INBO.R.2014.1909698

Wetenschappelijke instelling
van de Vlaamse overheid

Ontwikkelen methodiek steekproefopzet voor controle bosexploitatie

Thierry Onkelinx, Hans Van Calster, Toon Westra en Paul Quataert

inbo
Instituut voor
Natuur- en Bosonderzoek

Auteurs:
Thierry Onkelinx, Hans Van Calster, Toon Westra en Paul Quataert
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 Anderlecht
Kliniekstraat 25, 1070 Anderlecht
www.inbo.be

e-mail:
Thierry.Onkelinx@inbo.be

Wijze van citeren:
Onkelinx, T.; Van Calster, H.; Westra, T. en Quataert, P. (2014). Ontwikkelen methodiek steekproefopzet voor controle bosexploitatie. Rapporten van het Instituut voor Natuur- en Bosonderzoek 2014 (INBO.R.2014.1909698). Instituut voor Natuur- en Bosonderzoek, Brussel.

D/2014/3241/142
INBO.R.2014.1909698
ISSN: 1782-9054

Verantwoordelijke uitgever:
Jurgen Tack

Druk:
Managementondersteunende Diensten van de Vlaamse overheid

Foto cover:
Gestapeld brandhout. © Thierry Onkelinx

Deze studie werd uitgevoerd in opdracht van Agentschap voor Natuur en Bos

Ontwikkelen methodiek steekproefopzet voor controle bosexploitatie

Thierry Onkelinx, Hans Van Calster, Toon Westra en Paul Quataert

INBO.R.2014.1909698
INVERDE-2012-002-D

Inhoudsopgave

1 Inleiding

1.1 Visie op de opdracht

De opdracht vraagt om het ontwikkelen van een methodiek die toelaat met een gekende statistische zekerheid uitspraken te doen over de kwantiteit van de gevelde bomen en een eventuele meerkap. Het gaat dus om een controle op de uitvoering van houtexploitanten, waarbij men wil nagaan of enkel de aangeduide bomen geëxploiteerd zijn. Een belangrijk aandachtspunt is dat de controles eenvoudig en snel organiseerbaar en uitvoerbaar zijn.
Uitgaande van ervaringen met de 'Tweede regionale bosinventarisatie', die sinds 2009 lopend is, werd door ANB al gestart met een controle van de exploitatie die gebaseerd is op de veldprotocols die gebruikt worden voor de bosinventarisatie. Deze methode heeft het voordeel dat individuele bomen kunnen gelokaliseerd worden en, bij herhaalde metingen, met hoge betrouwbaarheid terug gevonden worden. Op het eerste zicht is het dus een geschikte methode om de controle van exploitatie mee uit te voeren. Een nadeel is echter dat het een erg arbeidsintensieve methode is. Bovendien verwachten we dat deze methode een vrij forse steekproef vereist. Daarom willen we een voorstel doen dat eenvoudiger en sneller kan worden uitgevoerd als screening van de exploitatie.

2 Uitspraken op basis van inventarisatie van individuele bomen

2.1 Inleiding

De methodologie van de bosinventarisatie laat toe om individuele bomen steekproefsgewijs te inventariseren. Een meting na het schalmen en voor de exploitatie laat toe om te weten hoeveel bomen in het proefvlak staan, welke daarvan geschalmd zijn en dus ook welke proportie geschalmd is. Een tweede meting na de exploitatie laat toe om de proportie gevelde bomen te schatten. Vervolgens kunnen we een statistische test uitvoeren waarbij we deze proporties met elkaar vergelijken. Hierbij is de nulhypothese dat beide proporties gelijk zijn: er werden evenveel bomen geveld als er geschalmd werden.
Op basis van deze informatie maken we een schatting van hoeveel bomen per lot we moeten meten om een bepaald verschil in proporties aan te tonen. Hierbij spelen volgende factoren een belangrijke rol:

1. De proportie geschalmde bomen
2. De grootte van het verschil tussen de proportie geschalmde bomen en de proportie gekapte bomen
3. De proportie van correct gekapte bomen (geschalmd en gekapt)
4. De kans om ten onrechte te stellen dat er een verschil is tussen beide proporties (de Type I-fout)
5. De kans om een bepaald verschil aan te tonen (de power)

We geven het nodige aantal bomen in de steekproef grafisch weer voor een aantal combinaties van bovenstaande variabelen. De relevante combinaties werden bepaald in overleg met de opdrachtgever. Onze ervaring leert dat het voor opdrachtgevers niet altijd eenvoudig is om te bepalen vanaf welk verschil ze een signaal wensen te krijgen. Daarom raden we aan om hiervoor een aantal verschillende waarden uit te proberen, gaande van een sterk verschil dat ook zonder statistiek te zien is tot een klein verschil dat eigenlijk niet meer relevant is. Dit geeft de opdrachtgever meer inzicht in de verhouding tussen de kosten voor de controle en het verlies aan kapitaal door overmatig kappen.
Uit ervaring weten we dat dergelijke testen vereisen dat een vrij groot aantal bomen gemeten moeten worden. Met als gevolg een groot aantal proefvlakken en/of grote proefvlakken, zeker in het geval van loten met een lager stamtal per ha. We verwachten daarom dat bij deze methode de kosten voor de controle niet in verhouding staan tot de gederfde opbrengsten.
Merk op dat alle uitspraken steeds gebaseerd zijn op het aantal bomen ongeacht de boomsoort. Het berekenen van het aantal bomen dat we moeten meten om verschillen in grondvlak of volume te bepalen, vereist *a priori* kennis van de diameterverdeling en de boomhoogte van het lot. Tevens zal de uiteindelijke steekproef in dat geval bomen moeten selecteren met een bepaalde diameter en hoogte. Dat is veel complexer en zal bijna tot een volinventaris leiden. Daarom beperken we ons tot een vergelijking gebaseerd op de aantallen. Als een lot uit meerdere boomsoorten / boomtypes[1] bestaat en een uitspraak per soort / type gewenst, dan is de steekproefgrootte per type.

2.2 Definities

- **geschalmd**: een boom die door de houtkoper gekapt mag worden. Hierbij wordt er een fysiek merkteken op de boom aangebracht. Het aantal is n_s.
- **gekapt**: een boom die geveld werd door de houtkoper. Het aantal is n_k.
- **correct gekapt**: een geschalmde boom die geveld werd. Het aantal is n_c.
- **overkap**: niet geschalmde bomen die geveld werden. Het aantal is $n_+ = n_k - n_c$.
- **onderkap**: geschalmde bomen die niet geveld werden. Het aantal is $n_- = n_s - n_c$.
- **foutief**: de combinatie van overkap en onderkap. Het aantal is $n_f = n_+ + n_- = n_s + n_k - n_c$
- **Type-I fout**: ten onrechte een effect aan te tonen. Tenzij anders vermeldt, gebruiken we steeds 10% als aanvaardbare kans op een Type I fout.
- **Type-II fout**: ten onrechte stellen dat er geen effect is.
- **power**: kans om geen Type-II fout te maken (en dus om een effect aan te tonen als het aanwezig). Tenzij anders vermeldt, gebruiken we steeds 90%.

[1] bijv. brandhout vs zaaghout

- **proportie**: het aantal van deze klasse gedeeld door het totaal aantal N.

In de tabel 2.1 geven we een overzicht van de vier mogelijke combinaties tussen de werkelijke toestand (rijen) en de uitspraak op van een steekproef (kolommen). Bij een Type-I fout stellen we ten onrechte dat er een probleem is. Dit kan zich voordoen wanneer er toevallig veel foutief gekapte bomen in het proefvlak bevinden. Het omgekeerde fenomeen is een Type-II fout: we stellen ten onrechte dat er geen probleem is. Bijvoorbeeld omdat er toevallig zeer weinig foutief gekapte bomen in het proefvlak liggen. De kans op een Type-I en een Type-II fout hangen af van de steekproefgrootte. Hoe groter de steekproefgrootte, hoe kleiner deze kansen. Immers, hoe meer bomen we meten, hoe beter de schatting van het werkelijke aandeel foutief gekapte bomen.

	niet vastgesteld	vastgesteld
geen probleem	correct	Type-I fout
probleem	Type-II fout	correct

Tabel 2.1: Mogelijke combinaties van werkelijke toestand en uitspraak op basis van een steekproef.

Op basis van een steekproef met een eerste meting tussen de dunning en de kap en een tweede meting na de kap, kunnen we de 2x2 kruistabel (Tabel 2.2) invullen.

	niet gekapt	gekapt	totaal
niet geschalmd	n_o	n_+	$n_o + n_+$
geschalmd	n_-	n_c	$n_s = n_- + n_c$
totaal	$n_o + n_-$	$n_k = n_+ + n_c$	$N = n_o + n_+ + n_c + n_-$

Tabel 2.2: De verschillende mogelijke combinaties in status voor geschalmd en gekapt met de bijhorende aantallen.

2.3 Steekproefgrootte voor verschil tussen overkap en onderkap op basis van een gepaarde steekproef

De methodologie van de bosinventarisatie laat toe om de bomen die voor de exploitatie gepositioneerd werden, na de exploitatie terug op te sporen en zo vast te stellen of ze al dan niet gekapt werden. Dit laat ons toe om de 2x2 kruistabel (Tabel 2.2) in te vullen op basis van een steekproef. Vervolgens kunnen hierop een McNemar test (Agresti 2002) toepassen. De nulhypothese van deze test is dat de proporties van de geschalmde en de gekapte bomen gelijk zijn. Aangezien het gepaarde waarnemingen zijn, kunnen we deze nulhypothese vereenvoudigen tot het aantal bomen in overkap is gelijk aan het aantal bomen in onderkap. Deze test heeft als test-statistiek z_0 en z_0^2 volgt een χ^2 verdeling met één vrijheidsgraad. In tabel 2.3 geven we de grenswaarde voor een aantal Type I fouten. Van zodra z_0^2 kleiner is dan de grenswaarde, dan kunnen we stellen dat er een verschil is tussen het aantal gekapt en het aantal geschalmde bomen. De Type I fout is de kans dat we deze uitspraak ten onrechte doen.

$$z_0 = \frac{n_+ - n_-}{\sqrt{n_+ + n_-}}$$

Type I fout	Grenswaarde
10.0%	0.0157908
5.0%	0.0039321

1.0%	0.0001571
0.1%	0.0000016

Tabel 2.3: Type I fout en bijhorende grenswaarden voor een McNemar test.

