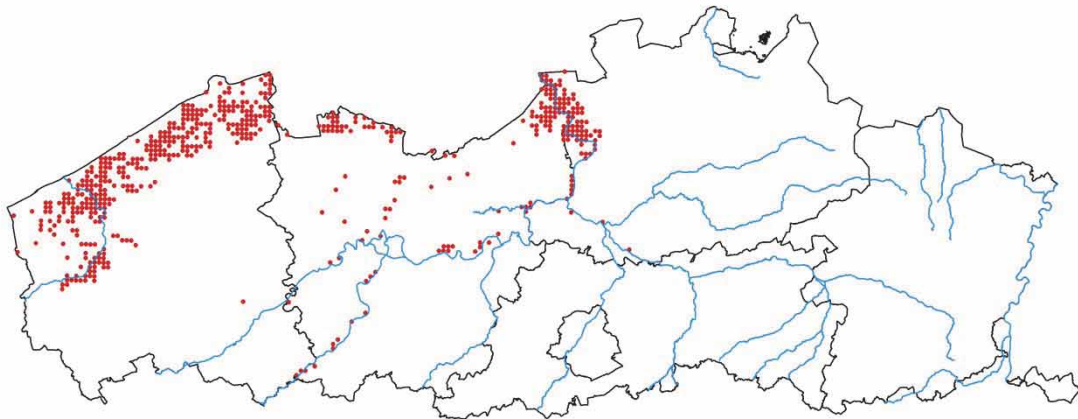


Kleine zilverreiger (foto Yves Adams – Vilda)

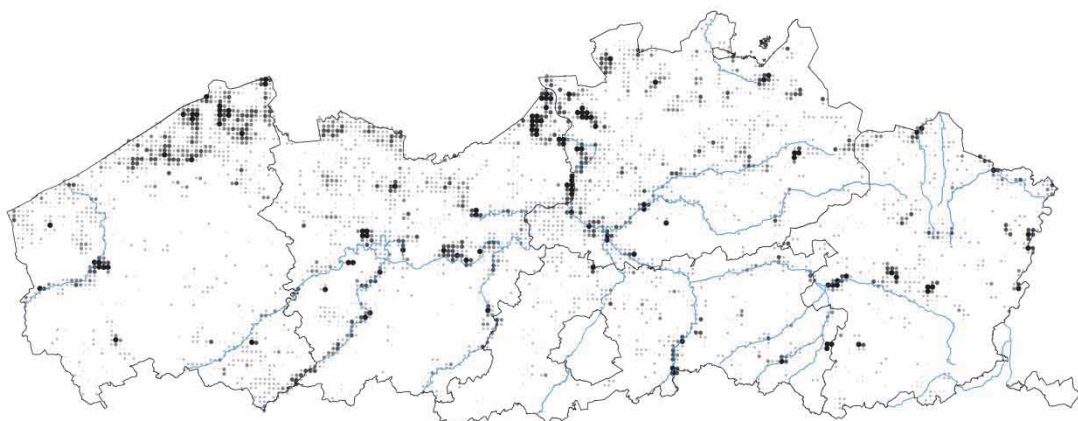
Kluut (*Recurvirostra avosetta*) – Adriaens & Vermeersch (in druk)



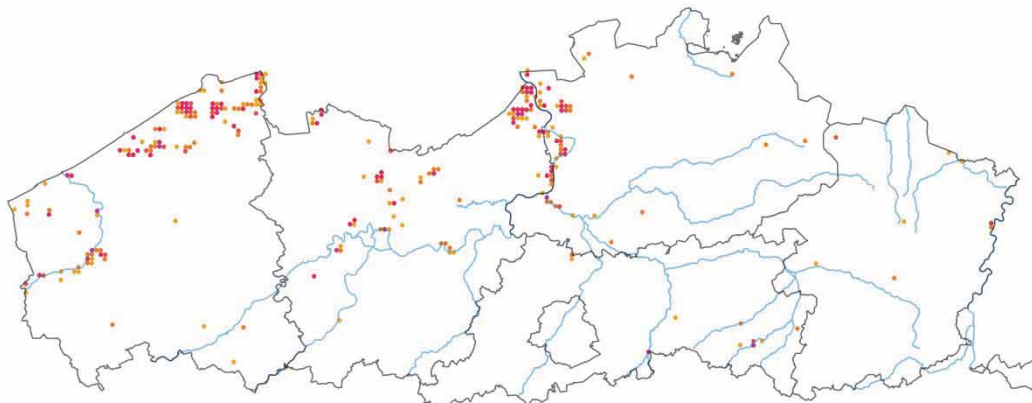
Potentiële leefgebiedenkaart voor Kluut op basis van het onderstaande gdx-script.
Groen = voortplantingsgebied, oranje = foerageergebied.



Potentiële verspreidingskaart voor Kluut op basis van correlatieve modellen.



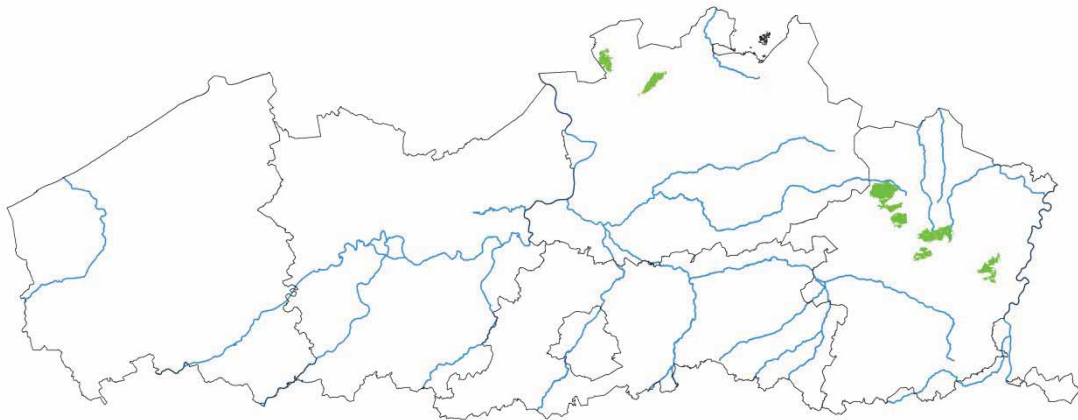
Som van de indicatorwaarde van begeleidende soorten van Kluut per kilometerhok in Vlaanderen (Tabel 7). Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Kluut uit het data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(kluut_voortplanting1, bwk(da%, ds%, hpr%, ku%, kz%))
// leefgebied mag geen open water of ruigte of moeras zijn
assign(openwater, bwk(ad%, ae%, aer%, aes%, aev%, ah%, am%, ao%, aom%, aoo%, ap%, apo%, app%, mz%))
assign(kluut_voortplanting2, and(kluut_voortplanting1, not(openwater)))
deleteVariable(kluut_voortplanting1)
deleteVariable(openwater)
// clustering: indien geschikte cel binnen 10 meter, dan hoort die cel bij het cluster
assign(kluut_voortplanting2_cluster, fuzzyClusterIdU(kluut_voortplanting2, 50, meter))
deleteVariable(kluut_voortplanting2)
// voortplantingsclusters moeten minimaal 5ha groot zijn
assign(kluut_voortplanting3, cFilterGeArea(kluut_voortplanting2_cluster, 5, hectare))
deleteVariable(kluut_voortplanting2_cluster)
// voortplantingscluster moet vrij dicht bij open water en vochtige weilanden liggen
assign(kluut_foerageer1, bwk(aer%, ah%, da%, ds%, hpr%, hpr+%, kn%, ku%, kz%))
// clustering: indien geschikte cel binnen 10 meter, dan hoort die cel bij het cluster
assign(kluut_foerageer1_cluster, fuzzyClusterIdU(kluut_foerageer1, 10, meter))
// foerageerclusters moeten minimaal 5ha groot zijn
assign(kluut_foerageer2, cFilterGeArea(kluut_foerageer1_cluster, 5, hectare))
// geschikte foerageergebieden liggen minder dan 2 km van voortplantingsgebieden en omgekeerd
assign(kluut_foerageer4, cFilter(kluut_foerageer2, cellDistanceLe(kluut_voortplanting3, 2000, meter)))
assign(kluut_voortplanting4, cFilter(kluut_voortplanting3, cellDistanceLe(kluut_foerageer2, 2000, meter)))
assign(kluut_leefgebied1, or(kluut_voortplanting4, kluut_foerageer4))
assign(kluut_leefgebied1_cluster, FuzzyClusterIdU(kluut_leefgebied1, 50, meter))
assign(kluut_leefgebied, cFilterGeArea(kluut_leefgebied1_cluster, 5, hectare))
// write('kluut_leefgebied.asc', if(kluut_leefgebied, kluut_leefgebied1_cluster, 0))
assign(kluut_foerageer, and(kluut_foerageer4, kluut_leefgebied))
// write('kluut_foerageer.asc', IsNonZero(kluut_foerageer))
assign(kluut_voortplanting, and(kluut_voortplanting4, kluut_leefgebied))
// write('kluut_voortplanting.asc', kluut_voortplanting)
assign(kluut_ecodistrict, OrEq('ecoregio.asc', 1, 2, 3, 4))
assign(kluut_voortplanting_ecodistrict, and(kluut_voortplanting, kluut_ecodistrict))
write('kluut_voortplanting_ecodistrict.asc', IsNonZero(kluut_voortplanting_ecodistrict))
assign(kluut_foerageer_ecodistrict, and(kluut_foerageer, kluut_ecodistrict))
write('kluut_foerageer_ecodistrict.asc', IsNonZero(kluut_foerageer_ecodistrict))
```



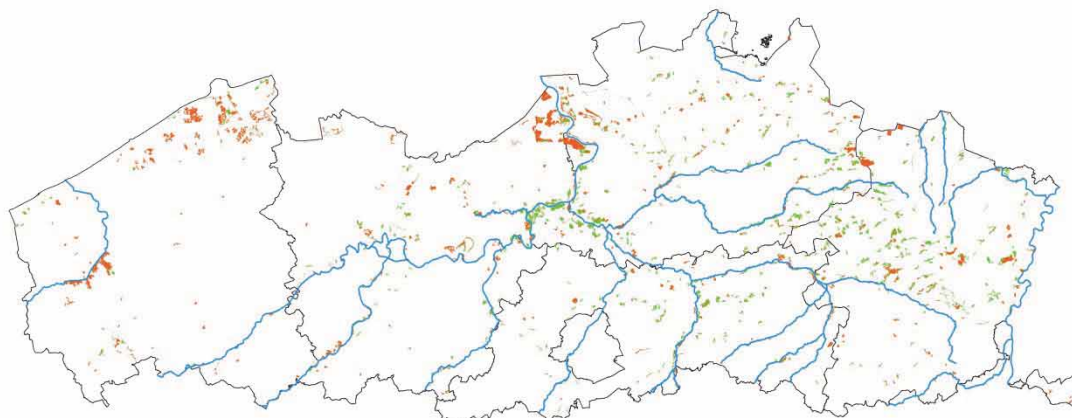
Potentiële leefgebiedenkaart voor Korhoen op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc  
// baken de voortplantingsgebieden af  
assign(korhoen_voortplanting, bwk(cd%, ce%, cg%, cm%, cp%, cv%, sg%, sm%, t%))  
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het voortplantingscluster  
assign(korhoen_leefgebied1_cluster, fuzzyClusterIdU(korhoen_voortplanting, 50, meter))  
// leefgebied moet groot genoeg zijn >300 ha  
assign(korhoen_leefgebied, cFilterGeArea(korhoen_leefgebied1_cluster, 300, hectare))  
write('korhoen_leefgebied.asc', IsNonZero(korhoen_leefgebied))
```



Korhoen (foto Rollin Verlinde – Vilda)



Potentiële leefgebiedenkaart voor Kwak op basis van het onderstaande gdx-script.
Groen = voortplantingsgebied, oranje = foerageergebied.

Gdx-script

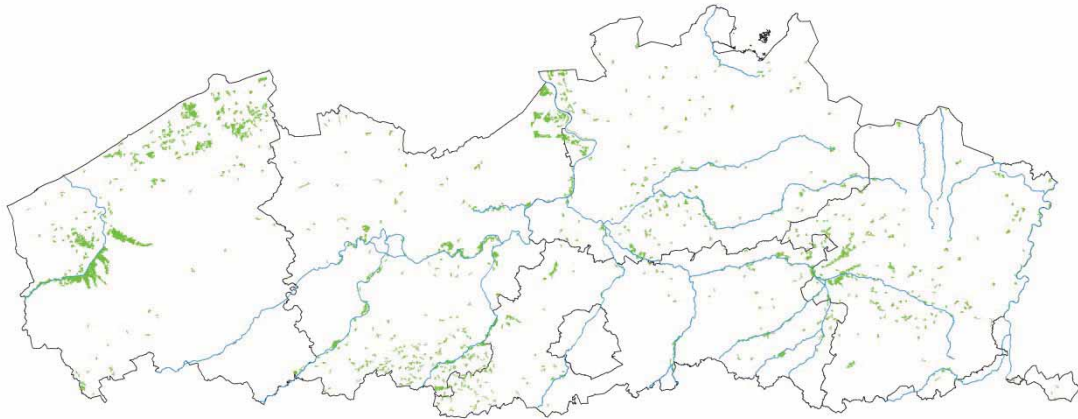
```
vlaanderen_20m.asc
// broedplaatsen: vochtige bossen
assign(kwak_voortplanting0, bwk(ru%, rud%, sf%, sm%, so%, sz%, va%, vc%, vf%, vm%, vn%, vo%, vt%, 91E0%,
  91F0%, rbbm%, rbbf%, rbbso%))
// clustering: indien geschikte cel binnen 100 meter, dan hoort die cel bij het cluster
assign(kwak_voortplanting0_cluster, fuzzyClusterIdU(kwak_voortplanting0, 100, meter))
deleteVariable(kwak_voortplanting0)
// clusters minimaal 5 ha
assign(kwak_voortplanting2, cFilterGeArea(kwak_voortplanting0_cluster, 5, hectare))
deleteVariable(kwak_voortplanting0_cluster)
assign(eikenbos, bwk(q%, f%))
assign(kwak_voortplanting3, and(kwak_voortplanting2, not(eikenbos)))
deleteVariable(kwak_voortplanting2)
deleteVariable(eikenbos)
// foerageergebieden:
// moerassen
assign(kwak_moeras, bwk(ad%, ae%, aer%, aev%, ah%, aom%, apo%, hf, hf+, hpr+, kn%, ku%, kub%, kz%, mc%,
  mcb%, md%, mdb%, mr%, mrb%, mru%, mz%, 1330%, 2180%, 2190%, 3110%, 3130%, 3140%, 3150%, 3160%,
  3260%, 3270%, 6430%, 7140%, 7230%, rbbmc%, rbbmr%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het cluster
assign(kwak_moeras_cluster, fuzzyClusterIdU(kwak_moeras, 50, meter))
deleteVariable(kwak_moeras)
// moerasclusters minimaal 10 ha
assign(kwak_moeras1, cFilterGeArea(kwak_moeras_cluster, 10, hectare))
deleteVariable(kwak_moeras_cluster)
// open water
assign(kwak_openwater, bwk(ad%, ae%, aev%, ah%, aom%, apo%))
assign(kwak_openwater_cluster, FuzzyClusterIdU(kwak_openwater, 50, meter))
deleteVariable(kwak_openwater)
// moerascluster moet minimaal 20% open water hebben, d.i. 2 ha
assign(kwak_foerageer1, cFilterGeArea(kwak_moeras1, 2, hectare))
deleteVariable(kwak_moeras1)
deleteVariable(kwak_openwater_cluster)
// geschikte foerageergebieden liggen minder dan 5 km van voortplantingsgebieden en omgekeerd
assign(kwak_voortplanting1, and(kwak_voortplanting3, cellDistanceLe(kwak_foerageer1, 5000, meter)))
assign(kwak_foerageer1, and(kwak_foerageer1, cellDistanceLe(kwak_voortplanting3, 5000, meter)))
deleteVariable(kwak_voortplanting3)
assign(kwak_leefgebied1, or(kwak_voortplanting1, kwak_foerageer1))
```

```
assign(kwak_leefgebied1_cluster, FuzzyClusterIdU(kwak_leefgebied1, 50, meter))
deleteVariable(kwak_leefgebied1)
assign(kwak_leefgebied, cFilterGeArea(kwak_leefgebied1_cluster, 10, hectare))
//write('kwak_leefgebied.asc', if(kwak_leefgebied, kwak_leefgebied1_cluster, 0))
write('kwak_leefgebied.asc', IsNonZero(kwak_leefgebied))
deleteVariable(kwak_leefgebied1_cluster)
assign(kwak_foerageer, and(kwak_foerageer1, kwak_leefgebied))
deleteVariable(kwak_foerageer1)
write('kwak_foerageer.asc', IsNonZero(kwak_foerageer))
deleteVariable(kwak_foerageer)
assign(kwak_voortplanting, and(kwak_voortplanting1, kwak_leefgebied))
deleteVariable(kwak_voortplanting1)
deleteVariable(kwak_leefgebied)
write('kwak_voortplanting.asc', kwak_voortplanting)
deleteVariable(kwak_voortplanting)
```

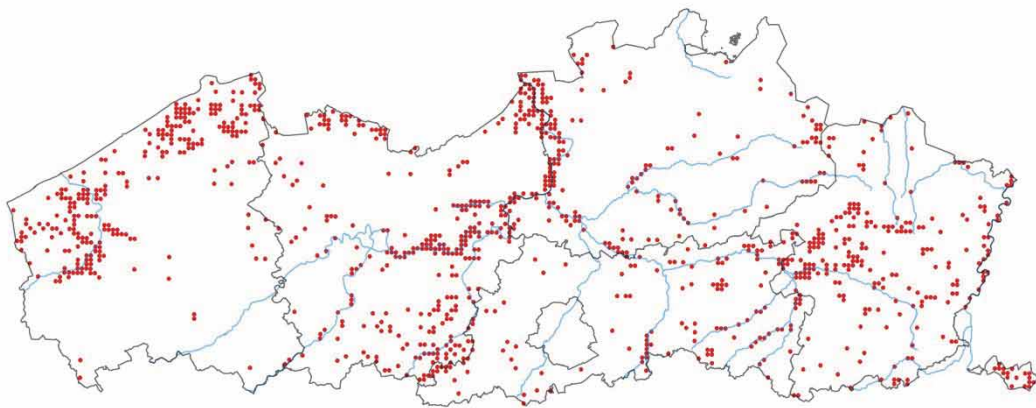


Kwak (foto Koen Devos)

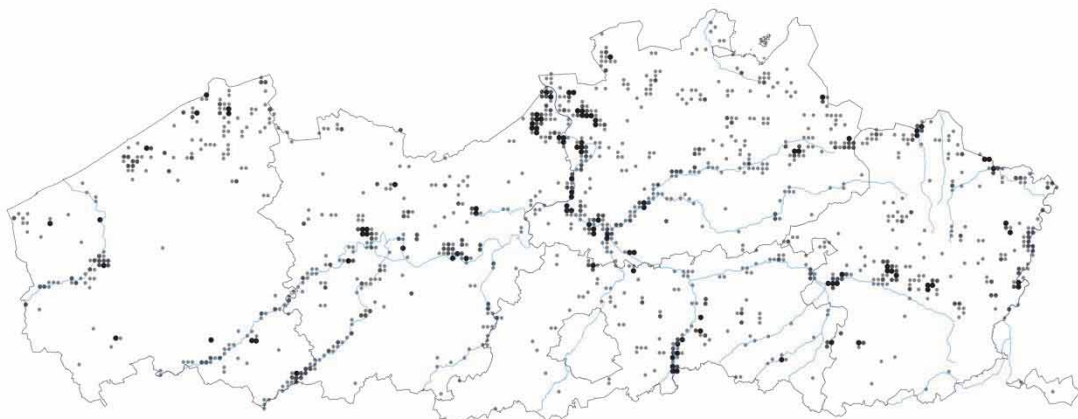
Kwartelkoning (*Crex crex*) – Adriaens & Vermeersch (in druk)



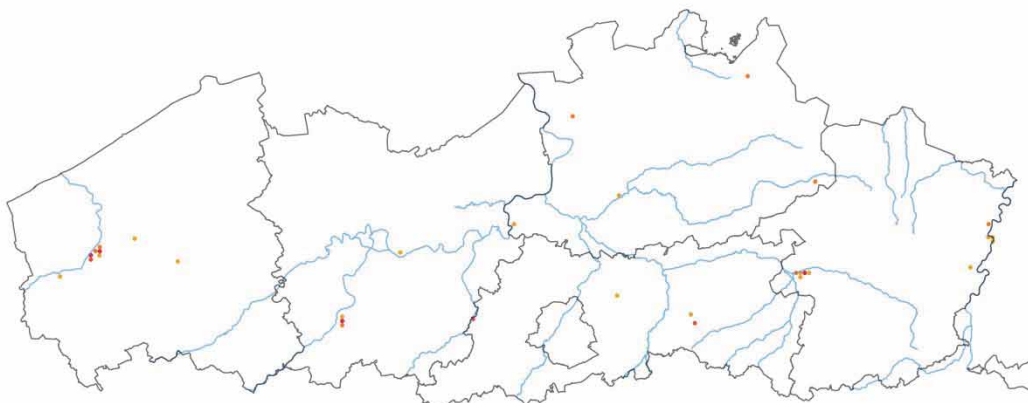
Potentiële leefgebiedenkaart voor Kwartelkoning op basis van het onderstaande gdx-script.



Potentiële verspreidingskaart voor Kwartelkoning op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Kwartelkoning per kilometerhok in Vlaanderen (Tabel 7). Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Kwartelkoning uit de Natuurpunt data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

Gdx-script

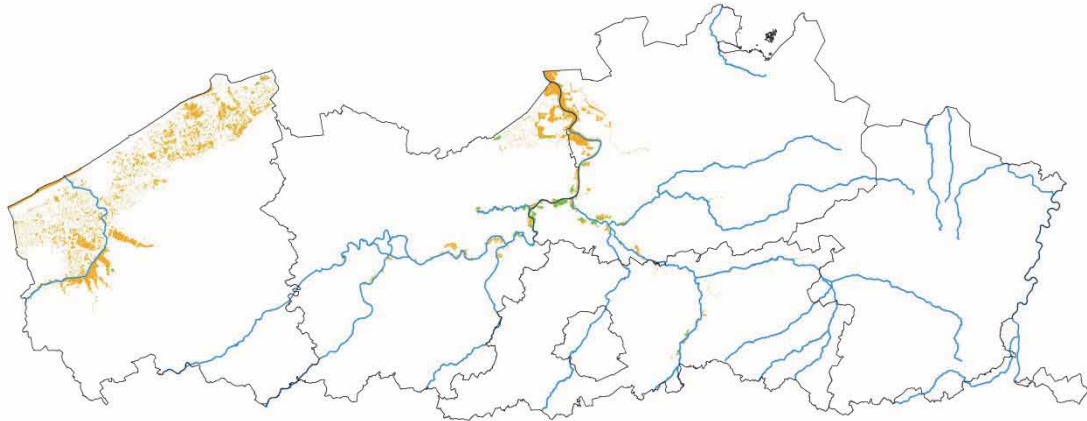
```
vlaanderen_20m.asc
// Externe lagen
assign(G, 'groenkaart2013_20m.asc')
assign(P, 'bodem_profiel.asc')
assign(D, 'bodem_drainage.asc')
assign(O, 'OVSTRGEV2014_20m.asc')
// Verbossing
// assign(verbossing, bwk(91%, gml, gmn, f%, n, n+, n-, l, ls%, lh%, p%, q%, ru%, v%, s%, kb%, kh%))
// minder streng kan alle hoog groen (>3m) uit de groenkaart2013 getrokken worden
assign(verbossing, or(eq(G, 1), bwk(91%)))
deletevariable(G)
// write('kwartelkoning_biotopen_verbossing_tris.asc', verbossing)
// Definitie geschikte biotopen
assign(grasland, and(bwk(rbbhc%, rbbhf%, rbbkam%, rbbzil%, 6120%, 6410%, 6510%, 6430%, hc, hc-, hc+, hf, hf-,
  hf+, hfc, hfc-, hfc+, hft, hft-, hft+, hj, hj+, hj-, hm, hm-, hm+, hme, hme-, hme+, hmm, hmm-, hmm+, hp+, hpr, hpr-,
  hpr+, hr, hr-, hr+, hu, hu-, hu+, ku, ku-, ku+), not(bwk(kg%)), not(verbossing)))
// write('kwartelkoning_biotopen_grasland_bis.asc', grasland)
assign(moerasvegetatie, and(bwk(2190%, 7140%, 7230%, rbbmc%, rbbmr%, rbbms%, mc-, mc+, mk, mk-, mk+, mp,
  mp-, mp+, mr, mr-, mr+, ms, ms-, ms+), not(grasland), not(verbossing)))
// write('kwartelkoning_biotopen_moerasvegetatie_tris.asc', moerasvegetatie)
// zeer geschikte biotopen:
// assign(kernbiotopen, and(bwk(hc, hc-, hc+, rbbhc%, hu, hu+, 6120%, 6510%, 2190%, 6410%), not(verbossing)))
// write('kwartelkoning_biotopen_kernbiotopen_tris.asc', kernbiotopen)
assign(kwartelkoning_geschikte_biotopen, or(grasland, moerasvegetatie))
// Beperking van geschikte biotopen tot de valleigronden en overstromingsgevoelige gebieden (profielontwikkeling
  p, of drainageklassen natter of gelijk aan d, of effectief of mogelijks overstromingsgevoelig)
assign(kwartelkoning_geschikte_biotopen_vallei, and(kwartelkoning_geschikte_biotopen,
  or(reclass('bodem_profiel_geenprofielontwikkeling.tab', P), reclass('bodem_drainage_d_en_natter.tab', D), ge(O,
  1))))
// write('kwartelkoning_geschikte_biotopen_valleigrond_tris.asc', kwartelkoning_geschikte_biotopen_vallei)
deletevariable(P)
deletevariable(D)
deletevariable(O)
// Wegfilteren van lintvormige biotopen (wegbermen autostraden, kanalen, etc.)
assign(kwartelkoning_geschikte_biotopen_filter_lint, MajorityFilter(kwartelkoning_geschikte_biotopen + 1, 1, 2) -
  1)
// write('kwartelkoning_geschikte_biotopen_majorityfilter_r2_tris.asc',
  kwartelkoning_geschikte_biotopen_filter_lint)
// wegfilteren van de clusters kleiner dan 3 ha
assign(x, clusterid(kwartelkoning_geschikte_biotopen))
```

```

assign(kwartelkoning_geschikte_biotopen_ge3ha, isnonzero(cfiltergearea(x, 3, hectare)))
// write('kwartelkoning_geschikte_biotopen_ge3ha_tris.asc', kwartelkoning_geschikte_biotopen_ge3ha)
// wegfilteren van clusters die geen enkele rastercel bevatten die op meer dan 100m ligt van hooggroen dat
// minstens uit 5 cellen bestaat (i.e. 100m lijnvormig of 20 are in oppervlakte) op een maximale onderlinge afstand
// van 20 meter; cfr. eis 200 m open zicht
assign(hooggroen_clusters_20m, fuzzyclusterid(verbossing, 20/cells))
deletevariable(verbossing)
assign(hooggroen_clusters_5_cells, cfiltergearea(hooggroen_clusters_20m, 5, cells))
deletevariable(hooggroen_clusters_20m)
assign(kwartelkoning_geschikte_biotopen_filter_openzicht, isnonzero(cfilter(x, ge(csum(x,
ge(celldistance(hooggroen_clusters_5_cells, 0), 100/cells)), 1))))
// write('kwartelkoning_geschikte_biotopen_hooggroen_ge100m_tris.asc',
// kwartelkoning_geschikte_biotopen_filter_openzicht)
// geschikt biotoop, exclusief hooggroen met een oppervlakte groter dan 20 are
assign(kwartelkoning_geschikte_biotopen_excl_hooggroen, and(kwartelkoning_geschikte_biotopen,
not(isnonzero(hooggroen_clusters_5_cells))))
deletevariable(hooggroen_clusters_5_cells)
// write('kwartelkoning_geschikte_biotopen_excl_hooggroen_tris.asc',
// kwartelkoning_geschikte_biotopen_excl_hooggroen)
// compileren van geschikte biotopen à la mode; nu moet geschikt voldoen aan al de bovenstaande eisen:
assign(kwartelkoning_geschikte_biotopen, and(kwartelkoning_geschikte_biotopen_vallei,
kwartelkoning_geschikte_biotopen_filter_lint, kwartelkoning_geschikte_biotopen_ge3ha,
kwartelkoning_geschikte_biotopen_filter_openzicht, kwartelkoning_geschikte_biotopen_excl_hooggroen))
// write('kwartelkoning_geschikte_biotopen_tris.asc', kwartelkoning_geschikte_biotopen)
deletevariable(kwartelkoning_geschikte_biotopen_vallei)
deletevariable(kwartelkoning_geschikte_biotopen_filter_lint)
deletevariable(kwartelkoning_geschikte_biotopen_ge3ha)
deletevariable(kwartelkoning_geschikte_biotopen_filter_openzicht)
deletevariable(kwartelkoning_geschikte_biotopen_excl_hooggroen)
// clustering
assign(id, fuzzyClusterId(kwartelkoning_geschikte_biotopen, 50/cellSize))
// cluster minimaal 10 ha
assign(y, cFiltergearea(id, 10, hectare))
assign(id, id*isnonzero(y))
// write('kwartelkoning_clusters_50m_10ha_tris.asc', y)
// Aandeel grasland minstens 75%
assign(grasland_fractie, cSum(id, grasland) / cSum(id, 1))
// write('kwartelkoning_clusters_50m_10ha_grasland_fractie_tris.asc', grasland_fractie)
assign(y, and(y, cFilter(id, ge(grasland_fractie, 0.75))))
// write('kwartelkoning_clusters_50m_10ha_grasland_ge75_tris.asc', y * id)
// Aandeel moerasvegetatie lager dan 25%
assign(moeras_fractie, cSum(id, moerasvegetatie) / cSum(id, 1))
// write('kwartelkoning_clusters_50m_10ha_moeras_fractie_tris.asc', moeras_fractie)
assign(y, and(y, cFilter(id, lt(moeras_fractie, 0.25))))
write('kwartelkoning_clusters_50m_10ha_grasland_ge75_moeras_lt25_tris.asc', y * id)

```

Lepelaar (*Platalea leucorodia*) – Adriaens & Vermeersch (in druk)



Potentiële leefgebiedenkaart voor Lepelaar op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m_met_maas.asc
// broedplaatsen: vochtige bossen
assign(lepelaar_voortplanting_bwk, bwk(ru%, rud%, sf%, sm%, so%, va%, vc%, vf%, vm%, vn%, vo%, vt%, 91E0%,
91F0%, rbbsf%, rbbsm%, rbbso%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het cluster
assign(lepelaar_voortplanting_cluster, fuzzyClusterIdU(lepelaar_voortplanting_bwk, 50, meter))
deleteVariable(lepelaar_voortplanting_bwk)
// clusters minimaal 10 ha
assign(lepelaar_voortplanting2, cFilterGeArea(lepelaar_voortplanting_cluster, 10, hectare))
deleteVariable(lepelaar_voortplanting_cluster)
// foerageergebieden:
// moerassen
assign(lepelaar_moeras, bwk(ae%, ah%, ds%, hpr%, hpr+%, ku%, kz%, 1130%, 1140%, 1310%, 1320%, 1330%,
3130%, 3140%, 3150%, 3270%))
// waterlopen
assign(lepelaar_waterlopen, OrEq('rastert_huetzon1.asc', 1, 2, 11, 12, 13, 14, 15, 16, 17, 18, 19))
assign(lepelaar_maas, OrEq('maas_20m.asc', 1))
assign(lepelaar_foerageer0, or(lepelaar_moeras, lepelaar_waterlopen, lepelaar_maas))
deleteVariable(lepelaar_maas)
deleteVariable(lepelaar_moeras)
deleteVariable(lepelaar_waterlopen)
// clustering: indien geschikte cel binnen 100 meter, dan hoort die cel bij het cluster
assign(lepelaar_foerageer0_cluster, fuzzyClusterIdU(lepelaar_foerageer0, 50, meter))
deleteVariable(lepelaar_foerageer0)
// moerasclusters minimaal 2500 ha
assign(lepelaar_foerageer01, cFilterGeArea(lepelaar_foerageer0_cluster, 2500, hectare))
deleteVariable(lepelaar_foerageer0_cluster)
// open water
assign(lepelaar_openwater, bwk(ae%, ah%, 1310%, 1320%, 1330%, 3130%, 3140%, 3150%))
//assign(lepelaar_openwater_cluster, FuzzyClusterIdU(lepelaar_openwater, 50, meter))
// moerascluster moet minimaal 20% open water hebben
assign(lepelaar_foerageer2, cFilterGeArea(lepelaar_foerageer01, 500, hectare))
deleteVariable(lepelaar_openwater)
deleteVariable(lepelaar_foerageer01)
// geschikte foerageergebieden liggen minder dan 15 km van voortplantingsgebieden en omgekeerd
assign(lepelaar_voortplanting3, and(lepelaar_voortplanting2, cellDistanceLe(lepelaar_foerageer2, 15000, meter)))
assign(lepelaar_foerageer3, and(lepelaar_foerageer2, cellDistanceLe(lepelaar_voortplanting2, 15000, meter)))
deleteVariable(lepelaar_foerageer2)
deleteVariable(lepelaar_voortplanting2)
assign(lepelaar_leefgebied0, or(lepelaar_voortplanting3, lepelaar_foerageer3))
assign(lepelaar_leefgebied0_cluster, FuzzyClusterIdU(lepelaar_leefgebied0, 50, meter))
```

```

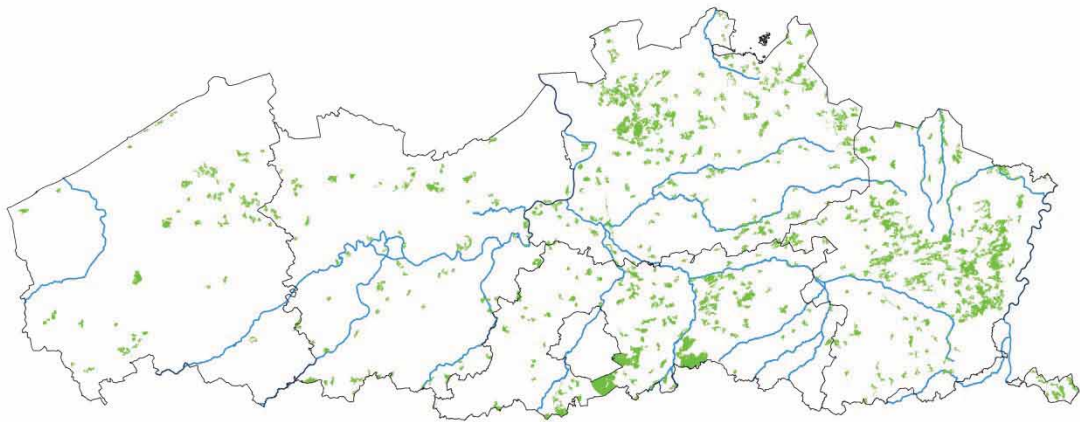
deleteVariable(lepelaar_leefgebied0)
assign(lepelaar_leefgebied, cFilterGeArea(lepelaar_leefgebied0_cluster, 500, hectare))
deleteVariable(lepelaar_leefgebied0_cluster)
assign(lepelaar_foerageer, and(lepelaar_foerageer3, lepelaar_leefgebied))
deleteVariable(lepelaar_foerageer3)
//write('lepelaar_foerageer.asc', IsNonZero(lepelaar_foerageer))
assign(lepelaar_voortplanting, and(lepelaar_voortplanting3, lepelaar_leefgebied))
deleteVariable(lepelaar_voortplanting3)
//write('lepelaar_voortplanting.asc', lepelaar_voortplanting)
assign(lepelaar_leefgebied1, or(lepelaar_voortplanting, lepelaar_foerageer))
deleteVariable(lepelaar_voortplanting)
deleteVariable(lepelaar_foerageer)
assign(lepelaar_leefgebied1_cluster, FuzzyClusterIdU(lepelaar_leefgebied1, 50, meter))
assign(lepelaar_leefgebied, cFilterGeArea(lepelaar_leefgebied1_cluster, 500, hectare))
//write('lepelaar_leefgebied.asc', if(lepelaar_leefgebied, lepelaar_leefgebied1_cluster, 0))
write('lepelaar_leefgebied.asc', IsNonZero(lepelaar_leefgebied))
deleteVariable(lepelaar_leefgebied1)
deleteVariable(lepelaar_leefgebied)
deleteVariable(lepelaar_leefgebied1_cluster)

```

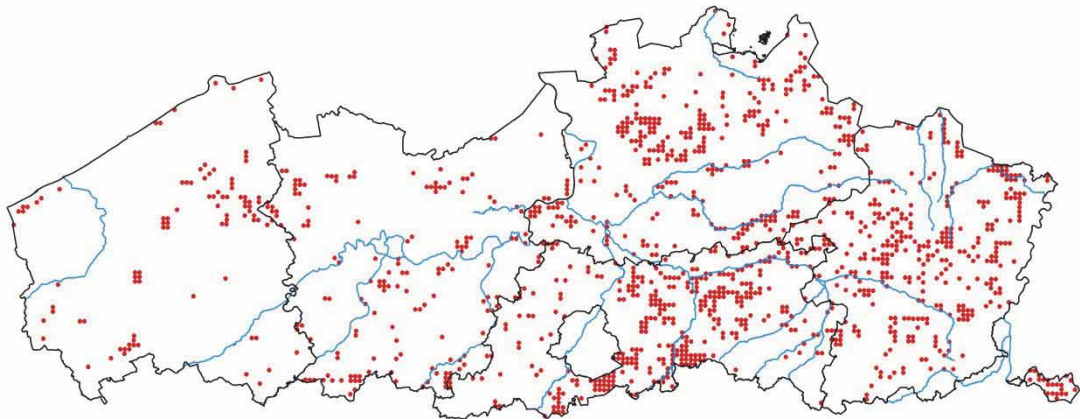


Lepelaar (foto Koen Devos)

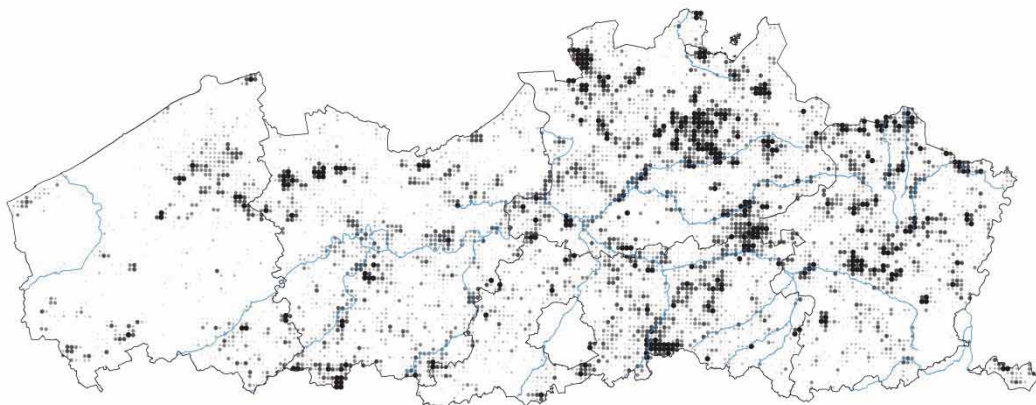
Middelste bonte specht (*Dendrocopos medius*) – Adriaens & Vermeersch (in druk)



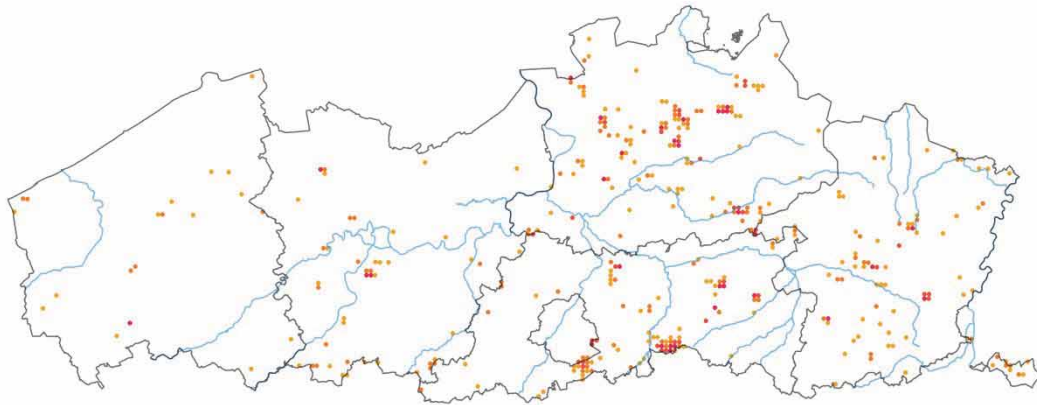
Potentiële leefgebiedenkaart voor Middelste bonte specht op basis van het onderstaande gdx-script.



Potentiële verspreidingskaart voor Middelste bonte specht op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Middelste bonte specht per kilometerhok in Vlaanderen (Tabel 7). Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.

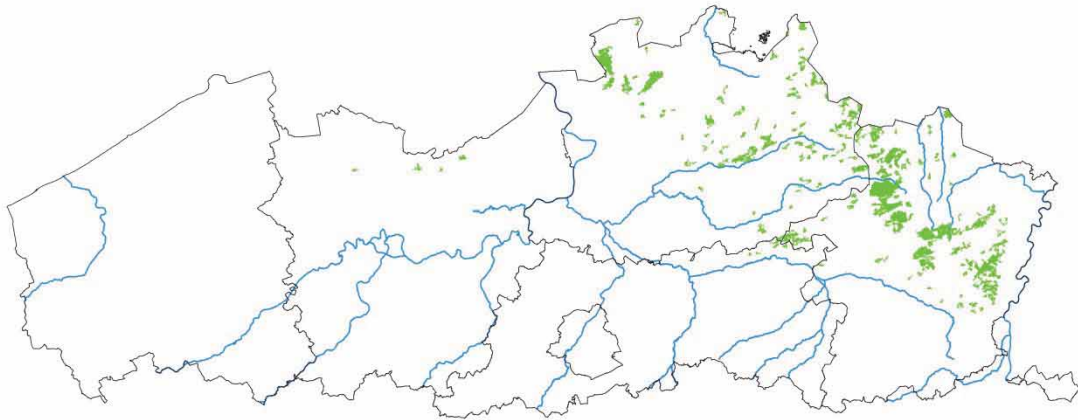


Waarnemingen van Middelste bonte specht uit het Natuurpunt data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

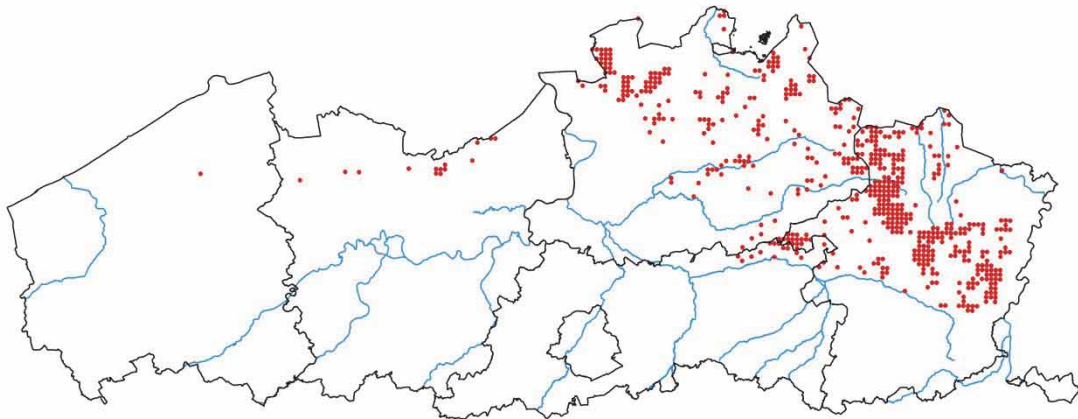
Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(middelstebontespecht_bwk, bwk(fa%, fe%, fk%, fl%, fm%, fs%, kp%, kpa%, kpk%, n, n+, n-, qa%, qb%, qe%,
  qk%, ql%, qs%, vf%))
// baken oude bossen af
assign(middelstebontespecht_bosinventaris, OrEq('bosinventaris_2001.asc', 12, 13, 14))
assign(middelstebontespecht_voortplanting, or(middelstebontespecht_bwk, middelstebontespecht_bosinventaris))
deleteVariable(middelstebontespecht_bwk)
deleteVariable(middelstebontespecht_bosinventaris)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het voortplantingscluster
assign(middelstebontespecht_voortplanting_cluster, fuzzyClusterId(middelstebontespecht_voortplanting,
  50/cellSize))
deleteVariable(middelstebontespecht_voortplanting)
// optimale voortplantingsclusters moeten minimaal 30ha groot zijn
assign(middelstebontespecht_leefgebied1, cFilterGeArea(middelstebontespecht_voortplanting_cluster, 30,
  hectare))
deleteVariable(middelstebontespecht_voortplanting_cluster)
assign(middelstebontespecht_ecodistrict, ge('ecoregio.asc', 1))
assign(middelstebontespecht_leefgebied_ecodistrict1, and(middelstebontespecht_leefgebied1,
  middelstebontespecht_ecodistrict))
deleteVariable(middelstebontespecht_leefgebied1)
deleteVariable(middelstebontespecht_ecodistrict)
assign(middelstebontespecht_leefgebied_ecodistrict1_cluster,
  FuzzyClusterId(middelstebontespecht_leefgebied_ecodistrict1, 50/cellSize))
deleteVariable(middelstebontespecht_leefgebied_ecodistrict1)
assign(middelstebontespecht_leefgebied_ecodistrict,
  cFilterGeArea(middelstebontespecht_leefgebied_ecodistrict1_cluster, 30, hectare))
// write('middelstebontespecht_leefgebied_ecodistrict.asc', if(middelstebontespecht_leefgebied_ecodistrict,
  middelstebontespecht_leefgebied_ecodistrict1_cluster, 0))
write('middelstebontespecht_leefgebied_ecodistrict.asc',
  IsNonZero(middelstebontespecht_leefgebied_ecodistrict))
deleteVariable(middelstebontespecht_leefgebied_ecodistrict1_cluster)
deleteVariable(middelstebontespecht_leefgebied_ecodistrict)
```

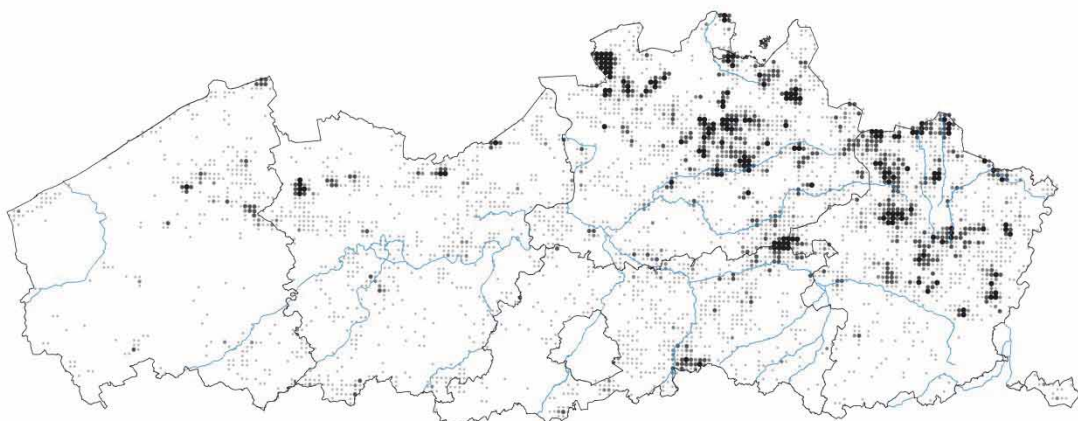
Nachtzwaluw (*Caprimulgus europaeus*) – Adriaens & Vermeersch (in druk)



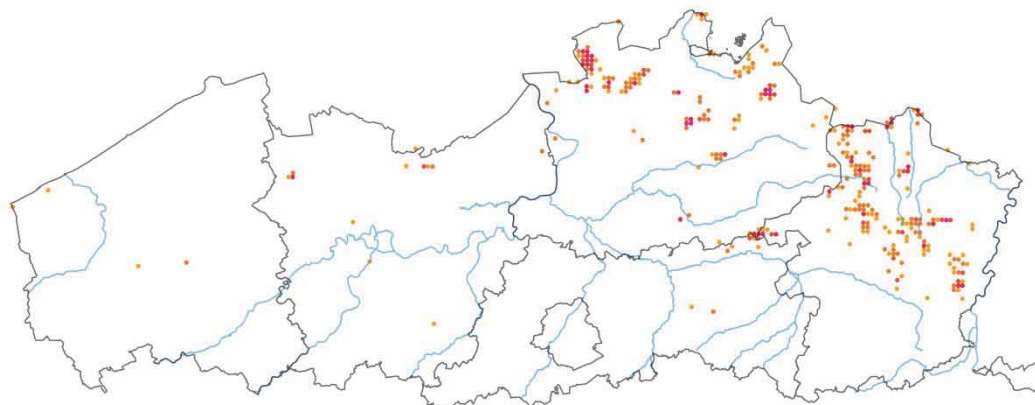
Potentiële leefgebiedenkaart voor Nachtzwaluw op basis van het onderstaande gdx-script.