De McNemar test kan uitgevoerd worden in R (R Core Team 2013) met het exact2x2 package (Fay 2010). Hieronder geven we een aantal voorbeelden met R code en bijhorende output. Een regel van de output start telkens met ##. De output van de McNemar-test geeft volgende waarden:

- **b**: het aantal bomen in overkap
- **c**: het aantal bomen in onderkap
- **p-value**: de kans dat het verschil tussen overkap en onderkap louter toeval is. Indien deze waarde kleiner is dan de vooropgestelde Type-I fout spreken we van een statistisch significant verschil.
- **confidence interval**: het betrouwbaarheidsinterval van de verhouding b/c. De dekkingsgraad van het interval is 100% - Type I-fout. Wanneer de test een niet-significante p-waarde geeft, zal de nul-hypothese ($b/c = 1$) binnen dit interval liggen. Bij een significante p-waarde zal deze er buiten liggen.
- **sample estimates**: de waargenomen verhouding b/c

In de voorbeelden meten we telkens 135 bomen waarvan 5 geschalmde bomen niet gekapt werden. Verder zijn er respectievelijk 10 en 20 niet geschalmde bomen gekapt. De eerste test besluit dat er geen statistische significant verschil is, bij de tweede test is er wel een statistisch significant verschil. Merk op dat de gewenste alfa de *aanvaardbare* kans op een Type-I fout is, deze hebben we *a priori* vastgelegd en geldt voor alle loten.. De p-waarde is de *werkelijke* kans op een Type-I fout. Deze geldt enkel voor het lot in kwestie en is gebaseerd op de resultaten van de steekproef. Indien de p-waarde kleiner is dan de gewenste alfa, hebben we een statistisch significant verschil.

```
library(exact2x2)
gewenste.Type.I <- 0.1
#eerste test
tabel <- rbind(
  "Niet geschalmd" = c(100, 10),
  "Geschalmd" = c(5, 20)
)
colnames(tabel) <- c("Niet gekapt", "Gekapt")
tabel
```

```
##                Niet gekapt Gekapt
## Niet geschalmd         100     10
## Geschalmd                5     20
```

```
mcnemar.exact(tabel, conf.level = 1 - gewenste.Type.I)
```

```
##
##  Exact McNemar test (with central confidence intervals)
##
## data:  tabel
## b = 10, c = 5, p-value = 0.3018
## alternative hypothesis: true odds ratio is not equal to 1
## 90 percent confidence interval:
##  0.7318 6.0590
## sample estimates:
## odds ratio
##          2
```

```
#p-waarde > gewenste Type I fout ==> geen significant verschil
```

```
#tweede test
tabel <- rbind(
  "Niet geschalmd" = c(90, 20),
  "Geschalmd" = c(5, 20)
)
colnames(tabel) <- c("Niet gekapt", "Gekapt")
tabel
```

```
##                Niet gekapt Gekapt
## Niet geschalmd          90     20
## Geschalmd                5     20
```

```
mcnemar.exact(tabel, conf.level = 1 - gewenste.Type.I)
```

```
##
##  Exact McNemar test (with central confidence intervals)
##
## data:  tabel
## b = 20, c = 5, p-value = 0.004077
## alternative hypothesis: true odds ratio is not equal to 1
## 90 percent confidence interval:
##   1.664 11.152
## sample estimates:
## odds ratio
##          4
```

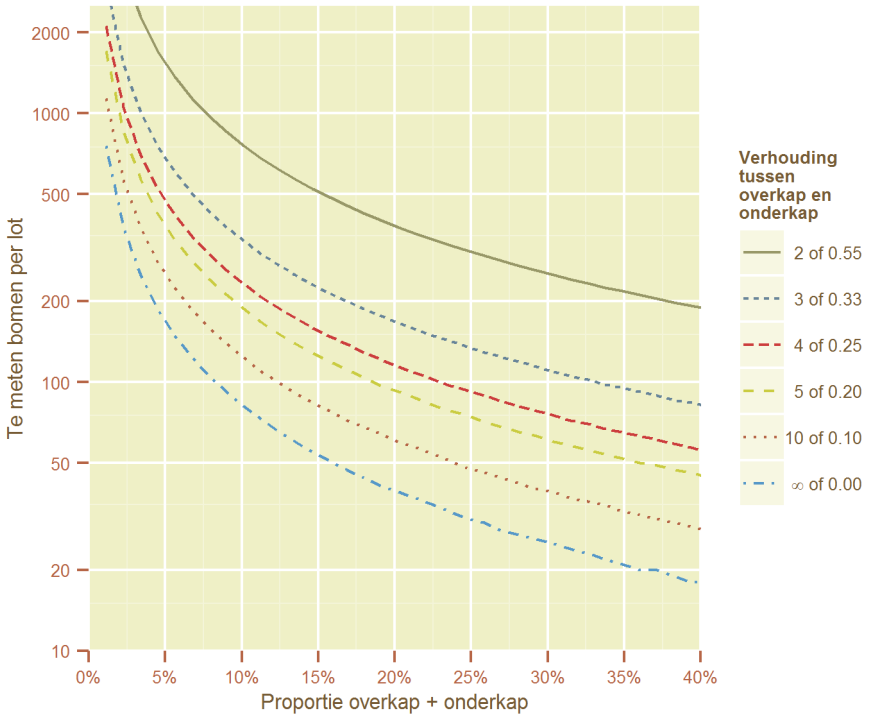
```
#p-waarde < gewenste Type I fout ==> significant verschil
```

Bij een steekproefgrootteberekening maken we *a priori* een inschatting van de steekproefgrootte die nodig is om een vooropgestelde kans (de power) te hebben een bepaald effect te waarnemen bij een gewenste Type I fout. In dit geval hangt de steekproefgrootte dus van 4 factoren af:

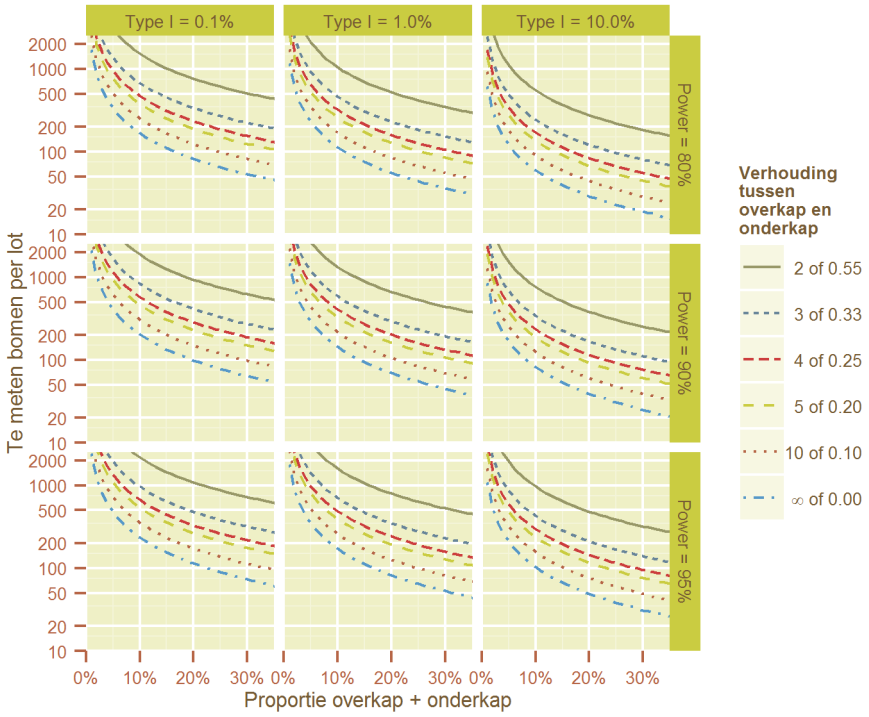
1. de verwachte proportie overkap
2. de verwachte proportie onderkap
3. de gewenste Type I fout
4. de gewenste Power

In figuur 2.1 geven we aan hoe de steekproefgrootte wijzigt in functie van de proportie overkap en de proportie onderkap bij een gewenste Type I van 10% en een gewenste power van 90%. Merk op dat de som van de proporties overkap en onderkap nooit meer dan 100% kan bedragen. De lijnen verbinden punten met een zelfde verhouding tussen overkap en onderkap. Indien er geen onderkap is, is de verhouding ∞. Hoe sterker de verhouding verschilt van 1, hoe kleiner de steekproefgrootte. Als we de verhouding constant houden, neemt de steekproefgrootte tevens af naarmate de som van over- en onderkap toeneemt. Merk op dat bij een constant verhouding, het absolute verschil tussen over- en onderkap toeneemt naarmate hun som toeneemt.
Veronderstel dat we een signaal wensen te krijgen als 10% van de bomen foutief zijn (som overkap en onderkap) en de overkap meer dan 10 keer groter is dan de onderkap. Uit de grafiek kunnen we aflezen dat we dan per lot tussen de 100 en 150 bomen moeten meten.
Om voeling te krijgen met de impact van de Type I fout en de power op de steekproefgrootte, tonen we in figuur 2.2 de steekproefgrootte voor 3 verschillende Type I fouten en 3 verschillende power waarden. Een lage Type I fout (of een hogere power) geeft meer zekerheid over de uitspraken. Meer zekerheid vereist een grotere steekproef. Een Type I fout van 10% en een power van 90% (=10% kans om een probleem niet te zien) zijn een redelijk afweging van beide fouten.
De nodige steekproefgrootte voor specifieke combinaties is vrij eenvoudig te bereken in `R` met het package `TrialSize` (E. Zhang et al. 2013). Hieronder geven we twee concrete voorbeelden van de `R` code met bijhorende output. De bekomen steekproefgrootte dient naar boven te worden afgerond.

```
library(TrialSize)
gewenste.Type.I <- 0.1 #gewenste Type I fout
gewenste.power <- 0.9 #gewenste power
```



Figuur 2.1: Aantal bomen nodig om een bepaald verschil of verhouding tussen de proportie overkap en de proportie onderkap aan te tonen met 90% power en 10% Type I fout.