Potentiële verspreidingskaart voor Nachtzwaluw op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Nachtzwaluw per kilometerhok in Vlaanderen (Tabel 7).
Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Nachtzwaluw uit het Natuurpunt data protaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

Gdx-script

vlaanderen_20m.asc

// baken de voortplantingsgebieden af

```
assign(nachtzwaluw_voortplanting, bwk(cd%, ce%, cg%, cm%, cp%, cv%, dm%, ha%, hn%, pi%, pms%, ppa%, ppi%,
  ppmh%, se%, sg%, 2310%, 2330%, 4010%, 4030%, 5130%, 6230%, rbbsg%, rbbppm%))
```

// baken de bossen af

```
assign(nachtzwaluw_bos, bwk(pa%, pm%, pp%, qs%, 9190%, rbbppm%))
```

// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het boscluster

```
assign(nachtzwaluw_bos_cluster, fuzzyClusterIdU(nachtzwaluw_bos, 50, meter))
```

```
deleteVariable(nachtzwaluw_bos)
```

// bosclusters moeten minimaal 5ha groot zijn

```
assign(nachtzwaluw_bos_cluster_5ha, cFilterGeArea(nachtzwaluw_bos_cluster, 5, hectare))
```

```
deleteVariable(nachtzwaluw_bos_cluster)
```

// zoek voortplantingsclusters die aan bosclusters grenzen

```
assign(nachtzwaluw_edge1, edge(IsNonZero(nachtzwaluw_bos_cluster_5ha),
  IsNonZero(nachtzwaluw_voortplanting)))
```

```
assign(nachtzwaluw_edge2, edge(IsNonZero(nachtzwaluw_voortplanting),
  IsNonZero(nachtzwaluw_bos_cluster_5ha)))
```

```
assign(nachtzwaluw_leefgebied1, cFilter(nachtzwaluw_voortplanting, or(nachtzwaluw_edge1,
  nachtzwaluw_edge2)))
```

```
deleteVariable(nachtzwaluw_voortplanting)
```

```
deleteVariable(nachtzwaluw_bos_cluster_5ha)
```

// lengte van bosrand tussen voortplantingsclusters en boscluster moet minstens 1500 meter zijn

```
assign(opp_bosrand, cSum(nachtzwaluw_leefgebied1, or(nachtzwaluw_edge1, nachtzwaluw_edge2)))
```

```
assign(nachtzwaluw_leefgebied_bosrand, cFilter(nachtzwaluw_leefgebied1, ge(opp_bosrand, 1500/cellSize)))
```

```
deleteVariable(nachtzwaluw_edge1)
```

```
deleteVariable(nachtzwaluw_edge2)
```

```
deleteVariable(nachtzwaluw_leefgebied1)
```

```
deleteVariable(opp_bosrand)
```

// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het voortplantingscluster

```
assign(nachtzwaluw_leefgebied_bosrand_cluster, fuzzyClusterIdU(nachtzwaluw_leefgebied_bosrand, 50, meter))
```

```
deleteVariable(nachtzwaluw_leefgebied_bosrand)
```

// voortplantingsclusters moeten minimaal 25ha groot zijn

```
assign(nachtzwaluw_leefgebied, cFilterGeArea(nachtzwaluw_leefgebied_bosrand_cluster, 25, hectare))
```

// baken ecoregio van ecodistrictgebieden af

```
assign(nachtzwaluw_ecodistrict, OrEq('ecoregio.asc', 5, 8, 12, 13, 14, 15, 16, 17))
```

```
assign(nachtzwaluw_leefgebied_ecodistrict, and(nachtzwaluw_leefgebied, nachtzwaluw_ecodistrict))
```

```
deleteVariable(nachtzwaluw_leefgebied)
```

```
deleteVariable(nachtzwaluw_ecodistrict)
```

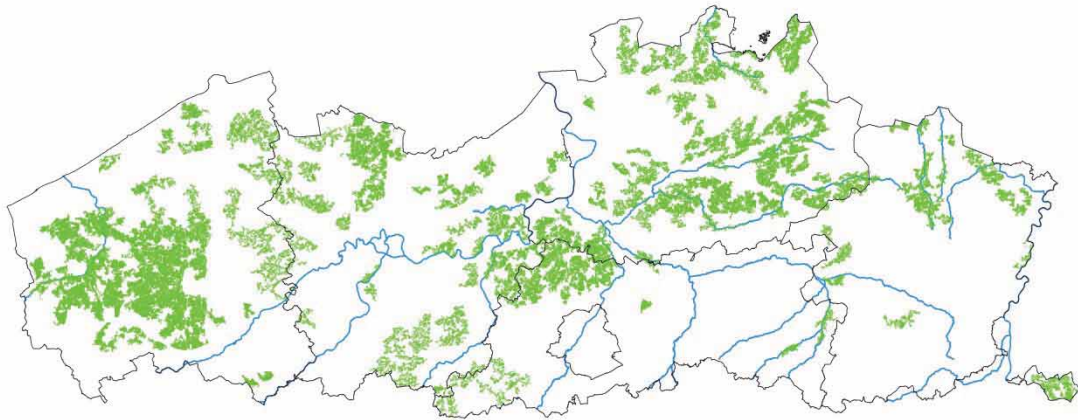
```
write('nachtzwaluw_leefgebied_ecodistrict.asc', if(nachtzwaluw_leefgebied_ecodistrict,  
  nachtzwaluw_leefgebied_bosrand_cluster, 0))  
// write('nachtzwaluw_leefgebied_ecodistrict.asc', IsNonZero(nachtzwaluw_leefgebied_ecodistrict))  
deleteVariable(nachtzwaluw_leefgebied_bosrand_cluster)  
deleteVariable(nachtzwaluw_leefgebied_ecodistrict)
```



Nachtzwaluw (foto Glenn Vermeersch)



Middelste bonte specht (foto Yves Adams – Vilda)



Potentiële leefgebiedenkaart voor Ooievaar op basis van het onderstaande gdx-script.

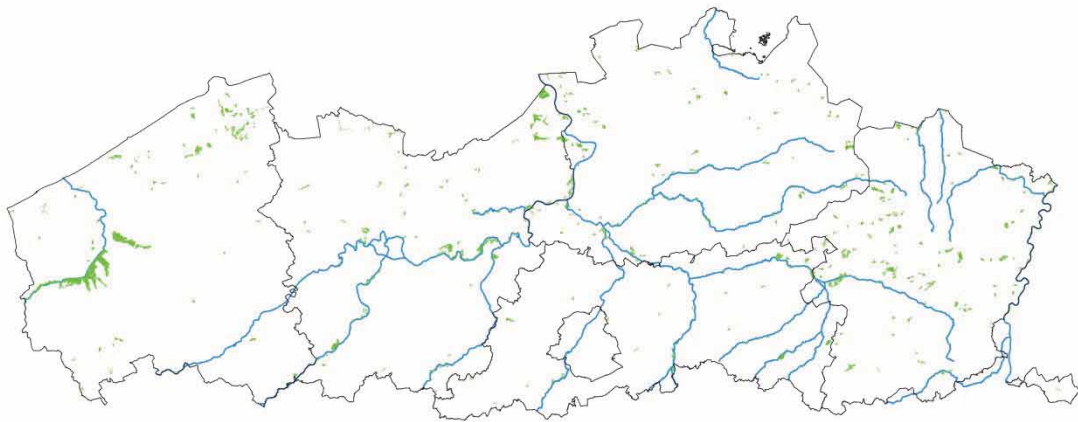
Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(ooievaar_voortplanting, bwk(hc%, hj%, hm%, hme%, hp%, hp+%, hpr%, hr%, hu%, hx%, kn%, rbbhc%, 6410%,
6510%, 3150%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij de geschikte leefgebiedcluster
assign(ooievaar_leefgebied2c, fuzzyClusterIdU(ooievaar_voortplanting, 50, meter))
deleteVariable(ooievaar_voortplanting)
// leefgebied moeten minimaal 300ha groot zijn
assign(ooievaar_leefgebied, cFilterGeArea(ooievaar_leefgebied2c, 300, hectare))
write('ooievaar_leefgebied.asc', if(ooievaar_leefgebied, ooievaar_leefgebied2c, 0))
// write('ooievaar_leefgebied.asc', IsNonZero(ooievaar_leefgebied))
deleteVariable(ooievaar_leefgebied2c)
deleteVariable(ooievaar_leefgebied)
```

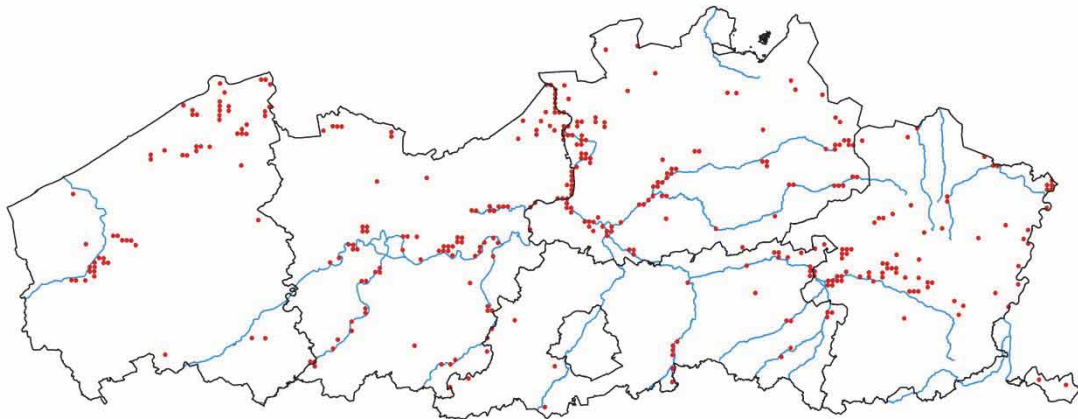


Ooievaar (foto Glenn Vermeersch)

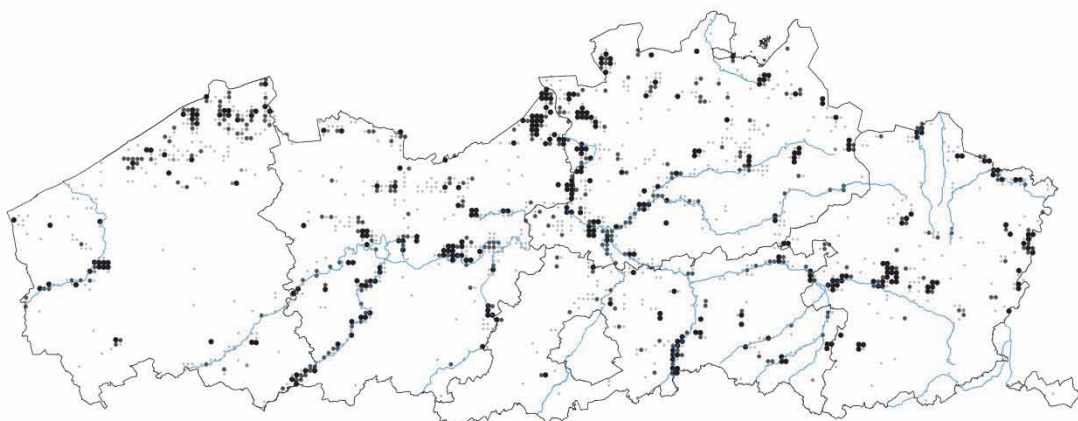
Porseleinhoen (*Porzana porzana*) – Adriaens & Vermeersch (in druk)



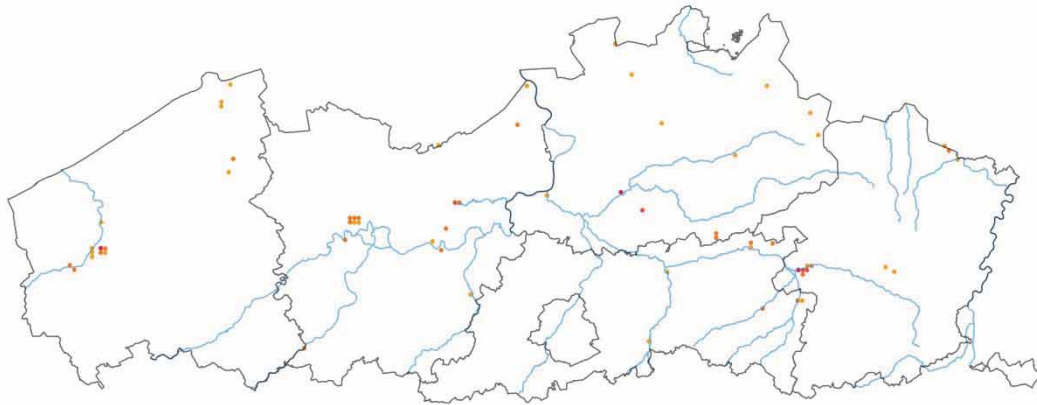
Potentiële leefgebiedenkaart voor Porseleinhoen op basis van het onderstaande gdx-script.



Potentiële verspreidingskaart voor Porseleinhoen op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Porseleinhoen per kilometerhok in Vlaanderen (Tabel 7). Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Porseleinhoen uit het Natuurpunt data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

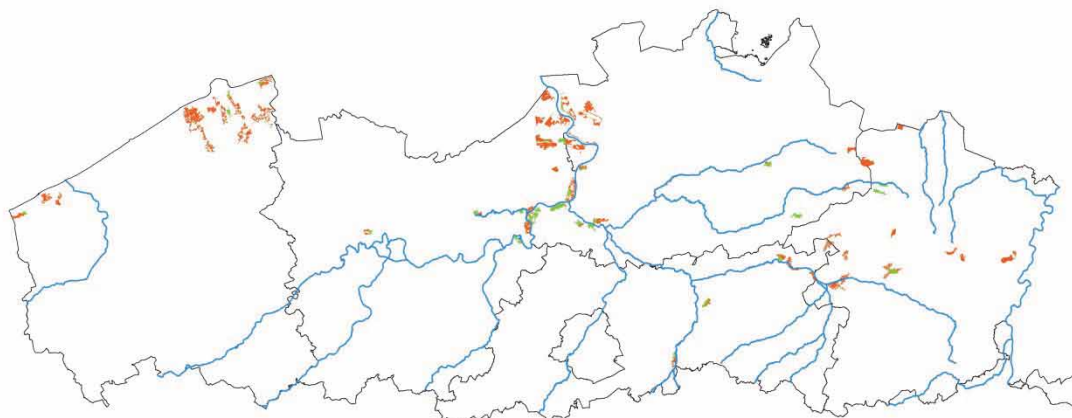
Gdx-script

```
vlaanderen_20m.asc
assign(OVSTR, 'OVSTRGEV2014_20m.asc')
assign(Groenkaart2013, 'groenkaart2013_20m.asc')
assign(wegen, 'GRB_Wegverbinding_at.asc')
// assign(vegetaties, bwk(ces, ces-, ces+, hc, hc-, hc+, rbbhc, 6410%, hf, hf-, hf+, rbbhf%, 6430%, hj, hj+, hj-, hm, hm-,
// hm+, hme, hme-, hme+, hmm, hmm-, hmm+, hpr, hpr-, hpr+, hu, hu-, hu+, ku, ku-, ku+, mc-, mc+, rbbmc%, md,
// md-, md+, mk, mk-, mk+, 7230%, mm, mm-, mm+, mp, mp-, mp+, mr, mr-, mr+, rbbmr%, ms, ms-, ms+, rbbms%))
assign(grasland, or(bwk(ces, ces-, ces+, 7140%, hj, hj+, hj-, hm, hm-, hm+, hme, hme-, hme+, hmm, hmm-, hmm+,
// 6410%, hu, hu-, hu+, 6510%, ku, ku-, ku+, 6430%), and(bwk(hpr, hpr-, hpr+), gt(OVSTR, 0))))
deleteVariable(OVSTR)
// write('porseleinhoen_biotopen_grasland_bis.asc', grasland)
assign(moeras, bwk(hc, hc-, hc+, rbbhc%, 6410%, hf, hf-, hf+, 6430%, rbbhf%, mc, mc-, mc+, rbbmc%, md, md-, md+,
// 7140%, mk, mk-, mk+, 7230%, mm, mm-, mm+, 7210%, mp, mp-, mp+, 2190%, mr, mr-, mr+, rbbmr%, ms, ms-,
// ms+, rbbms%))
// write('porseleinhoen_biotopen_moeras_bis.asc', moeras)
assign(plassen, and(bwk(ae, ae-, ae+, aer%, aom%, 3130%, 3150%), not(bwk(app%))))
// write('porseleinhoen_biotopen_plassen_bis.asc', plassen)
// assign(kernbiotoop, bwk(hc, hc+, hc-, rbbhc%, 6410%, hf, hf+, hf-, rbbhf%, 6430%, mc, mc+, mc-, rbbmc%, mk,
// mk+, mk-, 7230%, mm, mm+, mm-, ms, ms+, ms-, rbbms%, 7140%, hj, hj+, hj-))
// assign(verbosning, bwk(91%, gml, gmn, f%, n, n+, n-, l, ls%, lh%, p%, q%, ru%, v%, s%, kb%, kh%, kt(f%, kt(q%,
// kt(ru%, kt(v%, s%))
// verbosning afgeleid uit groenkaart, waarbij bossen gedefinieerd als aaneengesloten oppervlakte aan hooggroen
// van meer dan 0.25 ha (excl. kle)
assign(biotopen_kle_groenkaart_gr3m, edge(eq(Groenkaart2013, 1), gt(Groenkaart2013, 1)))
assign(biotopen_kle_groenkaart_gr3m, and(biotopen_kle_groenkaart_gr3m, ne(MajorityFilter(Groenkaart2013, 1,
// 2), 1)))
assign(verbosning, or(ge(csum(clusterid(and(eq(Groenkaart2013, 1), not(biotopen_kle_groenkaart_gr3m))), 1),
// 2500/cellsize2), bwk(91%)))
deleteVariable(Groenkaart2013)
deleteVariable(biotopen_kle_groenkaart_gr3m)
// write('porseleinhoen_biotopen_verbosning_bis.asc', verbosning)
// assign(porseleinhoen_geschikte_biotopen, and(or(moeras, and(plassen, moeras), and(grasland, or(moeras,
// plassen))), not(or(bwk(1130), verbosning))))
assign(porseleinhoen_geschikte_biotopen, and(or(moeras, plassen, grasland), not(or(bwk(1130), verbosning))))
deleteVariable(moeras)
deleteVariable(plassen)
deleteVariable(grasland)
deleteVariable(verbosning)
// afstand tot grote wegen, bebouwing en kanalen: 150m
```

```
// afstand tot bebouwing in rekening brengen, i.e. indien die clusters vormen van meer dan 10 ha (met
tussenafstand van 50m)
assign(bebouwing, bwk(u, ua, ua-, uc, ud, ui, ui-, un, un-, ur, ur-, ur+, uv, uv-, uv+))
assign(bebouwing_id, FuzzyClusterId(bebouwing, 50/cellSize))
deleteVariable(bebouwing)
assign(bebouwing_cluster_10ha_50m, cFiltergearea(bebouwing_id, 10, hectare))
deleteVariable(bebouwing_id)
assign(afstand_bebouwing, cellDistance(bebouwing_cluster_10ha_50m, 0))
deleteVariable(bebouwing_cluster_10ha_50m)
// afstand tot grote wegen (voorstel ANB); enkel autosnelwegen en tweevakbanen dan
assign(afstand_wegen, cellDistance(or(eq(wegen, 101), eq(wegen, 102)), 0))
deleteVariable(wegen)
assign(porseleinhoen_geschikte_biotopen, and(porseleinhoen_geschikte_biotopen, gt(afstand_bebouwing,
150/cellSize), gt(afstand_wegen, 150/cellSize)))
deleteVariable(afstand_bebouwing)
deleteVariable(afstand_wegen)
// a priori wegfilteren van de clusters kleiner dan 2 ha
assign(x, clusterid(porseleinhoen_geschikte_biotopen))
assign(porseleinhoen_geschikte_biotopen_ge2ha, isnonzero(cfiltergearea(x, 2, hectare)))
// write('porseleinhoen_geschikte_biotopen_ge2ha_bis.asc', porseleinhoen_geschikte_biotopen_ge2ha)
assign(porseleinhoen_geschikte_biotopen, porseleinhoen_geschikte_biotopen_ge2ha)
// write('porseleinhoen_geschikte_biotopen_bis.asc', porseleinhoen_geschikte_biotopen_ge2ha)
deleteVariable(porseleinhoen_geschikte_biotopen_ge2ha)
// clustering
assign(id, fuzzyClusterId(porseleinhoen_geschikte_biotopen, 100/cellSize))
// cluster minimaal 10 ha
assign(y, cFiltergearea(id, 10, hectare))
write('porseleinhoen_clusters_100m_10ha_bis.asc', y)
```



Porseleinhoen (foto Glenn Vermeersch)



Potentiële leefgebiedenkaart voor Purperreiger op basis van het onderstaande gdx-script.
Groen = voortplantingsgebied, oranje = foerageergebied.

Gdx-script

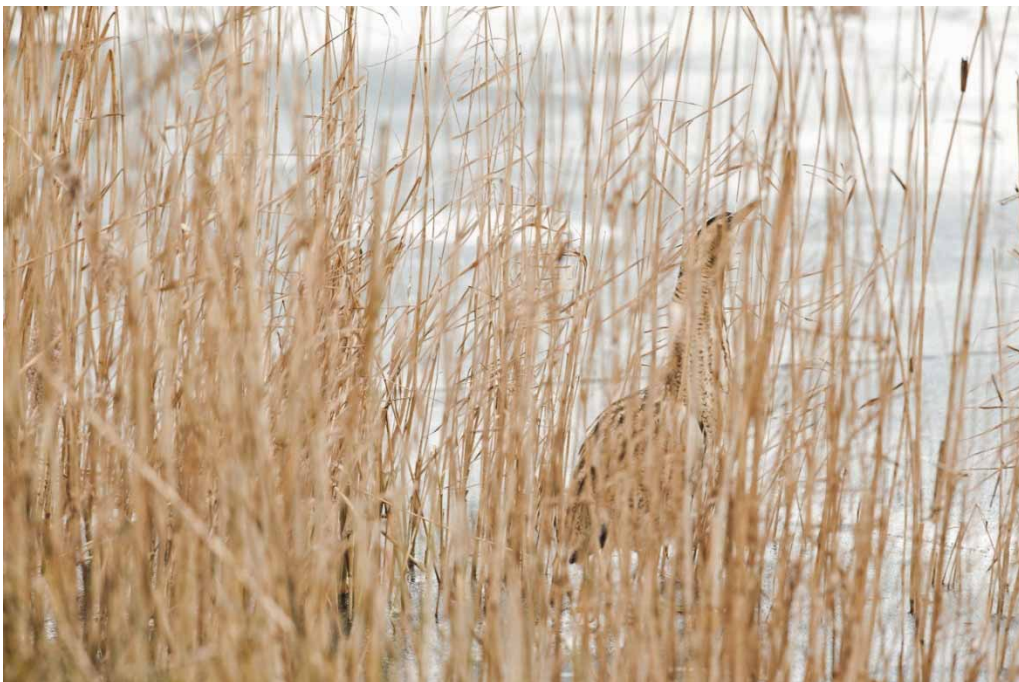
vlaanderen_20m.asc

```
assign(purperreiger_voortplanting_bwk, bwk(mc%, mcb%, mdb%, mr%, mrb%, mru%, ru%, rud%, sf%, sm%, so%,
    va%, vc%, vf%, vm%, vn%, vo%, vt%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het cluster
assign(purperreiger_voortplanting_cluster, fuzzyClusterIdU(purperreiger_voortplanting_bwk, 50, meter))
deleteVariable(purperreiger_voortplanting_bwk)
// clusters minimaal 30 ha
assign(purperreiger_voortplanting2, cFilterGeArea(purperreiger_voortplanting_cluster, 30, hectare))
deleteVariable(purperreiger_voortplanting_cluster)
assign(eikenbos, bwk(q%, f%))
assign(purperreiger_voortplanting3, and(purperreiger_voortplanting2, not(eikenbos)))
deleteVariable(purperreiger_voortplanting2)
deleteVariable(eikenbos)
// clustering: indien geschikte cel binnen 10 meter, dan hoort die cel bij het cluster
assign(purperreiger_voortplanting3_cluster, fuzzyClusterIdU(purperreiger_voortplanting3, 10, meter))
deleteVariable(purperreiger_voortplanting3)
assign(purperreiger_voortplanting4, cFilterGeArea(purperreiger_voortplanting3_cluster, 30, hectare))
deleteVariable(purperreiger_voortplanting3_cluster)
// foerageergebieden: moerassen
assign(purperreiger_moeras, bwk(ae%, aev%, ad%, ah%, aom%, hc%, hd%, hf%, hj%, kn%, ku%, kub%, k(mr%, mc%,
    mcb%, mdb%, mr%, mrb%, mru%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het cluster
assign(purperreiger_moeras_cluster, fuzzyClusterIdU(purperreiger_moeras, 50, meter))
deleteVariable(purperreiger_moeras)
// clusters minimaal 100 ha
assign(purperreiger_moeras1, cFilterGeArea(purperreiger_moeras_cluster, 100, hectare))
deleteVariable(purperreiger_moeras_cluster)
// geschikte foerageergebieden liggen minder dan 10 km van voortplantingsgebieden en omgekeerd
assign(purperreiger_voortplanting0, and(purperreiger_voortplanting4, cellDistanceLe(purperreiger_moeras1, 10,
    kilometer)))
assign(purperreiger_foerageer0, and(purperreiger_moeras1, cellDistanceLe(purperreiger_voortplanting4, 10,
    kilometer)))
deleteVariable(purperreiger_moeras1)
deleteVariable(purperreiger_voortplanting4)
assign(purperreiger_leefgebied1, or(purperreiger_voortplanting0, purperreiger_foerageer0))
assign(purperreiger_leefgebied1_cluster, FuzzyClusterIdU(purperreiger_leefgebied1, 100, meter))
deleteVariable(purperreiger_leefgebied1)
assign(purperreiger_leefgebied, cFilterGeArea(purperreiger_leefgebied1_cluster, 100, hectare))
// write('purperreiger_leefgebied.asc', if(purperreiger_leefgebied, purperreiger_leefgebied1_cluster, 0))
```

```
//write('purperreiger_leefgebied.asc', IsNonZero(purperreiger_leefgebied))
deleteVariable(purperreiger_leefgebied1_cluster)
assign(purperreiger_foerageer, and(purperreiger_foerageer0, purperreiger_leefgebied))
deleteVariable(purperreiger_foerageer0)
write('purperreiger_foerageer.asc', IsNonZero(purperreiger_foerageer))
deleteVariable(purperreiger_foerageer)
assign(purperreiger_voortplanting, and(purperreiger_voortplanting0, purperreiger_leefgebied))
deleteVariable(purperreiger_voortplanting0)
write('purperreiger_voortplanting.asc', purperreiger_voortplanting)
deleteVariable(purperreiger_voortplanting)
deleteVariable(purperreiger_leefgebied)
```

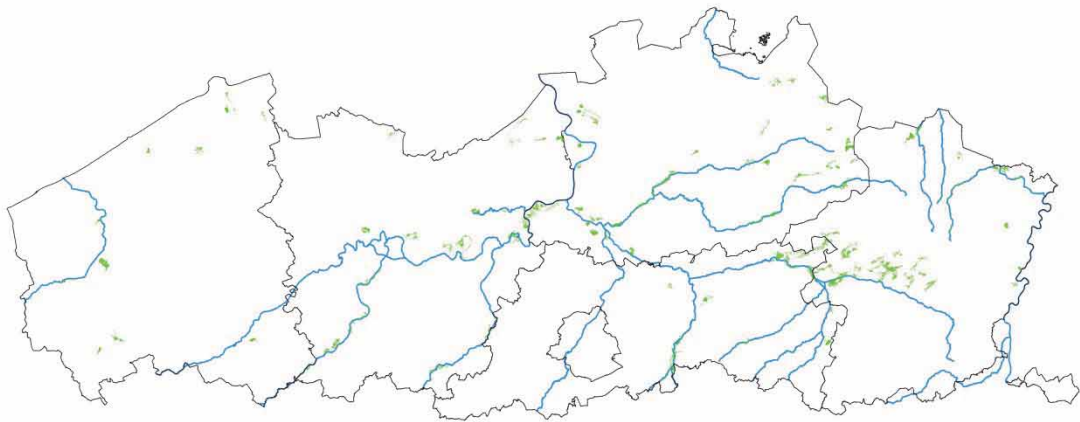


Purperreiger (foto Glenn Vermeersch)

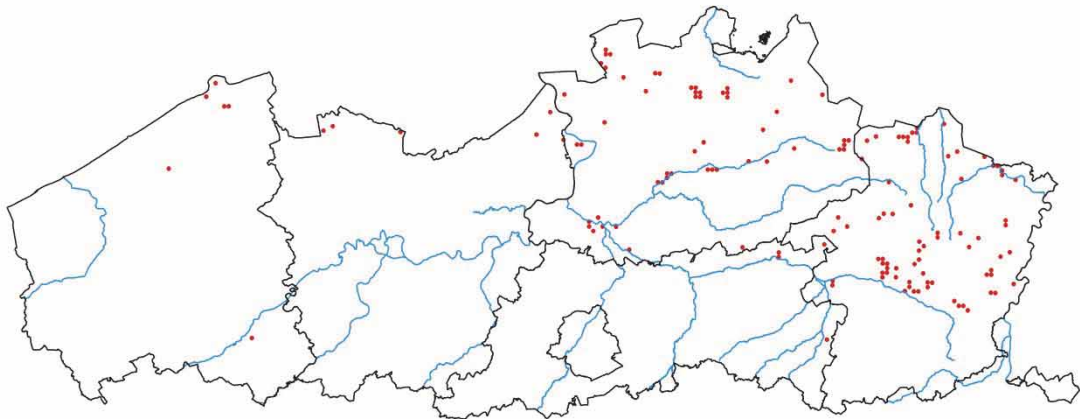


Roerdomp (foto Lars Soerink – Vilda)

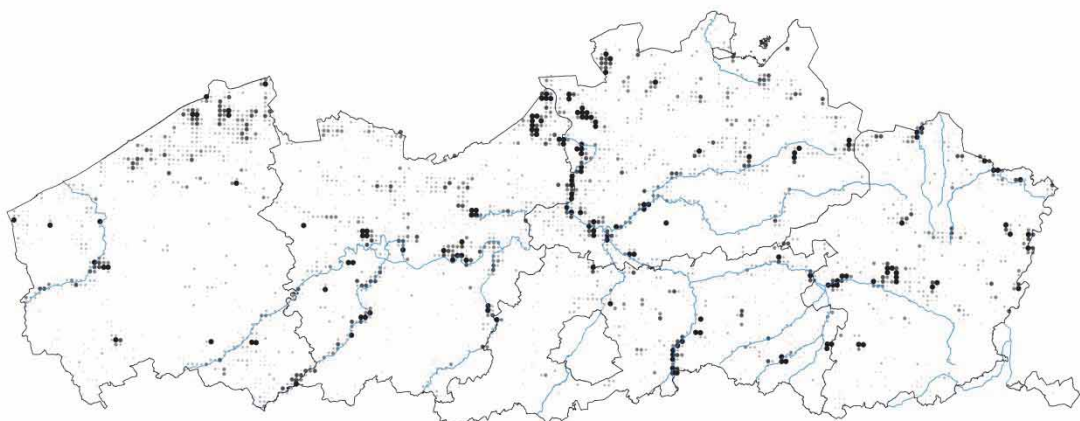
Roerdomp (*Botaurus stellaris*) – Adriaens & Vermeersch (in druk)



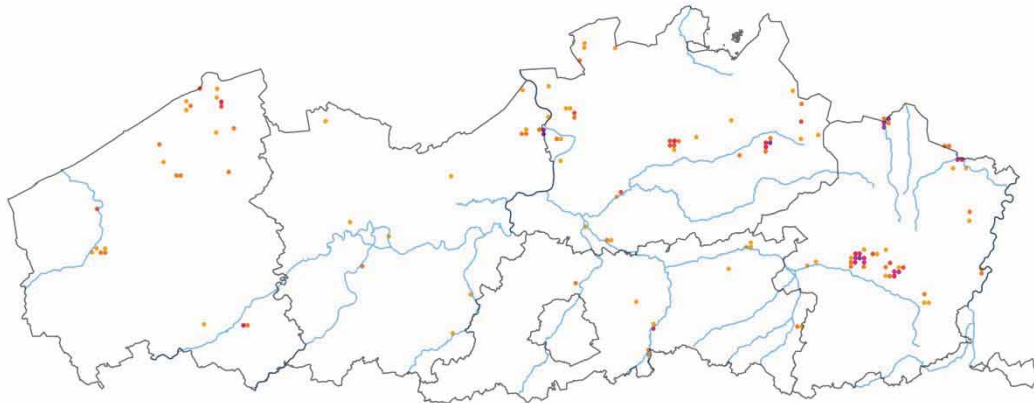
Potentiële leefgebiedenkaart voor Roerdomp op basis van het onderstaande gdx-script.



Potentiële verspreidingskaart voor Roerdomp op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Roerdomp per kilometerhok in Vlaanderen (Tabel 7).
Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Roerdomp uit het data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

Gdx-script

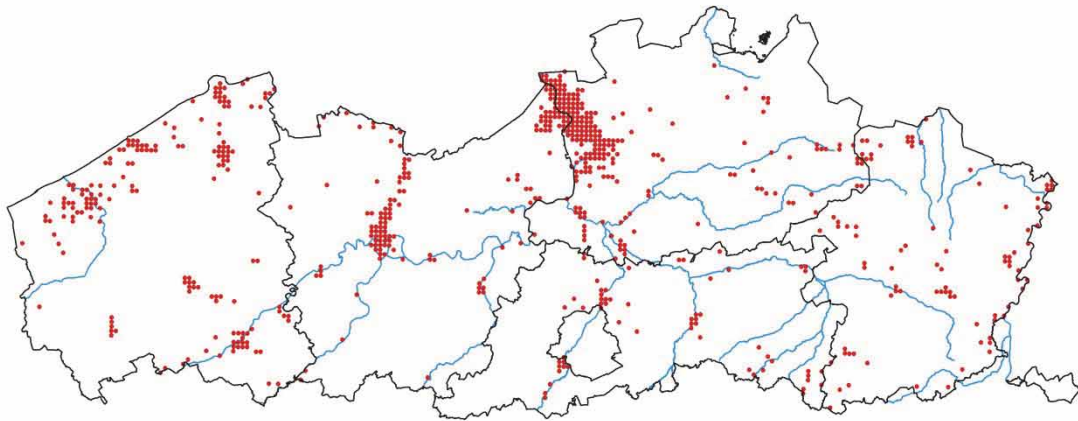
```
vlaanderen_20m.asc
assign(W, 'GRB_watergang_at.asc')
assign(G, 'groenkaart2013_20m.asc')
// Verboste stukken uitsluiten? verbossing afgeleid uit groenkaart, waarbij bossen gedefinieerd als aaneengesloten
// oppervlakte aan hooggroen van meer dan 0.25 ha?
assign(verbossing, or(ge(csum(clusterid(eq(G, 1)), 1), 2500/cellsize2), and(bwk(91%), not(bwk(91E0%)))))
deleteVariable(G)
// write('roerdomp_geschikte_biotopen_verbossing_bis.asc', verbossing)
// Eerst selectie cellen voor Roerdomp
assign(openwater, and(bwk(ae%, ao%, ap, ap+, ap-, apo, apo+, apo-, 3110%, 3130%, 3150%, 3160%, 3260%,
3270%), not(verbossing)))
assign(rietland, and(bwk(mr%, rbbmr%), not(bwk(mr-)), not(verbossing)))
assign(zegge, and(bwk(mc%, rbbmc%), not(bwk(mc-)), not(verbossing)))
assign(moeras, and(bwk(hf%, hc%, md%, ms%, mk%, mz%, rbbhf%, rbbhc%, rbbms%, 6430%, 7230%, 7140%),
not(verbossing)))
assign(grasland, and(bwk(hmo%, hj%), not(verbossing)))
deletevariable(verbossing)
assign(struweel, bwk(so%, sf%, rbbso%, rbbso%))
assign(roerdomp_geschikte_biotopen, and(or(openwater, rietland, zegge, moeras, grasland, struweel),
not(ht(1130%))))
deletevariable(zegge)
deletevariable(struweel)
deletevariable(grasland)
deletevariable(moeras)
// write('roerdomp_geschikte_biotopen_bis.asc', roerdomp_geschikte_biotopen)
// Daarna verdere selectie op grond van samenhang tussen die cellen
// clustering: indien geschikte cel binnen 200 meter, dan hoort die cel bij het cluster
assign(id, fuzzyClusterId(roerdomp_geschikte_biotopen, 200/cellSize))
// write('roerdomp_clusters_200m.asc', id)
// clusters minimaal 30 ha
assign(y, cFiltergearea(id, 30, hectare))
// write('roerdomp_clusters_200m_30ha_bis.asc', y)
assign(id, y)
// clusters minimaal 10 ha open water
// assign(wateropp, cSum(id, openwater))
assign(wateropp_grb, cSum(id, or(eq(W, 1), openwater)))
// write('roerdomp_clusters_200m_30ha_wateropp_ha_bis.asc', wateropp*cellsize2/10000)
// assign(roerdomp_clusters_200m_30ha_wateropp_10ha, cFilter(id, ge(wateropp, 10*10000/cellsize2)))
// write('roerdomp_clusters_200m_30ha_wateropp_10ha_bis.asc',
roerdomp_clusters_200m_30ha_wateropp_10ha)
```

```

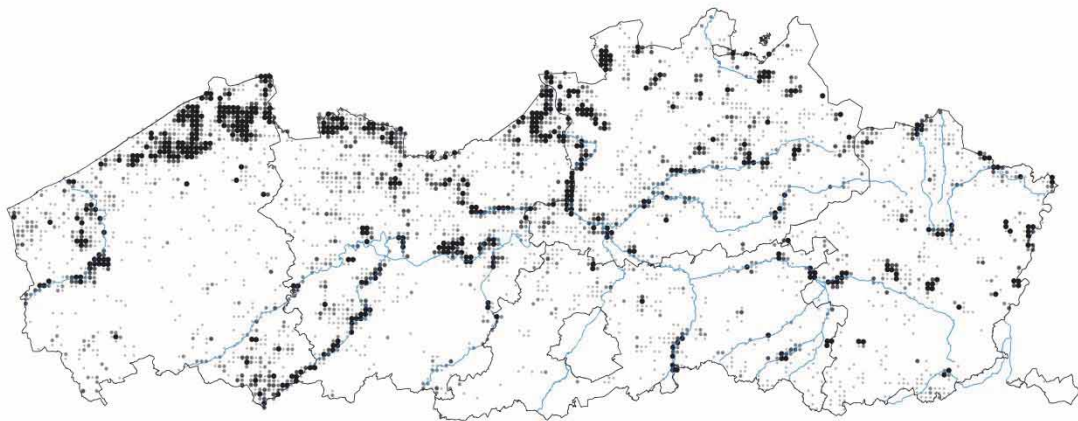
assign(roerdomp_clusters_200m_30ha_wateropp_10ha_grb, cFilter(id, ge(wateropp_grb, 10*10000/cellsize2)))
// write('roerdomp_clusters_200m_30ha_wateropp_10ha_grb_bis.asc',
  roerdomp_clusters_200m_30ha_wateropp_10ha_grb)
// clusters minimaal 500 meter waterkant
// assign(waterkant, cSum(id, IsNonZero(edge(rietland, openwater))))
assign(waterkant_grb, cSum(id, IsNonZero(edge(rietland, or(W, openwater)))))
deletevariable(W)
// write('roerdomp_clusters_200m_30ha_rand_water_riet_m_bis.asc', waterkant*cellsize)
// write('roerdomp_clusters_200m_30ha_rand_water_riet_grb_m_bis.asc', waterkant_grb*cellsize)
// assign(roerdomp_clusters_200m_30ha_rand_water_riet_500m, cFilter(id, ge(waterkant, 500/cellSize)))
assign(roerdomp_clusters_200m_30ha_rand_water_riet_grb_500m, cFilter(id, ge(waterkant_grb, 500/cellSize)))
// write('roerdomp_clusters_200m_30ha_rand_water_riet_500m_bis.asc',
  roerdomp_clusters_200m_30ha_rand_water_riet_500m)
// write('roerdomp_clusters_200m_30ha_rand_water_riet_grb_500m_bis.asc',
  roerdomp_clusters_200m_30ha_rand_water_riet_grb_500m)
// write('roerdomp_clusters_200m_30ha_wateropp_10ha_waterkant_ge500m_bis.asc',
  and(roerdomp_clusters_200m_30ha_wateropp_10ha,
  roerdomp_clusters_200m_30ha_rand_water_riet_500m)*id)
// write('roerdomp_clusters_200m_30ha_wateropp_10ha_waterkant_grb_ge500m_bis.asc',
  and(roerdomp_clusters_200m_30ha_wateropp_10ha,
  roerdomp_clusters_200m_30ha_rand_water_riet_grb_500m)*id)
// write('roerdomp_clusters_200m_30ha_wateropp_10ha_grb_waterkant_ge500m_bis.asc',
  and(roerdomp_clusters_200m_30ha_wateropp_10ha_grb,
  roerdomp_clusters_200m_30ha_rand_water_riet_500m)*id)
// write('roerdomp_clusters_200m_30ha_wateropp_10ha_grb_waterkant_grb_ge500m_bis.asc',
  and(roerdomp_clusters_200m_30ha_wateropp_10ha_grb,
  roerdomp_clusters_200m_30ha_rand_water_riet_grb_500m)*id)
// clusters minimaal 0.5 ha aaneengesloten rietland
assign(rietland_opp_max, cMax(id, cSum(fuzzyclusterid(rietland, 20/cellsize), 1)))
assign(roerdomp_clusters_200m_30ha_rietland_05ha, cFilter(id, ge(rietland_opp_max, 0.5*10000/cellsize2)))
// write('roerdomp_clusters_200m_30ha_rietland_05ha_bis.asc', roerdomp_clusters_200m_30ha_rietland_05ha)
// write('roerdomp_clusters_200m_30ha_wateropp_10ha_waterkant_ge500m_rietland_05ha_bis.asc',
  and(roerdomp_clusters_200m_30ha_wateropp_10ha, roerdomp_clusters_200m_30ha_rand_water_riet_500m,
  roerdomp_clusters_200m_30ha_rietland_05ha)*id)
// write('roerdomp_clusters_200m_30ha_wateropp_10ha_grb_waterkant_ge500m_rietland_05ha_bis.asc',
  and(roerdomp_clusters_200m_30ha_wateropp_10ha_grb,
  roerdomp_clusters_200m_30ha_rand_water_riet_500m, roerdomp_clusters_200m_30ha_rietland_05ha)*id)
write('roerdomp_clusters_200m_30ha_wateropp_10ha_grb_waterkant_grb_ge500m_rietland_05ha_bis.asc',
  and(roerdomp_clusters_200m_30ha_wateropp_10ha_grb,
  roerdomp_clusters_200m_30ha_rand_water_riet_grb_500m,
  roerdomp_clusters_200m_30ha_rietland_05ha)*id)

```

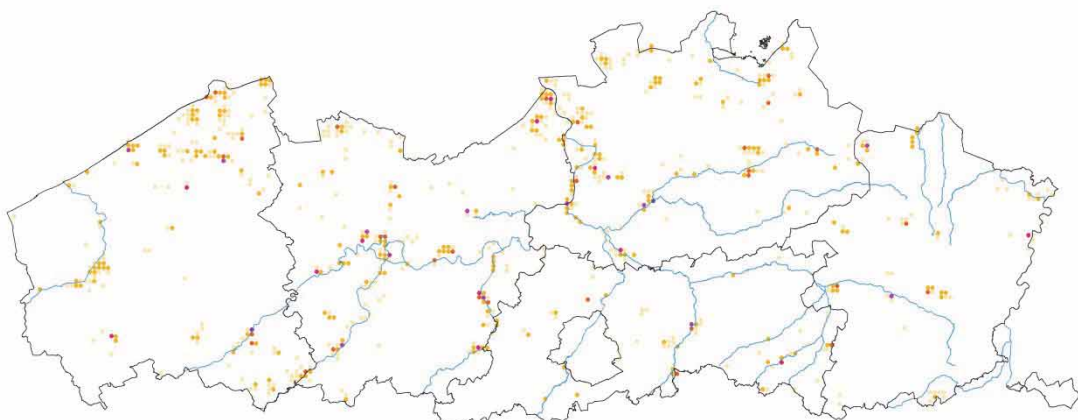
Slechtvalk (*Falco peregrinus*)



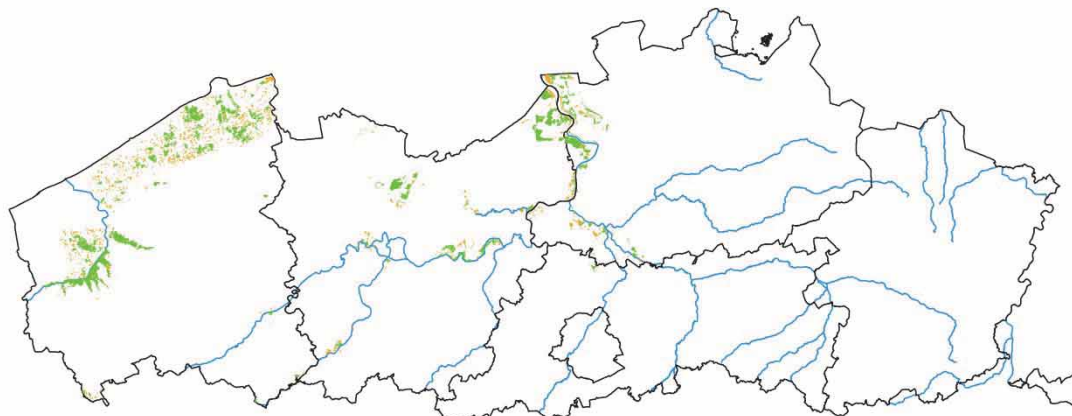
Potentiële verspreidingskaart voor Slechtvalk op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Roerdomp per kilometerhok in Vlaanderen (Tabel 7).
Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Slechtvalk uit het data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.



Potentiële leefgebiedenkaart voor Steltkluut op basis van het onderstaande gdx-script.
Groen = voortplantingsgebied, oranje = foerageergebied.

Gdx-script

vlaanderen_20m.asc

```
// baken de voortplantingsgebieden af
assign(steltkluut_voortplanting1, bwk(hpr%, hpr+%, kn%, ku%, kz%))
// clustering: indien geschikte cel binnen 20 meter, dan hoort die cel bij het cluster
assign(steltkluut_voortplanting1_cluster, fuzzyClusterIdU(steltkluut_voortplanting1, 50, meter))
deleteVariable(steltkluut_voortplanting1)
// voortplantingsclusters moeten minimaal 30ha groot zijn
assign(steltkluut_voortplanting2, cFilterGeArea(steltkluut_voortplanting1_cluster, 30, hectare))
deleteVariable(steltkluut_voortplanting1_cluster)
// voortplantingscluster moet vrij dicht bij open water en vochtige weilanden liggen
assign(steltkluut_foerageer1, bwk(ad%, ae%, aer%, ah%, da%, ds%, hpr%, hpr+%, kn%, ku%, kz%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het cluster
assign(steltkluut_foerageer1_cluster, fuzzyClusterIdU(steltkluut_foerageer1, 50, meter))
deleteVariable(steltkluut_foerageer1)
// foerageerclusters moeten minimaal 2ha groot zijn
assign(steltkluut_foerageer2, cFilterGeArea(steltkluut_foerageer1_cluster, 2, hectare))
deleteVariable(steltkluut_foerageer1_cluster)
// geschikte foerageergebieden liggen minder dan 2 km van voortplantingsgebieden en omgekeerd
assign(steltkluut_foerageer3, cFilter(steltkluut_foerageer2, cellDistanceLe(steltkluut_voortplanting2, 2000, meter)))
assign(steltkluut_voortplanting3, cFilter(steltkluut_voortplanting2, cellDistanceLe(steltkluut_foerageer2, 2000,
meter)))
deleteVariable(steltkluut_voortplanting2)
deleteVariable(steltkluut_foerageer2)
assign(steltkluut_leefgebied0, or(steltkluut_voortplanting3, steltkluut_foerageer3))
assign(steltkluut_leefgebied0_cluster, FuzzyClusterIdU(steltkluut_leefgebied0, 50, meter))
deleteVariable(steltkluut_leefgebied0)
assign(steltkluut_leefgebied, cFilterGeArea(steltkluut_leefgebied0_cluster, 2, hectare))
// write('steltkluut_leefgebied.asc', if(steltkluut_leefgebied, steltkluut_leefgebied0_cluster, 0))
// write('steltkluut_leefgebied.asc', IsNonZero(steltkluut_leefgebied))
deleteVariable(steltkluut_leefgebied0_cluster)
assign(steltkluut_foerageer, and(steltkluut_foerageer3, steltkluut_leefgebied))
// write('steltkluut_foerageer.asc', IsNonZero(steltkluut_foerageer))
deleteVariable(steltkluut_foerageer3)
assign(steltkluut_ecodistrict, OrEq('ecoregio.asc', 1, 2, 3, 4))
assign(steltkluut_foerageer_ecodistrict, and(steltkluut_foerageer, steltkluut_ecodistrict))
write('steltkluut_foerageer_ecodistrict.asc', IsNonZero(steltkluut_foerageer_ecodistrict))
assign(steltkluut_voortplanting, and(steltkluut_voortplanting3, steltkluut_leefgebied))
// write('steltkluut_voortplanting.asc', steltkluut_voortplanting)
deleteVariable(steltkluut_voortplanting3)
```

```
assign(steltkluut_ecodistrict, OrEq('ecoregio.asc', 1, 2, 3, 4))  
assign(steltkluut_voortplanting_ecodistrict, and(steltkluut_voortplanting, steltkluut_ecodistrict))  
write('steltkluut_voortplanting_ecodistrict.asc', IsNonZero(steltkluut_voortplanting_ecodistrict))
```

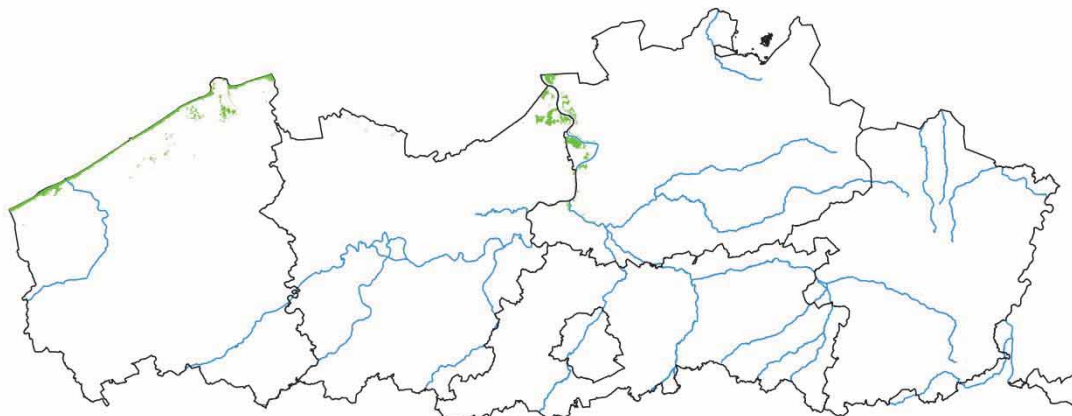


Steltkluut (foto Glenn Vermeersch)



Woudaap (foto Yves Adams – Vilda)

Strandplevier (*Charadrius alexandrinus*) – Adriaens & Vermeersch (in druk)



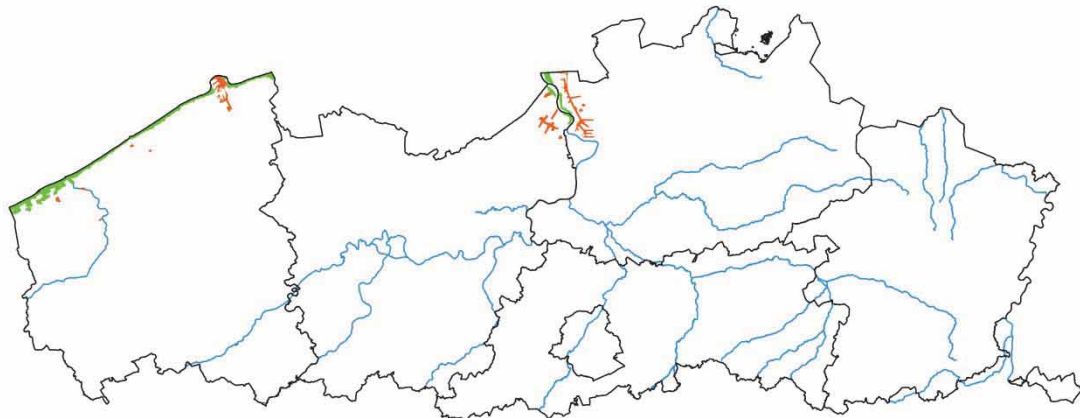
Potentiële leefgebiedenkaart voor Strandplevier op basis van het onderstaande gdx-script.
Groen = voortplantingsgebied, oranje = foerageergebied.

Gdx-script

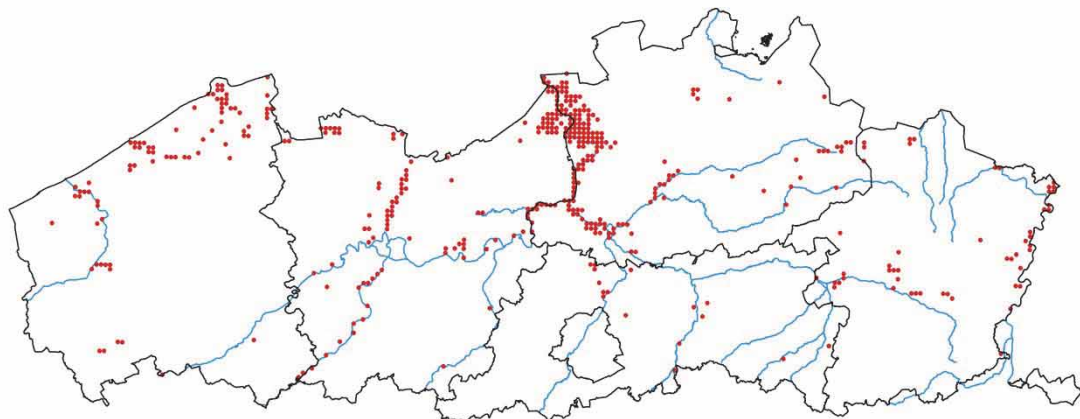
vlaanderen_20m.asc

```
// baken de voortplantingsgebieden af
assign(strandplevier_voortplanting1, bwk(da%, dd%, dl%, dla%, dz%, ku%, kz%))
// leefgebied mag geen open water of ruigte of moeras zijn
assign(openwater, bwk(ad%, ae, aes%, aev%, ah%, am%, ao%, aom%, aoo%, ap%, apo%, app%, mz%))
assign(strandplevier_voortplanting2, and(strandplevier_voortplanting1, not(openwater)))
deleteVariable(strandplevier_voortplanting1)
deleteVariable(openwater)
// clustering: indien geschikte cel binnen 10 meter, dan hoort die cel bij het voortplantingscluster
assign(strandplevier_voortplanting2_cluster, fuzzyClusterIdU(strandplevier_voortplanting2, 50, meter))
deleteVariable(strandplevier_voortplanting2)
// voortplantingsclusters moeten minimaal 2ha groot zijn
assign(strandplevier_voortplanting3, cFilterGeArea(strandplevier_voortplanting2_cluster, 2, hectare))
deleteVariable(strandplevier_voortplanting2_cluster)
// voortplantingscluster moet vrij dicht bij slik en strand liggen, dus baken slik af
assign(strandplevier_slik, bwk(da%, ds%, dz%, dl%))
// write('strandplevier_slik.asc', IsNonZero(strandplevier_slik))
// zoek voortplantingsgebieden die binnen de 200m van slik liggen en omgekeerd
assign(strandplevier_voortplanting4, cFilter(strandplevier_voortplanting3, cellDistanceLe(strandplevier_slik, 200,
meter)))
assign(strandplevier_foerageer4, cFilter(strandplevier_slik, cellDistanceLe(strandplevier_voortplanting3, 200,
meter)))
//write('strandplevier_voortplanting.asc', IsNonZero(strandplevier_voortplanting4))
assign(strandplevier_leefgebied0, and(strandplevier_voortplanting4, strandplevier_foerageer4))
assign(strandplevier_leefgebied0_cluster, FuzzyClusterIdU(strandplevier_leefgebied0, 50, meter))
assign(strandplevier_leefgebied, cFilterGeArea(strandplevier_leefgebied0_cluster, 2, hectare))
// write('strandplevier_leefgebied.asc', if(strandplevier_leefgebied, strandplevier_leefgebied0_cluster, 0))
// write('strandplevier_leefgebied.asc', IsNonZero(strandplevier_leefgebied))
assign(strandplevier_foerageer, and(strandplevier_foerageer4, strandplevier_leefgebied))
write('strandplevier_foerageer.asc', IsNonZero(strandplevier_foerageer))
assign(strandplevier_voortplanting2, and(strandplevier_voortplanting4, strandplevier_leefgebied))
write('strandplevier_voortplanting2.asc', strandplevier_voortplanting2)
deleteVariable(strandplevier_voortplanting4)
deleteVariable(strandplevier_foerageer4)
```

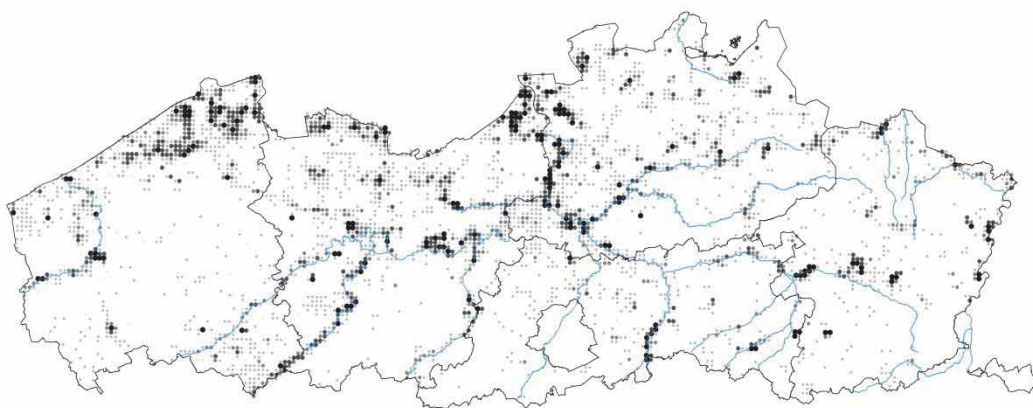
Visdief (*Sterna hirundo*) – Adriaens & Vermeersch (in druk)



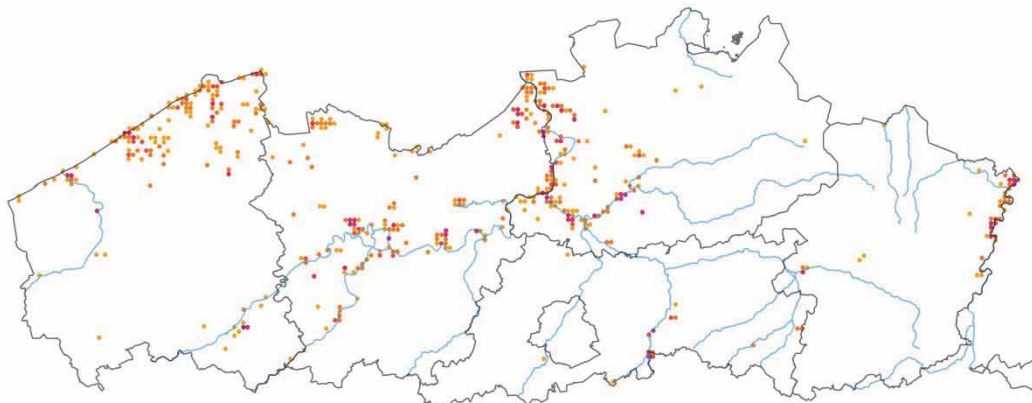
Potentiële leefgebiedenkaart voor Visdief op basis van het onderstaande gdx-script.
Groen = voortplantingsgebied, oranje = foerageergebied.



Potentiële verspreidingskaart voor Visdief op basis van correlatieve modellen.



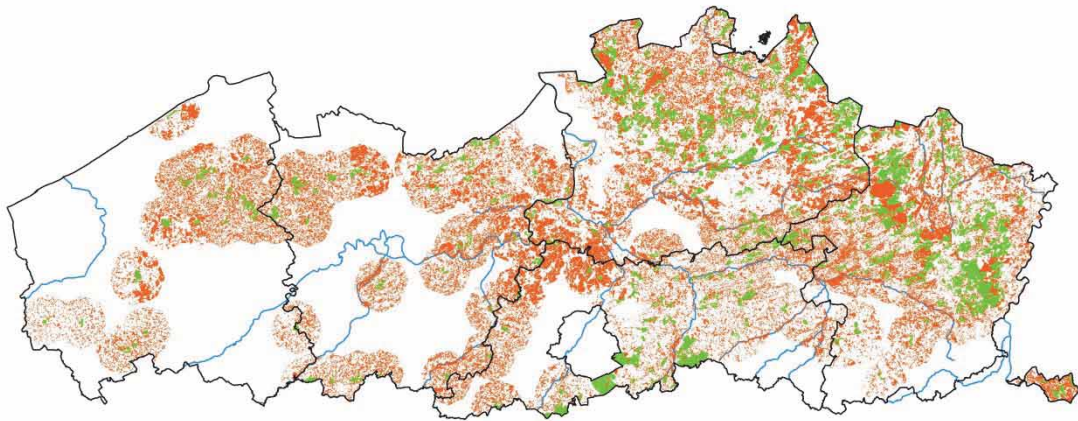
Som van de indicatorwaarde van begeleidende soorten van Visdief per kilometerhok in Vlaanderen (Tabel 7). Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



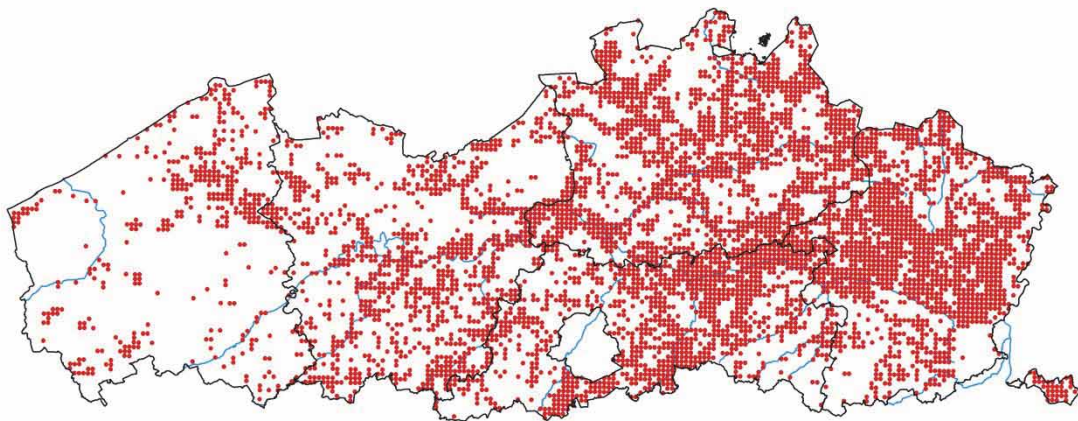
Waarnemingen van Visdief uit het data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

Gdx-script

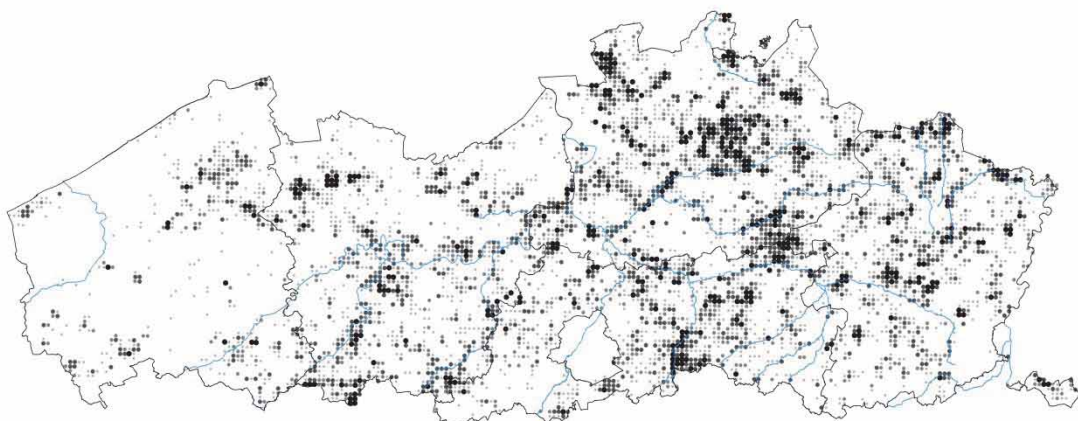
```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(visdief_voortplanting1, bwk(dd%, dl%, ds%, dz%))
// mag geen open water of ruigte of moeras zijn
assign(openwater, bwk(ad%, ae%, aer%, aes%, aev%, ah%, am%, ao%, aom%, aoo%, ap%, apo%, app%, ku%, mz%))
assign(visdief_voortplanting2, and(visdief_voortplanting1, not(openwater)))
// clustering: indien geschikte cel binnen 10 meter, dan hoort die cel bij de geschikte leefgebiedcluster
assign(visdief_voortplanting2_cluster, fuzzyClusterIdU(visdief_voortplanting2, 50, meter))
// leefgebied moeten minimaal 20ha groot zijn
assign(visdief_voortplanting3, cFilterGeArea(visdief_voortplanting2_cluster, 20, hectare))
// Zoek openwater in de kaarten van bwk, zee en waterlopen, die nog verrasterd moeten worden
assign(visdief_openwater, bwk(ad%, ae%, ah%, ap%, zee%))
// waterlopen
assign(visdief_waterlopen, OrEq('rastert_huetzon1.asc', 17, 19))
assign(visdief_foerageer1, or(visdief_openwater, visdief_waterlopen))
// clustering: indien geschikte cel binnen 10 meter, dan hoort die cel bij het openwaterscluster
assign(visdief_foerageer1cl, fuzzyClusterIdU(visdief_foerageer1, 10, meter))
// Open water moet minstens 1 ha groot zijn
assign(visdief_openwater, cFilterGeArea(visdief_foerageer1cl, 1, hectare))
// zoek voortplantingsgebieden die maximaal 5000 meter van groot open water liggen
assign(visdief_voortplanting4, cFilter(visdief_voortplanting3, cellDistanceLe(visdief_openwater, 5000, meter)))
assign(visdief_foerageer4, cFilter(visdief_openwater, cellDistanceLe(visdief_voortplanting3, 5000, meter)))
assign(visdief_leefgebied0, or(visdief_voortplanting4, visdief_foerageer4))
assign(visdief_leefgebied0_cluster, FuzzyClusterIdU(visdief_leefgebied0, 50, meter))
assign(visdief_leefgebied, cFilterGeArea(visdief_leefgebied0_cluster, 20, hectare))
// write('visdief_leefgebied.asc', if(visdief_leefgebied, visdief_leefgebied0_cluster, 0))
// write('visdief_leefgebied.asc', IsNonZero(visdief_leefgebied))
assign(visdief_foerageer, and(visdief_foerageer4, visdief_leefgebied))
write('visdief_foerageer.asc', IsNonZero(visdief_foerageer))
assign(visdief_voortplanting, and(visdief_voortplanting4, visdief_leefgebied))
write('visdief_voortplanting.asc', visdief_voortplanting)
```



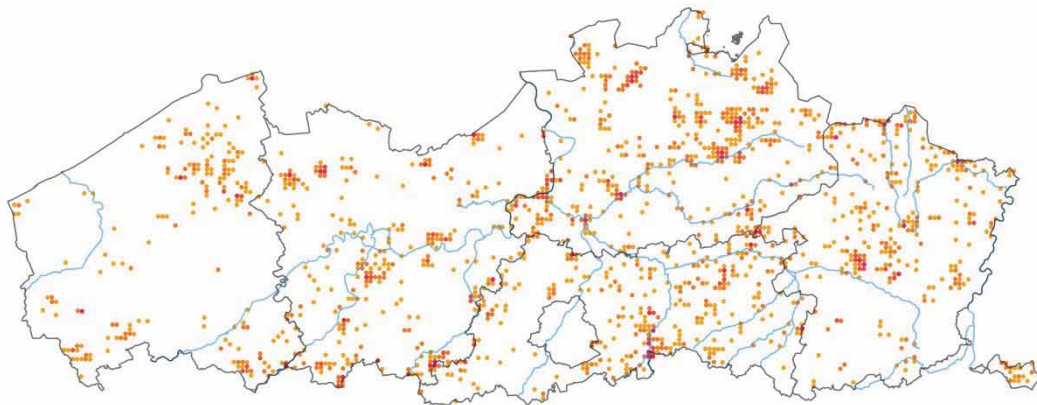
Potentiële leefgebiedenkaart voor Wespendief op basis van het onderstaande gdx-script.
Groen = voortplantingsgebied, oranje = foerageergebied.



Potentiële verspreidingskaart voor Wespendief op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Wespendief per kilometerhok in Vlaanderen (Tabel 7).
Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Wespandief uit het data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

Gdx-script

vlaanderen_20m.asc

// baken de voortplantingsgebieden af

```
assign(wespendief_voortplanting_bwk, bwk(fa%, fe%, fk%, fl%, fm%, fs%, lh%, ls%, pa%, pi%, pm%, pp%, ppa%,
    ppi%, ppmb%, ppmh%, ppms%, qa%, qb%, qe%, qk%, ql%, qs%, n, n-, n+, va%, vc%, vf%, vm%, vn%, vo%, vt%,
    9110%, 9120%, 9130%, 9150%, 9160%, 9190%, 91E0%, 91F0%, rbbppm%))
```

// baken oude bossen af

```
assign(wespendief_bosinventaris, OrEq('bosinventaris_2001.asc', 12, 13, 14, 22, 23, 24))
```

```
assign(wespendief_voortplanting1, and(wespendief_voortplanting_bwk, wespendief_bosinventaris))
```

```
deleteVariable(wespendief_bosinventaris)
```

```
deleteVariable(wespendief_voortplanting_bwk)
```

// clustering: indien geschikte cel binnen 100 meter, dan hoort die cel bij het voortplantingscluster

```
assign(wespendief_voortplanting_cluster, fuzzyClusterIdU(wespendief_voortplanting1, 100, meter))
```

```
deleteVariable(wespendief_voortplanting1)
```

// voortplantingscluster moeten minimaal 50ha groot zijn

```
assign(wespendief_bos, cFilterGeArea(wespendief_voortplanting_cluster, 50, hectare))
```

```
deleteVariable(wespendief_voortplanting_cluster)
```

// baken open foerageergebied af

```
assign(wespendief_foerageer_open, bwk(cd%, cg%, cm%, cp%, cv%, ha%, hc%, hk%, hm%, hme%, hmm%, hmo%,
    hn%, hu%, kb%, kh%, kn%, kp%, kpa%, kpk%, kw%, sf%, sg%, sk%, sm%, hp%, hp+, hpr%, hpr+, hr%, cdb%,
    cgb%, cmb%, cpb%, cvb%, hab%, hcb%, hmb%, n, n+, n-, hub%, mdb%, sgb%, 2190%, 2310%, 2330%, 3130%,
    3140%, 3160%, 4010%, 4030%, 6120%, 6210%, 6230%, 6410%, 6510%, 7140%, 7230%, rbbhc%, rbbkam%, rbbfsf%,
    rbbsg%, rbbsm%, rbbsp%))
```

// clustering: indien geschikte cel binnen 500 meter, dan hoort die cel bij het foerageercluster

```
assign(wespendief_foerageer_open_cluster, fuzzyClusterIdU(wespendief_foerageer_open, 500, meter))
```

// open foerageercluster moet minstens 50 ha zijn

```
assign(wespendief_openfoerageergebied, cFilterGeArea(wespendief_foerageer_open_cluster, 50, hectare))
```

```
deleteVariable(wespendief_foerageer_open)
```

```
deleteVariable(wespendief_foerageer_open_cluster)
```

```
assign(wespendief_foerageer1, or(wespendief_bos, wespendief_openfoerageergebied))
```

```
deleteVariable(wespendief_openfoerageergebied)
```

// clustering: indien geschikte cel binnen 500 meter, dan hoort die cel bij het foerageercluster open

```
assign(wespendief_foerageer1_cluster, fuzzyClusterIdU(wespendief_foerageer1, 500, meter))
```

```
deleteVariable(wespendief_foerageer1)
```

// optimale foerageerclusters moeten minimaal 250ha groot zijn

```
assign(wespendief_foerageer2, cFilterGeArea(wespendief_foerageer1_cluster, 250, hectare))
```

```
deleteVariable(wespendief_foerageer1_cluster)
```

// geschikte foerageergebieden liggen minder dan 5 km van voortplantingsgebieden en omgekeerd

```
assign(wespendief_voortplanting0, cFilter(wespendief_bos, cellDistanceLe(wespendief_foerageer2, 5000, meter)))
```

```
assign(wespendief_foerageer0, and(wespendief_foerageer2, cellDistanceLe(wespendief_bos, 5000, meter)))
```

```
deleteVariable(wespendief_foerageer2)
```

```

deleteVariable(wespendief_bos)
assign(wespendief_leefgebied1, or(wespendief_voortplanting0, wespendief_foerageer0))
assign(wespendief_leefgebied1_cluster, FuzzyClusterIdU(wespendief_leefgebied1, 500, meter))
assign(wespendief_leefgebied, cFilterGeArea(wespendief_leefgebied1_cluster, 250, hectare))
//write('wespendief_leefgebied.asc', if(wespendief_leefgebied, wespendief_leefgebied1_cluster, 0))
//write('wespendief_leefgebied.asc', IsNonZero(wespendief_leefgebied))
//deleteVariable(wespendief_leefgebied1)
//deleteVariable(wespendief_leefgebied1_cluster)
assign(wespendief_foerageer, and(wespendief_foerageer0, wespendief_leefgebied))
write('wespendief_foerageer.asc', wespendief_foerageer)
//deleteVariable(wespendief_foerageer0)
//deleteVariable(wespendief_foerageer)
assign(wespendief_voortplanting, and(wespendief_voortplanting0, wespendief_leefgebied))
//write('wespendief_voortplanting.asc', wespendief_voortplanting)
//deleteVariable(wespendief_leefgebied)
//deleteVariable(wespendief_voortplanting0)
//deleteVariable(wespendief_voortplanting)

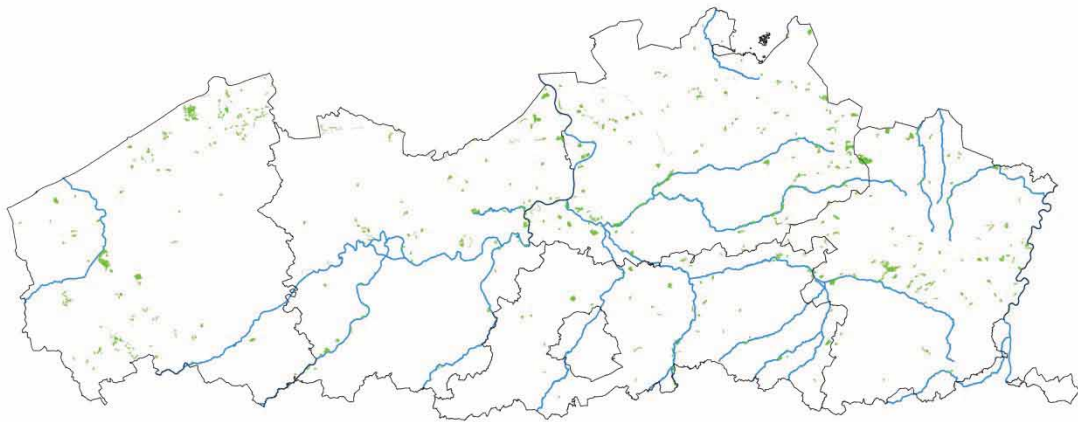
```



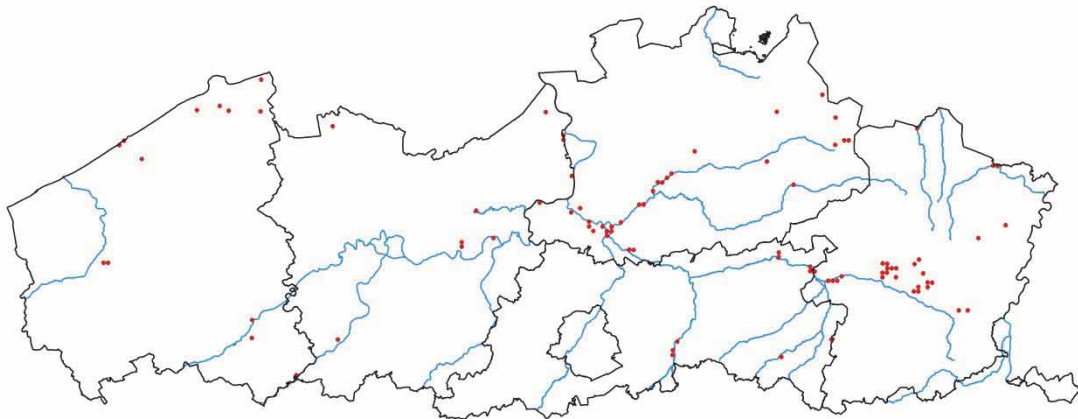
Wespendief (foto Glenn vermeersch)



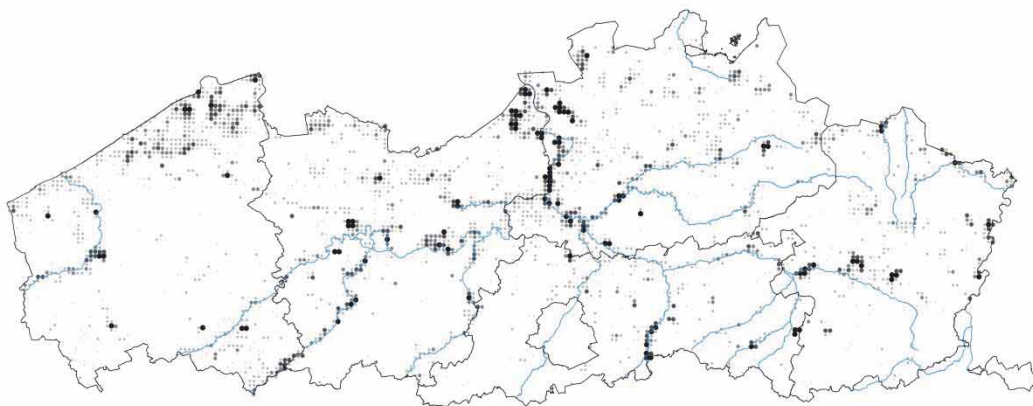
Visdief (foto Glenn Vermeersch)



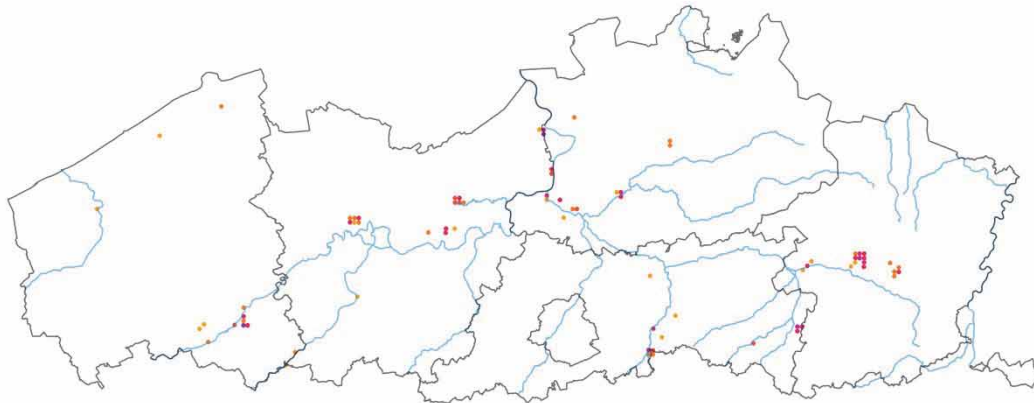
Potentiële leefgebiedenkaart voor Woudaap op basis van het onderstaande.gdx-script.



Potentiële verspreidingskaart voor Woudaap op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Woudaap per kilometerhok in Vlaanderen (Tabel 7).
Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.

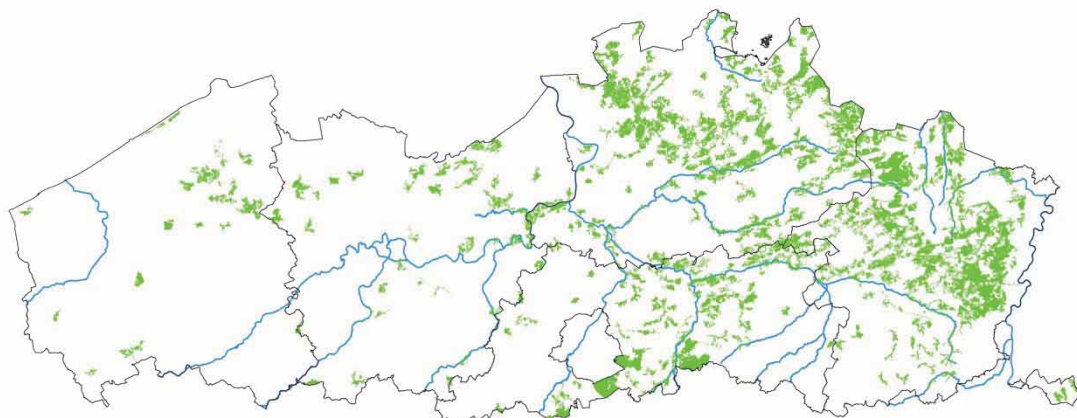


Waarnemingen van Woudaap uit het data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

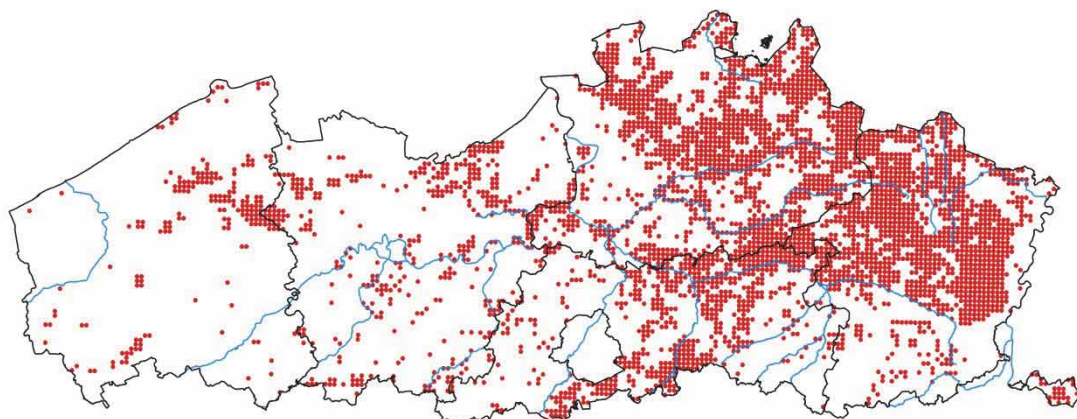
Gdx-script

```
vlaanderen_20m.asc
// broedplaatsen: rietland en moerassen
assign(woudaap_voortplanting_bwk, bwk(ad%, ae%, aer%, aev%, ah%, aom%, ap, ap+, apo, apo+, kn%, mc%, mcb%,
mr%, mrb%, sf%, 2190%, 3110%, 3130%, 3140%, 3150%, 3160%, 3260%, 3270%, 6430%, 7140%, 91E0%, rbbmc%,
rbbmr%, rbbmf%))
// mag geen populier zijn
assign(populier, bwk(l%))
assign(woudaap_voortplanting_bwk1, and(woudaap_voortplanting_bwk, not(populier)))
deleteVariable(woudaap_voortplanting_bwk)
deleteVariable(populier)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het cluster
assign(woudaap_voortplanting_cluster, fuzzyClusterIdU(woudaap_voortplanting_bwk1, 50, meter))
deleteVariable(woudaap_voortplanting_bwk1)
// clusters minimaal 10 ha
assign(woudaap_voortplanting2, cFilterGeArea(woudaap_voortplanting_cluster, 10, hectare))
deleteVariable(woudaap_voortplanting_cluster)
// er moet minimum 1 ha open water in het voortplantingsgebied liggen
assign(woudaap_openwater, bwk(ad%, ae%, aer%, aev%, ah%, aom%, ap, ap+, apo, apo+, 3110%, 3130%, 3140%,
3150%, 3160%, 3260%))
// clustering: indien geschikte cel binnen 10 meter, dan hoort die cel bij het cluster
assign(woudaap_openwater_cluster, fuzzyClusterIdU(woudaap_openwater, 10, meter))
deleteVariable(woudaap_openwater)
assign(woudaap_openwater1, cFilterGeArea(woudaap_openwater_cluster, 1, hectare))
deleteVariable(woudaap_openwater_cluster)
assign(woudaap_leefgebied1, cFilter(woudaap_voortplanting2, cellDistanceLe(woudaap_openwater1, 100, meter)))
deleteVariable(woudaap_voortplanting2)
deleteVariable(woudaap_openwater1)
assign(woudaap_leefgebied1_cluster, FuzzyClusterIdU(woudaap_leefgebied1, 100, meter))
deleteVariable(woudaap_leefgebied1)
assign(woudaap_leefgebied, cFilterGeArea(woudaap_leefgebied1_cluster, 10, hectare))
write('woudaap_leefgebied.asc', if(woudaap_leefgebied, woudaap_leefgebied1_cluster, 0))
// write('woudaap_leefgebied.asc', IsNonZero(woudaap_leefgebied))
deleteVariable(woudaap_leefgebied1_cluster)
deleteVariable(woudaap_leefgebied)
```

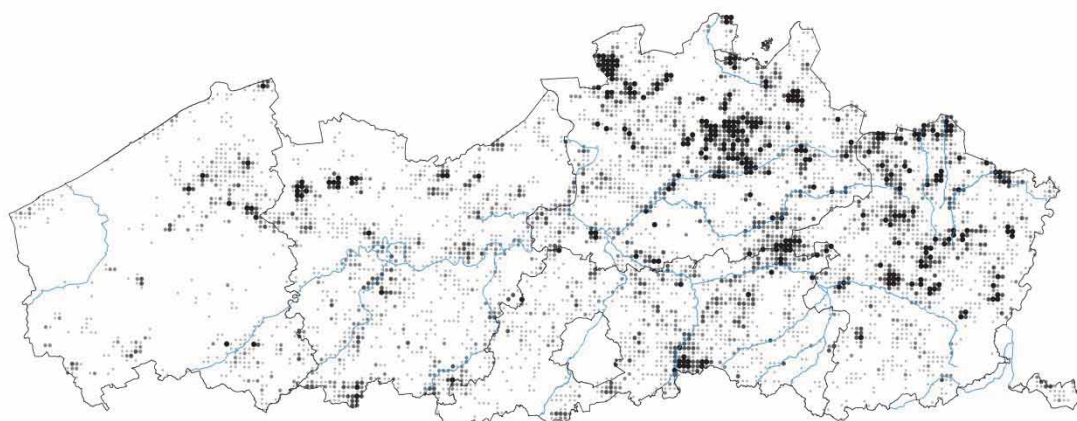
Zwarte specht (*Dryocopus martius*) – Adriaens & Vermeersch (in druk)



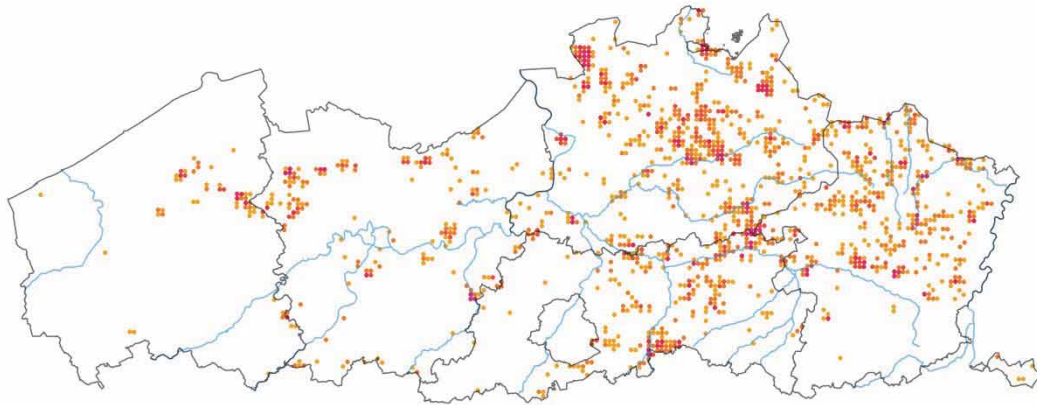
Potentiële leefgebiedenkaart voor Zwarte specht op basis van het onderstaande gdx-script.



Potentiële verspreidingskaart voor Zwarte specht op basis van correlatieve modellen.



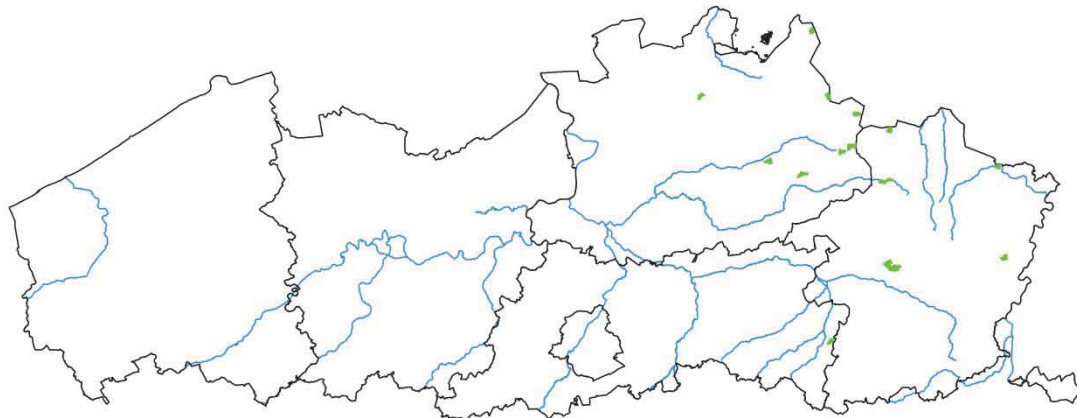
Som van de indicatorwaarde van begeleidende soorten van Zwarte specht per kilometerhok in Vlaanderen (Tabel 7). Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Zwarte specht uit het data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(zwartespecht_bwk, bwk(fa%, fe%, fk%, fl%, fm%, fs%, kp%, kpa%, kpk%, pa%, pi%, pm%, pmb%, pmh%,
    pms%, pop%, pp%, ppa%, ppi%, ppmb%, ppmh%, ppms%, qa%, qb%, qe%, qk%, ql%, qs%, ru%, rud%, sf%, va%,
    vc%, vf%, vm%, vn%, vo%, vt%, lh%, ls%, n, n+, n-))
// baken oude bossen af
assign(zwartespecht_bosinventaris, OrEq('bosinventaris_2001.asc', 12, 13, 14, 22, 23, 24))
assign(zwartespecht_voortplanting1, or(zwartespecht_bwk, zwartespecht_bosinventaris))
// clustering: indien geschikte cel binnen 100 meter, dan hoort die cel bij het voortplantingscluster
assign(zwartespecht_voortplanting1_cluster, fuzzyClusterId(zwartespecht_voortplanting1, 100/cellSize))
// optimale voortplantingsclusters moeten minimaal 300ha groot zijn
assign(zwartespecht_leefgebied_300ha, cFilterGeArea(zwartespecht_voortplanting1_cluster, 300, hectare))
// bossen tussen 50 en 300 ha zijn ook geschikt
assign(zwartespecht_leefgebied_kleinerdan300ha, cFilterLeArea(zwartespecht_voortplanting1_cluster, 300,
    hectare))
assign(zwartespecht_leefgebied_100_300ha, cFilterGeArea(zwartespecht_leefgebied_kleinerdan300ha, 50,
    hectare))
// minder dan 4 km van grotere bossen
assign(zwartespecht_leefgebied_100_300ha_geschikt, cFilter(zwartespecht_leefgebied_100_300ha,
    cellDistanceLe(zwartespecht_leefgebied_300ha, 4000, meter)))
// kleinere bossen minder dan 4 km van elkaar
assign(zwartespecht_leefgebied_100_300ha_geschikt2, cFilter(zwartespecht_leefgebied_100_300ha,
    cellDistanceLe(zwartespecht_leefgebied_100_300ha, 4000, meter)))
// zowel bossen groter dan 300ha zijn geschikt, maar ook bossen groter dan 100ha op voorwaarde dat ze op minder
    dan 4 km van een groot bos of van elkaar liggen
assign(zwartespecht_leefgebied1, or(zwartespecht_leefgebied_300ha,
    zwartespecht_leefgebied_100_300ha_geschikt2))
assign(zwartespecht_ecodistrict, ge('ecoregio.asc', 1))
assign(zwartespecht_leefgebied_ecodistrict1, and(zwartespecht_leefgebied1, zwartespecht_ecodistrict))
assign(zwartespecht_leefgebied_ecodistrict1_cluster, FuzzyClusterId(zwartespecht_leefgebied_ecodistrict1,
    100/cellSize))
assign(zwartespecht_leefgebied_ecodistrict, cFilterGeArea(zwartespecht_leefgebied_ecodistrict1_cluster, 100,
    hectare))
// write('zwartespecht_leefgebied_ecodistrict.asc', if(zwartespecht_leefgebied_ecodistrict,
    zwartespecht_leefgebied_ecodistrict1_cluster, 0))
write('zwartespecht_leefgebied_ecodistrict.asc', IsNonZero(zwartespecht_leefgebied_ecodistrict))
```



Potentiële leefgebiedenkaart voor Zwarte stern op basis van het onderstaande gdx-script.