Figuur 2.2: Aantal bomen nodig om een bepaalde verhouding tussen de proportie overkap en de proportie onderkap aan te tonen voor verschillende waarde van Type I fout en power.

```
#eerste voorbeeld
proportie.overkap <- 1/11
proportie.onderkap <- 1/110
#som overkap + onderkap = 10%
proportie.overkap + proportie.onderkap
```

```
## [1] 0.1
```

```
#verhouding overkap / onderkap = 10
proportie.overkap / proportie.onderkap
```

```
## [1] 10
```

```
n <- McNemar.Test(
    alpha = gewenste.Type.I,
    beta = 1 - gewenste.power,
    psai = proportie.overkap / proportie.onderkap,
    paid = proportie.overkap + proportie.onderkap
)
n
```

```
## [1] 124.1
```

```
#tweede voorbeeld
proportie.overkap <- 0.05
proportie.onderkap <- 0.01
#som overkap + onderkap = 6%
proportie.overkap + proportie.onderkap
```

```
## [1] 0.06
```

```
#verhouding overkap / onderkap = 10
proportie.overkap / proportie.onderkap
```

```
## [1] 5
```

```
n <- McNemar.Test(
    alpha = gewenste.Type.I,
    beta = 1 - gewenste.power,
    psai = proportie.overkap / proportie.onderkap,
    paid = proportie.overkap + proportie.onderkap
)
n
```

```
## [1] 317.4
```

2.4 Steekproefgrootte voor een specifieke proportie

Naast het testen voor een verschil tussen over- en onderkap, kunnen we tevens testen of een bepaalde proportie verschilt van een vooropgestelde streefwaarde. Deze streefwaarde is de proportie onder 'perfecte' omstandigheden. Daarnaast zullen we tevens moeten definiëren welke afwijking van deze streefwaarde te groot is om nog te kunnen toleren.
Idealiter doet zich noch overkap, noch onderkap voor. En dus zouden we 0% als streefwaarde kunnen gebruiken. Vanuit statistisch oogpunt heeft dat als grootste nadeel dat de test steeds een zeer significant zal zijn van zodra de steekproef één enkele foutieve boom bevat. En dat is vermoedelijk niet de bedoeling. Daarom nemen we best een streefwaarde die (iets) groter is dan 0%.
We kunnen dan testen of de proportie foutieve bomen[2] afwijkt van deze streefwaarde. De statistische analyse is een binomiale test, een van de standaard testen in R. Hieronder geven we een aantal uitgewerkte voorbeelden. Hierbij zijn drie waarden van belang:

1. Het aantal 'successen': het aantal gemeten bomen die tot de proportie horen. Bijvoorbeeld het aantal foutieve bomen in de steekproef.

[2] overkap + onderkap

2. Het aantal pogingen: het totaal aantal bomen in de steekproef.
3. De vooropgestelde proportie (streefwaarde) waarmee we willen vergelijken.
4. De gewenste dekkingsgraad van het betrouwbaarheidsinterval (typisch 100% - Type-I fout)

De test geeft volgende resultaten:

- **number of successes**: het aantal 'successen', bijvoorbeeld het aantal foutieve bomen in de steekproef van een bepaald lot.
- **number of trials**: het aantal pogingen: het aantal bomen in de steekproef van datzelfde lot.
- **p-value**: de p-waarde. De kans dat de waargenomen proportie (aandeel foutieve bomen in deze steekproef) door louter toeval afwijkt van de vooropgestelde proportie (streefwaarde). Wanneer deze waarde kleiner is dan de vooropgestelde Type I fout spreken we van een statistisch significant verschil.
- **confidence interval**: het betrouwbaarheidsinterval van de waargenomen proportie. Het vermelde percentage is de dekking van het interval. Dit interval geeft dus een idee van het bereik waarbinnen het werkelijke proportie foutieve bomen zich bevindt. De werkelijke proportie is de verhouding van het aantal foutieve bomen in het volledige lot gedeeld door het totaal aantal bomen in het volledige lot. Bijgevolg in de veronderstelling van een volinventaris.
- **probability of success**: de schatting van de proportie foutieve bomen op basis van de steekproef.

Bij de eerste test is 1 van de 200 gemeten bomen (0.5%) foutief en stellen we de aanvaardbare proportie op 0.1%. In dat geval is de p-waarde groter dan 0.1 (de vooropgestelde Type I fout). Dus besluiten we dat er geen statistisch significante afwijking is. Bij de tweede test verhogen we het aantal foutieve bomen naar 3, terwijl de andere drie waarde constant blijven. Nu kunnen we stellen dat de waargenomen proportie (1.5%) statistisch significant verschil van de vooropgestelde proportie (0.1%). Bij de derde test passen we de vooropgestelde proportie aan naar 0.6%. Deze proportie verschil niet langer statistisch significant van de waargenomen 1.5%.

```
#eerste test
gewenste.Type.I <- 0.1 # de gewenste Type I fout 0.1 = 10%
foutief <- 1 # aantal foutieve bomen in de steekproef van een lot
totaal <- 200 # aantal bomen in de steekproef van een lot
streefwaarde <- 0.001 # 0.001 = 0.1%
binom.test(x = foutief, n = totaal, p = streefwaarde, conf.level = 1 - gewenste.Type.I)
```

```
##
##  Exact binomial test
##
## data:  foutief and totaal
## number of successes = 1, number of trials = 200, p-value = 0.1814
## alternative hypothesis: true probability of success is not equal to 0.001
## 90 percent confidence interval:
##  0.0002564 0.0234985
## sample estimates:
## probability of success
##                  0.005
```

```
# p-value > gewenste.Type.I: niet significant
# probability of success: geschatte proportie = 5%
# confidence interval:
# werkelijke proportie ligt tussen 0.026% en 2.35%

#tweede test
foutief <- 3
totaal <- 200
streefwaarde <- 0.001
binom.test(x = foutief, n = totaal, p = streefwaarde, conf.level = 1 - gewenste.Type.I)
```

```
##
##  Exact binomial test
```

```
##
## data:  foutief and totaal
## number of successes = 3, number of trials = 200, p-value =
## 0.001134
## alternative hypothesis: true probability of success is not equal to 0.001
## 90 percent confidence interval:
##  0.004101 0.038310
## sample estimates:
## probability of success
##                  0.015
```

```
#p-value < gewenste.Type.I: significant verschil
#betrouwbaarheidsinterval (0.4%; 3.8%)

#derde test
foutief <- 3
totaal <- 200
streefwaarde <- 0.006
binom.test(x = foutief, n = totaal, p = streefwaarde, conf.level = 1 - gewenste.Type.I)
```

```
##
##  Exact binomial test
##
## data:  foutief and totaal
## number of successes = 3, number of trials = 200, p-value = 0.12
## alternative hypothesis: true probability of success is not equal to 0.006
## 90 percent confidence interval:
##  0.004101 0.038310
## sample estimates:
## probability of success
##                  0.015
```

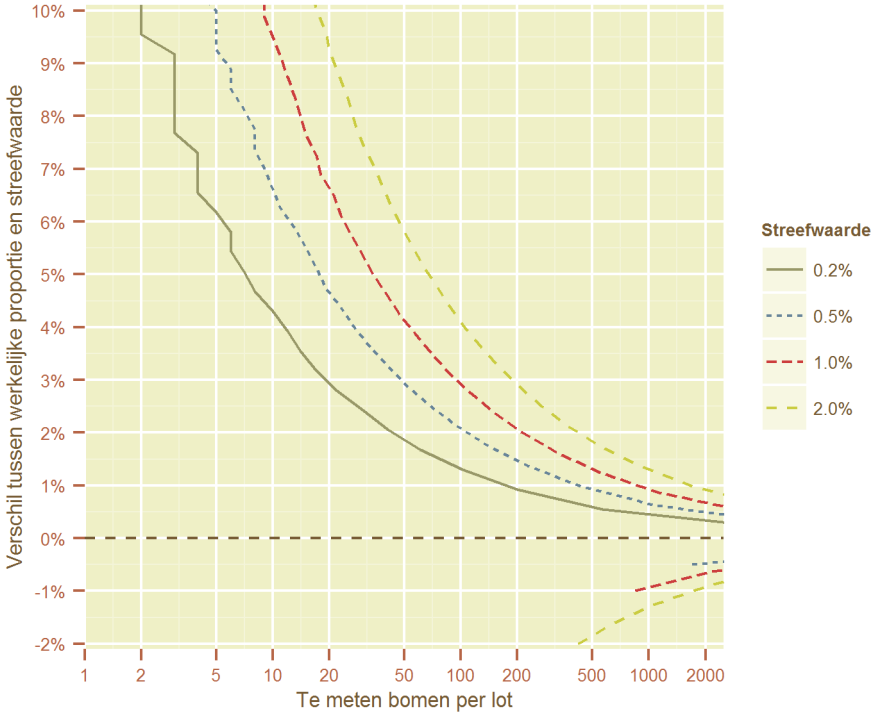
```
# p-value > gewenste.Type.I: niet significant
#betrouwbaarheidsinterval (0.4%; 3.8%)
```

Om de steekproefgrootte te berekenen hebben we 4 parameters nodig:

1. De streefwaarde waarmee we willen vergelijken.
2. De werkelijke proportie waarbij we een signaal moeten krijgen dat er iets aan de hand is.
3. De gewenste Type-I fout
4. De gewenste power