Gdx-script

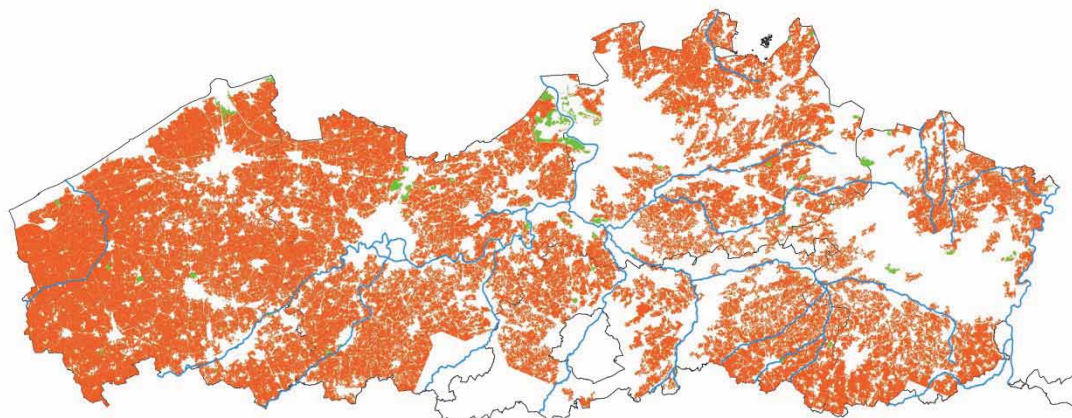
Vlaanderen_20m_met_maas.asc

```
// baken geschikte waterbiotopen af; zelf enkele moerastypes toegevoegd op basis van LSVI tekst
assign(zwartestern_waterbiotoop1, bwk(ae%, aev%, ao%, aom%, apo, apo+, md%, mc%, mk%, ms%))
assign(zwartestern_waterbiotoop1_cluster, fuzzyClusterIdU(zwartestern_waterbiotoop1, 50, meter))
// oppervlakte waterbiotoop moet minstens 250ha zijn
assign(zwartestern_leefgebied, cFilterGeArea(zwartestern_waterbiotoop1_cluster, 50, hectare))
//write('zwartestern_leefgebied.asc', IsNonZero(zwartestern_leefgebied))
// baken ecoregio van ecodistrictgebieden af
assign(zwartestern_ecodistrict, OrEq('ecoregio.asc', 12, 13, 14, 15, 16, 17))
assign(zwartestern_leefgebied_ecodistrict, and(zwartestern_leefgebied, zwartestern_ecodistrict))
write('zwartestern_leefgebied_ecodistrict.asc', IsNonZero(zwartestern_leefgebied_ecodistrict))
```

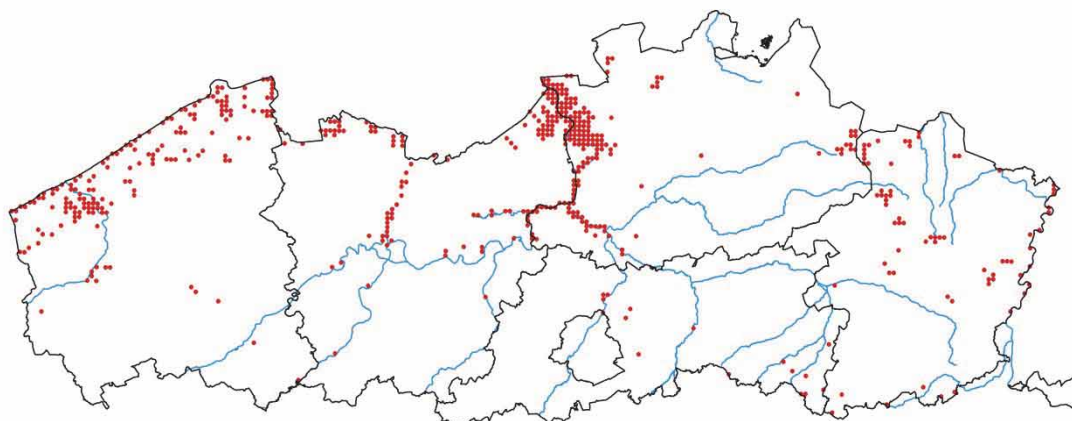


Zwarte stern (foto Glenn Vermeersch)

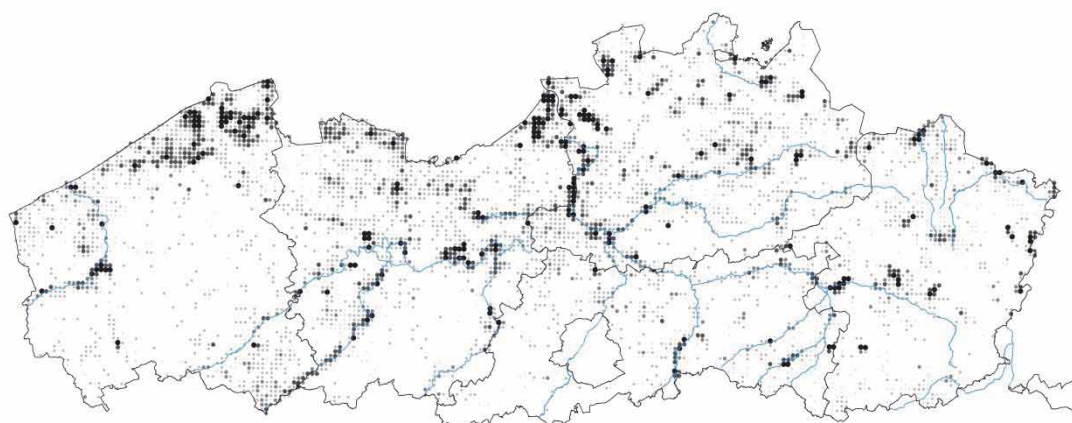
Zwartkopmeeuw (*Larus melanocephalus*) – Adriaens & Vermeersch (in druk)



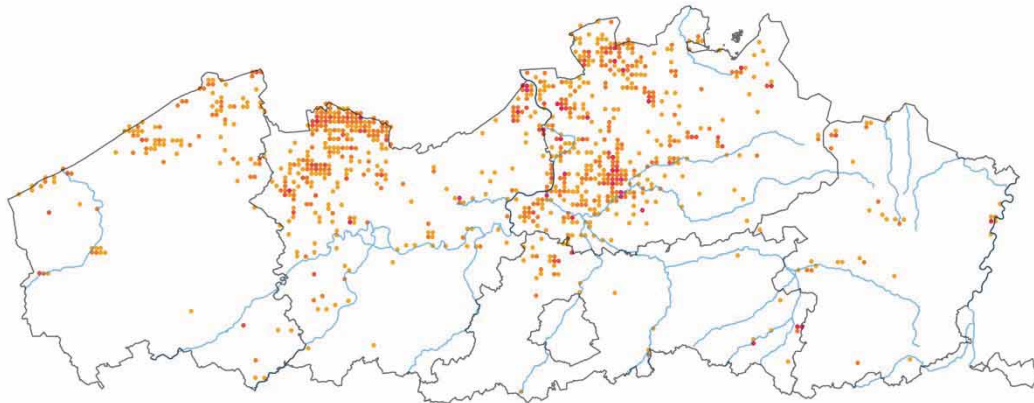
Potentiële leefgebiedenkaart voor Zwartkopmeeuw op basis van het onderstaande gdx-script.
Groen = voortplantingsgebied, oranje = foerageergebied.



Potentiële verspreidingskaart voor Zwartkopmeeuw op basis van correlatieve modellen.



Som van de indicatorwaarde van begeleidende soorten van Zwartkopmeeuw per kilometerhok in Vlaanderen (Tabel 7). Hoe groter en donkerder het symbool, hoe groter de som van de indicatorwaarde van de begeleidende soorten.



Waarnemingen van Zwartkopmeeuw uit het data portaal www.waarnemingen.be (enkel vermoedelijke broedgevallen in de periode 2007-2014). Hoe donkerder de kleur, hoe groter het aantal waarnemingen in het kilometerhok.

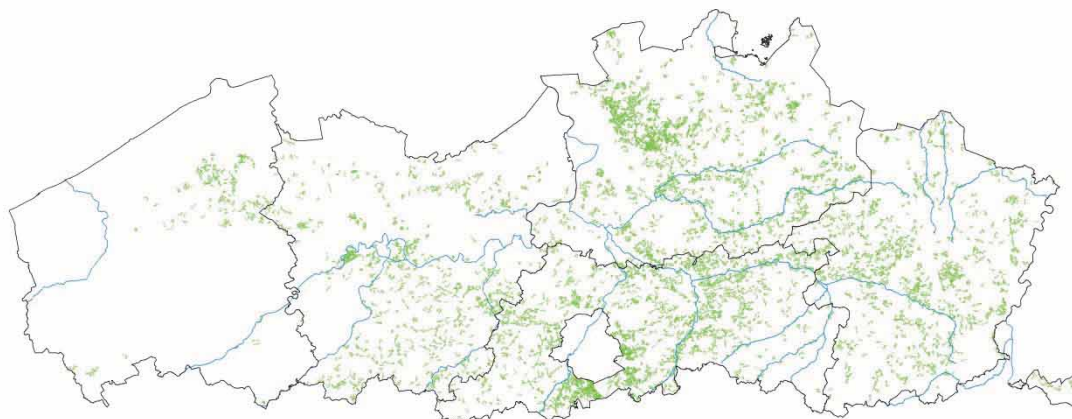
Gdx-script

vlaanderen_20m.asc

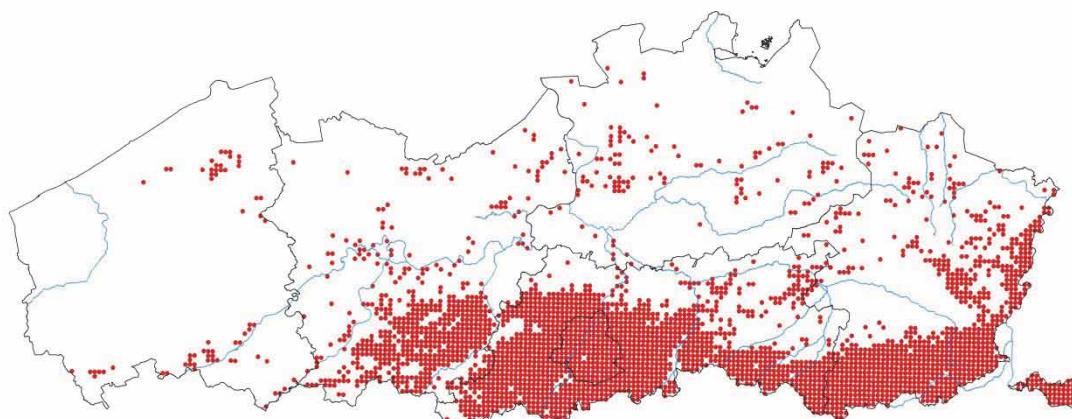
```
// baken de voortplantingsgebieden af
assign(zwartkopmeeuw_voortplanting1, bwk(ad%, ae%, da%, ku, ku+, ku-, kz%))
// voortplantingsclusters moeten minimaal 30ha groot zijn
assign(zwartkopmeeuw_voortplanting1_cluster, fuzzyClusterIdU(zwartkopmeeuw_voortplanting1, 50, meter))
assign(zwartkopmeeuw_voortplanting2, cFilterGeArea(zwartkopmeeuw_voortplanting1_cluster, 30, hectare))
// moet in de buurt van water liggen: 5 km?
assign(zwartkopmeeuw_water1, bwk(ad%, ae%, aer%, aev%, ah%, ao%, aom%, aoo%, ap%, apo%, app%, zee%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het cluster en water moet minstens 2 ha groot zijn
assign(zwartkopmeeuw_water1_cluster, fuzzyClusterIdU(zwartkopmeeuw_water1, 50, meter))
assign(zwartkopmeeuw_water2, cFilterGeArea(zwartkopmeeuw_water1_cluster, 2, hectare))
assign(zwartkopmeeuw_voortplanting3, and(zwartkopmeeuw_voortplanting2,
    cellDistanceLe(zwartkopmeeuw_water2, 5000, meter)))
assign(zwartkopmeeuw_water3, and(zwartkopmeeuw_water2, cellDistanceLe(zwartkopmeeuw_voortplanting3,
    5000, meter)))
// voortplantingscluster moet vrij dicht bij vochtige weilanden liggen, 5km?
assign(zwartkopmeeuw_foerageer1, bwk(bl%, bs%, bu%, hp%, hpr%, hx%, ku%))
// foerageerclusters moeten minimaal 200ha groot zijn
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het cluster
assign(zwartkopmeeuw_foerageer1_cluster, fuzzyClusterIdU(zwartkopmeeuw_foerageer1, 50, meter))
assign(zwartkopmeeuw_foerageer2, cFilterGeArea(zwartkopmeeuw_foerageer1_cluster, 200, hectare))
// geschikte foerageergebieden liggen minder dan 20 km van voortplantingsgebieden en omgekeerd
assign(zwartkopmeeuw_voortplanting4, and(zwartkopmeeuw_voortplanting3,
    cellDistanceLe(zwartkopmeeuw_foerageer2, 20000, meter)))
assign(zwartkopmeeuw_foerageer4, and(zwartkopmeeuw_foerageer2,
    cellDistanceLe(zwartkopmeeuw_voortplanting4, 20000, meter)))
assign(zwartkopmeeuw_leefgebied0, or(zwartkopmeeuw_voortplanting4, zwartkopmeeuw_foerageer4))
assign(zwartkopmeeuw_leefgebied0_cluster, FuzzyClusterIdU(zwartkopmeeuw_leefgebied0, 50, meter))
assign(zwartkopmeeuw_leefgebied, cFilterGeArea(zwartkopmeeuw_leefgebied0_cluster, 200, hectare))
// write('zwartkopmeeuw_leefgebied.asc', if(zwartkopmeeuw_leefgebied, zwartkopmeeuw_leefgebied0_cluster,
    0))
//write('zwartkopmeeuw_leefgebied.asc', IsNonZero(zwartkopmeeuw_leefgebied))
assign(zwartkopmeeuw_foerageer, and(zwartkopmeeuw_foerageer4, zwartkopmeeuw_leefgebied))
write('zwartkopmeeuw_foerageer.asc', IsNonZero(zwartkopmeeuw_foerageer))
assign(zwartkopmeeuw_voortplanting, and(zwartkopmeeuw_voortplanting4, zwartkopmeeuw_leefgebied))
write('zwartkopmeeuw_voortplanting.asc', zwartkopmeeuw_voortplanting)
```

KEVERS

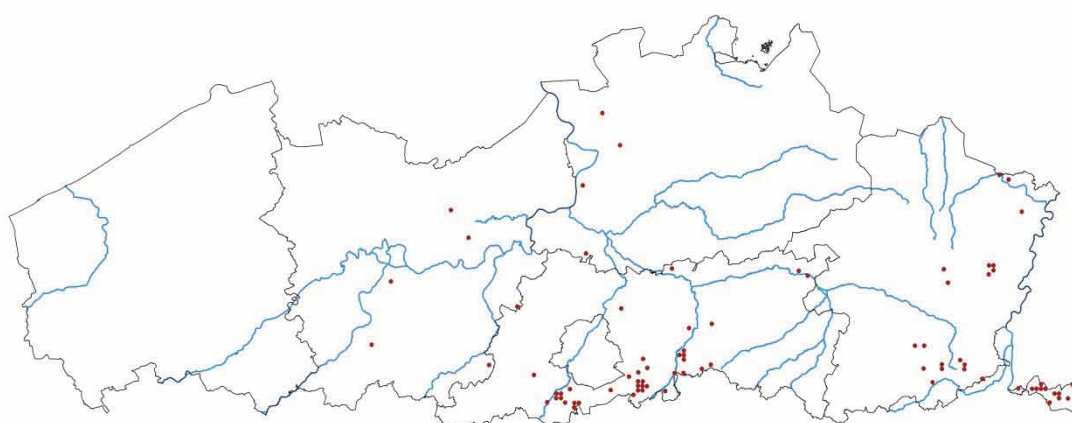
Vliegend hert (*Lucanus cervus*) – Thomaes (2015)



Potentiële leefgebiedenkaart voor Vliegend hert op basis van het onderstaande gdx-script.



Potentiële verspreidingskaart voor Vliegend hert op basis van correlatieve modellen.



Waarnemingen van Vliegend hert in Vlaanderen (Thomaes et al. 2015).

Gdx-script

```
vlaanderen_20m.asc
// baken hooggroen af
assign(vliegendhert_groenkaart1, Eq('groenkaart2013_20m.asc', 1))
// het mag geen naaldbos of broekbos zijn
assign(vliegendhert_naaldbos, bwk(pa%, pi%, pm%, pp%, vm%, vn%, vo%, vt%))
assign(vliegendhert_bossen1, and(vliegendhert_groenkaart1, not(vliegendhert_naaldbos)))
deleteVariable(vliegendhert_naaldbos)
deleteVariable(vliegendhert_groenkaart1)
// baken oude bossen af op bosleeftijdkaart
assign(vliegendhert_bosleeftijd12, OrEq('bosleeftijd.asc', 1, 2))
// leefgebied mag tot 1000m van oud bos liggen
assign(vliegendhert_leefgebied1, and(vliegendhert_bossen1, cellDistanceLe(vliegendhert_bosleeftijd12, 1000,
meter)))
//write('vliegendhert_leefgebied1.asc', vliegendhert_leefgebied1)
deleteVariable(vliegendhert_bosleeftijd12)
// zuidhellingen mogen tot op 1000m van bossen liggen
assign(zuidhelling, 'vliegend_hert_hellingen_zozw_4gr.asc')
assign(vliegendhert_zuidhelling, eq(zuidhelling, 1))
deleteVariable(zuidhelling)
assign(vliegendhert_leefgebied2, and(vliegendhert_bossen1, cellDistanceLe(vliegendhert_zuidhelling, 1000,
meter)))
//write('vliegendhert_leefgebied2.asc', vliegendhert_leefgebied2)
deleteVariable(vliegendhert_bossen1)
assign(vliegendhert_leefgebied3, or(vliegendhert_leefgebied1, vliegendhert_leefgebied2))
//write('vliegendhert_leefgebied3.asc', vliegendhert_leefgebied3)
deleteVariable(vliegendhert_zuidhelling)
deleteVariable(vliegendhert_leefgebied1)
deleteVariable(vliegendhert_leefgebied2)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het voortplantingscluster
assign(vliegendhert_leefgebied3_cluster, fuzzyClusterIdU(vliegendhert_leefgebied3, 50, meter))
deleteVariable(vliegendhert_leefgebied3)
// leefgebiedcluster moeten minimaal 0.5 ha groot zijn
assign(vliegendhert_leefgebied4, cFilterGeArea(vliegendhert_leefgebied3_cluster, 0.5, hectare))
//write('vliegendhert_leefgebied4.asc', vliegendhert_leefgebied4)
deleteVariable(vliegendhert_leefgebied3_cluster)
// baken open gebieden die naast bossen moet liggen af
assign(vliegendhert_open, OrEq('groenkaart2013_20m.asc', 2, 3, 4))
// clustering: indien geschikte cel binnen 100 meter, dan hoort die cel bij het opencluster
assign(vliegendhert_open_cluster, fuzzyClusterIdU(vliegendhert_open, 100, meter))
deleteVariable(vliegendhert_open)
// open cluster moeten minimaal 1 ha groot zijn
assign(vliegendhert_open1, cFilterGeArea(vliegendhert_open_cluster, 1, hectare))
//write('vliegendhert_open1.asc', vliegendhert_open1)
deleteVariable(vliegendhert_open_cluster)
// bos en open moeten elkaar raken
assign(vliegendhert_edge1, edgeDirected(vliegendhert_leefgebied4, vliegendhert_open1, 0,0,1,1,1,0,0,0))
//write('vliegendhert_edge1.asc', vliegendhert_edge1)
assign(vliegendhert_edge2, edgeDirected(vliegendhert_open1, vliegendhert_leefgebied4, 1,0,0,0,0,0,1,1))
//write('vliegendhert_edge2.asc', vliegendhert_edge2)
deleteVariable(vliegendhert_open1)
assign(vliegendhert_bosrand1, or(vliegendhert_edge1, vliegendhert_edge2))
//write('vliegendhert_bosrand1.asc', vliegendhert_bosrand1)
deleteVariable(vliegendhert_edge1)
deleteVariable(vliegendhert_edge2)
// lengte van bosrand tussen voortplantingsclusters en boscluster moet minstens 100 meter zijn en pixels moeten
    tegen elkaar liggen dus afstand <10m
assign(vliegendhert_bosrand_cluster, fuzzyClusterIdU(vliegendhert_bosrand1, 20, meter))
assign(vliegendhert_leefgebied5, cFilterGeArea(vliegendhert_bosrand_cluster, 0.2, hectare))
deleteVariable(vliegendhert_bosrand_cluster)
// aan geschikte bosrand, bos meenemen tot 40m in het bos
```

```

assign(vliegendhert_bosrand2, and(vliegendhert_leefgebied5, cellDistanceLe(vliegendhert_leefgebied4, 40,
meter)))
deleteVariable(vliegendhert_leefgebied5)
deleteVariable(vliegendhert_leefgebied4)
assign(vliegendhert_leefgebied, or(vliegendhert_bosrand1, vliegendhert_bosrand2))
deleteVariable(vliegendhert_bosrand1)
deleteVariable(vliegendhert_bosrand2)
assign(vliegendhert_leefgebied_cluster, fuzzyClusterIdU(vliegendhert_leefgebied, 50, meter))
deleteVariable(vliegendhert_leefgebied_cluster)
// eventueel nog ecodistricten afbakenen en gebruiken om verder te verfijnen
assign(vliegendhert_ecodistrict, OrEq('ecoregio.asc', 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 22, 23, 24, 25, 26, 27,
28, 29, 30, 31, 32, 33, 34, 35, 36))
assign(vliegendhert_leefgebied1_ecodistrict, and(vliegendhert_leefgebied, vliegendhert_ecodistrict))
deleteVariable(vliegendhert_leefgebied)
deleteVariable(vliegendhert_ecodistrict)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het voortplantingscluster
assign(vliegendhert_leefgebied1_ecodistrict_cluster, fuzzyClusterIdU(vliegendhert_leefgebied1_ecodistrict, 50,
meter))
// leefgebiedcluster moeten minimaal 10ha groot zijn
assign(vliegendhert_leefgebied_ecodistrict, cFilterGeArea(vliegendhert_leefgebied1_ecodistrict_cluster, 10,
hectare))
write('vliegendhert_leefgebied_ecodistrict_v2.asc', IsNonZero(vliegendhert_leefgebied_ecodistrict))
deleteVariable(vliegendhert_leefgebied1_ecodistrict)
deleteVariable(vliegendhert_leefgebied1_ecodistrict_cluster)
deleteVariable(vliegendhert_leefgebied_ecodistrict)

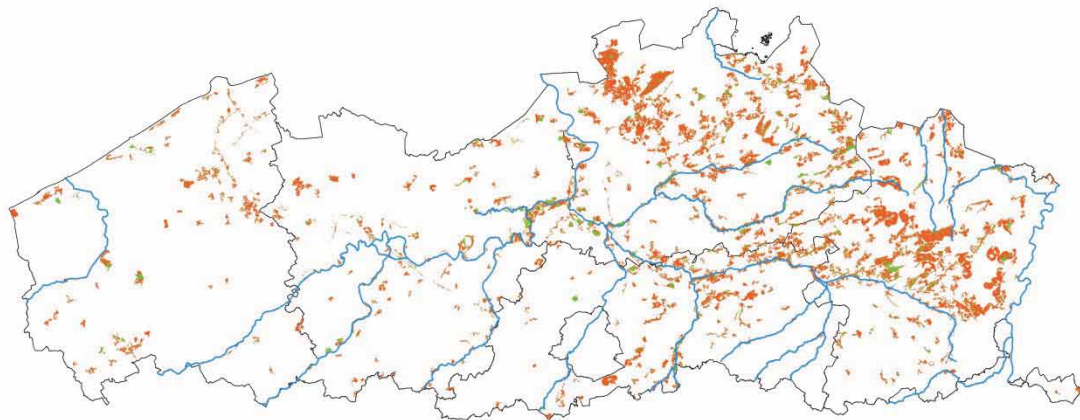
```



Vliegend hert (foto Lars Soerink – Vilda)

LIBELLEN

Gevlekte witsnuitlibel (*Leucorrhinia pectoralis*) – De Knijf & Adriaens (2015)



Potentiële leefgebiedenkaart voor Gevlekte witsnuitlibel op basis van het onderstaande gdx-script.
Groen = voortplantingsbiotoop, oranje = foerageerbiotoop.

Gdx-script

vlaanderen_20m.asc

// baken de voortplantingsgebieden af

```
assign(gevektewitsnuitlibel_voortplanting_bwk, bwk(ae, ae+, ae-, aer, aer+, aer-, aev, aev+, aev-, ao, ao+, ao-, aom, aom+, aom-, aoo, aoo-, 3110%, 3130%, 3140%, 3150%, 3160%))
```

// uit GRB-watgang, kleine waters proberen te filteren

```
assign(GRB_water, eq('GRB_watgang_at.asc', 1))
```

```
assign(GRB_water_cluster, fuzzyClusterIdU(GRB_water, 10, meter))
```

```
deleteVariable(GRB_water)
```

```
assign(gevektewitsnuitlibel_kleinwater, cFilterLeArea(GRB_water_cluster, 0.4, hectare))
```

```
deleteVariable(GRB_water_cluster)
```

```
assign(gevektewitsnuitlibel_waterbiotoop1, or(gevektewitsnuitlibel_voortplanting_bwk, gevektewitsnuitlibel_kleinwater))
```

```
deleteVariable(gevektewitsnuitlibel_voortplanting_bwk)
```

```
deleteVariable(gevektewitsnuitlibel_kleinwater)
```

// binnen een straal van 500m moeten minstens 3 (4-1 dwz 4 - de eigen poel) andere geschikte voortplantingsgebieden liggen

```
assign(gevektewitsnuitlibel_waterbiotoop1_cluster, clusterId(gevektewitsnuitlibel_waterbiotoop1))
```

```
deleteVariable(gevektewitsnuitlibel_waterbiotoop1)
```

```
assign(gevektewitsnuitlibel_waterbiotoop0, GeUniqueCountInBuffer(gevektewitsnuitlibel_waterbiotoop1_cluster, 500, meter, gevektewitsnuitlibel_waterbiotoop1_cluster, 4))
```

```
deleteVariable(gevektewitsnuitlibel_waterbiotoop1_cluster)
```

```
assign(gevektewitsnuitlibel_foerageer0, bwk(cd%, ce%, cg%, cm%, cp%, cv%, hf%, hmo%, kn%, ku%, mc%, md%, mk%, mm%, mp%, mr%, ms%, qb%, sm%, so%, sz%, t%, vf%, vm%, vn%, vo%, qs%, ppmb%, ppms%, ppmh%, pmb%, pmh%, hr%, hc%, hu%, va%, vc%, sf%, se%, sp%, sd%, ru%, qa%, qe%, fe%, fa%, fs%, fm%, n, n+, n-, lhb%, lhi%, qb%, sm%, so%, sz%, t%, vf%, vm%, vn%, vo%, qs%, ppmb%, ppms%, ppmh%, pmb%, pmh%, mr%, mc%, mm%, md%, va%, vc%, sf%, se%, sp%, sd%, ru%, qa%, qe%, fe%, fa%, fs%, fm%, n, n+, n-, lhb+, lhi+, 2180%, 2190%, 2310%, 3110%, 3130%, 3140%, 3150%, 3160%, 4010%, 4030%, 6230%, 6410%, 6430%, 7110%, 7140%, 7150%, 7210%, 7230%, 9120%, 9190%, 91E0%, rbbhf%, rbbmc%, rbbmr%, rbbms%, rbbsf%, rbbsm%, rbbso%, rbbppm%, 6510%, 5130%, 9110%, 9130%, 9150%, 9160%))
```

// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het foerageercluster

```
assign(gevektewitsnuitlibel_foerageer1_cluster, fuzzyClusterIdU(gevektewitsnuitlibel_foerageer0, 50, meter))
```

```
deleteVariable(gevektewitsnuitlibel_foerageer0)
```

// foerageergebied moet minstens 35ha groot zijn

```
assign(gevektewitsnuitlibel_foerageer2, cFilterGeArea(gevektewitsnuitlibel_foerageer1_cluster, 35, hectare))
```

```
deleteVariable(gevektewitsnuitlibel_foerageer1_cluster)
```

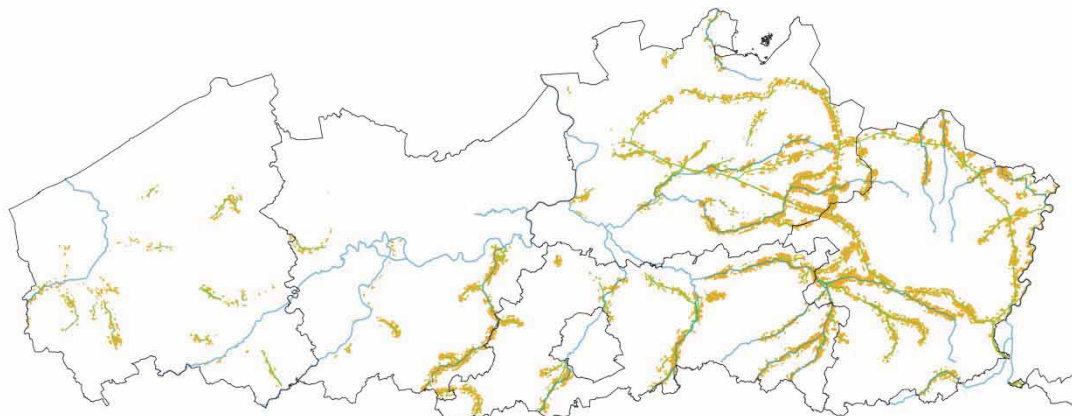
// baken struweel en bos af

```
assign(gevektewitsnuitlibel_struweel1, bwk(qb%, sm%, so%, sz%, t%, vf%, vm%, vn%, vo%, qs%, ppmb%, ppms%, ppmh%, pmb%, pmh%, mr%, mc%, mm%, md%, va%, vc%, sf%, se%, sp%, sd%, ru%, qa%, qe%, fe%, fa%, fs%, fm%,
```

```

n, n+, n-, lhb+, lhi+, 9110%, 9120%, 9130%, 9150%, 9160%, 9190%, rbbhf%, rbbmc%, rbbmr%, rbbms%, rbbppm%,
rbsf%, rbsm%, rbsso%, 91E0%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het struweelcluster
assign(gevektewitsnuitlibel_struweel1_cluster, fuzzyClusterIdU(gevektewitsnuitlibel_struweel1, 50, meter))
deleteVariable(gevektewitsnuitlibel_struweel1)
// minstens 30% van foerageergebied moet struweel of bos zijn (10ha)
assign(gevektewitsnuitlibel_struweel2, cFilterGeArea(gevektewitsnuitlibel_struweel1_cluster, 10, hectare))
deleteVariable(gevektewitsnuitlibel_struweel1_cluster)
assign(gevektewitsnuitlibel_foerageer3, and(gevektewitsnuitlibel_foerageer2,
  cellDistanceLe(gevektewitsnuitlibel_struweel2, 100, meter)))
deleteVariable(gevektewitsnuitlibel_foerageer2)
deleteVariable(gevektewitsnuitlibel_struweel2)
// zoek voortplantingsgebieden die binnen de 800m van foerageergebieden liggen
assign(gevektewitsnuitlibel_voortplanting1, and(gevektewitsnuitlibel_waterbiotoop0,
  cellDistanceLe(gevektewitsnuitlibel_foerageer3, 800, meter)))
assign(gevektewitsnuitlibel_foerageer1, and(gevektewitsnuitlibel_foerageer3,
  cellDistanceLe(gevektewitsnuitlibel_waterbiotoop0, 800, meter)))
deleteVariable(gevektewitsnuitlibel_foerageer3)
deleteVariable(gevektewitsnuitlibel_waterbiotoop0)
assign(gevektewitsnuitlibel_leefgebied1, or(gevektewitsnuitlibel_voortplanting1,
  gevektewitsnuitlibel_foerageer1))
assign(gevektewitsnuitlibel_leefgebied1_cluster, FuzzyClusterIdU(gevektewitsnuitlibel_leefgebied1, 50, meter))
deleteVariable(gevektewitsnuitlibel_leefgebied1)
assign(gevektewitsnuitlibel_leefgebied, cFilterGeArea(gevektewitsnuitlibel_leefgebied1_cluster, 35, hectare))
assign(gevektewitsnuitlibel_voortplanting, and(gevektewitsnuitlibel_leefgebied,
  gevektewitsnuitlibel_voortplanting1))
deleteVariable(gevektewitsnuitlibel_voortplanting1)
assign(gevektewitsnuitlibel_foerageer, and(gevektewitsnuitlibel_leefgebied, gevektewitsnuitlibel_foerageer1))
deleteVariable(gevektewitsnuitlibel_foerageer1)
// baken ecoregio van ecodistrictgebieden af
assign(gevektewitsnuitlibel_ecodistrict, ge('ecoregio.asc', 1))
assign(gevektewitsnuitlibel_leefgebied_ecodistrict1, and(gevektewitsnuitlibel_leefgebied,
  gevektewitsnuitlibel_ecodistrict))
deleteVariable(gevektewitsnuitlibel_leefgebied)
assign(gevektewitsnuitlibel_leefgebied_ecodistrict1_cluster,
  FuzzyClusterIdU(gevektewitsnuitlibel_leefgebied_ecodistrict1, 50, meter))
deleteVariable(gevektewitsnuitlibel_leefgebied_ecodistrict1)
assign(gevektewitsnuitlibel_leefgebied_ecodistrict,
  cFilterGeArea(gevektewitsnuitlibel_leefgebied_ecodistrict1_cluster, 35, hectare))
//write('gevektewitsnuitlibel_leefgebied_ecodistrict.asc', if(gevektewitsnuitlibel_leefgebied_ecodistrict,
  gevektewitsnuitlibel_leefgebied_ecodistrict1_cluster, 0))
deleteVariable(gevektewitsnuitlibel_leefgebied_ecodistrict1_cluster)
assign(gevektewitsnuitlibel_voortplanting_ecodistrict1, and(gevektewitsnuitlibel_voortplanting,
  gevektewitsnuitlibel_ecodistrict))
deleteVariable(gevektewitsnuitlibel_voortplanting)
assign(gevektewitsnuitlibel_voortplanting_ecodistrict, and(gevektewitsnuitlibel_voortplanting_ecodistrict1,
  gevektewitsnuitlibel_leefgebied_ecodistrict))
deleteVariable(gevektewitsnuitlibel_leefgebied1_cluster)
deleteVariable(gevektewitsnuitlibel_voortplanting_ecodistrict1)
write('gevektewitsnuitlibel_voortplanting_ecodistrict.asc', gevektewitsnuitlibel_voortplanting_ecodistrict)
deleteVariable(gevektewitsnuitlibel_voortplanting_ecodistrict)
assign(gevektewitsnuitlibel_foerageer_ecodistrict1, and(gevektewitsnuitlibel_foerageer,
  gevektewitsnuitlibel_ecodistrict))
deleteVariable(gevektewitsnuitlibel_foerageer)
assign(gevektewitsnuitlibel_foerageer_ecodistrict, and(gevektewitsnuitlibel_foerageer_ecodistrict1,
  gevektewitsnuitlibel_leefgebied_ecodistrict))
deleteVariable(gevektewitsnuitlibel_foerageer_ecodistrict1)
write('gevektewitsnuitlibel_foerageer_ecodistrict.asc', IsNonZero(gevektewitsnuitlibel_foerageer_ecodistrict))
deleteVariable(gevektewitsnuitlibel_foerageer_ecodistrict)

```



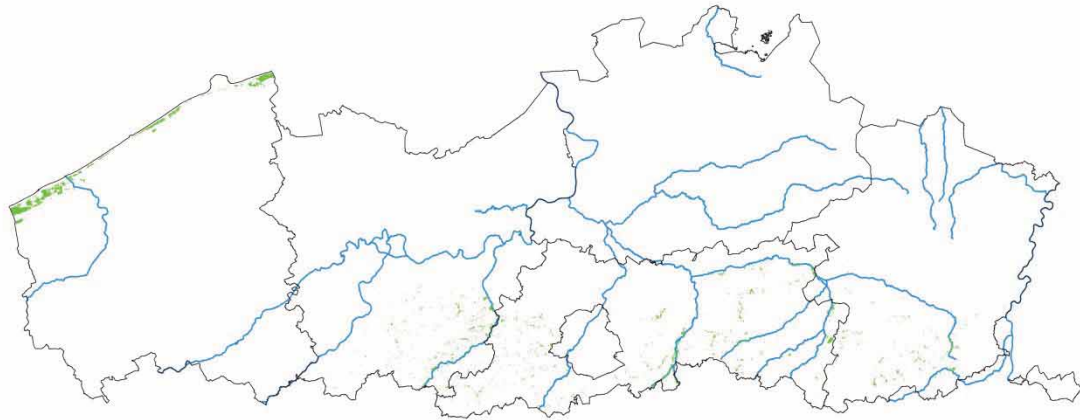
Potentiële leefgebiedenkaart voor Rivierrombout op basis van het onderstaande gdx-script.
Groen = voortplantingsbiotoop, oranje = foerageerbiotoop.

Gdx-script

```
vlaanderen_20m_met_maas.asc
// waterlopen
assign(rivierrombout_waterlopen1, OrEq('rastert_huetzon1.asc', 1, 2, 8, 9, 10, 17, 18, 19))
//write('rivierrombout_waterlopen1.asc', IsNonZero(rivierrombout_waterlopen1))
assign(rivierrombout_waterlopen2, eq('maas.asc', 1))
assign(rivierrombout_waterlopen, or(rivierrombout_waterlopen1, rivierrombout_waterlopen2))
deleteVariable(rivierrombout_waterlopen1)
deleteVariable(rivierrombout_waterlopen2)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het waterbiotoop
assign(rivierrombout_waterlopen_cluster, fuzzyClusterIdU(rivierrombout_waterlopen, 50, meter))
// waterbiotoop moet minimaal 5ha groot zijn
assign(rivierrombout_waterbiotoop, cFilterGeArea(rivierrombout_waterlopen_cluster, 5, hectare))
deleteVariable(rivierrombout_waterlopen_cluster)
// omgevend landbiotoop
assign(rivierrombout_landbiotoop_bwk, bwk(hf%, hr%, hu%, kb%, kh%, khw%, ku%, kz%, mr%, sf%, sz%, hc%, hj%,
    mc%, mm%, md%, sp%, sm%, vn%, vm%, va%, n, n+, n-, lhb%, lhi%, kd%, ks%, 3260%, 3270%, 6430%, 6510%,
    91E0%, rbbhc%, rbbhf%, rbbmr%, rbbf%, 7210%, 7230%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het landbiotoop
assign(rivierrombout_landbiotoop_cluster, fuzzyClusterIdU(rivierrombout_landbiotoop_bwk, 50, meter))
deleteVariable(rivierrombout_landbiotoop_bwk)
// landbiotoop moet minimaal 5ha groot zijn
assign(rivierrombout_landbiotoop, cFilterGeArea(rivierrombout_landbiotoop_cluster, 5, hectare))
deleteVariable(rivierrombout_landbiotoop_cluster)
// zoek voortplantingsgebieden die binnen de 1 km van foerageergebieden liggen
assign(rivierrombout_voortplanting1, and(rivierrombout_waterbiotoop, cellDistanceLe(rivierrombout_landbiotoop,
    1000, meter)))
//write('rivierrombout_voortplanting.asc', IsNonZero(rivierrombout_voortplanting1))
assign(rivierrombout_foerageer1, and(rivierrombout_landbiotoop, cellDistanceLe(rivierrombout_waterbiotoop,
    1000, meter)))
//write('rivierrombout_foerageer.asc', IsNonZero(rivierrombout_foerageer1))
assign(rivierrombout_ecodistrict, OrEq('ecoregio.asc', 6, 7, 8, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24,
    25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36))
assign(rivierrombout_voortplanting_ecodistrict, and(rivierrombout_voortplanting1, rivierrombout_ecodistrict))
write('rivierrombout_voortplanting_ecodistrict.asc', IsNonZero(rivierrombout_voortplanting_ecodistrict))
assign(rivierrombout_foerageer_ecodistrict, and(rivierrombout_foerageer1, rivierrombout_ecodistrict))
write('rivierrombout_foerageer_ecodistrict.asc', IsNonZero(rivierrombout_foerageer_ecodistrict))
```

MOLLUSKEN

Nauwe korfslak (*Vertigo angustior*) – Packet & Provoost (2015)



Potentiële leefgebiedenkaart voor Nauwe korfslak op basis van het onderstaande gdx-script.