In figuur 2.3 geven we de steekproefgrootte in functie van het verschil tussen de werkelijke proportie en de streefwaarde en dit voor verschillende streefwaardes en bij een gewenste Type I van 10% en een gewenste power van 90%. Opnieuw stellen we vast dat de steekproefgrootte sterk afhangt van het te detecteren verschil. Hoe kleinere verschillen we wensen op te sporen, hoe groter de steekproef moet zijn. De verwachte of aanvaardbare proportie is tevens een belangrijke parameter. De steekproefgrootte zal dalen naarmate de aanvaardbare proportie extremer wordt. Met extremer bedoelen we verder van 50%, dus meer richting 0% of 100%.
Indien de steekproefgrootte voor een specifieke combinatie gewenst is, kan deze eenvoudig in `R` worden berekend met het `TrialSize` package. De steekproefgrootte dient naar boven te worden afgerond.

```
library(TrialSize)
gewenste.Type.I <- 0.1 #Type I fout
gewenste.power <- 0.9
aanvaardbare.proportie <- 0.01
te.detecteren.verschil <- 0.04
n <- OneSampleProportion.Equality(
  alpha = gewenste.Type.I,
  beta = 1 - gewenste.power,
```



Figuur 2.3: Aantal bomen nodig om een verschil tussen de waargenomen proportie en de streefwaarde foutieve bomen aan te tonen met 90% power en 10% Type I fout. De proporties slaan op de verhouding van het aantal foutieve bomen t.o.v. het totaal aantal bomen in de steekproef.

```
  p = aanvaardbare.proportie,
  delta = te.detecteren.verschil
)
n
```

```
## [1] 52.99
```

2.5 Conclusie

Op basis van een steekproef van individuele bomen, moeten we al snel een paar 100 bomen per lot opvolgen. Een gemiddeld proefvlak van de bosinventarisatie bevat 20 à 25 bomen. 100 bomen komt dus *grosso modo* overeen met 4 à 5 proefvlakken. Dat is ongeveer het werk dat een ploeg van 2 mensen per dag kan opmeten. Stelt dat ANB kiest om 200 bomen per lot op te volgen. Aan 25 bomen per plot, komt dat neer op 8 plots die telkens voor en na de exploitatie gemeten worden. Dat vereist een inzet van ongeveer 8 mensdagen. Dat lijkt ons een behoorlijk forse inzet die mogelijk niet in verhouding staat tot de waarde van het hout.

3 Uitspraken op basis van inschatting van het grondvlak

3.1 Inleiding

De basisidee is dat we een model opstellen waarbij grondvlak na de exploitatie kan voorspeld worden door het grondvlak voor de exploitatie min het geschalmde grondvlak. In een ideale wereld is dit een gelijkheid, maar door meetfouten en mogelijke over- of onderexploitatie zal dit niet zo zijn. In een mixed model kan hier rekening mee gehouden worden door extra factoren op te nemen zoals effect van het lot en bestand en een effect voor de houtkoper.
Er bestaan verschillende methoden om grondvlak te meten:

- volinventaris / oppervlakte lot
- steekproefcirkel / oppervlakte cirkel
- plotless sampling (Bitterlich)

Omdat we als randvoorwaarde hebben dat we een snelle en haalbare methode willen om de controle uit te voeren, is de Bitterlich methode te verkiezen. Deze methode is een vorm van plotless sampling, wat betekent dat de vorm en grootte van het proefvlak niet relevant zijn.
De Bitterlich methode houdt in dat een plaatje met een bepaalde opening op een vaste afstand van het oog gehouden wordt. De waarnemer draait rond zijn as op het steekproefpunt en telt het aantal bomen dat breder is dan de opening (Figuur 3.1). Dit aantal vermenigvuldigen we met een factor die afhangt van de breedte van de opening om een schatting van het grondvlak te kennen. De opening wordt meestal zo gekozen dat ze met een eenvoudige factor overeenkomt: K = 1, 2, 3 of 4 m²/ha.boom.

Figuur 3.1: Voorbeeld van de meting van een boom. De boom wordt meegeteld indien een van de twee bovenste openingen gebruikt wordt. De boom wordt niet meegeteld indien de onderste opening gebruikt wordt.

De methode laat toe om zonder metingen van de omtrek van bomen visueel en snel een schatting te maken van het bestandsgrondvlak. Hierdoor kunnen we meer metingen doen. De aanbeveling is 2 à 3 per ha en minstens 4 per lot. Met de Bitterlich methode kunnen we het grondvlak voor en na exploitatie meten. Dit kan tijdens het schalmen of tijdens de controle na de exploitatie door de boswachter gebeuren.
Het geschalmde grondvlak kunnen we afleiden uit de catalogus en oppervlakte van het lot.
Het grondvlak door meerkap wordt beïnvloed door traditie, bestand, lot, houtkoper, . . . Dit grondvlak kan niet gemeten worden, maar zal voorspeld worden door het model (de residuele variatie en de random effecten). De modelmatige aanpak van alle loten samen laat toe om:

- individuele loten met sterke meerkap (of minderkap) te identificeren

- houtkopers te identificeren die systematisch voor meerkap of minderkap zorgen
- de gemiddelde meerkap kan geschat worden

De gegevens van alle domeinen en jaren kunnen gebruikt worden. Deze groeiende dataset zal toelaten om steeds betere schattingen te maken van de meerkap (globaal, per jaar, per koutkoper, ...). Randvoorwaarden zijn dat we unieke id van bestanden, unieke id van loten, unieke id van houtkopers moeten hebben en dat de gegevens centraal beheerd en verwerkt worden.
De modelmatige aanpak die we hier voorstellen zal dus toelaten om de loten te screenen voor potentiële problemen met overkap of onderkap. Het model laat eveneens toe om, op basis van de gegevens van de houtverkoop, de loten op te sporen met een hogere risico voor over- of onderkap. Deze informatie kan vervolgens gebruikt worden om dergelijke loten gerichter en met een 'gouden standaard' techniek (bv volinventarisatie) op te volgen.

3.2 Uitgewerkt concept

Indien we zouden werken met twee volinventarissen, kennen we de exacte grondvlakken van het lot i voor de kap ($G_i^{voor} = lot_i^{voor}$) en na de kap ($G_i^{na} = lot_i^{na}$). In de praktijk zullen we per lot op een aantal punten een schatting van het grondvlak maken met de Bitterlich methode. Aangezien de Bitterlich methode een onvertekende schatting levert kunnen we stellen dat de verwachte waarde overeenkomt met het bestandsgrondvlak: $E[G_{ij}^{voor}] = lot_i^{voor}$ en $E[G_{ij}^{na}] = lot_i^{na}$. G_{ij} staat voor de j Bitterlich meting in lot i. In de veronderstelling dat de exploitant alle geschalmde bomen en enkel de geschalmde bomen gekapt heeft, dat is het grondvlak voor de kap gelijk aan de som van het grondvlak na de kap en het geschalmde grondvlak $E[G_{ij}^{voor}] = E[G_{ij}^{na} + G_i^{schalm}]$. De De eventuele fouten in de houtcatalogus zullen in ieder geval beduidend kleiner zijn dan de meetfouten van de Bitterlich methode. Hierdoor kunnen we het geschalmd grondvlak van de houtcatalogus als een foutloze schatting aanzien. Bijgevolg is $E[G_{ij}^{na} + G_i^{schalm}] = E[G_{ij}^{na}] + G_i^{schalm} = lot_i^{na} + G_i^{schalm} = lot_i^{voor}$.
Een eenvoudig model voor de Bitterlich metingen is $E[G_{ij}] = lot_i^{na} + I_{ij}^{voor} G_i^{schalm}$, waarbij $I_{ij}^{voor} = 1$ als het een meting voor de kapping betreft en $I_{ij}^{voor} = 0$ bij een meting na de kapping. M.a.w. bij een meting na de kapping hangt de verwachte waarde van de meting enkel af van het grondvlak van het lot na de kapping. Bij een meting voor de kapping is de verwachte waarde van de meting het grondvlak van het lot na de kapping vermeerderd met het geschalmd grondvlak.
In de praktijk hebben we eerder te maken met een gekapt grondvlak G_i^{kap}. De vergelijking wordt dan $E[G_{ij}] = lot_i^{na} + I_{ij}^{voor} G_i^{kap}$. Veronderstel dat er een verschil is tussen het gekapt en het geschalmd grondvlak: $G_i^{kap} = G_i^{schalm} + \delta_i$. Dus als $\delta_i > 0$ is het gekapt grondvlak groter dan het geschalmd grondvlak. Het aangepaste model wordt dan $E[G_{ij}] = lot_i^{na} + I_{ij}^{voor} G_i^{schalm} + I_{ij}^{voor} \delta_i$. Het verschil δ_i wordt veroorzaakt door zowel lot specifieke effecten als effecten die spelen over meerdere loten. Voorbeelden van lot specifieke effecten zijn: overkap wegens exploitatieschade, aanleggen (extra) ruimingspistes, ...; onderkap wegens lokaal te moeilijke toegang, vergetelheid, ... Effecten die over meerdere loten kunnen spelen zijn bijvoorbeeld het effect van een koper die meerdere loten koopt en systematisch te veel of te weinig kapt.
Bovenstaande informatie laat toe om alles naar een *linear mixed model* te vertalen. De basis is een dataset met een rij per Bitterlich meting en vijf variabelen:

- `Grondvlak`: het gemeten grondvlak met de Bitterlich methode
- `Geschalmd`: het geschalmd grondvlak indien de meting voor de kap is, anders 0 m²/ha.
- `ExtraKap`: een indicatorvariabele: 1 voor de kap, 0 na de kap
- `Lot`: een unieke code per lot
- `Koper`: een unieke code per koper

De structuur van het *mixed model* ziet er dan als volgt uit: `Grondvlak` is de responsvariabele, `Geschalmd` een offsetvariabele, `ExtraKap` als *fixed effect*, `Lot` als *random effect* met een *random intercept* en een *random slope* volgens `ExtraKap`, `Koper` als *random effect* met een *random slope* volgens `ExtraKap`. Het *fixed effect* van `ExtraKap` geeft een indicatie van een systematisch verschil tussen het gekapt en het geschalmd grondvlak. De *random slope* van `ExtraKap` per `Lot` geeft een indicatie van het verschil tussen het gekapt en het geschalmd grondvlak in het betreffende lot. De *random slope* van `ExtraKap` per `Koper` geeft een indicatie van een systematisch verschil tussen het gekapt en het geschalmd grondvlak van de betreffende koper.

Een verschil tussen gekapt en geschalmd grondvlak kan door verschillende termen in het model gemodelleerd worden. Het effect van een koper zou bijvoorbeeld perfect gemodelleerd kunnen worden met het effect van het lot. Veronderstel dat een bepaalde koper gemiddeld 1 m²/ha te veel kapt. Op dat moment zal het model de afweging moeten maken tussen dat als een effect van die koper te modelleren of in rekening brengen bij alle loten van de koper in kwestie. Om de informatie op een goede manier over de verschillende termen te verdelen, maken mixed models gebruik van een 'penalty' voor de random effecten. Dat houdt in dat hoe sterker (in absolute termen) een random effect, hoe meer 'strafpunten' het krijgt bij de beoordeling van de model fit. Met als gevolg dat sterke effecten enkel behouden blijven als de verbetering van de model fit groter is dan de 'strafpunten' die ze opleveren. Stel dat een bepaalde koper 10 loten gekocht heeft en in elke van deze loten exact 1 m²/ha te weinig kapt. Op dat moment moet het model afwegen tussen a) 1 effect van -1 m²/ha voor de koper en 10 loten met elk een effect van 0 m²/ha en b) 1 effect van 0 m²/ha voor de koper en 10 loten met elk een effect van -1 m²/ha. De model fit zal in beide gevallen identiek zijn, het aantal 'strafpunten' echter niet. Ze bedragen $1 \times |-1| + 10 \times |0| = 1$ strafpunten voor geval (a) en $1 \times |0| + 10 \times |-1| = 10$ strafpunten voor geval (b). In dit voorbeeld zal geval (a) gebruikt worden omdat het dezelfde model fit oplevert met een kleiner aantal strafpunten. De methode zorgt er dus voor dat de informatie aan de meest generieke term toegewezen wordt.

3.3 Uittesten van het concept

3.3.1 Gesimuleerde dataset

Om een idee te krijgen van de werkwijze van het model hebben we een dataset gesimuleerd. De dataset bestaat uit 5 jaar met 200 loten per jaar. We hebben hierbij maximaal gebruik gemaakt van de beschikbare gegevens van de houtverkoop. We gebruiken het aantal bomen per lot, de omtrekverdeling en het type koper. De types loten werd arbitrair bepaald door de gegevens te clusteren op basis van totaal grondvlak en gemiddeld grondvlak per boom. Op die manier hebben we een onderscheid tussen kleine brandhoutloten, loten met een groot volume, loten met grote diameters, ... Elk van die loten trekt immers een ander soort kopers aan. Voor het model is vooral de frequentie van aankopen van belang. Loten met grote volumes worden meestal gekocht door kopers die meestal meerdere loten kopen. Terwijl kleinere brandhoutloten regelmatig gekocht worden door kopers die slechts sporadisch een lot kopen.
Het dunningspercentage en bestandsgrondvlak werd telkens uit een uniforme verdeling getrokken. Het dunningspercentage lieten we variëren van 10% tot 50%, het bestandsgrondvlak tussen 20 en 40 m²/ha.
In deze oefening veronderstellen we dat de kopers een constante intentie hebben m.b.t. de proportie correct gekapte bomen[1] en de proportie gekapte bomen[2]. De proportie correct gekapt lieten we variëren van 50% tot 100%. De proportie gekapt varieerde tussen de proportie correct gekapt[3] en 150%. Bovenop de effecten van de koper pasten we nog een effect van het lot toe. Hierbij werden de proporties correct gekapt en gekapt vermenigvuldigd met een willekeurig getal tussen 80% en 120%. We kiezen bewust voor vrij extreme proporties opdat de effecten duidelijk zichtbaar zullen zijn.
Op basis van bovenstaande gegevens werd de oppervlakte van het lot berekend, de positie en omtrek van de bomen gesimuleerd, een dunning aangeduid en een kapping uitgevoerd. Voor de eenvoud veronderstellen we een vierkant lot. In het lot voeren we op willekeurige plaatsen Bitterlich metingen uit. Hierbij blijven we steeds op 20 m van de rand om randeffecten te vermijden. We voeren 4 metingen per ha uit met een minimum van 4 metingen per lot. In figuur 3.2 tonen we een gesimuleerd lot met een bestandsgrondvlak van 25 m²/ha, 20% dunningspercentage, 80% correct gekapt en 120% van het geschalmde aantal bomen gekapt. De cirkels met kruis geven de ligging van de meetpunten weer. In elk van de meetpunten voeren we een Bitterlich meting uit zowel voor als na de kapping, en dit voor de conversiefactoren K = 1, 2, 3 en 4 m²/ha.boom. In tabel 3.1 geven we de metingen voor de kapping weer.

Meting	N (K=1)	N (K=2)	N (K=3)	N (K=4)	G (K=1)	G (K=2)	G (K=3)	G (K=4)

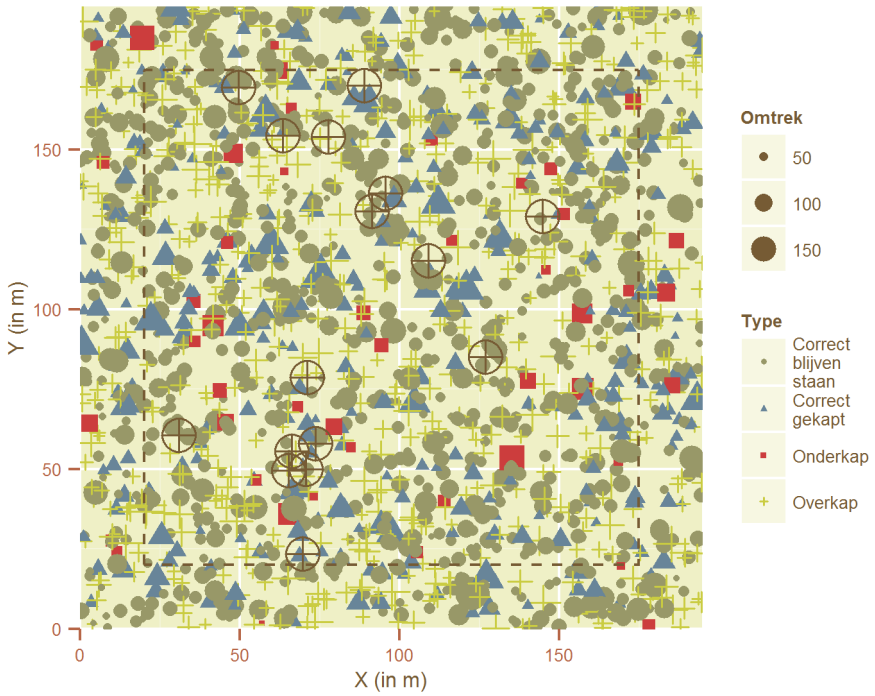
[1] aandeel effectief gekapte bomen van de geschalmde bomen
[2] verhouding tussen het aantal gekapte bomen en het aantal geschalmde bomen
[3] de proportie gekapt kan nooit lager zijn dan de proportie correct gekapt

1	24	11	6	5	*24*	*22*	*18*	*20*
2	24	11	9	7	*24*	*22*	*27*	*28*
3	23	12	8	5	*23*	*24*	*24*	*20*
4	24	13	4	1	*24*	*26*	*12*	*4*
5	34	15	11	8	*34*	*30*	*33*	*32*
6	24	13	9	7	*24*	*26*	*27*	*28*
7	20	9	7	6	*20*	*18*	*21*	*24*
8	23	13	9	8	*23*	*26*	*27*	*32*
9	29	13	6	4	*29*	*26*	*18*	*16*
10	28	17	10	10	*28*	*34*	*30*	*40*
11	23	11	8	6	*23*	*22*	*24*	*24*
12	26	12	10	10	*26*	*24*	*30*	*40*
13	26	15	12	7	*26*	*30*	*36*	*28*
14	21	12	6	6	*21*	*24*	*18*	*24*
15	24	9	8	7	*24*	*18*	*24*	*28*
16	25	14	7	5	*25*	*28*	*21*	*20*
Gem.	**24,9**	**12,5**	**8,1**	**6,4**	**24,9**	**25**	**24,4**	**25,5**

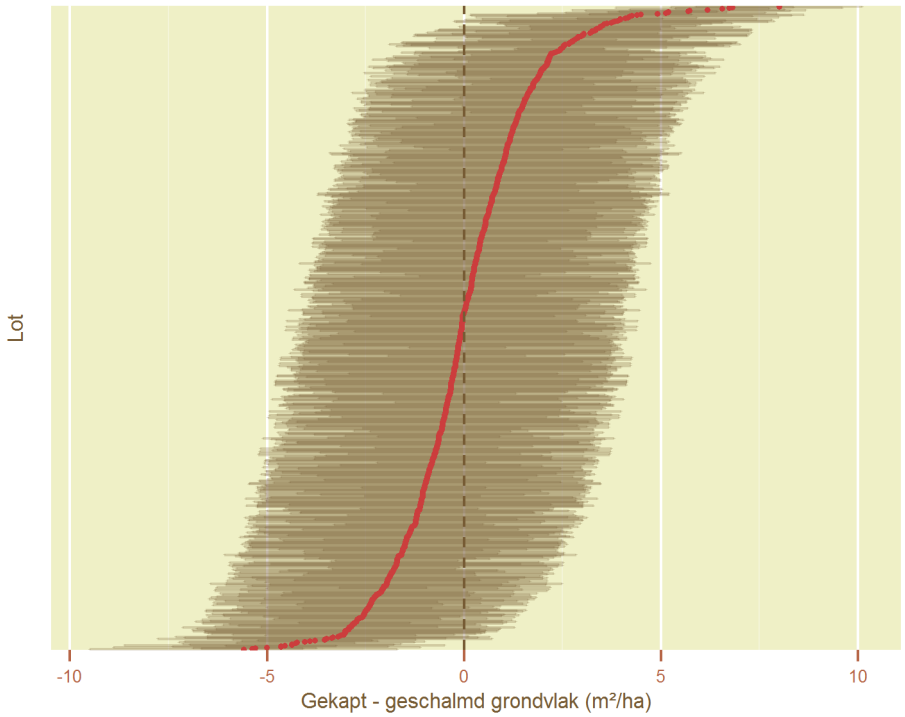
Tabel 3.1: Een tabel met de metingen voor de kapping van het voorbeeldlot uit figuur 3.2. N geeft het aantal bomen dat geteld werd met de Bitterlich methode, G het overeenkomstige grondvlak. K is de gebruikte grondvlakfactor (in m²/ha.boom).