Gdx-script

vlaanderen_20m.asc

```
// baken de voortplantingsgebieden af voor de duinen
assign(nauwekorfslak_voortplanting_duin, bwk(hd%, hdb%, ls%, lsi%, mp%, ru%, rud%, rud+%, sd%, sdb%, had%, n,
n+, n-, gml%, da%, 2130%, 2160%, 2170%, 2180%, 2190%, 91E0%, 91F0%, rbbmr%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het voortplantingscluster
assign(nauwekorfslak_voortplanting_duin_cluster, fuzzyClusterIdU(nauwekorfslak_voortplanting_duin, 50, meter))
deleteVariable(nauwekorfslak_voortplanting_duin)
// optimale voortplantingsclusters moeten minimaal 0.04ha groot zijn
assign(nauwekorfslak_leefgebied_duin1, cFilterGeArea(nauwekorfslak_voortplanting_duin_cluster, 0.04, hectare))
assign(nauwekorfslak_ecodistrict_duin, eq('ecoregio.asc', 1))
assign(nauwekorfslak_leefgebied_duin, and(nauwekorfslak_leefgebied_duin1, nauwekorfslak_ecodistrict_duin))
deleteVariable(nauwekorfslak_leefgebied_duin1)
deleteVariable(nauwekorfslak_ecodistrict_duin)
// write('nauwekorfslak_leefgebied_duin.asc', if(nauwekorfslak_leefgebied_duin,
nauwekorfslak_voortplanting_duin_cluster, 0))
deleteVariable(nauwekorfslak_voortplanting_duin_cluster)
// baken de voortplantingsgebieden af voor de rest van Vlaanderen, zelfde als zeggekorfslak!
// baken de voortplantingsgebieden af
assign(nauwekorfslak_voortplanting_leem, bwk(mc%, mcb%, mr%, vn%, rbbmc%, rbbmr%, 91E0%))
// andere biotopen zijn enkel geschikt als ze binnen de 100 meter van mc% liggen
assign(nauwekorfslak_ab, bwk(hfc%, md%, mk%, mp%, sf%, va%, vc%, vm%, hc%, 6410%, 6430%, 7230%, 91E0%,
91F0%, rbbhc%, rbbhf%, rbbmf%))
assign(nauwekorfslak_abgeschikt, and(nauwekorfslak_ab, cellDistanceLe(nauwekorfslak_voortplanting_leem, 100,
meter)))
assign(nauwekorfslak_leefgebied_leem1, or(nauwekorfslak_voortplanting_leem, nauwekorfslak_abgeschikt))
deleteVariable(nauwekorfslak_voortplanting_leem)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het voortplantingscluster
assign(nauwekorfslak_leefgebied_leem_cluster, fuzzyClusterIdU(nauwekorfslak_leefgebied_leem1, 100, meter))
// optimale voortplantingsclusters moeten minimaal 0.04ha groot zijn
assign(nauwekorfslak_leefgebied_leem2, cFilterGeArea(nauwekorfslak_leefgebied_leem_cluster, 0.04, hectare))
// write('nauwekorfslak_leefgebied_leem2.asc', IsNonZero(nauwekorfslak_leefgebied_leem2))
assign(nauwekorfslak_ecodistrict_leem, OrEq('ecoregio.asc', 25, 27, 28, 29, 30, 31, 32, 33, 34))
assign(nauwekorfslak_leefgebied_leem, and(nauwekorfslak_leefgebied_leem2, nauwekorfslak_ecodistrict_leem))
// write('nauwekorfslak_leefgebied_leem.asc', if(nauwekorfslak_leefgebied_leem,
nauwekorfslak_leefgebied_leem_cluster, 0))
assign(nauwekorfslak_leefgebied_ecodistrict1, or(nauwekorfslak_leefgebied_leem,
nauwekorfslak_leefgebied_duin))
assign(nauwekorfslak_leefgebied_ecodistrict1_cluster, fuzzyClusterIdU(nauwekorfslak_leefgebied_ecodistrict1,
100, meter))
```

```
assign(nauwekorfslak_leefgebied_ecodistrict, cFilterGeArea(nauwekorfslak_leefgebied_ecodistrict1_cluster, 0.04,
  hectare))
write('nauwekorfslak_leefgebied_ecodistrict.asc', if(nauwekorfslak_leefgebied_ecodistrict,
  nauwekorfslak_leefgebied_ecodistrict1_cluster, 0))
```

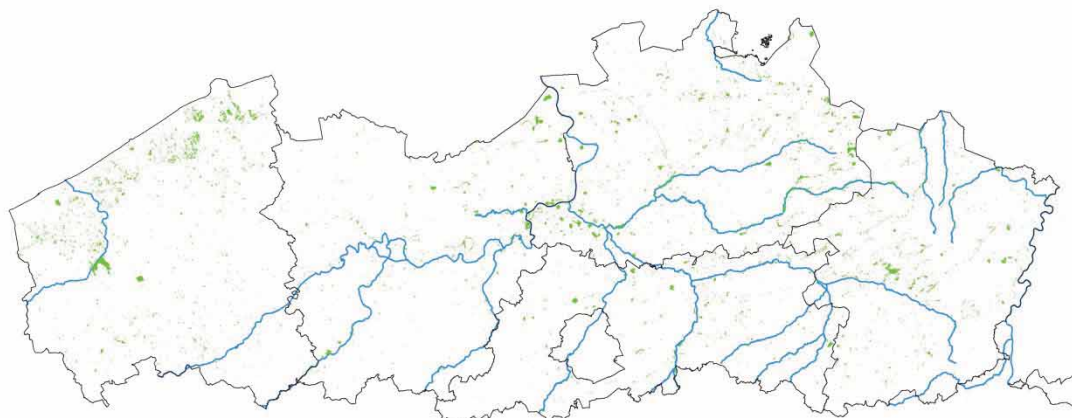


Nauwe korfslak (foto Rollin Verlinde – Vilda)



Platte schijfhoren (foto Yves Adams – Vilda)

Platte schijfhoren (*Anisus (Disculifer) vorticulus*) – Packet (2015a)

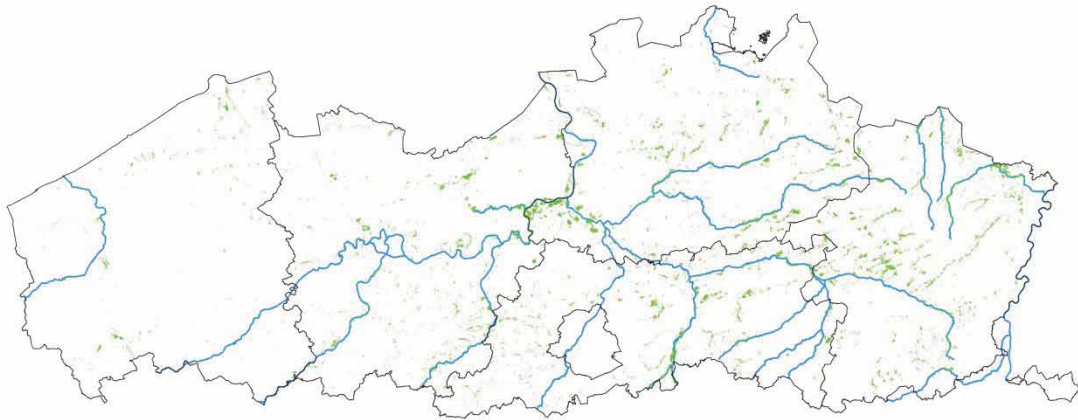


Potentiële leefgebiedenkaart voor Platte schijfhoren op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(platteschijfhoren_voortplanting_a, bwk(ae%, aom%, 3140%, 3130%, 2190%, 3150%))
assign(platteschijfhoren_voortplanting_h, bwk(hpr%, hpr+%, kn%, 3150%, 2190%))
// waterlopen
assign(platteschijfhoren_waterlopen, OrEq('rastert_huetzon1.asc', 3, 4, 5, 6, 15, 16))
assign(platteschijfhoren_voortplanting_wh, or(platteschijfhoren_voortplanting_h, platteschijfhoren_waterlopen))
deleteVariable(platteschijfhoren_voortplanting_h)
deleteVariable(platteschijfhoren_waterlopen)
// bodem_bodemtype
assign(platteschijfhoren_bodem, OrEq('bodem_bodemtype.asc', 89, 1460, 2830, 2913, 3089, 3381, 3513, 3640,
3687, 4011, 4036))
assign(platteschijfhoren_voortplanting2, and(platteschijfhoren_voortplanting_wh, platteschijfhoren_bodem))
deleteVariable(platteschijfhoren_bodem)
assign(platteschijfhoren_leefgebied, or(platteschijfhoren_voortplanting_a, platteschijfhoren_voortplanting2))
deleteVariable(platteschijfhoren_voortplanting_a)
deleteVariable(platteschijfhoren_voortplanting2)
deleteVariable(platteschijfhoren_voortplanting_wh)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het voortplantingscluster
assign(platteschijfhoren_leefgebied_cluster, fuzzyClusterIdU(platteschijfhoren_leefgebied, 50, meter))
// optimale voortplantingsclusters moeten minimaal 1ha groot zijn
assign(platteschijfhoren_leefgebied, cFilterGeArea(platteschijfhoren_leefgebied_cluster, 1, hectare))
write('platteschijfhoren_leefgebied.asc', if(platteschijfhoren_leefgebied, platteschijfhoren_leefgebied_cluster, 0))
deleteVariable(platteschijfhoren_leefgebied_cluster)
```

Zeggekorfslak (*Vertigo moulinsiana*) – Packet (2015b)



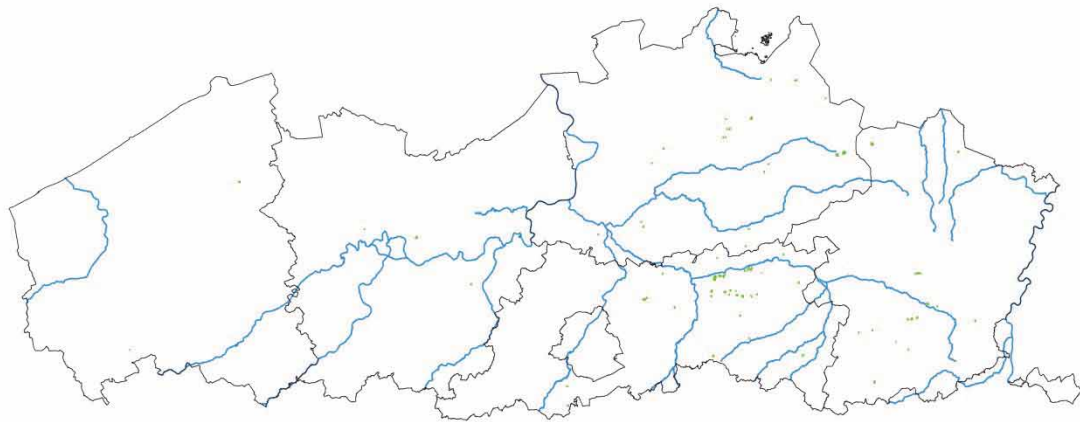
Potentiële leefgebiedenkaart voor Zeggekorfslak op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(zeggekorfslak_voortplanting, bwk(mc%, mcb%, mr%, vn%, vm%, md%, hfc%))
// andere biotopen zijn enkel geschikt als ze binnen de 100 meter van mc% liggen
assign(zeggekorfslak_ab, bwk(mk%, mp%, sf%, va%, vc%, hc%))
assign(zeggekorfslak_abgeschikt, and(zeggekorfslak_ab, cellDistanceLe(zeggekorfslak_voortplanting, 100, meter)))
deleteVariable(zeggekorfslak_ab)
assign(zeggekorfslak_leefgebied1, or(zeggekorfslak_voortplanting, zeggekorfslak_abgeschikt))
deleteVariable(zeggekorfslak_voortplanting)
deleteVariable(zeggekorfslak_abgeschikt)
// clustering: indien geschikte cel binnen 100 meter, dan hoort die cel bij het voortplantingscluster
assign(zeggekorfslak_leefgebied1_cluster, fuzzyClusterIdU(zeggekorfslak_leefgebied1, 100, meter))
deleteVariable(zeggekorfslak_leefgebied1)
// optimale voortplantingsclusters moeten minimaal 0.04ha groot zijn
assign(zeggekorfslak_leefgebied, cFilterGeArea(zeggekorfslak_leefgebied1_cluster, 0.04, hectare))
deleteVariable(zeggekorfslak_leefgebied1_cluster)
assign(zeggekorfslak_ecodistrict, ge('ecoregio.asc', 1))
assign(zeggekorfslak_leefgebied_ecodistrict1, and(zeggekorfslak_leefgebied, zeggekorfslak_ecodistrict))
deleteVariable(zeggekorfslak_leefgebied)
deleteVariable(zeggekorfslak_ecodistrict)
assign(zeggekorfslak_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(zeggekorfslak_leefgebied_ecodistrict1, 100,
meter))
assign(zeggekorfslak_leefgebied_ecodistrict, cFilterGeArea(zeggekorfslak_leefgebied_ecodistrict1_cluster, 0.04,
hectare))
write('zeggekorfslak_leefgebied_ecodistrict.asc', if(zeggekorfslak_leefgebied_ecodistrict,
zeggekorfslak_leefgebied_ecodistrict1_cluster, 0))
```

MOSSEN

Geel schorpioenmos (*Hamatocaulis vernicosus*) – Van Landuyt (2015a)



Potentiële leefgebiedenkaart voor Geel schorpioenmos op basis van het onderstaande gdx-script.

Gdx-script

vlaanderen_20m.asc

// baken de voortplantingsgebieden af

assign(geelschorpioenmos_voortplanting, bwk(hm, hm+, hme, hme+, hmm, hmm+, mk, mk+, 6410%, 7140%, 7230%))

// write('geelschorpioenmos_voortplanting.asc', IsNonZero(geelschorpioenmos_voortplanting))

// clustering: indien geschikte cel binnen 20 meter, dan hoort die cel bij het voortplantingscluster

assign(geelschorpioenmos_voortplanting_cluster, fuzzyClusterId(geelschorpioenmos_voortplanting, 20/cellSize))

// write('geelschorpioenmos_voortplanting_cluster.asc', geelschorpioenmos_voortplanting_cluster)

// optimale voortplantingsclusters moeten minimaal 0.16ha groot zijn

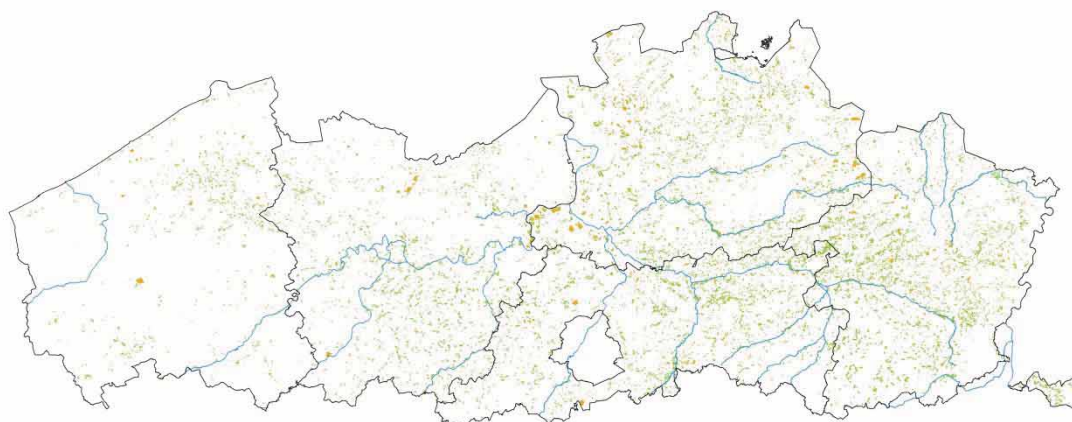
assign(geelschorpioenmos_voortplanting_oppervlakte, cSum(geelschorpioenmos_voortplanting_cluster, 1))

assign(geelschorpioenmos_leefgebied, cFilter(geelschorpioenmos_voortplanting_cluster, ge(geelschorpioenmos_voortplanting_oppervlakte, 0.16*10000/cellSize2)))

write('geelschorpioenmos_leefgebied.asc', if(geelschorpioenmos_leefgebied, geelschorpioenmos_voortplanting_cluster, 0))

NACHTVLINDERS

Spaanse vlag (*Euplagia quadripunctaria*) – Maes (2015)



Potentiële leefgebiedenkaart voor Spaanse vlag op basis van het onderstaande gdx-script.
Groen = rupsbiotoop, oranje = vlinderbiotoop.

Gdx-script

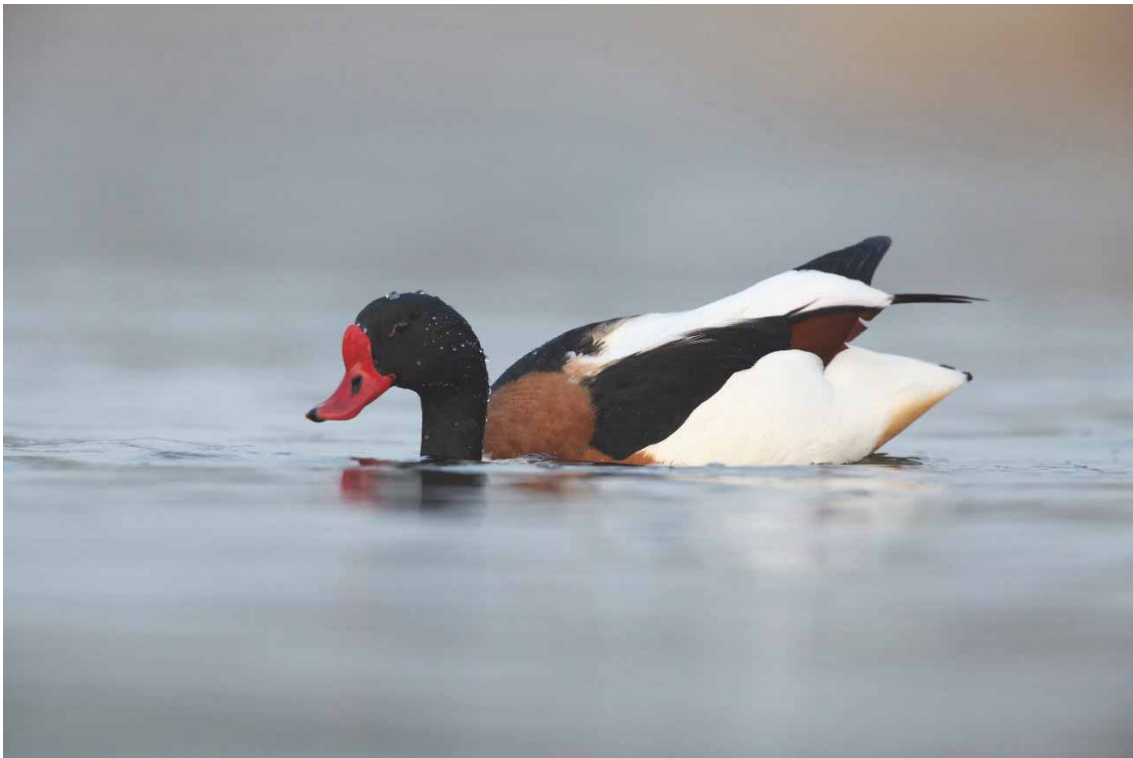
vlaanderen_20m.asc

```
// baken de voortplantingsgebieden af voor de rupsen
assign(spaansevlag_voortplanting_bwk, bwk(ha%, hc%, hf%, hj%, hp%, hp+%, hpr%, hr%, hu%, kc%, kl%, kp%, kpk%,
ks%, kt%, ku%, kub%, kw%, kz%, mr%, se%, sz%, 6430))
// voortplantingsplekken mogen klein zijn: <1ha groot zijn (Wallis de Vries & Groenendijk 2011)
assign(spaansevlag_voortplanting1_cluster, fuzzyClusterIdU(spaansevlag_voortplanting_bwk, 50, meter))
deleteVariable(spaansevlag_voortplanting_bwk)
assign(spaansevlag_voortplanting2, cFilterGeArea(spaansevlag_voortplanting1_cluster, 1, hectare))
deleteVariable(spaansevlag_voortplanting1_cluster)
// baken de bossen met bosranden af voor de adulte vlinders
assign(spaansevlag_bos_bwk, bwk(fs%, lhb%, n, n+, n-, pa%, pmb%, ppmb%, qa%, qb%, qs%, va%, vm%, 9160))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het boscluster
assign(spaansevlag_bos_bwk_cluster, fuzzyClusterIdU(spaansevlag_bos_bwk, 50, meter))
deleteVariable(spaansevlag_bos_bwk)
// bosclusters moeten minimaal 1ha groot zijn
assign(spaansevlag_bos, cFilterGeArea(spaansevlag_bos_bwk_cluster, 1, hectare))
deleteVariable(spaansevlag_bos_bwk_cluster)
// zoek voortplantingsclusters die aan bosclusters grenzen en die een zuidgeëxposeerde bosrand hebben (NO, O,
ZO, Z, ZW, W, NW, N)
assign(spaansevlag_edge1, edgeDirected(spaansevlag_bos, spaansevlag_voortplanting2, 0,0,0,1,0,0,0,0))
assign(spaansevlag_edge2, edgeDirected(spaansevlag_voortplanting2, spaansevlag_bos, 0,0,0,0,0,0,0,1))
deleteVariable(spaansevlag_bos)
assign(spaansevlag_bosrand1, or(spaansevlag_edge1, spaansevlag_edge2))
deleteVariable(spaansevlag_edge1)
deleteVariable(spaansevlag_edge2)
// lengte van bosrand tussen voortplantingsclusters en boscluster moet minstens 100 meter zijn en pixels moeten
tegen elkaar liggen dus afstand <10m
assign(spaansevlag_bosrand1_cluster, fuzzyClusterIdU(spaansevlag_bosrand1, 50, meter))
deleteVariable(spaansevlag_bosrand1)
assign(spaansevlag_bosrand2, cFilterGeArea(spaansevlag_bosrand1_cluster, 0.25, hectare))
deleteVariable(spaansevlag_bosrand1_cluster)
// geschikt leefgebied moet aanwezig zijn binnen de 100m van bosrand
assign(spaansevlag_leefgebied1, and(spaansevlag_voortplanting2, cellDistanceLe(spaansevlag_bosrand2, 100,
meter)))
write('spaansevlag_rups.asc', IsNonZero(spaansevlag_leefgebied1))
assign(spaansevlag_leefgebied2, and(spaansevlag_bosrand2, cellDistanceLe(spaansevlag_voortplanting2, 100,
meter)))
write('spaansevlag_vlinder.asc', IsNonZero(spaansevlag_leefgebied2))
```

```

deleteVariable(spaansevlag_voortplanting2)
deleteVariable(spaansevlag_bosrand2)
assign(spaansevlag_leefgebied3, or(spaansevlag_leefgebied1, spaansevlag_leefgebied2))
deleteVariable(spaansevlag_leefgebied1)
deleteVariable(spaansevlag_leefgebied2)
assign(spaansevlag_leefgebied3_cluster, FuzzyClusterIdU(spaansevlag_leefgebied3, 100, meter))
deleteVariable(spaansevlag_leefgebied3)
// oppervlakte moet minstens 1ha zijn
assign(spaansevlag_leefgebied, cFilterGeArea(spaansevlag_leefgebied3_cluster, 1, hectare))
deleteVariable(spaansevlag_leefgebied3_cluster)
// baken ecoregio van ecodistrictgebieden af
assign(spaansevlag_ecodistrict, ge('ecoregio.asc', 1))
assign(spaansevlag_leefgebied_ecodistrict1, and(spaansevlag_leefgebied, spaansevlag_ecodistrict))
assign(spaansevlag_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(spaansevlag_leefgebied_ecodistrict1, 100,
meter))
assign(spaansevlag_leefgebied_ecodistrict, cFilterGeArea(spaansevlag_leefgebied_ecodistrict1_cluster, 1, hectare))
//write('spaansevlag_leefgebied_ecodistrict.asc', if(spaansevlag_leefgebied_ecodistrict,
spaansevlag_leefgebied_ecodistrict1_cluster, 0))

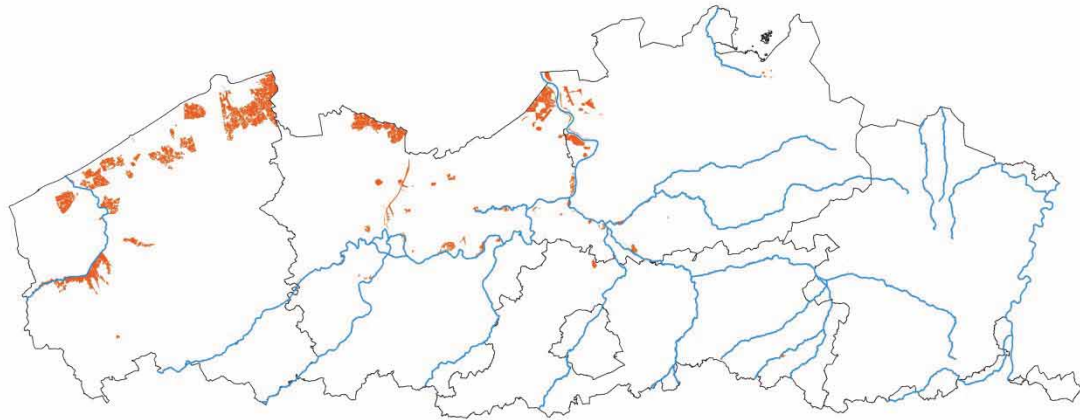
```



Bergeend (foto Glenn Vermeersch)

OVERWINTERENDE WATERVOGELS

Bergeend (*Tadorna tadorna*) – Adriaens & Vermeersch (in druk)



Potentiële overwinteringskaart voor Bergeend op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

vlaanderen_20m.asc

// lees de shape file in uit de vogelgebiedendatabank

assign(bergeend_watervogelgebieden, 'leefgebied_bergeend_20m.asc')

assign(bergeend_watervogelgebied1, and(ge(bergeend_watervogelgebieden, 1), le(bergeend_watervogelgebieden, 192)))

deleteVariable(bergeend_watervogelgebieden)

// baken de foerageergebieden binnen de telgebieden af

assign(bergeend_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, ap%, bu%, da%, ds%, dz%, hp+%, hpr%, kn%, kz%, hpr%, ku%, 1140%, 1310%, 1320%, 1330%, 2190%, 3130%, 3140%, 3150%, 3270%, 6430%))

// waterlopen

assign(bergeend_waterlopen, OrEq('rastert_huetzon1.asc', 11, 12, 13, 14, 17))

assign(bergeend_foerageer1, or(bergeend_foerageer_bwk, bergeend_waterlopen))

deleteVariable(bergeend_foerageer_bwk)

deleteVariable(bergeend_waterlopen)

// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied

assign(bergeend_foerageer1_cluster, fuzzyClusterIdU(bergeend_foerageer1, 50, meter))

deleteVariable(bergeend_foerageer1)

// leefgebiedcluster moeten minimaal 5ha groot zijn

assign(bergeend_leefgebied1, cFilterGeArea(bergeend_foerageer1_cluster, 5, hectare))

deleteVariable(bergeend_foerageer1_cluster)

assign(bergeend_leefgebied, and(bergeend_watervogelgebied1, bergeend_leefgebied1))

deleteVariable(bergeend_watervogelgebied1)

deleteVariable(bergeend_leefgebied1)

// baken ecoregio van ecodistrictgebieden af

assign(bergeend_ecodistrict, ge('ecoregio.asc', 1))

assign(bergeend_leefgebied_ecodistrict1, and(bergeend_leefgebied, bergeend_ecodistrict))

assign(bergeend_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(bergeend_leefgebied_ecodistrict1, 50, meter))

assign(bergeend_leefgebied_ecodistrict, cFilterGeArea(bergeend_leefgebied_ecodistrict1_cluster, 5, hectare))

write('bergeend_leefgebied_ecodistrict.asc', if(bergeend_leefgebied_ecodistrict, bergeend_leefgebied_ecodistrict1_cluster, 0))



Potentiële overwinteringskaart voor Goudplevier op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

vlaanderen_20m.asc

```
// lees de shape file in uit de vogelgebiedendatabank
```

```
assign(goudplevier_watervogelgebieden, 'leefgebied_goudplevier_20m.asc')
```

```
assign(goudplevier_watervogelgebied1, and(ge(goudplevier_watervogelgebieden, 1),
```

```
le(goudplevier_watervogelgebieden, 30)))
```

```
deleteVariable(goudplevier_watervogelgebieden)
```

```
// baken de foerageergebieden af
```

```
assign(goudplevier_foerageer, bwk(bu%, da%, ds%, hp%, hp+%, hpr%, hpr+%, hx%))
```

```
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open gebied zijn!
```

```
assign(goudplevier_foerageer_cluster, fuzzyClusterIdU(goudplevier_foerageer, 50, meter))
```

```
deleteVariable(goudplevier_foerageer)
```

```
// leefgebiedcluster moeten minimaal 80ha groot zijn
```

```
assign(goudplevier_leefgebied1, cFilterGeArea(goudplevier_foerageer_cluster, 80, hectare))
```

```
deleteVariable(goudplevier_foerageer_cluster)
```

```
assign(goudplevier_leefgebied, and(goudplevier_watervogelgebied1, goudplevier_leefgebied1))
```

```
deleteVariable(goudplevier_watervogelgebied1)
```

```
// baken ecoregio van ecodistrictgebieden af
```

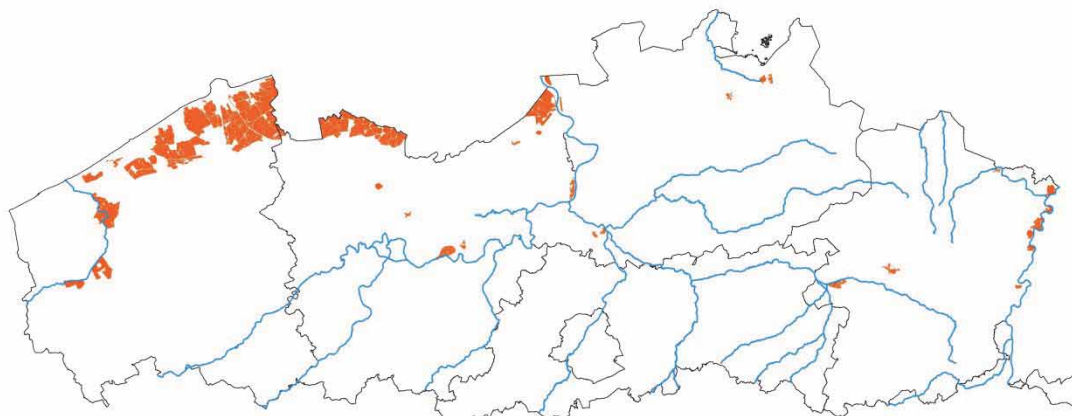
```
assign(goudplevier_ecodistrict, ge('ecoregio.asc', 1))
```

```
assign(goudplevier_leefgebied_ecodistrict1, and(goudplevier_leefgebied, goudplevier_ecodistrict))
```

```
assign(goudplevier_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(goudplevier_leefgebied_ecodistrict1, 50, meter))
```

```
assign(goudplevier_leefgebied_ecodistrict, cFilterGeArea(goudplevier_leefgebied_ecodistrict1_cluster, 80, hectare))
```

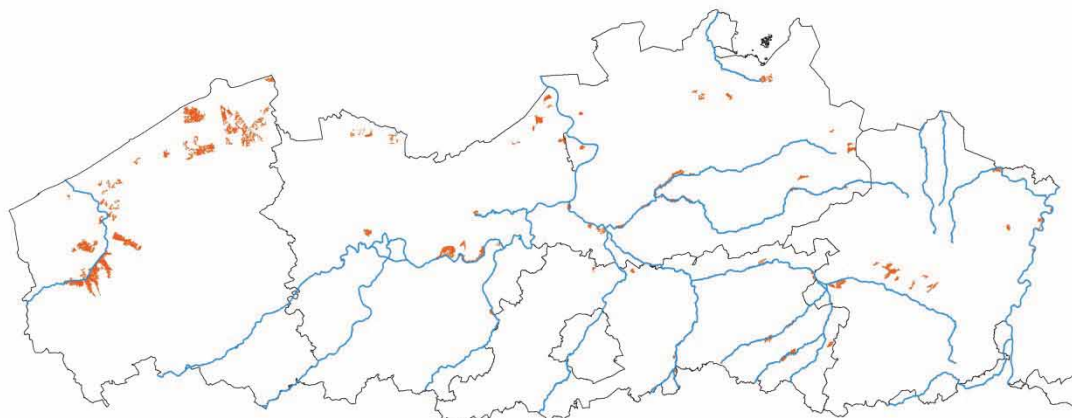
```
write('goudplevier_leefgebied_ecodistrict.asc', if(goudplevier_leefgebied_ecodistrict, goudplevier_leefgebied_ecodistrict1_cluster, 0))
```



Potentiële overwinteringskaart voor Grauwe gans op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m_met_maas.asc
// lees de shape file in uit de vogelgebiedendatabank
assign(grauwegans_watervogelgebieden, 'leefgebied_grauwegans_20m.asc')
assign(grauwegans_watervogelgebied1, and(ge(grauwegans_watervogelgebieden, 1),
    le(grauwegans_watervogelgebieden, 95)))
// baken de foerageergebieden binnen de telgebieden af
assign(grauwegans_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, ap%, bk%, bl%, bs%, bu%, da%, ds%, hp%, hp+%,
    hpr%, hpr+%, hx%, ku%, kz%, mr%, mz%))
// waterlopen
assign(grauwegans_waterlopen, OrEq('rastert_huetzon1.asc', 11, 12, 13, 14, 17, 19))
assign(grauwegans_maas, eq('maas.asc', 1))
assign(grauwegans_foerageer_epr, OrEq('epr2013.asc', 6, 38, 100, 18, 8, 4, 89, 40, 120, 7, 1))
assign(grauwegans_foerageer1, or(grauwegans_foerageer_bwk, grauwegans_waterlopen, grauwegans_maas,
    grauwegans_foerageer_epr))
deleteVariable(grauwegans_foerageer_bwk)
deleteVariable(grauwegans_waterlopen)
deleteVariable(grauwegans_maas)
deleteVariable(grauwegans_foerageer_epr)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
// gebied zijn!
assign(grauwegans_foerageer1_cluster, fuzzyClusterIdU(grauwegans_foerageer1, 50, meter))
deleteVariable(grauwegans_foerageer1)
// leefgebiedcluster moeten minimaal 80ha groot zijn
assign(grauwegans_leefgebied1, cFilterGeArea(grauwegans_foerageer1_cluster, 30, hectare))
deleteVariable(grauwegans_foerageer1_cluster)
assign(grauwegans_leefgebied, and(grauwegans_watervogelgebied1, grauwegans_leefgebied1))
deleteVariable(grauwegans_watervogelgebied1)
deleteVariable(grauwegans_leefgebied1)
// baken ecoregio van ecodistrictgebieden af
assign(grauwegans_ecodistrict, ge('ecoregio.asc', 1))
assign(grauwegans_leefgebied_ecodistrict1, and(grauwegans_leefgebied, grauwegans_ecodistrict))
assign(grauwegans_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(grauwegans_leefgebied_ecodistrict1, 50,
    meter))
assign(grauwegans_leefgebied_ecodistrict, cFilterGeArea(grauwegans_leefgebied_ecodistrict1_cluster, 80,
    hectare))
write('grauwegans_leefgebied_ecodistrict.asc', if(grauwegans_leefgebied_ecodistrict,
    grauwegans_leefgebied_ecodistrict1_cluster, 0))
```

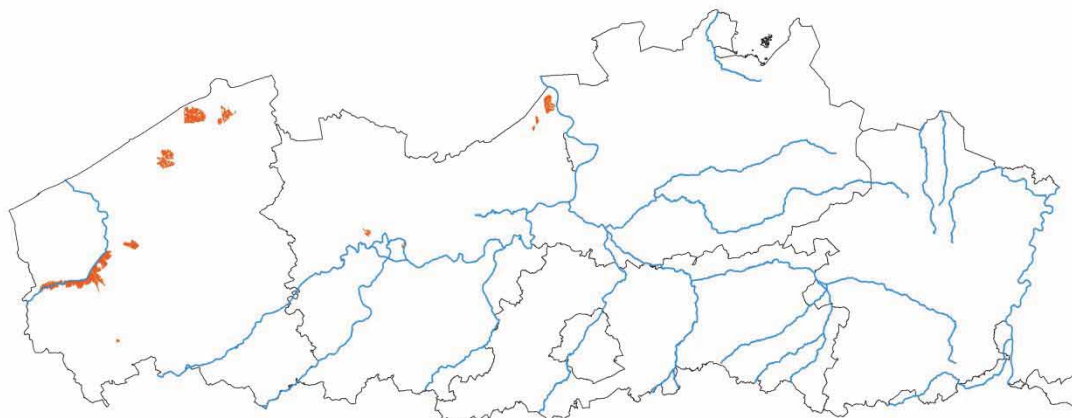


Potentiële overwinteringskaart voor Grote zilverreiger op basis van de watervogeltellingen en het onderstaande.gdx-script.

Gdx-script

vlaanderen_20m.asc

```
// lees de shape file in uit de vogelgebiedendatabank
assign(grotezilverreiger_watervogelgebieden, 'leefgebied_grotezilverreiger_20m.asc')
assign(grotezilverreiger_watervogelgebied1, and(ge(grotezilverreiger_watervogelgebieden, 1),
  le(grotezilverreiger_watervogelgebieden, 169)))
deleteVariable(grotezilverreiger_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(grotezilverreiger_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, ao%, aom%, aoo%, da%, hc%, hcb%, hp%,
  hp+%, hpr%, hpr+%, mc%, mcb%, mr%, mrb%, sf%, sz%, va%, vc%, vf%, vm%, vn%, vo%, vt%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
// gebied zijn!
assign(grotezilverreiger_foerageer1_cluster, fuzzyClusterIdU(grotezilverreiger_foerageer_bwk, 50, meter))
// leefgebiedcluster moeten minimaal 30ha groot zijn
assign(grotezilverreiger_leefgebied1, cFilterGeArea(grotezilverreiger_foerageer1_cluster, 30, hectare))
assign(grotezilverreiger_leefgebied, and(grotezilverreiger_watervogelgebied1, grotezilverreiger_leefgebied1))
deleteVariable(grotezilverreiger_watervogelgebied1)
deleteVariable(grotezilverreiger_leefgebied1)
// baken ecoregio van ecodistrictgebieden af
assign(grotezilverreiger_ecodistrict, ge('ecoregio.asc', 1))
assign(grotezilverreiger_leefgebied_ecodistrict1, and(grotezilverreiger_leefgebied, grotezilverreiger_ecodistrict))
assign(grotezilverreiger_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(grotezilverreiger_leefgebied_ecodistrict1,
  50, meter))
assign(grotezilverreiger_leefgebied_ecodistrict, cFilterGeArea(grotezilverreiger_leefgebied_ecodistrict1_cluster, 30,
  hectare))
write('grotezilverreiger_leefgebied_ecodistrict.asc', if(grotezilverreiger_leefgebied_ecodistrict,
  grotezilverreiger_leefgebied_ecodistrict1_cluster, 0))
```

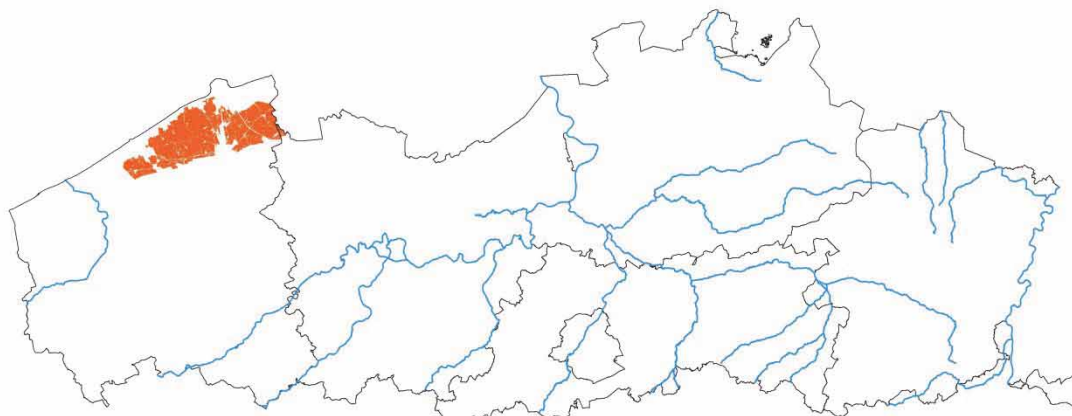


Potentiële overwinteringskaart voor Kemphaan op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// lees de shape file in uit de vogelgebiedendatabank
assign(kemphaan_watervogelgebieden, 'leefgebied_kemphaan_20m.asc')
assign(kemphaan_watervogelgebied, and(ge(kemphaan_watervogelgebieden, 1),
  le(kemphaan_watervogelgebieden, 26)))
deleteVariable(kemphaan_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(kemphaan_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, bl%, bs%, bu%, da%, ds%, hc%, hp+, hpr%, hpr+,
  hx%, kn%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
// gebied zijn!
assign(kemphaan_foerageer1_cluster, fuzzyClusterIdU(kemphaan_foerageer_bwk, 50, meter))
// leefgebiedcluster moeten minimaal 5ha groot zijn
assign(kemphaan_leefgebied1, cFilterGeArea(kemphaan_foerageer1_cluster, 5, hectare))
assign(kemphaan_leefgebied, and(kemphaan_watervogelgebied, kemphaan_leefgebied1))
deleteVariable(kemphaan_watervogelgebied)
assign(kemphaan_ecodistrict, ge('ecoregio.asc', 1))
assign(kemphaan_leefgebied_ecodistrict1, and(kemphaan_leefgebied1, kemphaan_ecodistrict))
deleteVariable(kemphaan_leefgebied1)
assign(kemphaan_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(kemphaan_leefgebied_ecodistrict1, 50, meter))
assign(kemphaan_leefgebied_ecodistrict, cFilterGeArea(kemphaan_leefgebied_ecodistrict1_cluster, 30, hectare))
write('kemphaan_leefgebied_ecodistrict.asc', if(kemphaan_leefgebied_ecodistrict,
  kemphaan_leefgebied_ecodistrict1_cluster, 0))
```

Kleine rietgans (*Anser brachyrhynchus*) – Adriaens & Vermeersch (in druk)

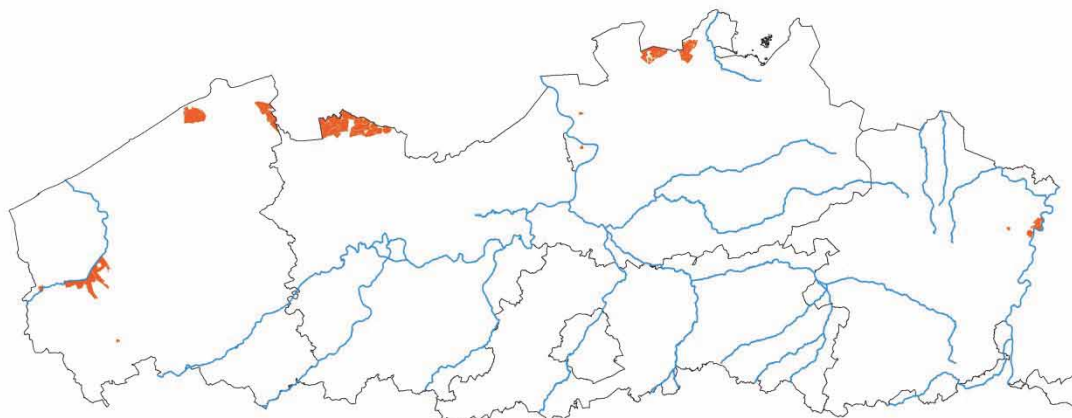


Potentiële overwinteringskaart voor Kleine rietgans op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

vlaanderen_20m.asc

```
// lees de shape file in uit de vogelgebiedendatabank
assign(kleinerietgans_watervogelgebieden, 'leefgebied_kleinerietgans_20m.asc')
assign(kleinerietgans_watervogelgebied1, and(ge(kleinerietgans_watervogelgebieden, 1),
le(kleinerietgans_watervogelgebieden, 10)))
deleteVariable(kleinerietgans_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(kleinerietgans_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, bu%, da%, hp%, hp+%, hpr%, hpr+%, hx%))
assign(kleinerietgans_foerageer_epr, OrEq('epr2013.asc', 6, 38, 100, 18, 8, 4, 89, 40, 120, 7, 1))
assign(kleinerietgans_foerageer, or(kleinerietgans_foerageer_bwk, kleinerietgans_foerageer_epr))
deleteVariable(kleinerietgans_foerageer_epr)
deleteVariable(kleinerietgans_foerageer_bwk)
assign(kleinerietgans_foerageer_cluster, fuzzyClusterIdU(kleinerietgans_foerageer, 50, meter))
deleteVariable(kleinerietgans_foerageer)
// leefgebiedcluster moeten minimaal 50ha groot zijn
assign(kleinerietgans_foerageer2, cFilterGeArea(kleinerietgans_foerageer_cluster, 50, hectare))
// 80% van oppervlakte moet grasland zijn, d.i. 40ha
// baken graslanden af
assign(kleinerietgans_grasland_bwk, bwk(hp%, hp+%, hpr%, hpr+%, hx%))
// write('kleinerietgans_grasland_bwk.asc', kleinerietgans_grasland_bwk)
assign(kleinerietgans_grasland_cluster, fuzzyClusterIdU(kleinerietgans_grasland_bwk, 50, meter))
deleteVariable(kleinerietgans_grasland_bwk)
// graslandcluster moeten minimaal 40ha groot zijn
assign(kleinerietgans_leefgebied1, cFilterGeArea(kleinerietgans_foerageer2, 40, hectare))
deleteVariable(kleinerietgans_foerageer2)
deleteVariable(kleinerietgans_grasland_cluster)
assign(kleinerietgans_leefgebied, and(kleinerietgans_leefgebied1, kleinerietgans_watervogelgebied1))
write('kleinerietgans_leefgebied.asc', if(kleinerietgans_leefgebied, kleinerietgans_foerageer_cluster, 0))
deleteVariable(kleinerietgans_foerageer_cluster)
deleteVariable(kleinerietgans_leefgebied1)
deleteVariable(kleinerietgans_watervogelgebied1)
deleteVariable(kleinerietgans_leefgebied)
```



Potentiële overwinteringskaart voor Kleine zwaan op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

vlaanderen_20m.asc

// lees de shape file in uit de vogelgebiedendatabank

assign(kleinezwaan_watervogelgebieden, 'leefgebied_kleinezwaan_20m.asc')

assign(kleinezwaan_watervogelgebied1, and(ge(kleinezwaan_watervogelgebieden, 1),

le(kleinezwaan_watervogelgebieden, 24)))

// write('kleinezwaan_watervogelgebied1.asc', IsNonZero(kleinezwaan_watervogelgebied1))

// baken de foerageergebieden binnen de telgebieden af

assign(kleinezwaan_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, ap%, bl%, bs%, bu%, da%, hp%, hp+%, hpr%,
hpr+%, hx%))

// write('kleinezwaan_foerageer_bwk.asc', IsNonZero(kleinezwaan_foerageer_bwk))

// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open gebied zijn!

assign(kleinezwaan_foerageer1_cluster, fuzzyClusterId(kleinezwaan_foerageer_bwk, 50/cellSize))

// write('kleinezwaan_foerageer1_cluster.asc', kleinezwaan_foerageer1_cluster)

// leefgebiedcluster moeten minimaal 30ha groot zijn

assign(kleinezwaan_foerageer1_cluster_oppervlakte, cSum(kleinezwaan_foerageer1_cluster, 1))

assign(kleinezwaan_leefgebied1, cFilter(kleinezwaan_foerageer1_cluster,

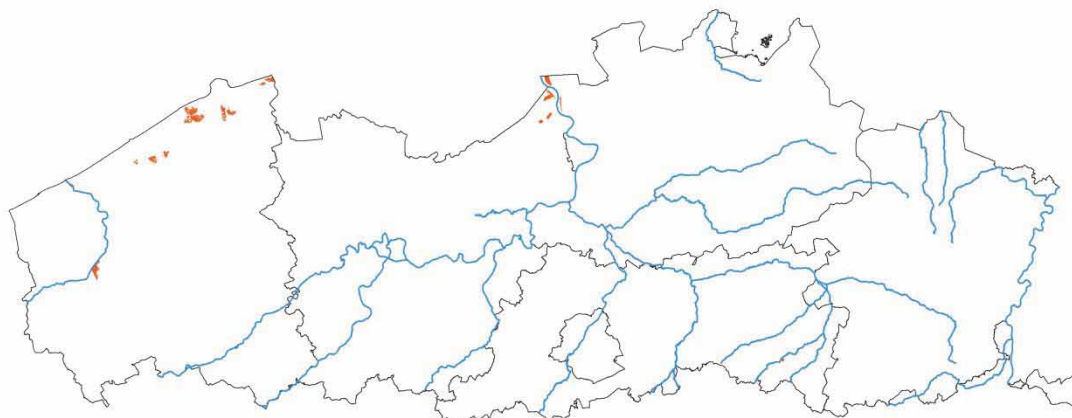
ge(kleinezwaan_foerageer1_cluster_oppervlakte, 30*10000/cellSize2)))

// write('kleinezwaan_leefgebied1.asc', IsNonZero(kleinezwaan_leefgebied1))

assign(kleinezwaan_leefgebied, and(kleinezwaan_watervogelgebied1, kleinezwaan_foerageer_bwk))

write('kleinezwaan_leefgebied.asc', IsNonZero(kleinezwaan_leefgebied))

Kluut (*Recurvirostra avosetta*) – Adriaens & Vermeersch (in druk)



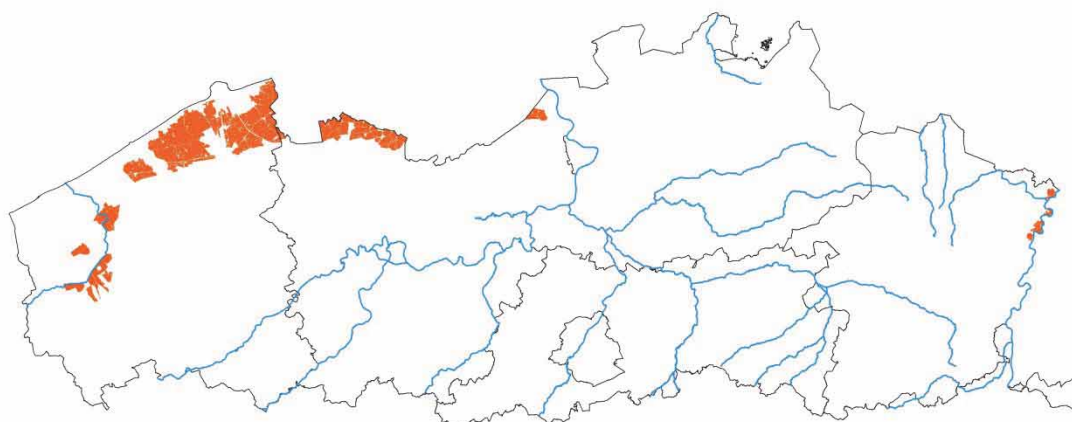
Potentiële overwinteringskaart voor Kluut op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// lees de shape file in uit de vogelgebiedendatabank
assign(kluut_watervogelgebieden, 'leefgebied_kluut_20m.asc')
assign(kluut_watervogelgebied1, and(ge(kluut_watervogelgebieden, 1), le(kluut_watervogelgebieden, 24)))
deleteVariable(kluut_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(kluut_foerageer_bwk, bwk(ad%, ae%, aer%, aev%, ah%, da%, ds%, hpr%, hpr+%, kz%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
gebied zijn!
assign(kluut_foerageer1_cluster, fuzzyClusterIdU(kluut_foerageer_bwk, 50, meter))
// leefgebiedcluster moeten minimaal 30ha groot zijn
assign(kluut_leefgebied1, cFilterGeArea(kluut_foerageer1_cluster, 30, hectare))
assign(kluut_leefgebied, and(kluut_watervogelgebied1, kluut_leefgebied1))
deleteVariable(kluut_watervogelgebied1)
assign(kluut_ecodistrict, ge('ecoregio.asc', 1))
assign(kluut_leefgebied_ecodistrict1, and(kluut_leefgebied1, kluut_ecodistrict))
deleteVariable(kluut_leefgebied1)
assign(kluut_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(kluut_leefgebied_ecodistrict1, 50, meter))
assign(kluut_leefgebied_ecodistrict, cFilterGeArea(kluut_leefgebied_ecodistrict1_cluster, 30, hectare))
write('kluut_leefgebied_ecodistrict.asc', if(kluut_leefgebied_ecodistrict, kluut_leefgebied_ecodistrict1_cluster, 0))
```



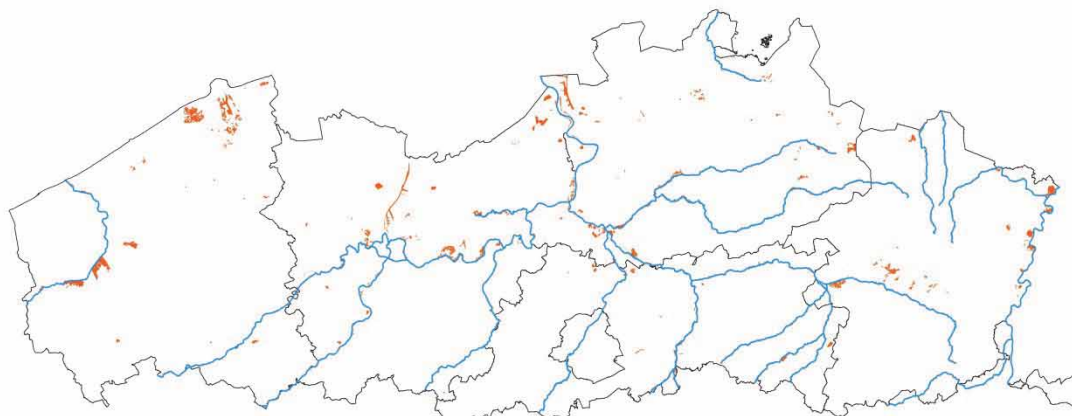
Kluut (foto Lars Soerink – Vilda))



Potentiële overwinteringskaart voor Kolangans op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

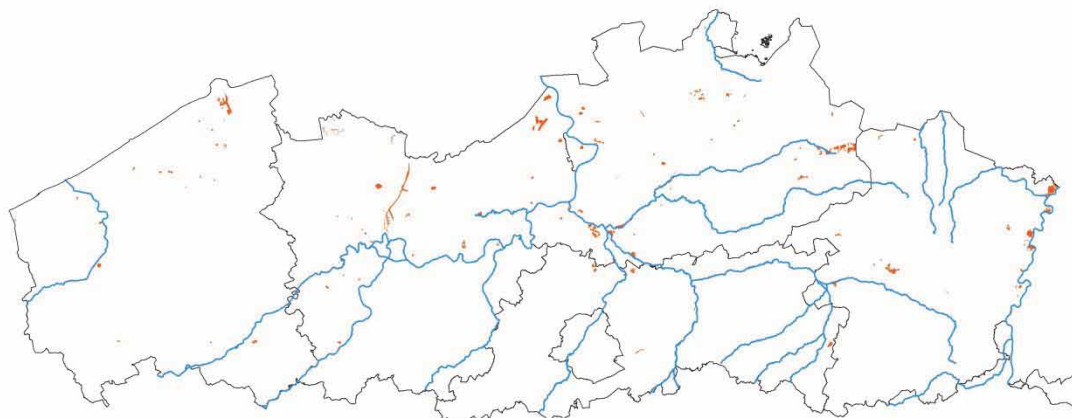
```
vlaanderen_20m_met_maas.asc
// lees de shape file in uit de vogelgebiedendatabank
assign(kolgans_watervogelgebieden, 'leefgebied_kolgans_20m.asc')
assign(kolgans_watervogelgebied1, and(ge(kolgans_watervogelgebieden, 1), le(kolgans_watervogelgebieden, 35)))
deleteVariable(kolgans_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(kolgans_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, ap%, bl%, bs%, bu%, da%, hp%, hp+%, hpr%, hpr+%,
  hx%))
// waterlopen
assign(kolgans_waterlopen, eq('rastert_huetzon1.asc', 19))
assign(kolgans_foerageer_epr, OrEq('epr2013.asc', 6, 38, 100, 18, 8, 4, 89, 40, 120, 7, 1))
assign(kolgans_maas, eq('maas.asc', 1))
assign(kolgans_foerageer1, or(kolgans_foerageer_bwk, kolgans_waterlopen, kolgans_maas,
  kolgans_foerageer_epr))
deleteVariable(kolgans_foerageer_bwk)
deleteVariable(kolgans_waterlopen)
deleteVariable(kolgans_foerageer_epr)
deleteVariable(kolgans_maas)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
// gebied zijn!
assign(kolgans_foerageer1_cluster, fuzzyClusterIdU(kolgans_foerageer1, 50, meter))
// leefgebiedcluster moeten minimaal 80ha groot zijn
assign(kolgans_leefgebied1, cFilterGeArea(kolgans_foerageer1_cluster, 80, hectare))
assign(kolgans_leefgebied, and(kolgans_watervogelgebied1, kolgans_foerageer1))
deleteVariable(kolgans_watervogelgebied1)
//write('kolgans_leefgebied.asc', if(kolgans_leefgebied, kolgans_foerageer1_cluster, 0))
// baken ecoregio van ecodistrictgebieden af
assign(kolgans_ecodistrict, ge('ecoregio.asc', 1))
assign(kolgans_leefgebied_ecodistrict1, and(kolgans_leefgebied, kolgans_ecodistrict))
// write('kolgans_leefgebied_ecodistrict1.asc', IsNonZero(kolgans_leefgebied_ecodistrict1))
assign(kolgans_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(kolgans_leefgebied_ecodistrict1, 50, meter))
// write('kolgans_leefgebied_ecodistrict1_cluster.asc', IsNonZero(kolgans_leefgebied_ecodistrict1_cluster))
assign(kolgans_leefgebied_ecodistrict, cFilterGeArea(kolgans_leefgebied_ecodistrict1_cluster, 80, hectare))
write('kolgans_leefgebied_ecodistrict.asc', if(kolgans_leefgebied_ecodistrict,
  kolgans_leefgebied_ecodistrict1_cluster, 0))
```



Potentiële overwinteringskaart voor Krakeend op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// lees de shape file in uit de vogelgebiedendatabank
assign(krakeend_watervogelgebieden, 'leefgebied_krakeend_20m.asc')
assign(krakeend_watervogelgebied1, and(ge(krakeend_watervogelgebieden, 1), le(krakeend_watervogelgebieden,
283)))
deleteVariable(krakeend_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(krakeend_foerageer_bwk, bwk(ad%, ae%, aer%, aev%, ah%, aom%, ap%, ds%, hp+%, hpr%, hpr+%))
// waterlopen
assign(krakeend_waterlopen, OrEq('rastert_huetzon1.asc', 11, 12, 13, 14, 17, 18))
assign(krakeend_foerageer1, or(krakeend_foerageer_bwk, krakeend_waterlopen))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
gebied zijn!
assign(krakeend_foerageer1_cluster, fuzzyClusterIdU(krakeend_foerageer1, 50, meter))
// leefgebiedcluster moeten minimaal 5ha groot zijn
assign(krakeend_leefgebied1, cFilterGeArea(krakeend_foerageer1_cluster, 5, hectare))
assign(krakeend_leefgebied, and(krakeend_watervogelgebied1, krakeend_leefgebied1))
// baken ecoregio van ecodistrictgebieden af
assign(krakeend_ecodistrict, ge('ecoregio.asc', 1))
assign(krakeend_leefgebied_ecodistrict1, and(krakeend_leefgebied, krakeend_ecodistrict))
assign(krakeend_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(krakeend_leefgebied_ecodistrict1, 50, meter))
assign(krakeend_leefgebied_ecodistrict, cFilterGeArea(krakeend_leefgebied_ecodistrict1_cluster, 5, hectare))
write('krakeend_leefgebied_ecodistrict.asc', if(krakeend_leefgebied_ecodistrict,
krakeend_leefgebied_ecodistrict1_cluster, 0))
```

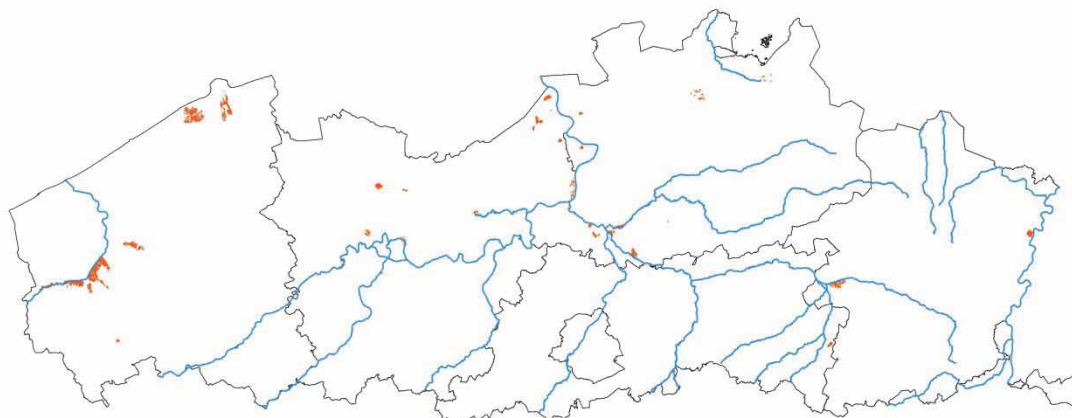


Potentiële overwinteringskaart voor Kuifeend op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m_met_maas.asc
// lees de shape file in uit de vogelgebiedendatabank
assign(kuifeend_watervogelgebieden, 'leefgebied_kuifeend_20m.asc')
assign(kuifeend_watervogelgebied1, and(ge(kuifeend_watervogelgebieden, 1), le(kuifeend_watervogelgebieden,
197)))
deleteVariable(kuifeend_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(kuifeend_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, ap%))
// waterlopen
assign(kuifeend_waterlopen, OrEq('rastert_huetzon1.asc', 17, 19))
assign(kuifeend_maas, eq('maas.asc', 1))
assign(kuifeend_foerageer1, or(kuifeend_foerageer_bwk, kuifeend_waterlopen, kuifeend_maas))
deleteVariable(kuifeend_foerageer_bwk)
deleteVariable(kuifeend_waterlopen)
deleteVariable(kuifeend_maas)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
gebied zijn!
assign(kuifeend_foerageer1_cluster, fuzzyClusterIdU(kuifeend_foerageer1, 50, meter))
// leefgebiedcluster moeten minimaal 5ha groot zijn
assign(kuifeend_leefgebied1, cFilterGeArea(kuifeend_foerageer1_cluster, 5, hectare))
assign(kuifeend_leefgebied, and(kuifeend_watervogelgebied1, kuifeend_leefgebied1))
deleteVariable(kuifeend_watervogelgebied1)
// baken ecoregio van ecodistrictgebieden af
assign(kuifeend_ecodistrict, ge('ecoregio.asc', 1))
assign(kuifeend_leefgebied_ecodistrict1, and(kuifeend_leefgebied, kuifeend_ecodistrict))
assign(kuifeend_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(kuifeend_leefgebied_ecodistrict1, 50, meter))
assign(kuifeend_leefgebied_ecodistrict, cFilterGeArea(kuifeend_leefgebied_ecodistrict1_cluster, 5, hectare))
write('kuifeend_leefgebied_ecodistrict.asc', if(kuifeend_leefgebied_ecodistrict,
kuifeend_leefgebied_ecodistrict1_cluster, 0))
```

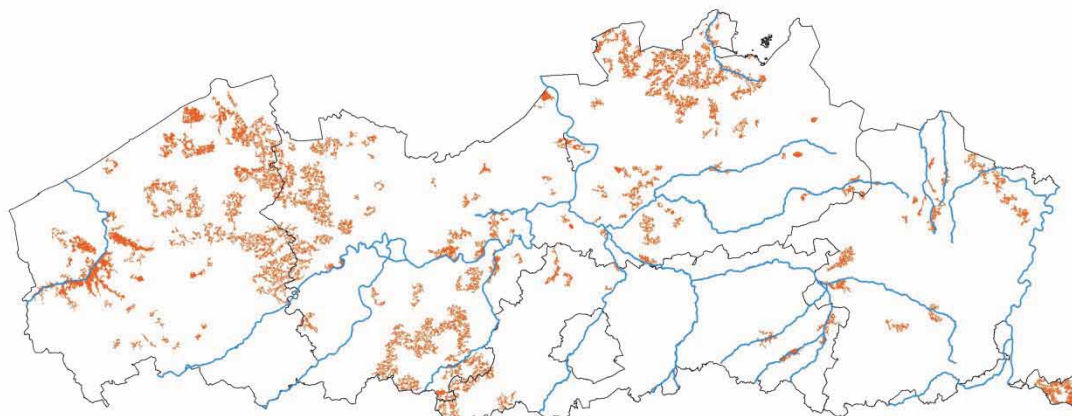
Pijlstaart (*Anas acuta*) – Adriaens & Vermeersch (in druk)



Potentiële overwinteringskaart voor Pijlstaart op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// lees de shape file in uit de vogelgebiedendatabank
assign(pijlstaart_watervogelgebieden, 'leefgebied_pijlstaart_20m.asc')
assign(pijlstaart_watervogelgebied1, and(ge(pijlstaart_watervogelgebieden, 1), le(pijlstaart_watervogelgebieden,
67)))
// baken de foerageergebieden binnen de telgebieden af
assign(pijlstaart_foerageer_bwk, bwk(ae%, aer%, aev%, ap%, da%, ds%, hp+%, hpr%, hpr+%))
// waterlopen
assign(pijlstaart_waterlopen, OrEq('rastert_huetzon1.asc', 11, 12, 13, 14, 17, 18))
assign(pijlstaart_foerageer1, or(pijlstaart_foerageer_bwk, pijlstaart_waterlopen))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
gebied zijn!
assign(pijlstaart_foerageer1_cluster, fuzzyClusterIdU(pijlstaart_foerageer1, 50, meter))
// leefgebiedcluster moeten minimaal 5ha groot zijn
assign(pijlstaart_leefgebied1, cFilterGeArea(pijlstaart_foerageer1_cluster, 5, hectare))
assign(pijlstaart_leefgebied, and(pijlstaart_watervogelgebied1, pijlstaart_leefgebied1))
//write('pijlstaart_leefgebied.asc', if(pijlstaart_leefgebied, pijlstaart_foerageer1_cluster, 0))
// baken ecoregio van ecodistrictgebieden af
assign(pijlstaart_ecodistrict, ge('ecoregio.asc', 1))
assign(pijlstaart_leefgebied_ecodistrict1, and(pijlstaart_leefgebied, pijlstaart_ecodistrict))
assign(pijlstaart_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(pijlstaart_leefgebied_ecodistrict1, 50, meter))
assign(pijlstaart_leefgebied_ecodistrict, cFilterGeArea(pijlstaart_leefgebied_ecodistrict1_cluster, 5, hectare))
write('pijlstaart_leefgebied_ecodistrict.asc', if(pijlstaart_leefgebied_ecodistrict,
pijlstaart_leefgebied_ecodistrict1_cluster, 0))
```



Potentiële overwinteringskaart voor Regenwulp op basis van de watervogeltellingen en het onderstaande gdx-script.

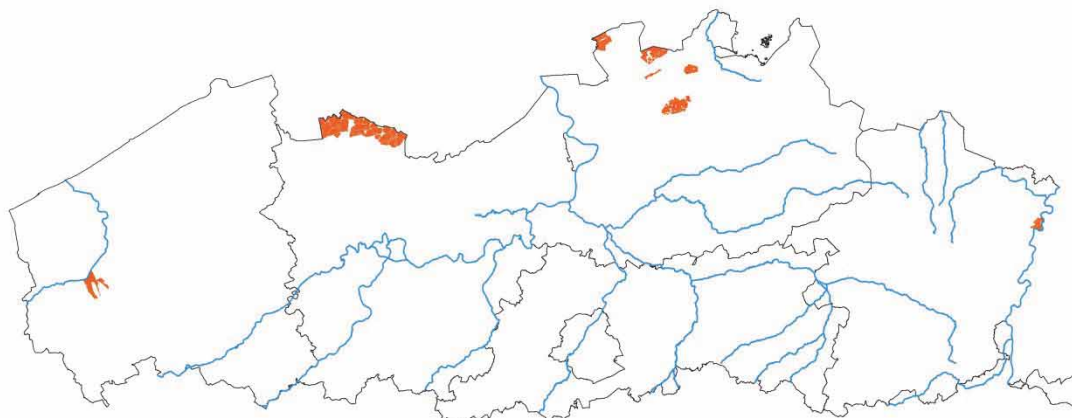
Gdx-script

```
vlaanderen_20m_met_maas.asc
// baken potentieel foerageergebied in graslanden af
assign(regenwulp_foerageer, bwk(ha, ha+, 6230%, 2330%, hc, hc+, rbbhc%, hj, hj+, hp%, hp+, hpr%, hpr+, hx%,
ku%, 6430%))
// clustering: indien geschikte cel binnen 20 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
gebied zijn!
assign(regenwulp_foerageer_cluster, fuzzyClusterIdU(regenwulp_foerageer, 50, meter))
deleteVariable(regenwulp_foerageer)
// leefgebiedcluster moeten minimaal 300ha groot zijn
assign(regenwulp_leefgebied1, cFilterGeArea(regenwulp_foerageer_cluster, 300, hectare))
deleteVariable(regenwulp_foerageer_cluster)
// baken goede foerageergebieden af
assign(regenwulp_foerageer_goed, bwk(hp, hp+, hpr, hpr+))
// clustering: indien geschikte cel binnen 20 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
gebied zijn!
assign(regenwulp_foerageer_goed_cluster, fuzzyClusterIdU(regenwulp_foerageer_goed, 50, meter))
// goed foerageercluster moeten minimaal 100ha groot zijn
assign(regenwulp_foerageer2, cFilterGeArea(regenwulp_foerageer_goed_cluster, 100, hectare))
deleteVariable(regenwulp_foerageer_goed)
deleteVariable(regenwulp_foerageer_goed_cluster)
// in foerageergebied moet minstens 100ha goed gebied zitten
assign(regenwulp_leefgebied3, cFilter(regenwulp_leefgebied1, cellDistanceLe(regenwulp_foerageer2, 50, meter)))
deleteVariable(regenwulp_leefgebied1)
deleteVariable(regenwulp_foerageer2)
// in het foerageergebied er moet ook een ondiepe plas in liggen als slaapplek
assign(regenwulp_plas, bwk(ae%, aer%, aev%, ah%, ao%, aom%, apo%, 3130%, 3140%, 3150%, da%, 1310%, 1320%,
1330%, ds%, 1140%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het plasgebied
assign(regenwulp_plas_cluster, fuzzyClusterIdU(regenwulp_plas, 50, meter))
deleteVariable(regenwulp_plas)
// plas moet minimaal 0.5ha groot zijn
assign(regenwulp_plas1, cFilterGeArea(regenwulp_plas_cluster, 0.5, hectare))
deleteVariable(regenwulp_plas_cluster)
assign(regenwulp_leefgebied4a, cFilter(regenwulp_leefgebied3, cellDistanceLe(regenwulp_plas1, 50, meter)))
deleteVariable(regenwulp_plas1)
// slikken en schorren mogen kleiner zijn
assign(regenwulp_slik, bwk(da%, 1310%, 1320%, 1330%, ds%, 1140%))
// clustering: indien geschikte cel binnen 20 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
gebied zijn!
```

```

assign(regenwulp_slik_cluster, fuzzyClusterIdU(regenwulp_slik, 50, meter))
deleteVariable(regenwulp_slik)
assign(regenwulp_slik1, cFilterGeArea(regenwulp_slik_cluster, 0.25, hectare))
deleteVariable(regenwulp_slik_cluster)
assign(regenwulp_leefgebied4b, cFilter(regenwulp_leefgebied3, cellDistanceLe(regenwulp_slik1, 100, meter)))
deleteVariable(regenwulp_leefgebied3)
deleteVariable(regenwulp_slik1)
assign(regenwulp_leefgebied5, or(regenwulp_leefgebied4a, regenwulp_leefgebied4b))
deleteVariable(regenwulp_leefgebied4a)
deleteVariable(regenwulp_leefgebied4b)
assign(regenwulp_leefgebied5_cluster, fuzzyClusterIdU(regenwulp_leefgebied5, 50, meter))
deleteVariable(regenwulp_leefgebied5)
// leefgebiedcluster moeten minimaal 50ha groot zijn
assign(regenwulp_leefgebied6, cFilterGeArea(regenwulp_leefgebied5_cluster, 50, hectare))
deleteVariable(regenwulp_leefgebied5_cluster)
// op heide gaan ze slapen als er een ven aanwezig is van >5ha
assign(regenwulp_heide, bwk(ce, ce+, ct, ct+, 4010%, 7150%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
// gebied zijn!
assign(regenwulp_heide_cluster, fuzzyClusterIdU(regenwulp_heide, 50, meter))
deleteVariable(regenwulp_heide)
// leefgebiedcluster moeten minimaal 50ha groot zijn
assign(regenwulp_heide1, cFilterGeArea(regenwulp_heide_cluster, 50, hectare))
// er moet ook een ondiepe plas in liggen
assign(regenwulp_heideplas, bwk(ae%, aer%, aev%, ah%, ao%, aom%, apo%, 3130%, 3140%, 3150%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het plasgebied
assign(regenwulp_heideplas_cluster, fuzzyClusterIdU(regenwulp_heideplas, 50, meter))
deleteVariable(regenwulp_heideplas)
// plas moet minimaal 0.5ha groot zijn
assign(regenwulp_heideplas1, cFilterGeArea(regenwulp_heideplas_cluster, 0.5, hectare))
deleteVariable(regenwulp_heide_cluster)
deleteVariable(regenwulp_heideplas_cluster)
assign(regenwulp_heide2, cFilter(regenwulp_heide1, cellDistanceLe(regenwulp_heideplas1, 100, meter)))
deleteVariable(regenwulp_heide1)
deleteVariable(regenwulp_heideplas1)
assign(regenwulp_leefgebied0, or(regenwulp_leefgebied6, regenwulp_heide2))
deleteVariable(regenwulp_leefgebied6)
deleteVariable(regenwulp_heide2)
// geen akkers
assign(regenwulp_akkers, bwk(b%))
assign(regenwulp_leefgebied, and(regenwulp_leefgebied0, not(regenwulp_akkers)))
deleteVariable(regenwulp_leefgebied0)
deleteVariable(regenwulp_akkers)
// baken ecoregio van ecodistrictgebieden af
assign(regenwulp_ecodistrict, ge('ecoregio.asc', 1))
assign(regenwulp_leefgebied_ecodistrict1, and(regenwulp_leefgebied, regenwulp_ecodistrict))
assign(regenwulp_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(regenwulp_leefgebied_ecodistrict1, 50, meter))
assign(regenwulp_leefgebied_ecodistrict, cFilterGeArea(regenwulp_leefgebied_ecodistrict1_cluster, 50, hectare))
write('regenwulp_leefgebied_ecodistrict.asc', if(regenwulp_leefgebied_ecodistrict,
    regenwulp_leefgebied_ecodistrict1_cluster, 0))

```

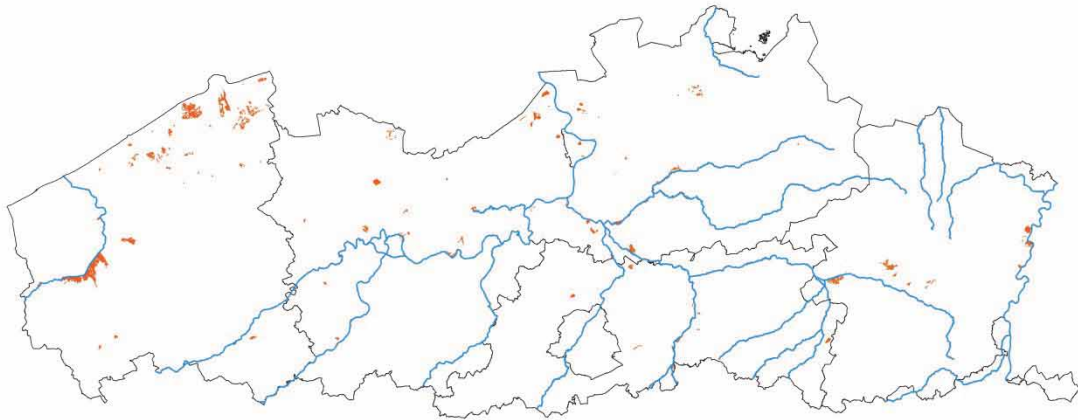


Potentiële overwinteringskaart voor Rietgans op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m_met_maas.asc
// lees de shape file in uit de vogelgebiedendatabank
assign(riegtans_watervogelgebieden, 'leefgebied_riegtans_20m.asc')
assign(riegtans_watervogelgebied1, and(ge(riegtans_watervogelgebieden, 1), le(riegtans_watervogelgebieden,
24)))
deleteVariable(riegtans_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(riegtans_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, bk%, bl%, bs%, bu%, hc%, hp%, hp+%, hpr%, hpr+%,
hx%))
// write('riegtans_foerageer_bwk.asc', IsNonZero(riegtans_foerageer_bwk))
// waterlopen
assign(riegtans_waterlopen, eq('rastert_huetzon1.asc', 19))
assign(riegtans_maas, eq('maas.asc', 1))
assign(riegtans_foerageer_epr, OrEq('epr2013.asc', 6, 38, 100, 18, 8, 4, 89, 40, 120, 7, 1))
assign(riegtans_foerageer1, or(riegtans_foerageer_bwk, rietgans_waterlopen, rietgans_maas,
riegtans_foerageer_epr))
deleteVariable(riegtans_foerageer_bwk)
deleteVariable(riegtans_waterlopen)
deleteVariable(riegtans_foerageer_epr)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
gebied zijn!
assign(riegtans_foerageer1_cluster, fuzzyClusterIdU(riegtans_foerageer1, 50, meter))
// leefgebiedcluster moeten minimaal 80ha groot zijn
assign(riegtans_leefgebied1, cFilterGeArea(riegtans_foerageer1_cluster, 80, hectare))
assign(riegtans_leefgebied, and(riegtans_watervogelgebied1, rietgans_leefgebied1))
// baken ecoregio van ecodistrictgebieden af
assign(riegtans_ecodistrict, ge('ecoregio.asc', 1))
assign(riegtans_leefgebied_ecodistrict1, and(riegtans_leefgebied, rietgans_ecodistrict))
// write('riegtans_leefgebied_ecodistrict1.asc', IsNonZero(riegtans_leefgebied_ecodistrict1))
assign(riegtans_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(riegtans_leefgebied_ecodistrict1, 50, meter))
// write('riegtans_leefgebied_ecodistrict1_cluster.asc', IsNonZero(riegtans_leefgebied_ecodistrict1_cluster))
assign(riegtans_leefgebied_ecodistrict, cFilterGeArea(riegtans_leefgebied_ecodistrict1_cluster, 80, hectare))
write('riegtans_leefgebied_ecodistrict.asc', if(riegtans_leefgebied_ecodistrict,
riegtans_leefgebied_ecodistrict1_cluster, 0))
```

Slobeend (*Anas clypeata*) – Adriaens & Vermeersch (in druk)



Potentiële overwinteringskaart voor Slobeend op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

vlaanderen_20m.asc

// lees de shape file in uit de vogelgebiedendatabank

assign(slobeend_watervogelgebieden, 'leefgebied_slobeend_20m.asc')

assign(slobeend_watervogelgebied1, and(ge(slobeend_watervogelgebieden, 1), le(slobeend_watervogelgebieden, 145)))

deleteVariable(slobeend_watervogelgebieden)

// baken de foerageergebieden binnen de telgebieden af

assign(slobeend_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, aom%, ap%, hp+%, hpr%, hpr+%))

// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open gebied zijn!

assign(slobeend_foerageer_bwk_cluster, fuzzyClusterIdU(slobeend_foerageer_bwk, 50, meter))

deleteVariable(slobeend_foerageer_bwk)

// leefgebiedcluster moeten minimaal 5ha groot zijn

assign(slobeend_leefgebied1, cFilterGeArea(slobeend_foerageer_bwk_cluster, 5, hectare))

deleteVariable(slobeend_foerageer_bwk_cluster)

assign(slobeend_leefgebied, and(slobeend_watervogelgebied1, slobeend_leefgebied1))

deleteVariable(slobeend_watervogelgebied1)

deleteVariable(slobeend_leefgebied1)

// baken ecoregio van ecodistrictgebieden af

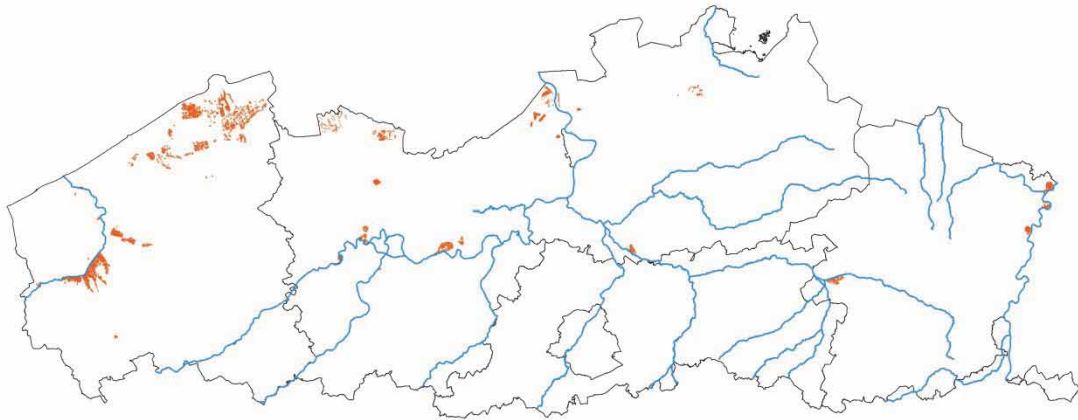
assign(slobeend_ecodistrict, ge('ecoregio.asc', 1))

assign(slobeend_leefgebied_ecodistrict1, and(slobeend_leefgebied, slobeend_ecodistrict))

assign(slobeend_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(slobeend_leefgebied_ecodistrict1, 50, meter))

assign(slobeend_leefgebied_ecodistrict, cFilterGeArea(slobeend_leefgebied_ecodistrict1_cluster, 5, hectare))

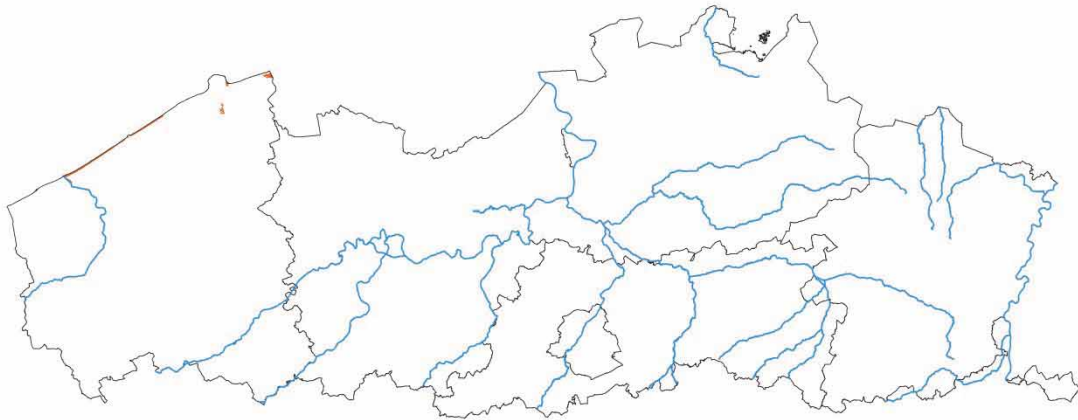
write('slobeend_leefgebied_ecodistrict.asc', if(slobeend_leefgebied_ecodistrict, slobeend_leefgebied_ecodistrict1_cluster, 0))



Potentiële overwinteringskaart voor Smient op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// lees de shape file in uit de vogelgebiedendatabank
assign(smient_watervogelgebieden, 'leefgebied_smient_20m.asc')
assign(smient_watervogelgebied1, and(ge(smient_watervogelgebieden, 1), le(smient_watervogelgebieden, 82)))
deleteVariable(smient_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(smient_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, ap%, da%, ds%, hp%, hp+%, hpr%, hpr+%))
// waterlopen
assign(smient_waterlopen, OrEq('rastert_huetzon1.asc', 11, 12, 13, 14, 17))
assign(smient_foerageer1, or(smient_foerageer_bwk, smient_waterlopen))
deleteVariable(smient_foerageer_bwk)
deleteVariable(smient_waterlopen)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
// gebied zijn!
assign(smient_foerageer1_cluster, fuzzyClusterIdU(smient_foerageer1, 50, meter))
deleteVariable(smient_foerageer1)
// leefgebiedcluster moeten minimaal 5ha groot zijn
assign(smient_leefgebied1, cFilterGeArea(smient_foerageer1_cluster, 5, hectare))
deleteVariable(smient_foerageer1_cluster)
assign(smient_leefgebied, and(smient_watervogelgebied1, smient_leefgebied1))
deleteVariable(smient_watervogelgebied1)
deleteVariable(smient_leefgebied1)
// baken ecoregio van ecodistrictgebieden af
assign(smient_ecodistrict, ge('ecoregio.asc', 1))
assign(smient_leefgebied_ecodistrict, and(smient_leefgebied, smient_ecodistrict))
deleteVariable(smient_ecodistrict)
assign(smient_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(smient_leefgebied_ecodistrict1, 50, meter))
assign(smient_leefgebied_ecodistrict, cFilterGeArea(smient_leefgebied_ecodistrict1_cluster, 5, hectare))
write('smient_leefgebied_ecodistrict.asc', if(smient_leefgebied_ecodistrict, smient_leefgebied_ecodistrict1_cluster,
0))
```

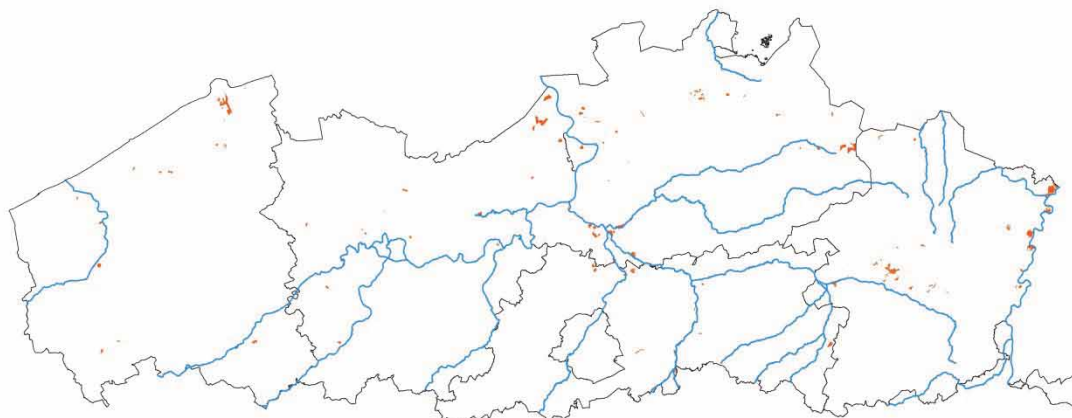


Potentiële overwinteringskaart voor Steenloper op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

vlaanderen_20m.asc

```
// lees de shape file in uit de vogelgebiedendatabank
assign(steenloper_watervogelgebieden, 'leefgebied_steenloper_20m.asc')
assign(steenloper_watervogelgebied1, and(ge(steenloper_watervogelgebieden, 1),
    le(steenloper_watervogelgebieden, 14)))
deleteVariable(steenloper_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(steenloper_foerageer_bwk, bwk(da%, dl%, dla%, dls%, ds%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
// gebied zijn!
assign(steenloper_foerageer1_cluster, fuzzyClusterIdU(steenloper_foerageer_bwk, 50, meter))
// leefgebiedcluster moeten minimaal 30ha groot zijn
assign(steenloper_leefgebied0, cFilterGeArea(steenloper_foerageer1_cluster, 30, hectare))
assign(steenloper_leefgebied1, and(steenloper_watervogelgebied1, steenloper_leefgebied0))
deleteVariable(steenloper_watervogelgebied1)
deleteVariable(steenloper_leefgebied0)
assign(steenloper_ecodistrict, ge('ecoregio.asc', 1))
assign(vlaanderen, 'vlaanderen_20m.asc')
assign(steenloper_leefgebied_ecodistrict1, and(steenloper_leefgebied1, vlaanderen))
deleteVariable(vlaanderen)
deleteVariable(steenloper_leefgebied1)
assign(steenloper_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(steenloper_leefgebied_ecodistrict1, 50, meter))
assign(steenloper_leefgebied_ecodistrict, cFilterGeArea(steenloper_leefgebied_ecodistrict1_cluster, 30, hectare))
write('steenloper_leefgebied_ecodistrict.asc', if(steenloper_leefgebied_ecodistrict,
    steenloper_leefgebied_ecodistrict1_cluster, 0))
```