3.3.2 Analyse van gesimuleerde dataset

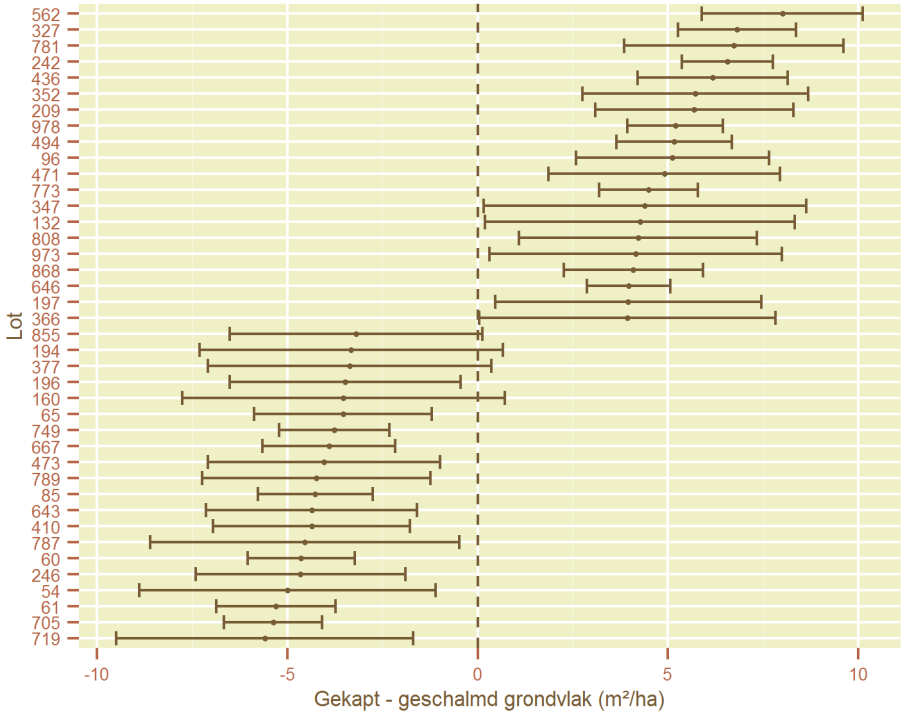
We demonstreren het concept aan de hand van de gegevens van alle gesimuleerde loten voor de metingen met K = 1 m²/ha per boom. De 'fixed effects' van het model bevatten twee termen: het intercept en extra kap. Het intercept is te interpreteren als het gemiddeld grondvlak na de kap over alle loten. Voor deze fictieve dataset wordt het geschat op (14.9; 15.7) m²/ha. Extrakap is het gemiddeld verschil tussen gekapt en geschalmd grondvlak. Dit geval is positief als het gekapte grondvalk groter is dan het geschalmde grondvlak. Voor de fictieve dataset wordt het geschat op (4.6; 6.1) m²/ha. Merk op dat deze fictieve dataset enkel tot doel heeft om het concept toe te lichten. De cijfers in kwestie zijn enkel bruikbaar om de bruikbaarheid van het concept na te gaan.
In figuur 3.3 geven we de schattingen van het verschil tussen gekapt en geschalmd grondvlak per lot, na correctie voor het globale verschil en het effect van de koper. Deze grafiek geeft inzicht in de verdeling van de effecten. Bij een beperkt aantal loten is het effect extreem laag en bij andere extreem hoog. Het zijn deze extreme loten waar de grootste problemen te verwachten zijn en bijgevolg het meeste aandacht verdienen. Door het hoge aantal loten is het echter niet mogelijk om de uit deze grafiek af te leiden over welke loten het gaat. Daarom selecteren we de loten met 20 laagste en 20 hoogste effecten. Die kunnen we weergeven in een tabel of figuur (bijv. figuur 3.4). Merk op dat we de analyse uitgevoerd hebben op de volledige dataset en dat deze loten van 5 verschillende jaren omvat. In de praktijk zouden we daarom kunnen kiezen om de selectie te beperken tot de extreemste loten van het afgelopen jaar.



Figuur 3.2: Voorbeeld van een gesimuleerd lot. De kruisje in een cirkel geven de meetpunten voor de Bitterlich methode aan.



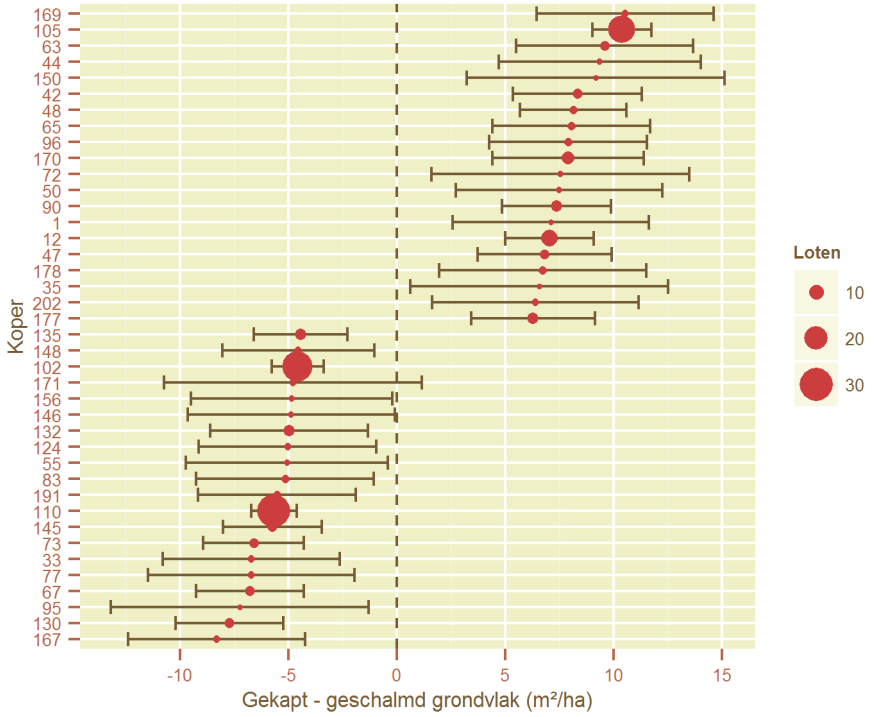
Figuur 3.3: Verschil tussen gekapt en geschalmd grondvlak per fictieve lot in de fictieve dataset (in m²/ha). Hierbij werd reeds het globale verschil en het effect van de koper in rekening gebracht.



Figuur 3.4: Verschil tussen gekapt en geschalmd grondvlak per lot voor de 40 extreemste fictieve loten in de fictieve dataset (in m²/ha). Hierbij werd reeds het globale verschil en het effect van de koper in rekening gebracht.

Op een gelijkaardige manier kunnen we kijken naar het verschil tussen gekapt en geschalmd grondvlak per koper (Figuur 3.5). Ook bij de kopers kan best gefocust worden op de extremen. In figuur 3.6 geven we de kopers met de 20 hoogste en 20 laagste effecten. De grens van 20 hoge en 20 lage is arbitrair gekozen. Het ANB zal steeds de bijhorende waarden moeten interpreteren. Ook als geen enkele koper meer kapt dan geschalmd, zullen er kopers zijn die het hoogste effect hebben. Deze tool is en blijft een screeningstool die aangeeft waar mogelijk de grootste problemen zijn. Indien nummer 10 een probleem blijkt te zijn, dan zullen nummers 1 tot 9 dat waarschijnlijk ook zijn. Indien nummer 10 geen probleem is, dan zullen nummers 11 en verder dat waarschijnlijk ook niet zijn. Merk op dat de breedte van de betrouwbaarheidsintervallen verschilt naargelang de koper. Hoe vaker we een bepaalde koper opvolgen, hoe smaller het betrouwbaarheidsinterval zal zijn. De breedte van het interval hangt tevens af van de werkwijze van de koper. Een koper die zeer systematisch een bepaald percentage te veel of te weinig kapt, zal een smaller betrouwbaarheidsinterval krijgen in vergelijking met een koper die de ene keer te weinig en de andere keer te veel kapt.

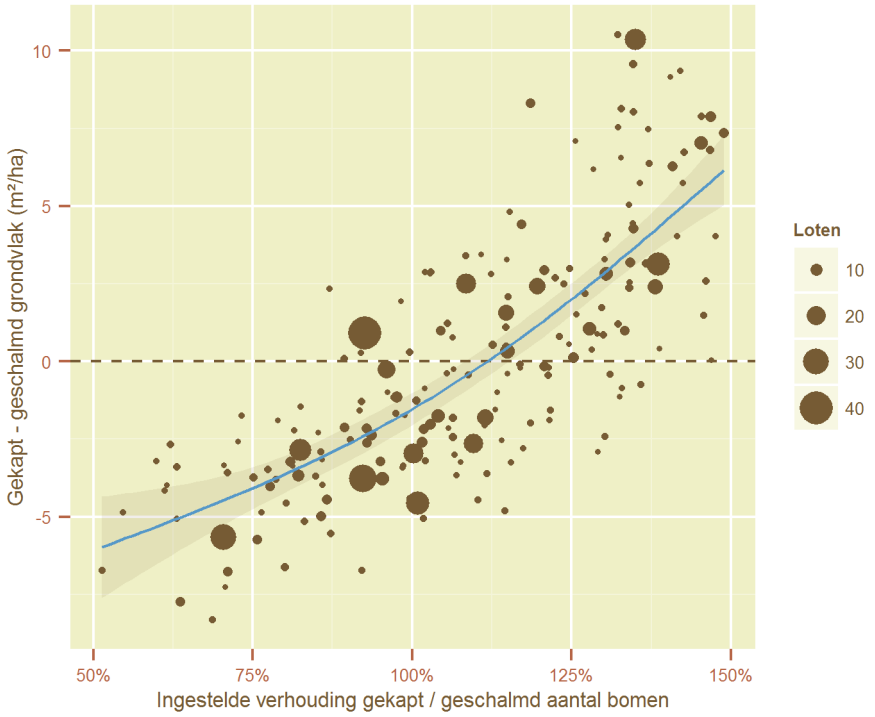
Figuur 3.5: Gemiddeld verschil tussen gekapt en geschalmd grondvlak per fictieve koper in de fictieve dataset (in m²/ha). Hierbij werd reeds het globale verschil en het effect van het lot in rekening gebracht.