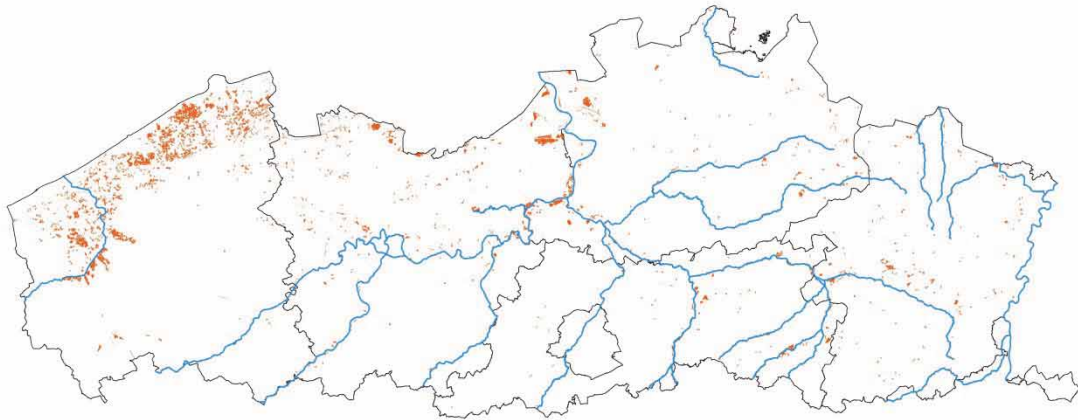


Potentiële overwinteringskaart voor Tafeleend op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// lees de shape file in uit de vogelgebiedendatabank
assign(tafeleend_watervogelgebieden, 'leefgebied_tafeleend_20m.asc')
assign(tafeleend_watervogelgebied1, and(ge(tafeleend_watervogelgebieden, 1), le(tafeleend_watervogelgebieden,
138)))
deleteVariable(tafeleend_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(tafeleend_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, aom%, ap%, ds%))
// waterlopen
assign(tafeleend_waterlopen, OrEq('rastert_huetzon1.asc', 17, 18))
assign(tafeleend_foerageer1, or(tafeleend_foerageer_bwk, tafeleend_waterlopen))
deleteVariable(tafeleend_foerageer_bwk)
deleteVariable(tafeleend_waterlopen)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
gebied zijn!
assign(tafeleend_foerageer1_cluster, fuzzyClusterIdU(tafeleend_foerageer1, 50, meter))
// leefgebiedcluster moeten minimaal 5ha groot zijn
assign(tafeleend_leefgebied1, cFilterGeArea(tafeleend_foerageer1_cluster, 5, hectare))
assign(tafeleend_leefgebied, and(tafeleend_watervogelgebied1, tafeleend_leefgebied1))
deleteVariable(tafeleend_watervogelgebied1)
deleteVariable(tafeleend_leefgebied1)
// baken ecoregio van ecodistrictgebieden af
assign(tafeleend_ecodistrict, ge('ecoregio.asc', 1))
assign(tafeleend_leefgebied_ecodistrict1, and(tafeleend_leefgebied, tafeleend_ecodistrict))
assign(tafeleend_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(tafeleend_leefgebied_ecodistrict1, 50, meter))
assign(tafeleend_leefgebied_ecodistrict, cFilterGeArea(tafeleend_leefgebied_ecodistrict1_cluster, 5, hectare))
write('tafeleend_leefgebied_ecodistrict.asc', if(tafeleend_leefgebied_ecodistrict,
tafeleend_leefgebied_ecodistrict1_cluster, 0))
```

Waterrietzanger (*Acrocephalus paludicola*) – Adriaens & Vermeersch (in druk)

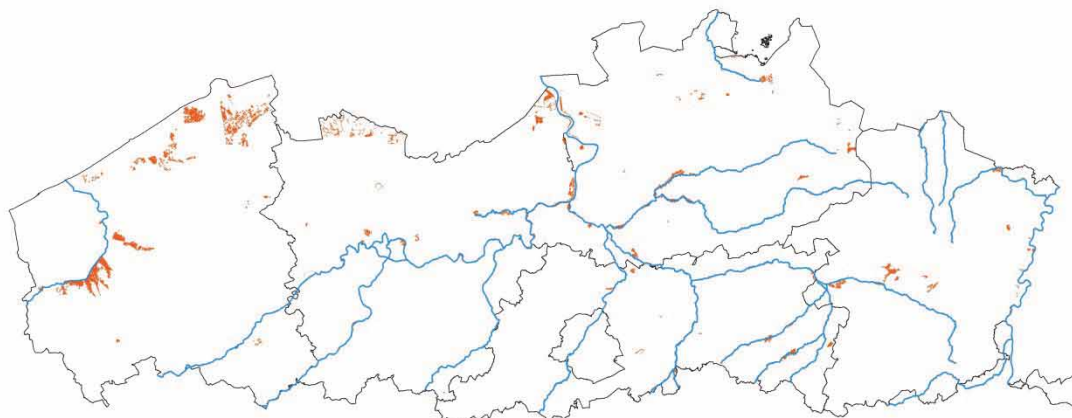


Potentiële overwinteringskaart voor Waterrietzanger op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

vlaanderen_20m.asc

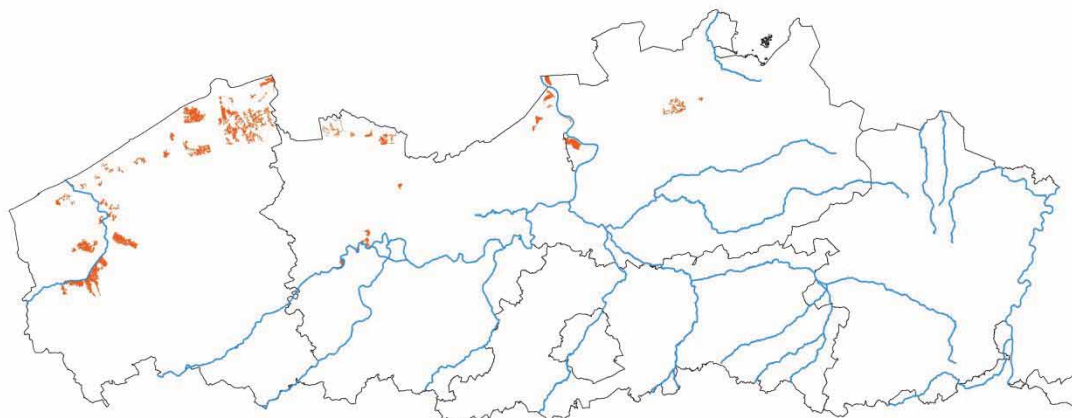
```
// baken de foerageergebieden binnen de telgebieden af
assign(waterrietzanger_foerageer_bwk, bwk(hj, hj+, mc, mc+, mr, mr+, k(mc%, k(mr%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
gebied zijn!
assign(waterrietzanger_foerageer_bwk_cluster, fuzzyClusterIdU(waterrietzanger_foerageer_bwk, 50, meter))
deleteVariable(waterrietzanger_foerageer_bwk)
// leefgebiedcluster moeten minimaal 5ha groot zijn
assign(waterrietzanger_leefgebied1, cFilterGeArea(waterrietzanger_foerageer_bwk_cluster, 5, hectare))
deleteVariable(waterrietzanger_foerageer_bwk_cluster)
assign(waterrietzanger_ecodistrict, ge('ecoregio.asc', 1))
assign(waterrietzanger_leefgebied_ecodistrict1, and(waterrietzanger_leefgebied1, waterrietzanger_ecodistrict))
deleteVariable(waterrietzanger_leefgebied1)
deleteVariable(waterrietzanger_ecodistrict)
assign(waterrietzanger_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(waterrietzanger_leefgebied_ecodistrict1,
50, meter))
assign(waterrietzanger_leefgebied_ecodistrict, cFilterGeArea(waterrietzanger_leefgebied_ecodistrict1_cluster, 5,
hectare))
write('waterrietzanger_leefgebied_ecodistrict.asc', if(waterrietzanger_leefgebied_ecodistrict,
waterrietzanger_leefgebied_ecodistrict1_cluster, 0))
```



Potentiële overwinteringskaart voor Wintertaling op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// lees de shape file in uit de vogelgebiedendatabank
assign(wintertaling_watervogelgebieden, 'leefgebied_wintertaling_20m.asc')
assign(wintertaling_watervogelgebied1, and(ge(wintertaling_watervogelgebieden, 1),
    le(wintertaling_watervogelgebieden, 204)))
deleteVariable(wintertaling_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(wintertaling_foerageer_bwk, bwk(ae%, aer%, aev%, ah%, aom%, da%, ds%, ds%, hp%, hp+%, hpr%, hpr+%,
    mr%, sf%, vm%, vn%))
// waterlopen
assign(wintertaling_waterlopen, OrEq('rastert_huetzon1.asc', 11, 12, 13, 14, 17, 18))
assign(wintertaling_foerageer1, or(wintertaling_foerageer_bwk, wintertaling_waterlopen))
deleteVariable(wintertaling_foerageer_bwk)
deleteVariable(wintertaling_waterlopen)
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
// gebied zijn!
assign(wintertaling_foerageer1_cluster, fuzzyClusterIdU(wintertaling_foerageer1, 50, meter))
// leefgebiedcluster moeten minimaal 5ha groot zijn
assign(wintertaling_leefgebied1, cFilterGeArea(wintertaling_foerageer1_cluster, 5, hectare))
deleteVariable(wintertaling_foerageer1_cluster)
assign(wintertaling_leefgebied, and(wintertaling_watervogelgebied1, wintertaling_leefgebied1))
deleteVariable(wintertaling_watervogelgebied1)
deleteVariable(wintertaling_leefgebied1)
// baken ecoregio van ecodistrictgebieden af
assign(wintertaling_ecodistrict, ge('ecoregio.asc', 1))
assign(wintertaling_leefgebied_ecodistrict1, and(wintertaling_leefgebied, wintertaling_ecodistrict))
assign(wintertaling_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(wintertaling_leefgebied_ecodistrict1, 50,
    meter))
assign(wintertaling_leefgebied_ecodistrict, cFilterGeArea(wintertaling_leefgebied_ecodistrict1_cluster, 5, hectare))
write('wintertaling_leefgebied_ecodistrict.asc', if(wintertaling_leefgebied_ecodistrict,
    wintertaling_leefgebied_ecodistrict1_cluster, 0))
```



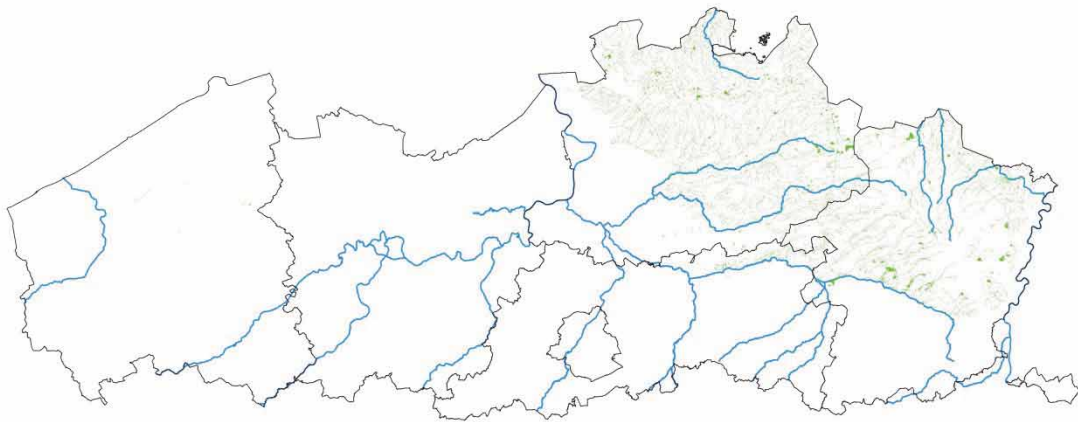
Potentiële overwinteringskaart voor Wulp op basis van de watervogeltellingen en het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// lees de shape file in uit de vogelgebiedendatabank
assign(wulp_watervogelgebieden, 'leefgebied_wulp_20m.asc')
assign(wulp_watervogelgebied1, and(ge(wulp_watervogelgebieden, 1), le(wulp_watervogelgebieden, 88)))
deleteVariable(wulp_watervogelgebieden)
// baken de foerageergebieden binnen de telgebieden af
assign(wulp_foerageer_bwk, bwk(aer%, aev%, ah%, apo%, aom%, aoo%, da%, ds%, hp%, hp+%, hpr%, hpr+%, ku%,
kz%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied - er moet voldoende open
gebied zijn!
assign(wulp_foerageer1_cluster, fuzzyClusterIdU(wulp_foerageer_bwk, 50, meter))
// leefgebiedcluster moeten minimaal 30ha groot zijn
assign(wulp_leefgebied0, cFilterGeArea(wulp_foerageer1_cluster, 30, hectare))
assign(wulp_leefgebied1, and(wulp_watervogelgebied1, wulp_leefgebied0))
deleteVariable(wulp_watervogelgebied1)
deleteVariable(wulp_leefgebied0)
assign(wulp_ecodistrict, ge('ecoregio.asc', 1))
assign(wulp_leefgebied_ecodistrict1, and(wulp_leefgebied1, wulp_ecodistrict))
deleteVariable(wulp_leefgebied1)
deleteVariable(wulp_ecodistrict)
assign(wulp_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(wulp_leefgebied_ecodistrict1, 50, meter))
assign(wulp_leefgebied_ecodistrict, cFilterGearea(wulp_leefgebied_ecodistrict1_cluster, 30, hectare))
write('wulp_leefgebied_ecodistrict.asc', if(wulp_leefgebied_ecodistrict, wulp_leefgebied_ecodistrict1_cluster, 0))
```

PLANTEN

Drijvende waterweegbree (*Luronium natans*) – Denys et al. (2015)

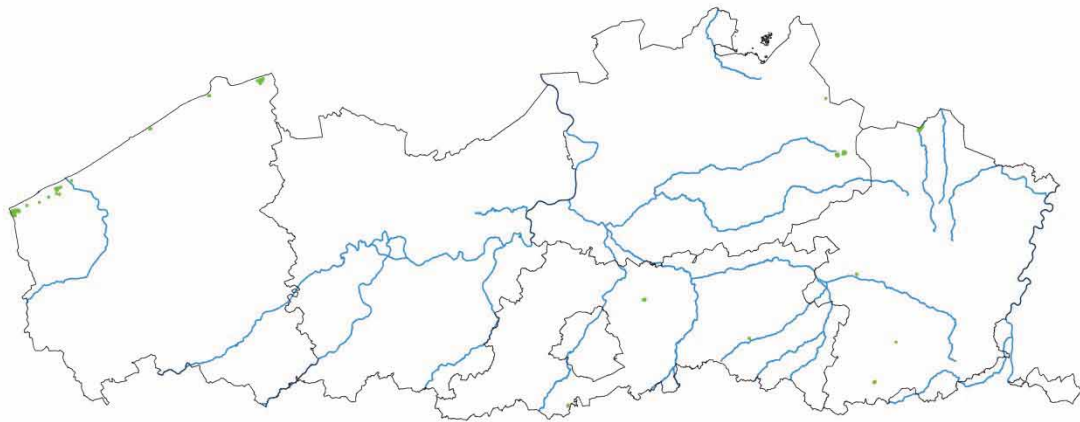


Potentiële leefgebiedenkaart voor Drijvende waterweegbree op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(drijvendewaterweegbree_voortplanting_bwk, bwk(ao, ao+, aom, aom+, ap, ap+, k(ao), k(aom), 3110%,
  3130%, 3140%, 3150%, 3260%))
// waterlopen
assign(drijvendewaterweegbree_waterlopen, or('rastert_huetzon1.asc', 5, 6, 7, 8, 9, 10))
assign(drijvendewaterweegbree_voortplanting, or(drijvendewaterweegbree_voortplanting_bwk,
  drijvendewaterweegbree_waterlopen))
deleteVariable(drijvendewaterweegbree_voortplanting_bwk)
deleteVariable(drijvendewaterweegbree_waterlopen)
// clustering: indien geschikte cel binnen 20 meter, dan hoort die cel bij het voortplantingscluster
assign(drijvendewaterweegbree_voortplanting_cluster, fuzzyClusterIdU(drijvendewaterweegbree_voortplanting,
  20, meter))
// optimale voortplantingsclusters moeten minimaal 0.16ha groot zijn
assign(drijvendewaterweegbree_leefgebied, cFilterGeArea(drijvendewaterweegbree_voortplanting_cluster, 0.16,
  hectare))
// beperken tot de Kempen en zandig Vlaanderen
assign(drijvendewaterweegbree_ecodistrict, or('ecoregio.asc', 7, 8, 12, 13, 14, 15, 16, 17, 11, 29))
assign(drijvendewaterweegbree_leefgebied_ecodistrict, and(drijvendewaterweegbree_leefgebied,
  drijvendewaterweegbree_ecodistrict))
write('drijvendewaterweegbree_leefgebied_ecodistrict.asc', if(drijvendewaterweegbree_leefgebied_ecodistrict,
  drijvendewaterweegbree_voortplanting_cluster, 0))
```

Groenknolorchis (*Liparis loeselii*) – Van Landuyt (2015b)



Potentiële leefgebiedenkaart voor Groenknolorchis op basis van het onderstaande gdx-script.

Gdx-script

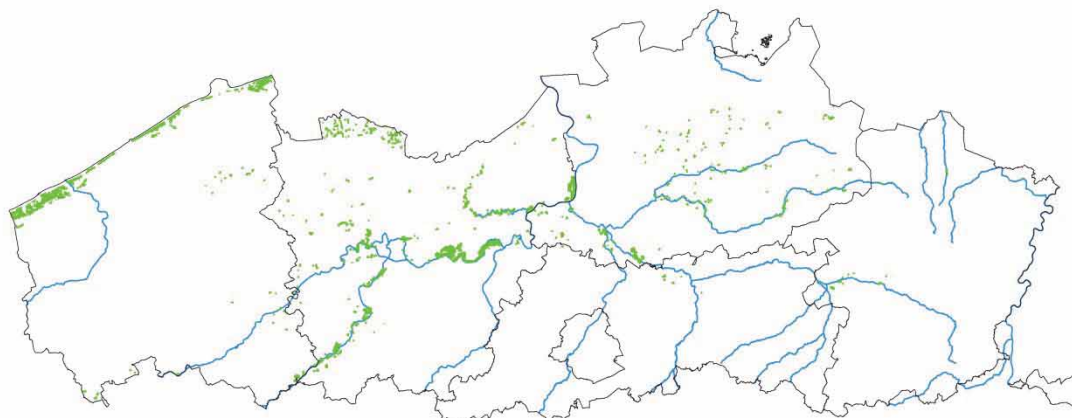
vlaanderen_20m.asc

```
// baken de voortplantingsgebieden af (zonder opgespoten terreinen want dat zijn artificiële leefgebieden)
assign(groenknolorchis_voortplanting_bwk, bwk(mk, mk+, mm, mm+, mp, mp-, 2190%, 7140%, 7210%, 7230%))
// clustering: indien geschikte cel binnen 20 meter, dan hoort die cel bij het voortplantingscluster
assign(groenknolorchis_voortplanting_cluster, fuzzyClusterIdU(groenknolorchis_voortplanting_bwk, 20, meter))
deleteVariable(groenknolorchis_voortplanting_bwk)
// optimale voortplantingsclusters moeten minimaal 0.16ha groot zijn
assign(groenknolorchis_leefgebied, cFilterGeArea(groenknolorchis_voortplanting_cluster, 0.16, hectare))
write('groenknolorchis_leefgebied.asc', if(groenknolorchis_leefgebied, groenknolorchis_voortplanting_cluster, 0))
```



Groenknolorchis (foto Jeroen Mentens – Vilda)

Kruipend moerasscherm (*Apium repens*) – Van Landuyt (2015c)



Potentiële leefgebiedenkaart voor Kruipend moerasscherm op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af buiten de duinen
assign(kruipendmoerasscherm_voortplanting_rest1, bwk(hpr, hpr+))
// overstromingskaart
assign(kruipendmoerasscherm_osg, ge('OVSTRGEV2014_20m.asc', 1))
assign(kruipendmoerasscherm_voortplanting_rest, and(kruipendmoerasscherm_voortplanting_rest1,
  kruipendmoerasscherm_osg))
deleteVariable(kruipendmoerasscherm_voortplanting_rest1)
deleteVariable(kruipendmoerasscherm_osg)
// clustering: indien geschikte cel binnen 20 meter, dan hoort die cel bij het voortplantingscluster
assign(kruipendmoerasscherm_voortplanting_rest_cluster,
  fuzzyClusterIdU(kruipendmoerasscherm_voortplanting_rest, 20, meter))
deleteVariable(kruipendmoerasscherm_voortplanting_rest)
// optimale voortplantingsclusters moeten minimaal 0.16ha groot zijn
assign(kruipendmoerasscherm_leefgebied_rest, cFilterGeArea(kruipendmoerasscherm_voortplanting_rest_cluster,
  0.16, hectare))
assign(kruipendmoerasscherm_ecodistrict_rest, OrEq('ecoregio.asc', 3, 4, 13))
assign(kruipendmoerasscherm_leefgebied_ecodistrict_rest, and(kruipendmoerasscherm_leefgebied_rest,
  kruipendmoerasscherm_ecodistrict_rest))
deleteVariable(kruipendmoerasscherm_leefgebied_rest)
// write('kruipendmoerasscherm_leefgebied_ecodistrict_rest.asc',
  if(kruipendmoerasscherm_leefgebied_ecodistrict_rest, kruipendmoerasscherm_voortplanting_rest_cluster, 0))
deleteVariable(kruipendmoerasscherm_voortplanting_rest_cluster)
deleteVariable(kruipendmoerasscherm_ecodistrict_rest)
// baken de voortplantingsgebieden af binnen de duinen
assign(kruipendmoerasscherm_voortplanting_duin, bwk(mp, mp+, mp-, kn, hd%, hpr, hpr+, 2130%, 2190%))
// clustering: indien geschikte cel binnen 20 meter, dan hoort die cel bij het voortplantingscluster
assign(kruipendmoerasscherm_voortplanting_duin_cluster,
  fuzzyClusterIdU(kruipendmoerasscherm_voortplanting_duin, 20, meter))
deleteVariable(kruipendmoerasscherm_voortplanting_duin)
// optimale voortplantingsclusters moeten minimaal 0.16ha groot zijn
assign(kruipendmoerasscherm_leefgebied_duin,
  cFilterGeArea(kruipendmoerasscherm_voortplanting_duin_cluster, 0.16, hectare))
assign(kruipendmoerasscherm_ecodistrict_duin, eq('ecoregio.asc', 1))
assign(kruipendmoerasscherm_leefgebied_ecodistrict_duin, and(kruipendmoerasscherm_leefgebied_duin,
  kruipendmoerasscherm_ecodistrict_duin))
deleteVariable(kruipendmoerasscherm_ecodistrict_duin)
// write('kruipendmoerasscherm_leefgebied_ecodistrict_duin.asc',
  if(kruipendmoerasscherm_leefgebied_ecodistrict_duin, kruipendmoerasscherm_voortplanting_duin_cluster, 0))
deleteVariable(kruipendmoerasscherm_voortplanting_duin_cluster)
```

```
write('kruipendmoerasscherm_leefgebied.asc',  
      fuzzyclusterIdU(or(kruipendmoerasscherm_leefgebied_ecodistrict_rest,  
                        kruipendmoerasscherm_leefgebied_ecodistrict_duin), 20, meter))
```



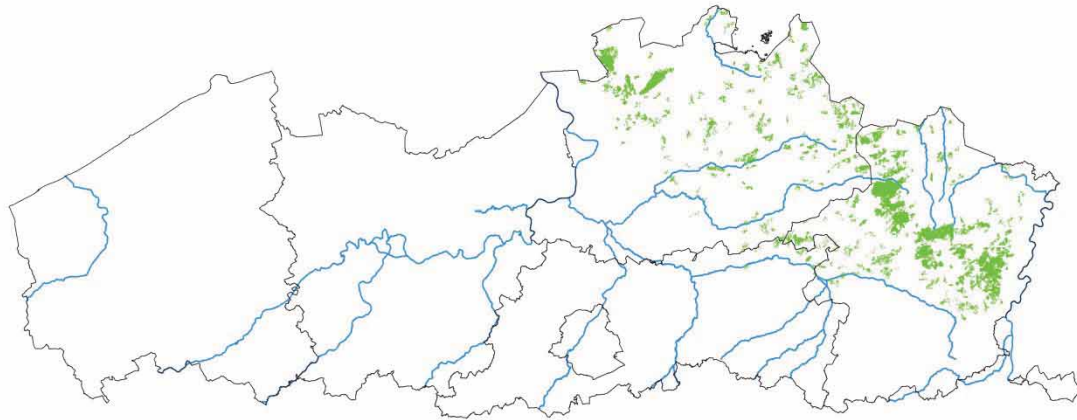
Kruipend moerasscherm (foto Yves Adams – Vilda)



Gladde slang (foto Lars Soerink)

REPTIELEN

Gladde slang (*Coronella austriaca*) – Speybroeck (2015)



Potentiële leefgebiedenkaart voor Gladde slang op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
assign(sluitingsgraad_bos, 'bosinventaris_2001_sluitingsgraad.asc')
assign(G, 'groenkaart2013_20m.asc')
assign(wegen, 'GRB_Wegverbinding_at.asc')
assign(Ecoregio, 'Ecoregio.asc')
// Geschikte biotopen
// c?b, ces (natte hei met hoogveenelementen), hab, pmb, pmh, pms (naalldhoutaanplant niet grove den met
// ondergroei), ppmh, ppms, qs (zuur eikenbos), kc (ontginning) nog toegevoegd
assign(biotopen_geslacht_vlakdekkend, bwk(cd%, ce%, cg%, cm%, cp%, cv%, dm%, ha, ha-, ha+, hab, hab-, hab+, hc,
// hc-, hc+, hk%, hm%, hn%, kc, kg, ks, kt, kt-, kt+, ppmb, ppmb-, ppmb+, ppmh, ppmh-, ppmh+, ppms, ppms-,
// ppms+, pmb, pmb-, pmb+, pmh, pmh-, pmh+, pms, pms-, pms+, qb, qb-, qb+, qs, qs-, qs+, se, se-, se+, sg, sg-, sg+,
// sk, sk-, sk+, sm, sm-, sm+, 2310%, 2330%, 4010%, 4030%, 5130%, 6210%, 6230%, 6410%, 7150%, rbbhc%, rbbsg%,
// rbbism%, rbbppm%))
// write('gladdeslang_biotopen_geslacht_vlakdekkend.asc', biotopen_geslacht_vlakdekkend)
// lijnvormige varianten gedefinieerd als kt(), k() of kh() van vlakdekkende biotopen hierboven
assign(biotopen_geslacht_lijnvormig, bwk(k(cd%, k(ce%, k(cg%, k(cm%, k(cp%, k(cv%, k(dm%, k(ha), k(ha-), k(ha+),
// k(hab), k(hab-), k(hab+), k(hc), k(hc-), k(hc+), k(hk%, k(hm%, k(hn%, k(kc), k(kg), k(ks), k(kt), k(kt-), k(kt+), k(ppmb),
// k(ppmb-), k(ppmb+), k(ppmh), k(ppmh-), k(ppmh+), k(ppms), k(ppms-), k(ppms+), k(pmb), k(pmb-), k(pmb+),
// k(pmh), k(pmh-), k(pmh+), k(pms), k(pms-), k(pms+), k(qb), k(qb-), k(qb+), k(qs), k(qs-), k(qs+), k(se), k(se-), k(se+),
// k(sg), k(sg-), k(sg+), k(sk), k(sk-), k(sk+), k(sm), k(sm-), k(sm+), kh(cd%, kh(ce%, kh(cg%, kh(cm%, kh(cp%, kh(cv%,
// kh(dm%, kh(ha), kh(ha-), kh(ha+), kh(hab), kh(hab-), kh(hab+), kh(hc), kh(hc-), kh(hc+), kh(hk%, kh(hm%, kh(hn%,
// kh(kc), kh(kg), kh(ks), kh(kt), kh(kt-), kh(kt+), kh(ppmb), kh(ppmb-), kh(ppmb+), kh(ppmh), kh(ppmh-), kh(ppmh+),
// kh(ppms), kh(ppms-), kh(ppms+), kh(pmb), kh(pmb-), kh(pmb+), kh(pmh), kh(pmh-), kh(pmh+), kh(pms), kh(pms-),
// kh(pms+), kh(qb), kh(qb-), kh(qb+), kh(qs), kh(qs-), kh(qs+), kh(se), kh(se-), kh(se+), kh(sg), kh(sg-), kh(sg+), kh(sk),
// kh(sk-), kh(sk+), kh(sm), kh(sm-), kh(sm+), kt(cd%, kt(ce%, kt(cg%, kt(cm%, kt(cp%, kt(cv%, kt(dm%, kt(ha), kt(ha-),
// kt(ha+), kt(hab), kt(hab-), kt(hab+), kt(hc), kt(hc-), kt(hc+), kt(hk%, kt(hm%, kt(hn%, kt(kc), kt(kg), kt(ks), kt(kt),
// kt(kt-), kt(kt+), kt(ppmb), kt(ppmb-), kt(ppmb+), kt(ppmh), kt(ppmh-), kt(ppmh+), kt(ppms), kt(ppms-), kt(ppms+),
// kt(pmb), kt(pmb-), kt(pmb+), kt(pmh), kt(pmh-), kt(pmh+), kt(pms), kt(pms-), kt(pms+), kt(qb), kt(qb-), kt(qb+),
// kt(qs), kt(qs-), kt(qs+), kt(se), kt(se-), kt(se+), kt(sg), kt(sg-), kt(sg+), kt(sk), kt(sk-), kt(sk+), kt(sm), kt(sm-), kt(sm+)))
// write('gladdeslang_biotopen_geslacht_lijnvormig.asc', biotopen_geslacht_lijnvormig)
// geschikte biotopen zijn de som van de vlakdekkende en de lijnvormige varianten van de opgegeven bwk-codes
assign(biotopen_geslacht, or(biotopen_geslacht_vlakdekkend, biotopen_geslacht_lijnvormig))
deletevariable(biotopen_geslacht_lijnvormig)
deletevariable(biotopen_geslacht_vlakdekkend)
// verbossing is alles wat volgens de groenkaart 2013 geclassificeerd werd als "hooggroen"
assign(verbossing, or(eq(G, 1), bwk(91%)))
deletevariable(G)
// droge, open en lijnvormige biotopen binnen de geschikte biotopen: niet alleen de echte -en volgens de
// groenkaart niet verboste-, open biotopen, maar ook de bossen met een sluitingsgraad van <= 2/3
```

```

assign(biotopen_geschikt_droog_open_lijn, or(and(biotopen_geschikt, bwk(c%, d%, h%, k%, s%, 2310%, 2330%,
4010%, 4030%, 5130%, 6210%, 6230%, 6410%, 7150%, rbbhc%, rbbsg%, rbbism%), not(verbossing)),
and(and(biotopen_geschikt, bwk(p%, q%, rbbppm%)), or(eq(sluitingsgraad_bos, 2), eq(sluitingsgraad_bos, 1))))
deletevariable(verbossing)
deletevariable(sluitingsgraad_bos)
// write('gladdeslang_biotopen_geschikt_droog_open_lijn.asc', biotopen_geschikt_droog_open_lijn)
// biotopen_geschikt bestaat subset van vlakdekkende en lijnvormige biotopen, met name uit
// biotopen_geschikt_droog_open_lijn enerzijds en een 50 m buffer hierrond in de niet-open bossen of verboste
// varianten van open biotopen; op die manier verdwijnen aaneengesloten, dichte bossen of verboste percelen uit
// de leefgebiedenkaart, tenzij ze dus op korte afstand (i.e. 50 m) liggen van de meer open biotopen waar de meeste
// waarnemingen zich concentreren
assign(biotopen_geschikt, or(biotopen_geschikt_droog_open_lijn, and(biotopen_geschikt,
le(celldistance(edge(biotopen_geschikt, biotopen_geschikt_droog_open_lijn), 0), 50/cellsizesize))))
// write('gladdeslang_biotopen_geschikt_b.asc', biotopen_geschikt)
// clustering; bufferafstand van 100 m genomen
assign(id_buffer, ClusterId(le(CellDistance(biotopen_geschikt, 0), 100/cellsizesize)))
assign(id, biotopen_geschikt * id_buffer)
deletevariable(id_buffer)
assign(y, cFilterGeArea(id, 20, hectare))
// write('gladdeslang_clusters_100m_20ha_b.asc', y)
assign(id, y * isnonzero(y))
// clustering met alle typen wegen als barrière, m.u.v. wandel- en fietswegen die niet toegankelijk zijn voor andere
// voertuigen; minimale oppervlakte is 20 ha
assign(barrier, and(ne(wegen, 114), ne(wegen, 0)))
assign(barrier, or(barrier, edge(eq(wegen, 114), and(ne(wegen, 0), ne(wegen, 114)))))
deletevariable(wegen)
assign(x, and(y, not(barrier)))
assign(x, x * y)
assign(z, clusterid(not(barrier)))
deletevariable(barrier)
assign(id_barrier, x * z)
assign(id_barrier_20ha, cFilterGeArea(id_barrier, 20, hectare))
deletevariable(id_barrier)
// assign(id_barrier_20ha_opp, cSum(id_barrier_20ha, 1))
// write('gladdeslang_clusters_100m_20ha_barriersnelwegen_b.asc', id_barrier_20ha)
// write('gladdeslang_clusters_100m_20ha_barriersnelwegen_opp_ha_b.asc', id_barrier_20ha_opp * cellsizesize2 /
// 10000)
// indien keuze met bovenvermelde wegen als barrière
assign(id, id_barrier_20ha)
// de cluster mag niet bestaan uit louter biotopen die niet ergens een open karakter hebben
// (biotopen_geschikt_droog_open_lijn)
// assign(droog_open_lijn_fractie, cSum(id, biotopen_geschikt_droog_open_lijn)/cSum(id, 1))
// write('gladdeslang_clusters_100m_20ha_droog_open_fractie_b.asc', droog_open_lijn_fractie)
assign(y, cFilterGeArea(id, 1, cellsizesize))
// write('gladdeslang_clusters_100m_20ha_droog_open_gt0_b.asc', cSum(y, 1)*cellsizesize2/10000)
// enkel de leefgebieden die overlappen met de ecoregio Kempen
assign(z, cFilter(y, ge(cSum(y, and(ge(Ecoregio, 12), le(Ecoregio, 17))), 1)))
write('gladdeslang_leefgebied.asc', z)

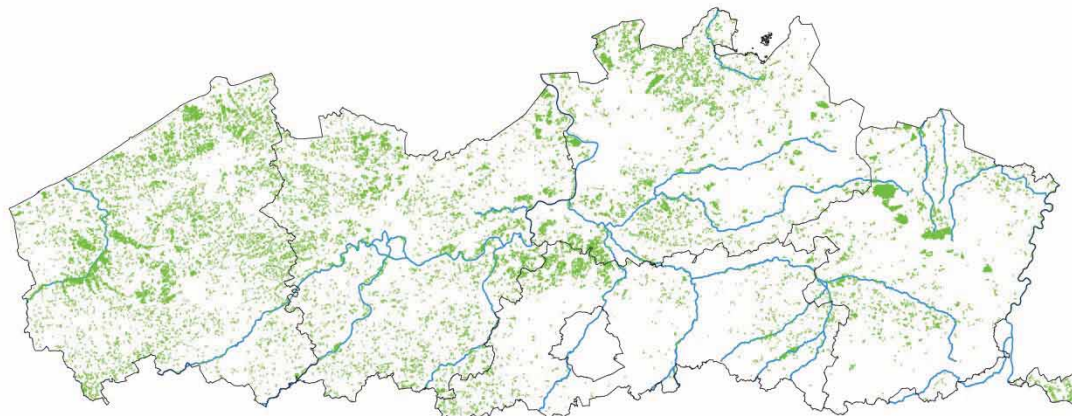
```

VELDVOGELS

Graspieper (*Anthus pratensis*)

Ecologisch profiel (mondelinge mededeling T'jollyn & Vermeersch)

Een koppel Graspiepers heeft een leefgebied nodig van minstens 5ha dat bestaat uit weilanden in grote weidecomplexen in polders en riviervalleien, grazige vochtige heide, opgespoten en braakterreinen, grazige terrils, brede wegbermen, dijken en rivieroeveren. Dit leefgebied moet minstens 100m van de bosrand afliggen en er mogen geen hoge bomen in staan.



Potentiële leefgebiedenkaart voor Graspieper op basis van het onderstaande gdx-script.

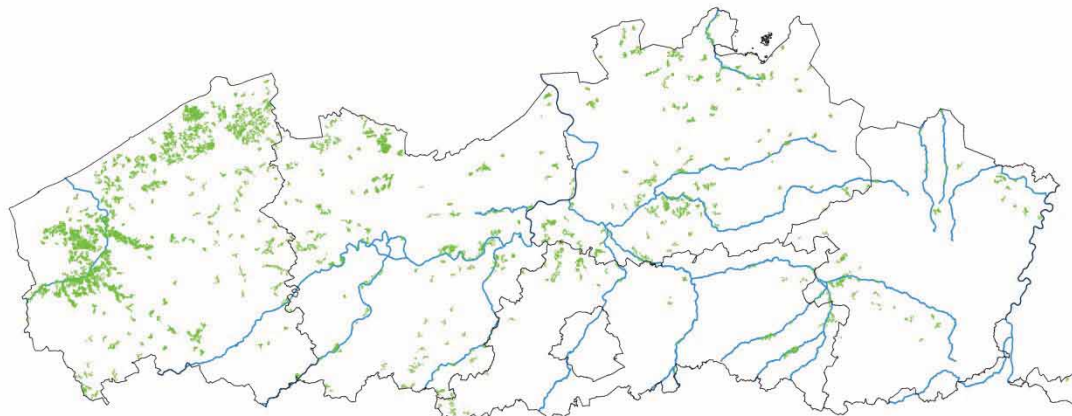
Gdx-script

```
vlaanderen_20m.asc
// Geschikte biotopen
assign(graspieper_bwk, bwk(hd, hd+, hd-, had%, ce, ce+, ce-, ces, ces+, ces-, cm, cm+, cm-, cd, cd+, cd-, ha, ha+, ha-,
  hc, hc+, hc-, hk, hk+, hk-, hm, hm+, hm-, hmm%, hme%, hmo%, hn, hn+, hn-, hu, hu+, hu-, hj, hj+, hj-, hp+, hpr,
  hpr+, hz, hz+, hz-, ku, ku+, ku-, kz))
// afstand van 100m tot bosrand
assign(graspieper_bos, eq('groenkaart2013_20m.asc', 1))
// beperking van impact van bos door te stellen dat de afstand van 100 m enkel geldt als de bossen/bomen een
  aaneengesloten oppervlakte van minstens 1 ha hebben
assign(graspieper_bos_cluster, fuzzyClusterIdU(graspieper_bos, 20, meter))
assign(graspieper_bos1, cFilterGeArea(graspieper_bos_cluster, 1, hectare))
assign(graspieper_leefgebied1, and(graspieper_bwk, celldistanceGe(graspieper_bos1, 100, meter)))
// Clustering met bufferafstand 50m
assign(graspieper_leefgebied1_cluster, fuzzyClusterIdU(graspieper_leefgebied1, 50, meter))
// leefgebied moet minstens 5 ha groot zijn
assign(graspieper_leefgebied, cFilterGeArea(graspieper_leefgebied1_cluster, 5, hectare))
write('graspieper_leefgebied.asc', if(graspieper_leefgebied, graspieper_leefgebied1_cluster, 0))
```

Grutto (*Limosa limosa*)

Ecologisch profiel (mondelinge mededeling T'jollyn & Vermeersch)

Een koppel grutto's heeft nood aan minstens 100 ha aaneengesloten leefgebied dat bij voorkeur bestaat uit soortenrijk, permanent grasland met verspreide kleine poelen en vochtige laagtes ingebed in een groter landbouwgebied van minstens 1000 ha. Het is belangrijk dat het leefgebied erg open is met bij voorkeur geen bomenrijen, houtkanten of bosranden langs de perceelsranden. Variatie in hoogte van de grassen is belangrijk en er dient doorheen het leefgebied een mozaiek beschikbaar te zijn van cultuurgraslanden en hooilanden.



Potentiële leefgebiedenkaart voor Grutto op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// Externe lagen
assign(W, 'GRB_watergang_at.asc')
// assign(EPR, 'epr2013.asc')
assign(OVSTR, 'OVSTRGEV2014_20m.asc')
assign(Groenkaart2013, 'groenkaart2013_20m.asc')
assign(Drainage, 'bodem_drainage.asc')
assign(Textuur, 'bodem_textuur.asc')
// geschikte biotopen
// afstand tot bebouwing; met bebouwing bestaande uit clusters van min. 10ha aaneengesloten bebouwing
// (bufferafstand 50 m)
assign(bebouwing, and(bwk(u%), not(bwk(ulm))))
assign(bebouwing_cluster_10ha_50m, ge(csum(FuzzyClusterId(bebouwing, 50/cellsz), 1), 10*10000/cellsz2))
assign(bebouwing_afstand, celldistance(bebouwing_cluster_10ha_50m, 0))
//write('grutto_bebouwing_cluster_10ha_50m_bis.asc', bebouwing_cluster_10ha_50m)
// verbossing
assign(verbossing, bwk(91%, gml, gmn, f%, n, n+, n-, l, ls%, lh%, p%, q%, ru%, v%, s%, kb%, kh%))
assign(hooggroen, eq(Groenkaart2013, 1))
assign(hooggroen_afstand, celldistance(ge(csum(fuzzyclusterid(hooggroen, 50/cellsz), 1), 5), 0))
// soortenrijk en permanent grasland
assign(cultuurgrasland, and(bwk(hj+, hj, hp+, hpr+, hpr, hc-, hm-, hme-, hmo-, hu-), not(hooggroen)))
//write('grutto_cultuurgrasland_bis.asc', cultuurgrasland)
assign(hooiland, and(bwk(hc, hc+, hm, hm+, hme, hme+, hmm, hmm+, hmo, hmo+, hu, hu+), not(hooggroen)))
//write('grutto_hooiland_bis.asc', hooiland)
// kleine eutrofe poelen; overlay maken van externe laag GRB_watergang én BWK-polygonen met vermelding van
// ae% of kn%; mag code ah, en evt. da, ook meegenomen worden?
assign(poelen, or(and(W, and(bwk(ae%, kn%), not(bwk(aes))))), and(le(clustersize(W), 10), gt(clustersize(W), 0),
not(bebouwing), not(hooggroen))))
//write('grutto_poelen_le10_bis.asc', poelen)
// kernbiotoop bestaat uit hpr, hpr+, hp+ en hj+, of uit hp en hj op een maximale afstand van 100m van effectief
// overstromingsgevoelig gebied; kernbiotopen moeten daarnaast:
// even nat of natter zijn dan drainageklasse d
// op minstens 100 m van bebouwing gelegen zijn
```

```

// op meer dan 50 m van aaneengesloten oppervlakte van hooggroen (>3m hoog, >20 are groot) gelegen zijn
// geen hooggroen bevatten (>3m hoog)
assign(kernbiotoop, and(or(bwk(hpr, hpr+, hp+, hj+), and(bwk(hp, hj), lt(celldistance(eq(OVSTR, 1), 0),
100/cellsize))), or(OrEq('bodem_drainage.asc', 1,2,3,7,10,11,12,14,15), and(or(eq('bodem_textuur.asc', 1),
eq('bodem_textuur.asc', 2)), eq('bodem_drainage.asc', 0))), ge(bebouwing_afstand, 100/cellsize),
ge(hooggroen_afstand, 50/cellsize), not(hooggroen)))
//write('grutto_kernbiotoop_bis.asc', kernbiotoop)
// extra hp, hx, hr bovenop cultuurgrasland en hooiland
assign(hp_hr_hx, and(bwk(hx%, hp, hr+, hr-, hr), not(or(hooiland, cultuurgrasland)), not(hooggroen)))
//write('grutto_hp_hr_hx_extra_bis.asc', hp_hr_hx)
// kernbiotoop, met binnen een afstand van 200 meter ook nog het aanwezige cultuurgrasland, hooiland, of extra
hp/hx/hr op minstens 100m van bebouwing, en op minstens 50m van aaneengesloten hooggroen van minstens 20
are groot
assign(kernbiotoop_overig_biotoop_200m, or(kernbiotoop, and(and(or(cultuurgrasland, hooiland, poelen,
hp_hr_hx), ge(bebouwing_afstand, 100/cellsize), ge(hooggroen_afstand, 50/cellsize)),
le(celldistance(kernbiotoop, 0), 200/cellsize))))
//write('grutto_kernbiotoop_overig_biotoop_200m.asc', kernbiotoop_overig_biotoop_200m)
// clustering
// clustering van kernbiotoop met ander omliggend (binnen 200m) biotoop, met bufferafstand 50m
assign(cluster_id, fuzzyclusterid(kernbiotoop_overig_biotoop_200m, 50/cellsize))
assign(cluster_opp_ha, csum(cluster_id, 1)*cellsize2/10000)
//write('grutto_cluster_opp_ha.asc', cluster_opp_ha)
// enkel de clusters overhouden die
// groter zijn dan 25 ha én minstens 5 ha kernbiotoop bevatten
// tussen de 10 en 25 ha groot zijn én voor 90% uit kernbiotoop bestaan
// kleiner dan 10 ha zijn maar op afstand van minder dan 200 m liggen van clusters die groter zijn dan 25 ha én
minstens 5 ha kernbiotoop bevatten
assign(cluster_finaal, cfilter(cluster_id, or(and(lt(csum(cluster_id, 1), 25*10000/cellsize), ge(csum(cluster_id, 1),
10*10000/cellsize2), ge(csum(cluster_id, kernbiotoop)/csum(cluster_id, 1), 0.90)), and(ge(csum(cluster_id, 1),
25*10000/cellsize2), ge(csum(cluster_id, kernbiotoop), 5*10000/cellsize2)), and(lt(csum(cluster_id, 1),
10*10000/cellsize2), lt(celldistance(ge(csum(cluster_id, 1), 25*10000/cellsize2), 0), 200/cellsize),
ge(csum(cluster_id, kernbiotoop), 2*10000/cellsize2))))))
//write('grutto_cluster_bis.asc', cluster_finaal)
// finale leefgebieden
write('grutto_leefgebied.asc', cluster_finaal)

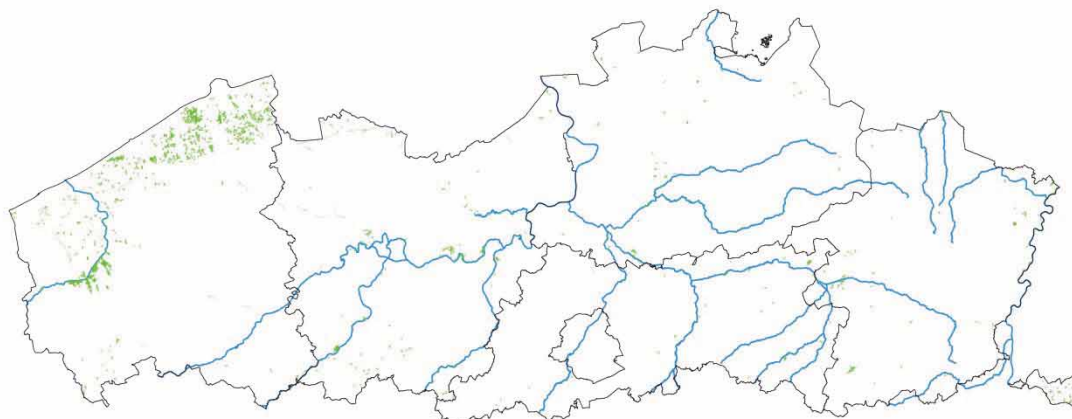
```



Grutto (foto Koen Devos)

Paapje (*Saxicola rubetra*)**Ecologisch profiel (mondelinge mededeling T'jollyn & Vermeersch)**

Een koppel Paapjes heeft nood aan een leefgebied van 5 ha bestaande uit vochtige structuurrijke graslanden, bloemrijke hooilanden, heides en venen en jonge bosaanplanten. Dit leefgebied moet minstens 100m van de bosrand afliggen.



Potentiële leefgebiedenkaart voor Paapje op basis van het onderstaande gdx-script.

Gdx-script

vlaanderen_20m.asc