Figuur 3.6: Verschil tussen gekapt en geschalmd grondvlak per koper voor de 40 extreemste fictieve kopers in de fictieve dataset (in m²/ha). Hierbij werd reeds het globale verschil en het effect van het lot in rekening gebracht.

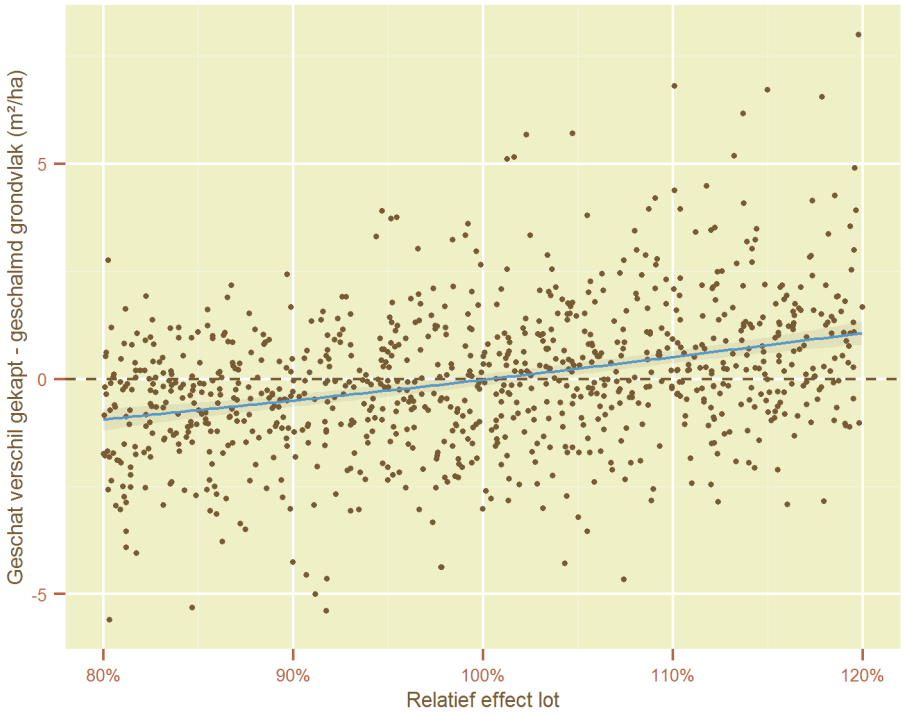
3.3.3 Vergelijking van resultaten met gekende effecten

Het voordeel aan gesimuleerde gegevens is dat we weten wat de werkelijke effecten achter de gegevens zijn. We hebben voor elke koper een streefcijfer ingesteld voor de verhouding tussen het aantal gekapte en geschalmde bomen. In figuur 3.7 zetten we deze waarde uit tegen de gemodelleerde effect van de koper uit figuur 3.5. We stellen vast dat er een duidelijk verband is tussen het gemodelleerde en het ingestelde effect. Het verband is niet perfect om een aantal redenen: 1) het ingestelde effect betreft het aantal bomen terwijl het gemodelleerde effect het grondvlak betreft. 2) het gemodelleerde effect houdt ook rekening met het effect van de gemiddelde koper en het lot.



Figuur 3.7: Vergelijking tussen ingestelde verhouding gekapt versus geschalmd en het verschil tussen gekapt en geschalmd grondvlak per koper in de fictieve dataset (in m²/ha).

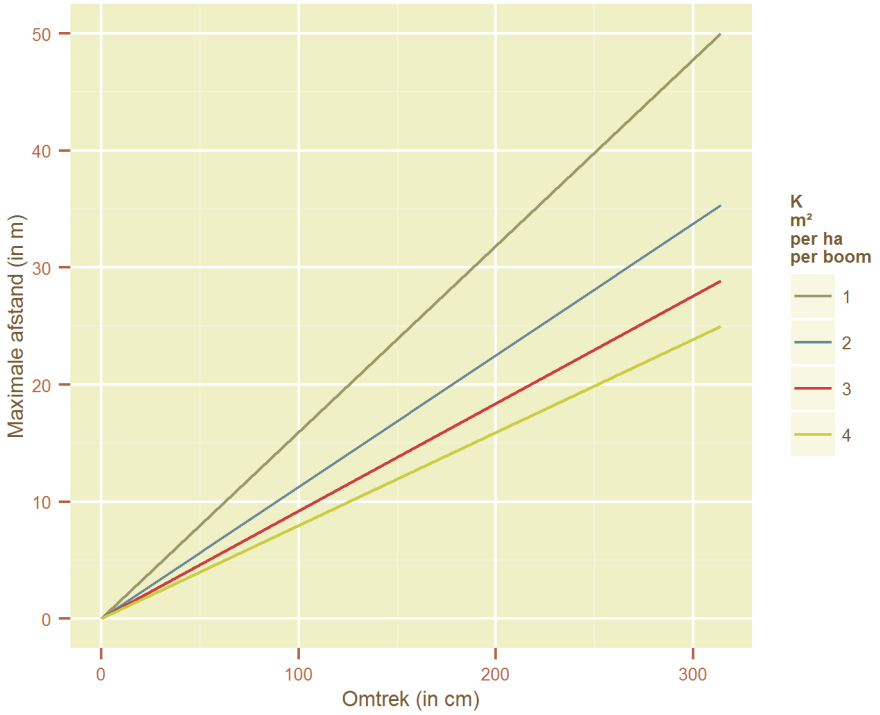
Wanneer we de vergelijking maken tussen de proportie gekapt in een lot en het geschatte effect van het lot (Figuur 3.8), stellen we vast dat er nog steeds een verband is doch minder duidelijk. Dit is enerzijds te wijten aan de meetfouten op de individuele grondvlakmetingen met Bitterlich. Anderzijds wordt in deze dataset het verschil tussen het gekapt en het geschalmd grondvlak gedomineerd door het effect van de koper.



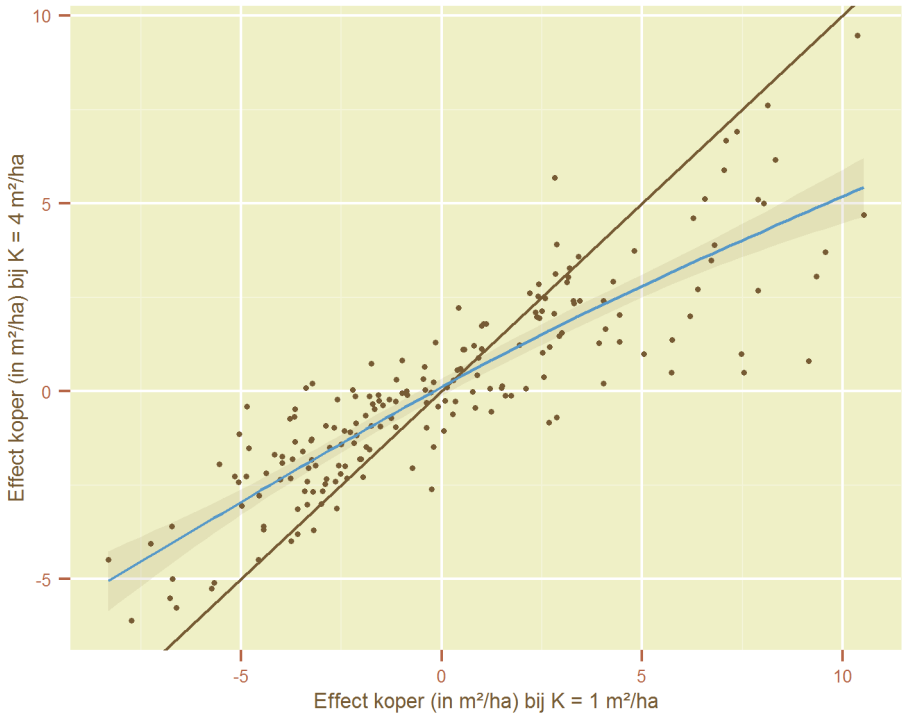
Figuur 3.8: Vergelijking tussen het relatieve lot effect versus geschalmd en het verschil tussen gekapt en geschalmd grondvlak per lot in de fictieve dataset (in m²/ha).