```
assign(biotopen, bwk(hpr+, hf, hf+, hf-, hu, hu+, hu-, hc, hc+, hc-, hm, hm+, hm-, hmm, hmm+, hmm-, hme, hme+,  
  hme-, hmo, hmo+, hmo-, hj+, hr+, t, t+, tm, tm+, ms, ms+, ms-, mm, mm+, mm-, mk, mk+, mk-, md, md+, md-, mp,  
  mp+, mp-))
```

```
// 100 m van de bosrand
```

```
assign(biotopen, and(biotopen, ge(celldistance(bwk(91%, gml, gmn, f%, n, n+, n-, l, ls%, lh%, p%, q%, ru%, v%), 0),  
  100/cellsize)))
```

```
//write('paapje_geschikte_biotopen.asc', biotopen)
```

```
// Clustering
```

```
assign(id, fuzzyclusterid(biotopen, 100/cellsize))
```

```
assign(cluster, cFilter(id, ge(cSum(id, 1), 5*10000/cellsize2)))
```

```
write('paapje_leefgebied.asc', cluster)
```

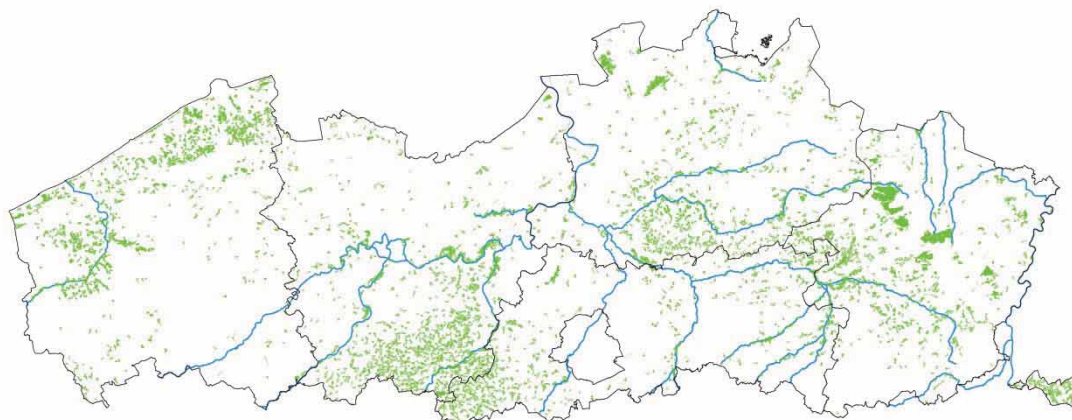


Paapje (foto Glenn Vermeersch)

Roodborsttapuit (*Saxicola torquata*)

Ecologisch profiel (mondelinge mededeling T'jollyn & Vermeersch)

Het leefgebied van roodborsttapuit in Vlaanderen bestaat ofwel uit open heidegebieden van minimaal 5 ha waarbij een beperkte mate van verbossing wordt getolereerd ofwel uit extensief landbouwgebied met veel perceelsranden en houtkanten. De soort vertoont binnen het landbouwgebied een voorkeur voor soortenrijke graslanden in vallei- of poldergebieden met veel sloten en beken, maar kan ook in raaigrasland en zelfs intensief akkerland gedijen indien er microreliëf en een hoog aandeel perceelsranden aanwezig is. Een gemiddeld territorium in landbouwgebied in Vlaanderen is ca. 10 ha groot, in heidegebieden bedraagt de gemiddelde territoriumgrootte 5 ha.



Potentiële leefgebiedenkaart voor Roodborsttapuit op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// Externe lagen
assign(G, 'groenkaart2013_20m.asc')
assign(W, 'GRB_watergang_at.asc')
// biotopen
// heidebiotopen, met lijnvormige varianten erbij:
assign(biotoop_heide, bwk(cd%, cg%, cm%, cv%, dm%, ha%, hd%, k(cd%, k(cg%, k(cm%, k(cv%, k(dm%, k(ha%,
  k(hd%, kt(cd%, kt(cg%, kt(cm%, kt(cv%, kt(dm%, kt(ha%, kt(hd%))
// soortenrijke graslanden:
assign(biotoop_grasland_soortenrijk, bwk(hj, hp+, hpr, hpr+))
// kle's op basis van bwk (dus volledige polygonen uit bwk gaan mee als er ergens kle's in aanwezig zijn)
assign(biotoop_kle_bwk, bwk(k, kb%, kh%, kp%, kj%, kt%, kw%, k(% , kh(% , kt(%))
// kle's op basis van groenkaart
// lijnvormig hooggroen
assign(biotoop_kle_groenkaart_gr3m, edge(eq(G, 1), gt(G, 1)))
assign(biotoop_kle_groenkaart_gr3m, and(biotoop_kle_groenkaart_gr3m, ne(MajorityFilter(G, 1, 2), 1)))
// lijnvormig laaggroen dat geen landbouw is
assign(biotoop_kle_groenkaart_lt3m, edge(eq(G, 2), and(ge(G, 1), ne(G, 2))))
assign(biotoop_kle_groenkaart_lt3m, and(biotoop_kle_groenkaart_lt3m, ne(MajorityFilter(G, 1, 2), 2)))
// kle's die minstens 100 m lang zijn (bufferafstand 2 rastercellen; indien kle enkel bestaan uit laaggroen, moet er op
  een afstand van 200m ook kle met hooggroen aanwezig zijn)
assign(id_kle_100m, cfilter(fuzzyclusterid(or(biotoop_kle_groenkaart_lt3m, biotoop_kle_groenkaart_gr3m), 2),
  ge(csum(fuzzyclusterid(or(biotoop_kle_groenkaart_lt3m, biotoop_kle_groenkaart_gr3m), 2), 1), 5)))
assign(id_kle_100m_met_gr3m, cfilter(id_kle_100m, ge(csum(id_kle_100m, biotoop_kle_groenkaart_gr3m), 1)))
assign(id_kle_100m_zonder_gr3m_ok, cfilter(id_kle_100m, and(eq(csum(id_kle_100m,
  biotoop_kle_groenkaart_gr3m), 0), lt(celldistance(id_kle_100m_met_gr3m, 0), 200/cellsize))))
assign(biotoop_kle_groenkaart, or(id_kle_100m_met_gr3m, id_kle_100m_zonder_gr3m_ok))
//write('roodborsttapuit_kle_groenkaart_100m_40m_bis.asc', biotoop_kle_groenkaart)
deletevariable(id_kle_100m)
deletevariable(id_kle_100m_met_gr3m)
deletevariable(id_kle_100m_zonder_gr3m_ok)
deletevariable(biotoop_kle_groenkaart_gr3m)
```

```

deletevariable(biotoop_kle_groenkaart_lt3m)
// keuze kle (bwk of groenkaart)
assign(biotoop_kle, biotoop_kle_groenkaart)
// assign(biotoop_kle, biotoop_kle_bwk)
// intensief landbouwgebied bestaat uit hp, hp-, hx% en hr% onder voorwaarde dat er ofwel kleine
// landschapelementen aanwezig zijn, ofwel de afstand tot kle kleiner is dan 100 m, ofwel er water aanwezig is op
// minder dan 100 meter
assign(biotoop_landbouw_intensief, and(bwk(hp, hp-, hx%, hr%), le(celldistance(or(biotoop_kle, W), 0),
100/cellsize)))
// bos: vanaf aaneengesloten oppervlakte hooggroen van 0.5ha
assign(bos, cfilter(clusterid(eq(G, 1)), ge(csum(clusterid(eq(G, 1)), 1), 0.5*10000/cellsize2)))
// kernbiotopen bestaan uit ofwel (1) heidebiotopen, ofwel (2) soortenrijk grasland met kle's op minder dan 200m,
// ofwel (3) soortenrijk grasland met op minder dan 100m afstand intensieve landbouwbiotopen zoals hierboven
// gedefinieerd
// en geen aaneengesloten stukken bos bevatten van meer dan 0.5 ha
assign(kernbiotopen_heide, and(biotoop_heide, not(bos)))
assign(kernbiotopen_overig, and(or(and(biotoop_grasland_soortenrijk, le(celldistance(biotoop_kle, 0),
200/cellsize)), and(biotoop_grasland_soortenrijk, le(celldistance(biotoop_landbouw_intensief, 0), 100/cellsize))),
not(bos)))
//write('roodborsttapuit_kernbiotopen_bis.asc', or(kernbiotopen_heide, kernbiotopen_overig))
// satellietbiotopen bestaan uit intensieve landbouwbiotopen (zie hierboven voor inhoud) op een afstand van
// maximaal 100m van kernbiotoop
// die geen aaneengesloten stukken bos bevatten van meer dan 0.5 ha
assign(satellietbiotopen, and(biotoop_landbouw_intensief, le(celldistance(or(kernbiotopen_heide,
kernbiotopen_overig), 0), 100/cellsize), not(bos)))
//write('roodborsttapuit_satellietbiotopen_bis.asc', satellietbiotopen)
assign(biotoop_geschikt, or(kernbiotopen_heide, kernbiotopen_overig, satellietbiotopen))
// Clustering met bufferafstand 50 m
assign(id, Fuzzyclusterid(biotoop_geschikt, 50/cellsize))
// Minimale oppervlakte is functie van verhouding heide/overig biotoop; puur heide, dan minimaal 5 ha; helemaal
// geen heide, dan minimaal 10 ha
assign(min_cluster_opp, 10*10000/cellsize2 - 5*10000/cellsize2*cSum(id, biotoop_heide)/cSum(id, 1))
assign(cluster, cFilter(id, ge(cSum(id, 1), min_cluster_opp)))
write('roodborsttapuit_leefgebied.asc', cluster)

```

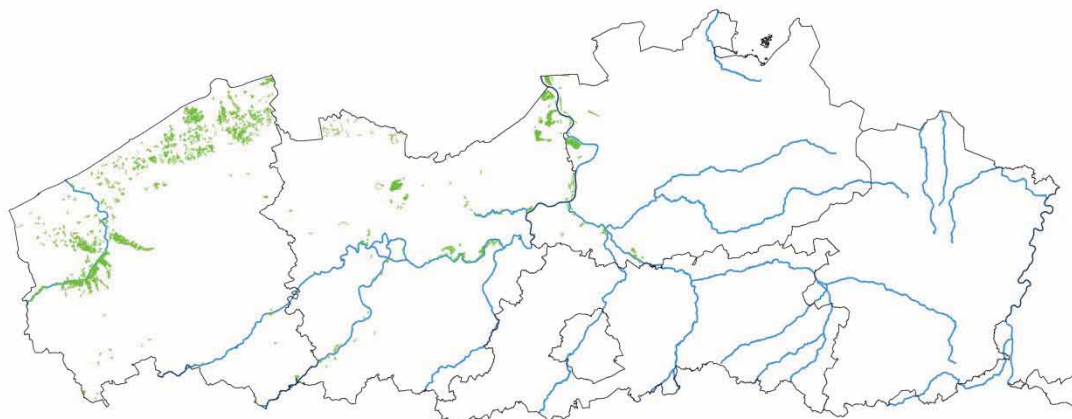


Roodborsttapuit (foto Glenn Vermeersch)

Tureluur (*Tringa totanus*)

Ecologisch profiel (mondelinge mededeling T'jollyn & Vermeersch)

Een koppel Tureluurs heeft een leefgebied nodig van 10-15ha dat bestaat uit graslanden, heides met vennen, oude opgespoten terreinen, schorren en vochtige en zilte graslanden. Hogere dichtheden worden gehaald in slikken en schorregebieden: 5-10ha per broedpaar.



Potentiële leefgebiedenkaart voor Tureluur op basis van het onderstaande gdx-script.

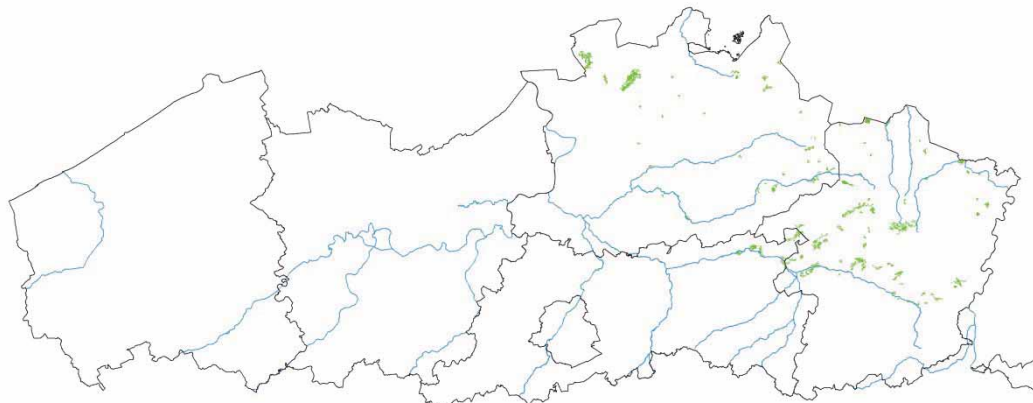
Gdx-script

```
vlaanderen_20m.asc
assign(biotopen, bwk(kz, da%, k(da%, ds%, ah%, k(ah%, ao%, k(ao%, hpr, hpr+))
assign(biotopen_slik_schor, bwk(da%, k(da%, ds%, ah%, k(ah%))
//write('tureluur_geschikte_biotopen.asc', biotopen)
// Clustering
assign(id, fuzzyclusterid(biotopen, 100/cellsize))
assign(min_cluster_opp, 10*10000/cellsize2 - 5*10000/cellsize2*cSum(id, biotopen_slik_schor)/cSum(id, 1))
assign(cluster, cFilter(id, ge(cSum(id, 1), min_cluster_opp)))
//write('tureluur_clusters_100m.asc', cluster)
// In kempen blijken heel wat clusters te liggen die enkel uit ao% bestaan; die moeten er nog uit
assign(cluster_ao_lt95, cFilter(cluster, lt(cSum(cluster, bwk(ae%, ao%, k(ao%))/cSum(cluster, 1), 0.95)))
//write('tureluur_clusters_100m_ao_lt95.asc', cluster_ao_lt95)
// beperking tot ecodistricten in Polders, IJzervallei, Meetjesland en Schelde (LO): Kustpoldersdistrict,
// Getijdenschelde- en -poldersdistrict, Pleistoceen riviervalleiendistrict, Lemig IJzer-Leie interfluviumdistrict
assign(ecodistrict, or(and(ge('Ecoregio.asc', 2), le('Ecoregio.asc', 4)), eq('Ecoregio.asc', 20)))
assign(cluster_ecodistrict, cfilter(cluster_ao_lt95, ge(csum(cluster_ao_lt95, ecodistrict), 1)))
//write('tureluur_clusters_100m_ao_lt95_ecodistrict.asc', cluster_ecodistrict)
write('tureluur_leefgebied.asc', cluster_ecodistrict)
```

Watersnip (*Gallinago gallinago*)

Ecologisch profiel (mondelinge mededeling T'jollyn & Vermeersch)

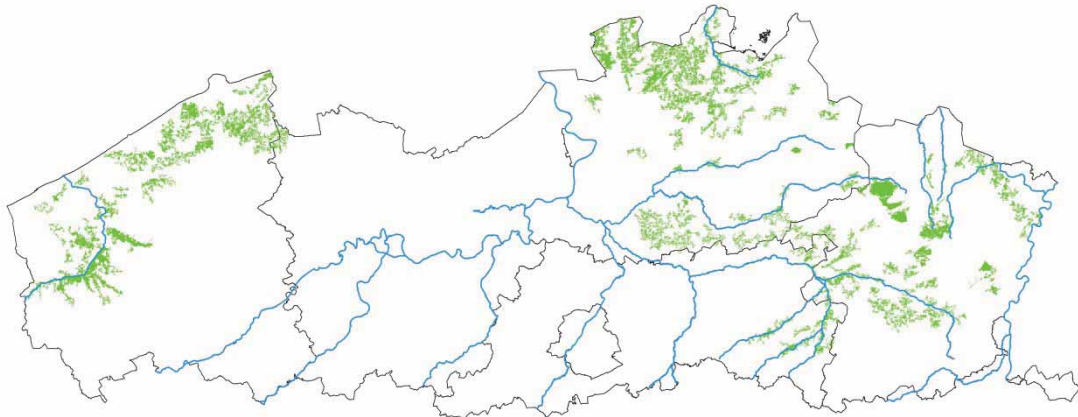
Een koppel Watersnippen heeft 10-15ha leefgebied nodig dat bestaat uit laagvenen, verlandingszones, zegges en biezenvelden, structuurrijke natte graslanden & natte heide.



Potentiële leefgebiedenkaart voor Watersnip op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// Biotopen (zonder kle-vorm)
assign(biotopen, bwk(ao%, mc%, hj%, hc%, hf%, md%, mk%, mm%, ms%, ce%, hm%, hpr, hpr+))
// Enkel even nat of natter dan drainageklasse d
assign(biotopen, and(biotopen, or(OrEq('bodem_drainage.asc', 1,2,3,7,10,11,12,14,15), eq('bodem_drainage.asc',
0))))
//write('watersnip_geschikte_biotopen.asc', biotopen)
// Clustering met minimale oppervlakte van 10 ha en bufferafstand van 100m
assign(id, fuzzyclusterid(biotopen, 100/cellsz))
assign(clusters, cfilter(id, ge(csum(id, 1), 10*10000/cellsz2)))
//write('watersnip_clusters_10ha_100m.asc', clusters)
assign(id, clusters)
// Extra eisen aan clusters
// overlay overstromingsgevoelige gebieden.
// openheid biotopen. minder relevant, aangezien de soort ook kan broeden in (open) broekbossen; verbossing is
// dus niet echt een probleem; aandeel mag echter niet te hoog liggen: max. 75%
assign(y, cfilter(id, lt(csum(id, eq('groenkaart2013_20m.asc', 1))/csum(id, 1), 0.75)))
//write('watersnip_clusters_10ha_100m_bos_lt75.asc', y)
assign(id, y)
// een cluster mag maximaal voor 90% uit open water bestaan. Open water evalueren met grb_watergang, eerder
// dan met bwk, vanwege problematiek complexen in bwk
assign(y, cfilter(id, lt(csum(id, eq('GRB_watergang_at.asc', 1))/csum(id, 1), 0.90)))
//write('watersnip_clusters_10ha_100m_bos_lt75_openwater_lt90.asc', y)
assign(id, y)
// er moet minstens één aaneengesloten blok van terrestrisch biotoop van 10 ha aanwezig zijn
assign(y, cfilter(id, ge(cmax(id, csum(clusterid(and(biotopen, not(bwk(ao%))))), 1), 10*10000/cellsz2)))
//write('watersnip_clusters_10ha_100m_bos_lt75_openwater_lt90_terrest_contig_10ha.asc', y)
assign(id, y)
// beperken tot ecodistricten van de ecoregio Kempen
assign(y, cfilter(id, ge(csum(id, and(ge('Ecoregio.asc', 12), le('Ecoregio.asc', 17))), 1)))
write('watersnip_leefgebied.asc', y)
```



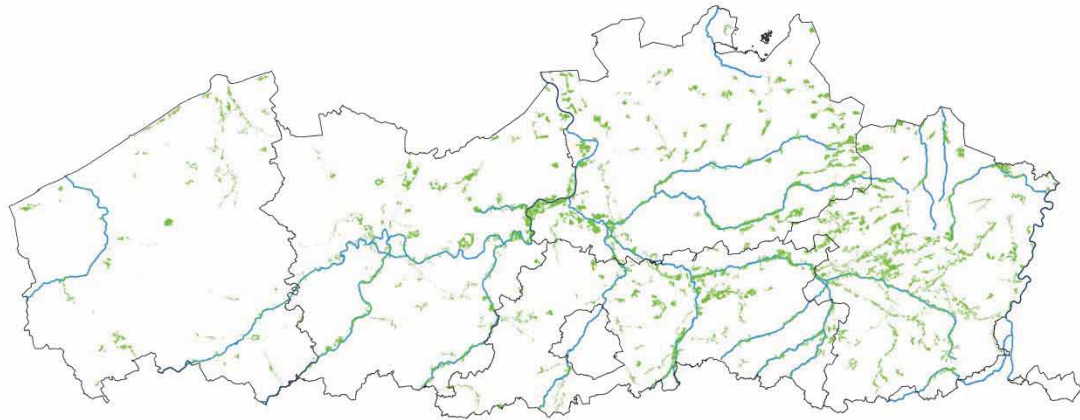
Potentiële leefgebiedenkaart voor Wulp op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
//Externe lagen
assign(Textuur, 'bodem_textuur.asc')
assign(Profontw, 'bodem_profiel.asc')
assign(Drainage, 'bodem_drainage.asc')
assign(Ecoregio, 'Ecoregio.asc')
assign(OVSTR, 'OVSTRGEV2014_20m.asc')
assign(Groenkaart, 'groenkaart2013_20m.asc')
//Biotopen
assign(biotoop_grasland_nat, bwk(hpr, hpr+, hp+))
assign(biotoop_overig, bwk(ha, ha+, ha-, da%, ds%, ah%, ao%, cg%, ce%, dm, dm+, dm-, t, t+))
//extra biotoop meenemen
assign(biotoop_extra_nat, and(bwk(hp, hx, hx+), not(bwk(b%))))
//enkel graslanden en extra biotoop dat gelegen is in valleigebieden (geen profielontwikkeling, of drainageklassen
natter dan of gelijk aan d, of in overstromingsgevoelig gebied)
assign(biotoop_grasland_nat, and(biotoop_grasland_nat, or(ge(OVSTR, 1),
OrEq('bodem_drainage.asc', 1,2,3,7,10,11,12,14,15), eq(Profontw, 1), eq(Profontw, 6))))
assign(biotoop_extra_nat, and(biotoop_extra_nat, or(ge(OVSTR, 1),
OrEq('bodem_drainage.asc', 1,2,3,7,10,11,12,14,15), eq(Profontw, 1), eq(Profontw, 6))))
assign(verbossing, isnonzero(cfilter(clusterid(eq(Groenkaart, 1)), ge(csum(clusterid(eq(Groenkaart, 1)), 1), 5))))
assign(biotoop_geschikt, and(or(biotoop_grasland_nat, biotoop_extra_nat, biotoop_overig), not(verbossing)))
//write('wulp_geschikte_biotopen_d_bis.asc', biotoop_geschikt)
//Clustering met bufferafstand van 100 m
assign(id, fuzzyclusterid(biotoop_geschikt, 100/cellsz))
//oppervlaktevereiste van 100 ha
assign(id_100ha, cFilter(id, ge(cSum(id, 1), 100*10000/cellsz2)))
//write('wulp_clusters_100ha_100m_d_bis.asc', id_100ha)
//enkel clusters die overlappen met de ecodistricten van kempen, polders van IJzervallei, Vochtig-Haspengouw
district, of Velp-Gete district
assign(id_100ha_ecodistrict, cFilter(id_100ha, ge(cSum(id_100ha, or(and(ge(Ecoregio, 12), le(Ecoregio, 17)),
eq(Ecoregio, 2), eq(Ecoregio, 30), eq(Ecoregio, 31))), 1)))
write('wulp_leefgebied.asc', id_100ha_ecodistrict)
```

ZOOGDIEREN

Europese bever (*Castor fiber*) – Van Den Berge et al. (2015)



Potentiële leefgebiedenkaart voor Europese bever op basis van het onderstaande.gdx-script.

Gdx-script

vlaanderen_20m_met_maas.asc

// baken de leefgebieden af

```
assign(europesebever_leefgebied_bwk, bwk(ae%, aer%, aev%, apo%, app%, gml%, k(a%, kn%, ku%, lh%, n, n-, n+, pop%, mc%, md%, mr%, ms%, ru%, rud%, sf%, sm%, sz%, va%, vc%, vf%, vm%, vn%, vo%, vt%))
```

// waterlopen

```
assign(europesebever_waterlopen66, OrEq('rastert_huetzon1.asc', 1, 17, 18, 19))
```

```
assign(europesebever_maas, eq('maas.asc', 1))
```

```
assign(europesebever_waterlopen, or(europesebever_waterlopen66, europesebever_maas))
```

```
deleteVariable(europesebever_waterlopen66)
```

```
deleteVariable(europesebever_maas)
```

```
assign(europesebever_leefgebied0, or(europesebever_leefgebied_bwk, europesebever_waterlopen))
```

```
deleteVariable(europesebever_leefgebied_bwk)
```

// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het leefgebied

```
assign(europesebever_leefgebied0_cluster, fuzzyClusterIdU(europesebever_leefgebied0, 50, meter))
```

```
deleteVariable(europesebever_leefgebied0)
```

// leefgebiedcluster moeten minimaal 50ha groot zijn

```
assign(europesebever_leefgebied1, cFilterGeArea(europesebever_leefgebied0_cluster, 50, hectare))
```

```
// write('europesebever_leefgebied1.asc', if(europesebever_leefgebied1, europesebever_leefgebied0_cluster, 0))
```

// > 1 km oever

```
// assign(europesebever_waterlopen_cluster, fuzzyClusterIdU(europesebever_waterlopen, 100, meter))
```

// oever moet minimaal 1 km lang zijn (20 x 20 x 1000 = 2 ha)

```
// assign(europesebever_oever, cFilterGeArea(europesebever_waterlopen_cluster, 1, hectare))
```

```
// assign(europesebever_leefgebied1a, and(europesebever_leefgebied1, cellDistanceLe(europesebever_oever, 50, meter)))
```

// 10-20 m oeverzones met 15-30% houtige opslag (wilgen, populieren, elzen of berken met <8 cm)

// overige: 50% struweel en rietland/biezen, 50% grasland

```
deleteVariable(europesebever_waterlopen)
```

```
assign(europesebever_houtig, bwk(gml%, kb%, lh%, n, n+, n-, pop%, ru%, rud%, sf%, sm%, sz%, va%, vc%, vf%, vm%, vn%, vo%, vt%))
```

// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het houtig leefgebied

```
assign(europesebever_houtig_cluster, fuzzyClusterIdU(europesebever_houtig, 50, meter))
```

```
deleteVariable(europesebever_houtig)
```

```
assign(europesebever_houtig1, cFilterGeArea(europesebever_houtig_cluster, 5, hectare))
```

```
// write('europesebever_houtig1.asc', if(europesebever_houtig1, europesebever_houtig_cluster, 0))
```

```
deleteVariable(europesebever_houtig_cluster)
```

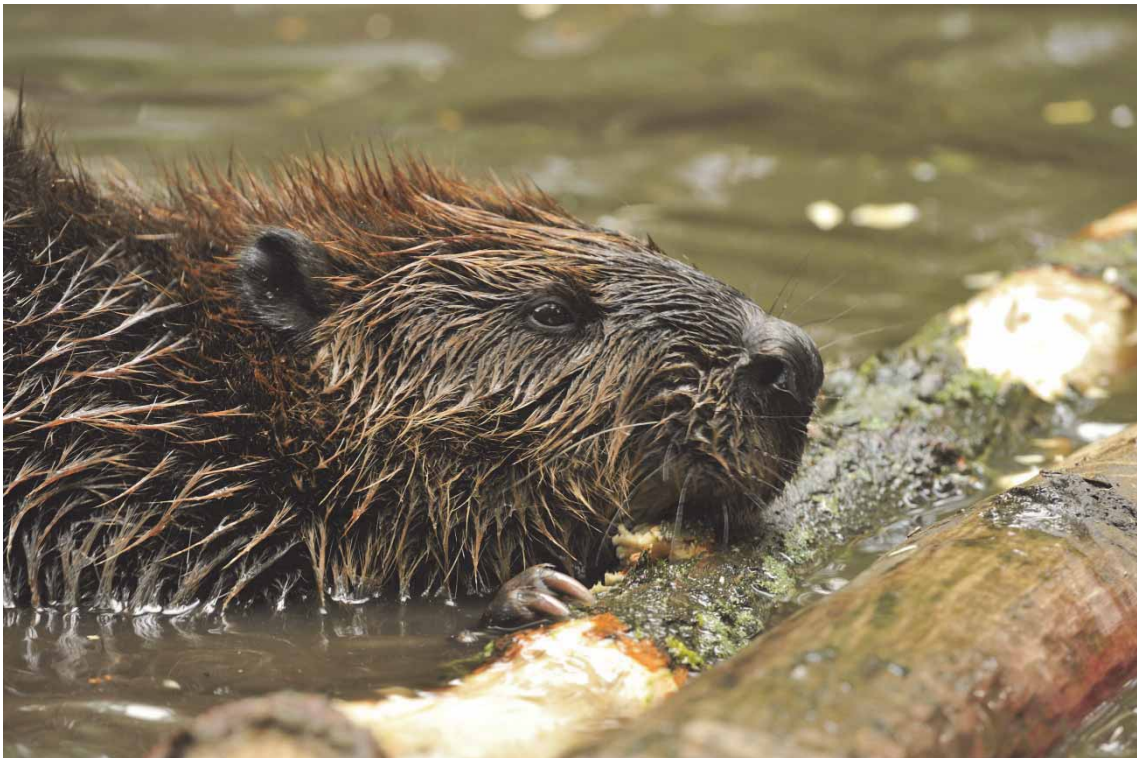
```
assign(europesebever_leefgebied2, and(europesebever_leefgebied1, cellDistanceLe(europesebever_houtig1, 50, meter)))
```

```
assign(europesebever_houtig2, and(europesebever_houtig1, cellDistanceLe(europesebever_leefgebied1, 50, meter)))
```

```

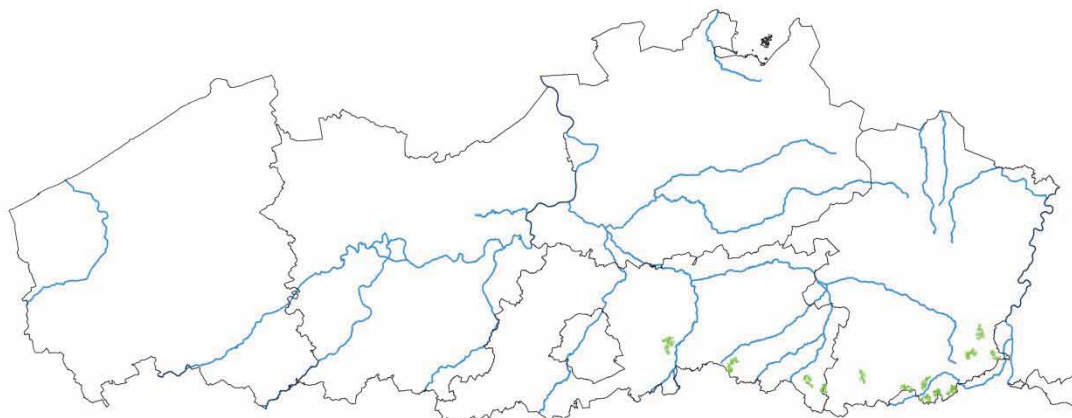
deleteVariable(europesebever_houtig1)
deleteVariable(europesebever_leefgebied1)
assign(europesebever_leefgebied3, or(europesebever_leefgebied2, europesebever_houtig2))
write('europesebever_leefgebied.asc', if(europesebever_leefgebied3, europesebever_leefgebied0_cluster, 0))
deleteVariable(europesebever_leefgebied2)
deleteVariable(europesebever_leefgebied0_cluster)
deleteVariable(europesebever_houtig2)
assign(europesebever_ecodistrict, ge('ecoregio.asc', 1))
assign(europesebever_leefgebied_ecodistrict1, and(europesebever_leefgebied3, europesebever_ecodistrict))
deleteVariable(europesebever_leefgebied3)
deleteVariable(europesebever_ecodistrict)
assign(europesebever_leefgebied_ecodistrict1_cluster, FuzzyClusterIdU(europesebever_leefgebied_ecodistrict1,
50, meter))
deleteVariable(europesebever_leefgebied_ecodistrict1)
assign(europesebever_leefgebied_ecodistrict, cFilterGeArea(europesebever_leefgebied_ecodistrict1_cluster, 50,
hectare))
write('europesebever_leefgebied_ecodistrict.asc', if(europesebever_leefgebied_ecodistrict,
europesebever_leefgebied_ecodistrict1_cluster, 0))
deleteVariable(europesebever_leefgebied_ecodistrict1_cluster)

```



Europese bever (foto Rollin Verlinde – Vilda)

Europese hamster (*Cricetus cricetus*) – Van Den Berge et al. (2015)

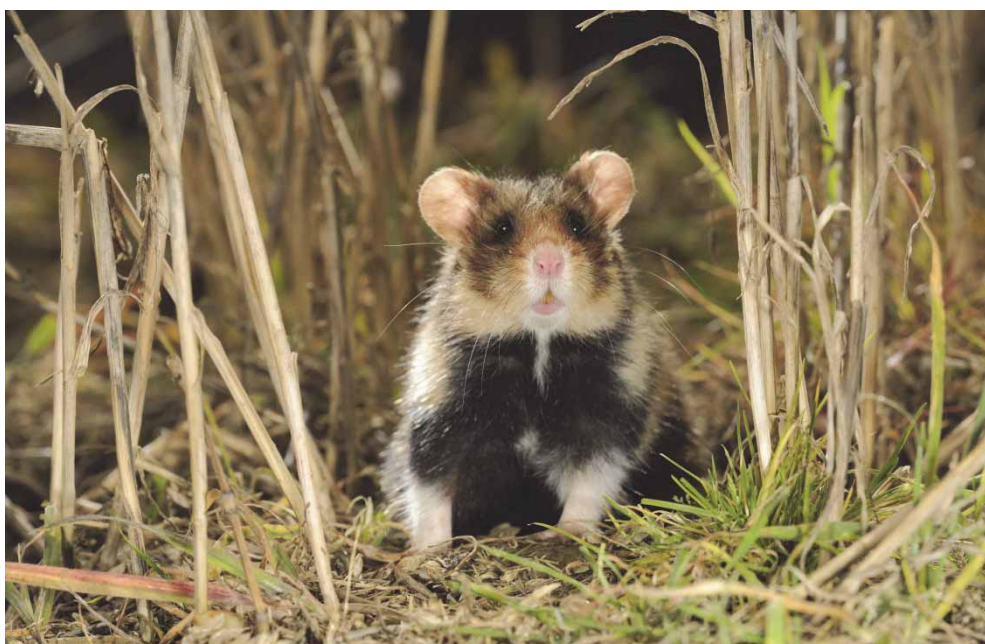


Potentiële leefgebiedenkaart voor Europese hamster op basis van het onderstaande gdx-script.

Gdx-script

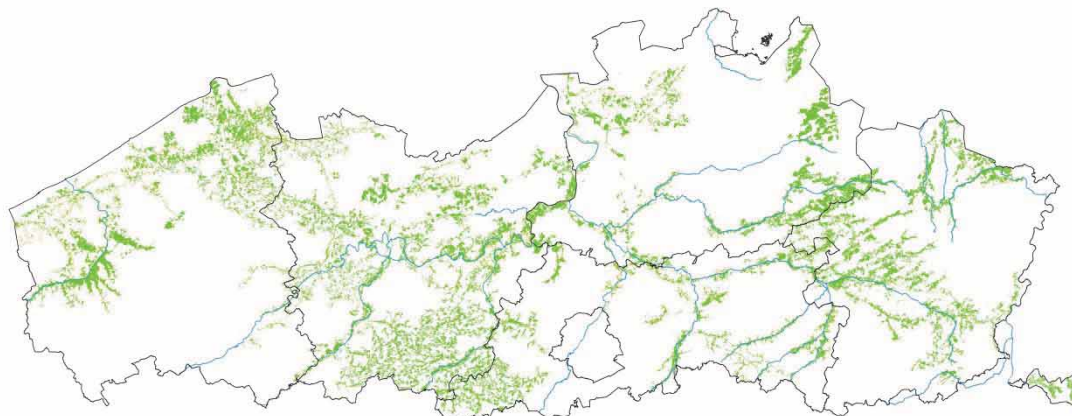
vlaanderen_20m.asc

```
// baken de voortplantingsgebieden af met behulp van de gewassenkaart
assign(europesehamster_graan, OrEq('epr2013.asc', 6, 49, 38, 36, 63, 100, 219, 21, 19, 225, 241, 74, 174, 45))
assign(europesehamster_nietoverstroombaar, Eq('OVSTRGEV2014_20m.asc', 0))
assign(europesehamster_bodem, Eq('bodem_textuur.asc', 6))
assign(europesehamster_leefgebied1, and(europesehamster_graan, europesehamster_bodem))
deleteVariable(europesehamster_graan)
deleteVariable(europesehamster_nietoverstroombaar)
deleteVariable(europesehamster_bodem)
// als pixels minder dan 50 meter van elkaar liggen behoren ze tot dezelfde cluster
assign(europesehamster_leefgebied1_cluster, fuzzyClusterIdU(europesehamster_leefgebied1, 50, meter))
deleteVariable(europesehamster_leefgebied1)
// oppervlakte moet minstens 125 ha groot zijn
assign(europesehamster_leefgebied2, cFilterGeArea(europesehamster_leefgebied1_cluster, 125, hectare))
write('europesehamster_leefgebied.asc', if(europesehamster_leefgebied2, europesehamster_leefgebied1_cluster,
0))
```



Europese hamster (foto Rollin Verlinde – Vilda)

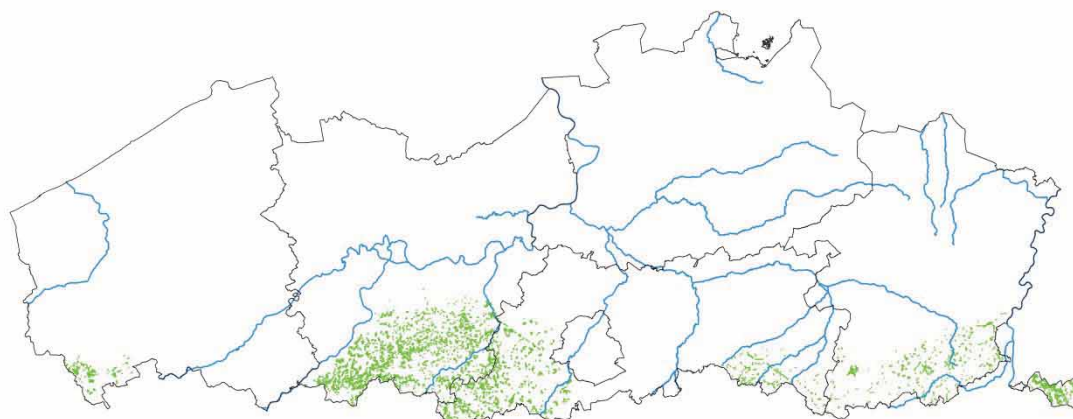
Europese otter (*Lutra lutra*) – Van Den Berge & Gouwy (2015)



Potentiële leefgebiedenkaart voor Europese otter op basis van het onderstaande gdx-script.
Groen = moerasbiotoop, oranje = waterlopen

Gdx-script

```
vlaanderen_20m_nieuw.asc
// baken waterlopen af
assign(europeseotter_waterlopen1, Ge('rastert_huetzon1.asc', 1))
assign(europeseotter_maas, OrEq('maas.asc', 1))
assign(europeseotter_waterlopen, or(europeseotter_waterlopen1, europeseotter_maas))
// write('europeseotter_waterlopen.asc', europeseotter_waterlopen)
deleteVariable(europeseotter_waterlopen1)
deleteVariable(europeseotter_maas)
// baken moerassen af
assign(europeseotter_moeras1, bwk(da%, ds%, hc%, hf%, hj%, hm%, hme%, hmm%, hpr%, hu%, ka%, kb%, kh%,
    khw%, kn%, kp%, ku%, mc%, md%, mk%, mm%, mp%, mr%, ms%, mz%, ru%, sf%, sg%, sm%, sp%, t%, va%, vc%,
    vf%, vm%, vn%, vo%, vt%))
// clustering: indien geschikte cel binnen 100 meter, dan hoort die cel bij het moerasgebied
assign(europeseotter_moeras1_cluster, fuzzyClusterIdU(europeseotter_moeras1, 100, meter))
// write('europeseotter_moeras1_cluster.asc', europeseotter_moeras1_cluster)
deleteVariable(europeseotter_moeras1)
// moerasgebied moeten minimaal 100 km² groot zijn = 1000 ha
assign(europeseotter_moeras2, cFilterGeArea(europeseotter_moeras1_cluster, 1000, hectare))
// write('europeseotter_moeras100km.asc', europeseotter_moeras2)
// moeras- en waterlopen mogen niet verder dan 500 meter van elkaar liggen
assign(europeseotter_leefgebied3, and(europeseotter_moeras2, cellDistanceLe(europeseotter_waterlopen, 500,
    meter)))
write('europeseotter_moeras.asc', IsNonZero(europeseotter_leefgebied3))
assign(europeseotter_leefgebied4, and(europeseotter_waterlopen, cellDistanceLe(europeseotter_moeras2, 500,
    meter)))
write('europeseotter_waterlopen.asc', IsNonZero(europeseotter_leefgebied4))
assign(europeseotter_leefgebied, or(europeseotter_leefgebied3, europeseotter_leefgebied4))
//write('europeseotter_leefgebied.asc', IsNonZero(europeseotter_leefgebied))
deleteVariable(europeseotter_leefgebied3)
deleteVariable(europeseotter_leefgebied4)
deleteVariable(europeseotter_moeras1_cluster)
```



Potentiële leefgebiedenkaart voor Hazelmuis op basis van het onderstaande gdx-script.

Gdx-script

```
vlaanderen_20m.asc
// baken de voortplantingsgebieden af
assign(hazelmuis_voortplanting, bwk(cp%, fa%, fe%, fk%, fl%, fm%, fs%, kh(q%, kh(s%, qa%, qb%, qe%, qk%, ql%,
  qs%, ru%, sf%, sm%, sz%, va%, vc%, vf%, vm%, vn%, vo%, vt%, 9110%, 9120%, 9130%, 9150%, 9160%, 91E0%,
  rbbm%))
// clustering: indien geschikte cel binnen 50 meter, dan hoort die cel bij het voortplantingscluster
assign(hazelmuis_voortplanting_cluster, fuzzyClusterIdU(hazelmuis_voortplanting, 100, meter))
deleteVariable(hazelmuis_voortplanting)
// voortplantingsclusters moeten minimaal 50ha groot zijn
assign(hazelmuis_voortplanting1, cFilterGeArea(hazelmuis_voortplanting_cluster, 50, hectare))
deleteVariable(hazelmuis_voortplanting_cluster)
// zoek open gebieden die aan voortplantingsclusters grenzen
assign(hazelmuis_open, bwk(cdb%, cg%, ha%, hc%, hf%, hj%, hp%, hp+%, hpr%, hr%, hu%, 2310%, 4030%, 2330%,
  6120%, 6230%, 6410%, 6430%, 6510%, rbbhc%, rbbhf%))
// clustering: indien geschikte cel binnen 100 meter, dan hoort die cel bij het open cluster
assign(hazelmuis_open_cluster, fuzzyClusterIdU(hazelmuis_open, 100, meter))
deleteVariable(hazelmuis_open)
// open clusters moeten minimaal 5ha groot zijn
assign(hazelmuis_open1, cFilterGeArea(hazelmuis_open_cluster, 5, hectare))
deleteVariable(hazelmuis_open_cluster)
// zoek bosranden tussen voortplanting en open gebieden
assign(edge1, edgeDirected(hazelmuis_voortplanting1, hazelmuis_open1, 0,0,1,1,0,0,0))
assign(edge2, edgeDirected(hazelmuis_open1, hazelmuis_voortplanting1, 1,0,0,0,0,0,1,1))
assign(hazelmuis_zuidelijkebosrand, or(edge1, edge2))
deleteVariable(hazelmuis_open1)
// lengte van bosrand tussen voortplantingsclusters en boscluster moet minstens 100 meter zijn
assign(hazelmuis_zuidelijkebosrand_cluster, fuzzyClusterIdU(hazelmuis_zuidelijkebosrand, 100, meter))
assign(hazelmuis_zuidelijkebosrand1, cFilterGeArea(hazelmuis_zuidelijkebosrand_cluster, 0.2, hectare))
deleteVariable(edge1)
deleteVariable(edge2)
// baken lineaire voortplantingsgebieden af
assign(hazelmuis_voortplanting_lineair, bwk(kt(sz%, kt, k(ku%, kh(vc%))
// clustering: indien geschikte cel binnen 30 meter, dan hoort die cel bij het lineaire voortplantingscluster
assign(hazelmuis_voortplanting_lineair_cluster, fuzzyClusterIdU(hazelmuis_voortplanting_lineair, 100, meter))
deleteVariable(hazelmuis_voortplanting_lineair)
// lineaire voortplantingsclusters moeten minimaal 100m lang zijn, d.i. 0.2ha groot zijn
assign(hazelmuis_leefgebied_lineair1, cFilterGeArea(hazelmuis_voortplanting_lineair_cluster, 0.2, hectare))
deleteVariable(hazelmuis_voortplanting_lineair_cluster)
// alle leefgebied = bosranden + lineaire landschapselementen
assign(hazelmuis_leefgebied1, or(hazelmuis_zuidelijkebosrand1, hazelmuis_leefgebied_lineair1))
```

```

assign(hazelmuis_leefgebied1_cluster, fuzzyClusterIdU(hazelmuis_leefgebied1, 100, meter))
deleteVariable(hazelmuis_leefgebied1)
// leefgebied moet minstens 0.3 ha groot zijn
assign(hazelmuis_leefgebied, cFilterGeArea(hazelmuis_leefgebied1_cluster, 0.3, hectare))
// externe kaarten bevragen
// baken ecoregio van krijtgebieden af
assign(hazelmuis_ecodistrict, OrEq('ecoregio.asc', 24, 25, 32, 33, 34, 35))
assign(hazelmuis_leefgebied_ecodistrict, and(hazelmuis_leefgebied, hazelmuis_ecodistrict))
write('hazelmuis_leefgebied_ecodistrict.asc', if(hazelmuis_leefgebied_ecodistrict, hazelmuis_leefgebied1_cluster,
0))
deleteVariable(hazelmuis_leefgebied)
deleteVariable(hazelmuis_leefgebied1_cluster)

```



Hazelmuis (foto Rollin Verlinde – Vilda)