3.3.4 Geschikte grondvlakconversiefactor K

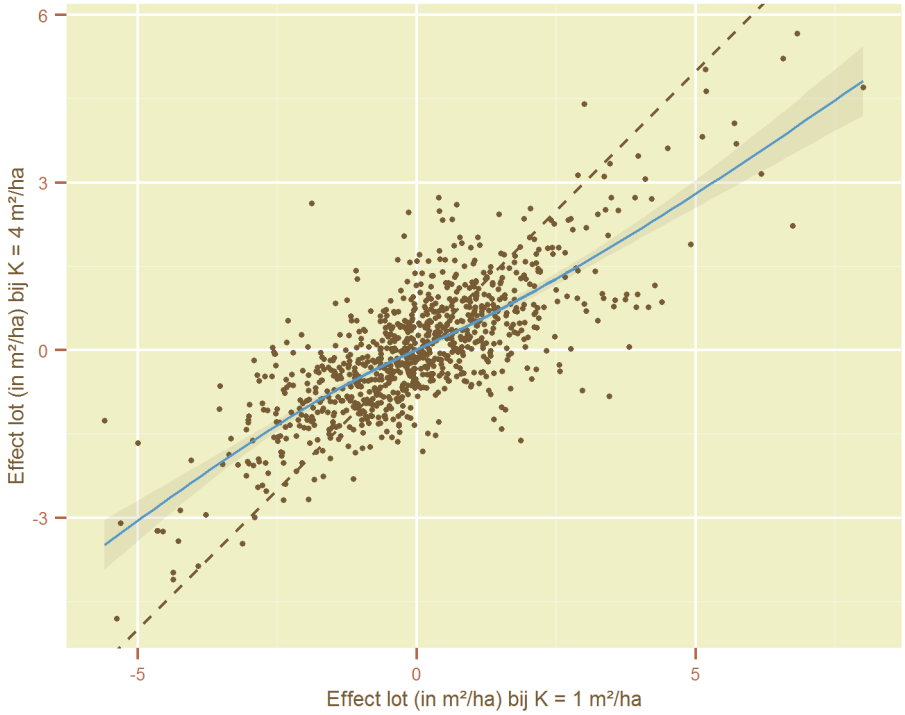
Een van de belangrijke keuze die gemaakte moet worden bij grondvlakmetingen met de Bitterlich methode is de keuze van de grondvlakconversiefactor K. Deze K waarde is gerelateerd aan de breedte van de opening in het plaatje en bijgevolg aan de kijkhoek. Bij een lage K waarde is de kijkhoek smal. Hierdoor zal een boom relatief snel voldoende breed zijn om tot de steekproef gerekend te worden. We hebben dus meer bomen in de steekproef. Om dit te compenseren is de K waarde lager. Een lage K waarde heeft als voordeel dat we meer resolutie hebben in de grondvlakmetingen. Bij K = 1 m²/ha/boom zal het grondvlakken in stappen van 1m²/ha toenemen, bij K = 4 m²/ha.boom zal dat in stappen van 4 m²/ha zijn. Een lage K waarde heeft echter ook nadelen. Ten eerste moeten we meer bomen tellen, hetgeen lastig kan zijn in loten met een vrij hoog stamtal. Ten tweede impliceert een lage K waarde dat we over een grotere afstand moeten kijken. In figuur 3.9 geven we de maximale afstand weer waarop de boom mag staan om tot de steekproef te behoren. Het verband tussen de maximale afstand en de omtrek is $afstand = \frac{omtrek}{2\pi\sqrt{K}}$. Hoe dikker de boom, hoe groter deze afstand is. Dat geeft problemen wanneer een lot een goed ontwikkelde struiklaag heeft. Verder lopen we het risico dat we bomen meten die in een ander perceel staan.
In figuur 3.10 en 3.11 gaan we na wat het effect is van de K waarde op de geschatte effecten van de koper en het lot. De invloed op de effecten van de koper zijn beperkter dan die op de effecten van het lot. In beide gevallen stellen we vast dat er meer ruis op de effecten zit en dat de effecten minder extreem worden geschat. Door een hogere K waarde te gebruiken, wordt het grondvlak met een lagere resolutie geschat waardoor we minder informatie hebben. Minder informatie geeft meest ruis en maakt het moeilijker om de effecten van de koper of het lot te onderscheiden van de ruis.
De invloed op de standaardfouten van het effect van de koper blijkt eerder beperkt te zijn (Figuur 3.12). De invloed bij de loten (Figuur 3.13) is duidelijk groter.



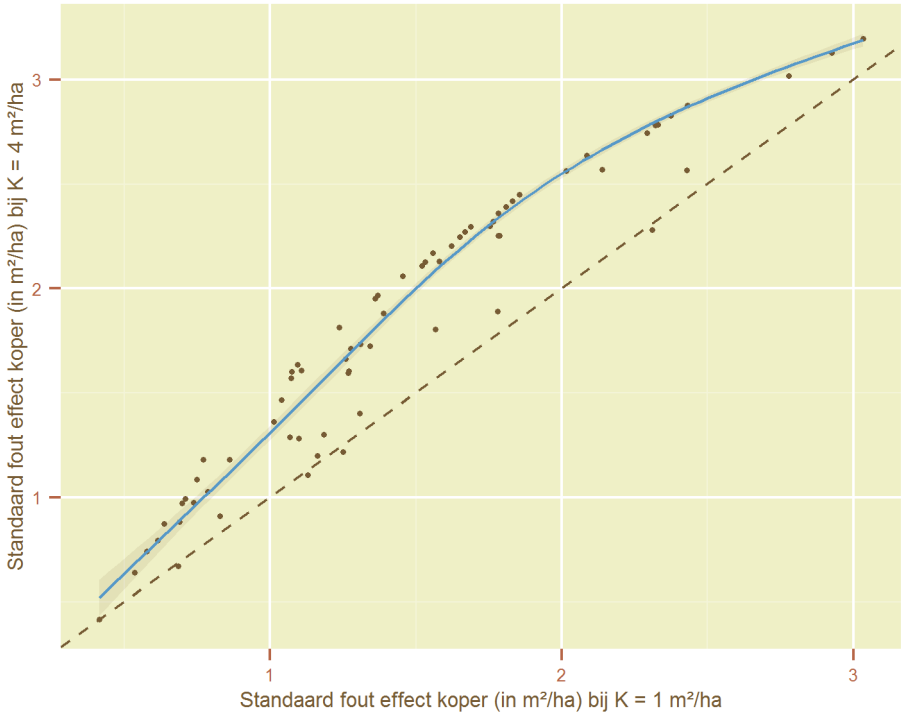
Figuur 3.9: Maximale afstand waarop een boom minstens even breed is als de kijkhoek in functie van zijn omtrek en de grondvlakconversiefactor K.



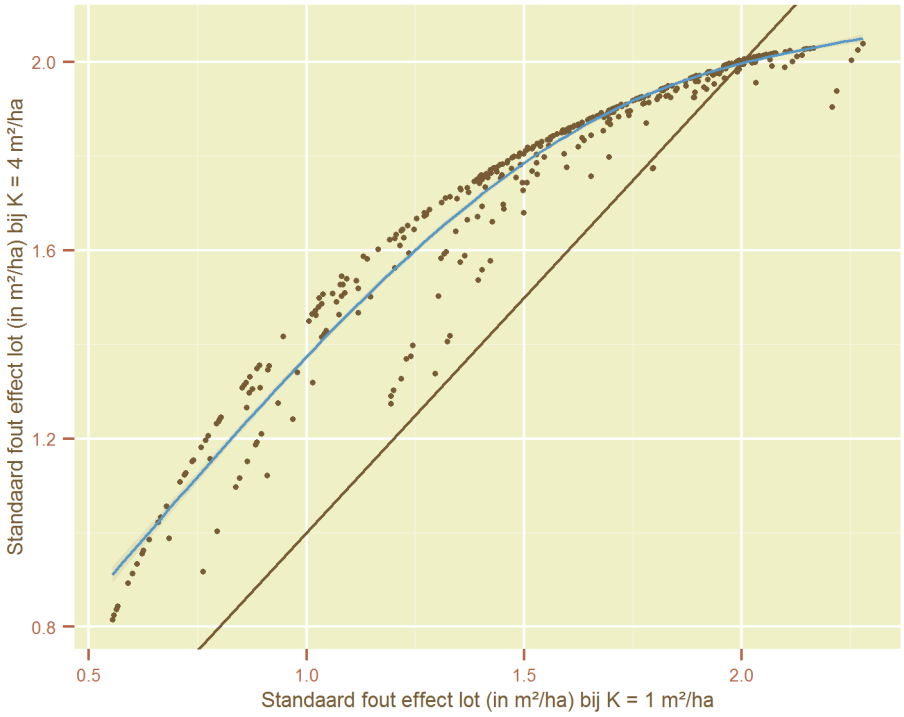
Figuur 3.10: Verband tussen het geschat effect van de koper in de fictieve dataset bij K = 1 m²/ha.boom en K = 4 m²/ha.boom.



Figuur 3.11: Verband tussen het geschat effect van het lot in de fictieve dataset bij K = 1 m²/ha.boom en K = 4 m²/ha.boom.



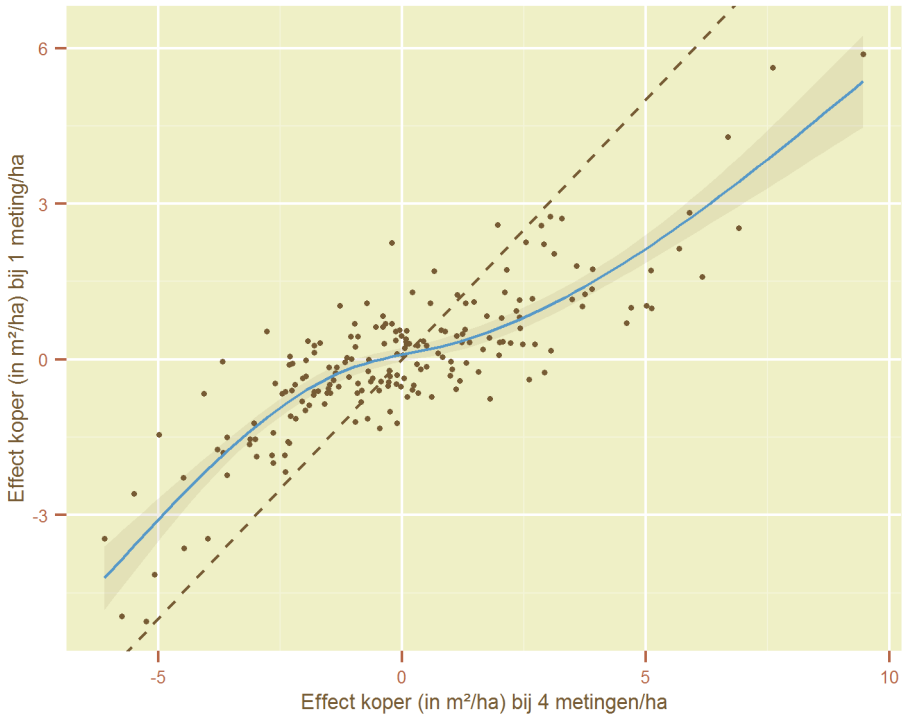
Figuur 3.12: Verband tussen de standaard fouten van het geschat effect van de koper in de fictieve dataset bij K = 1 m²/ha.boom en K = 4 m²/ha.boom.



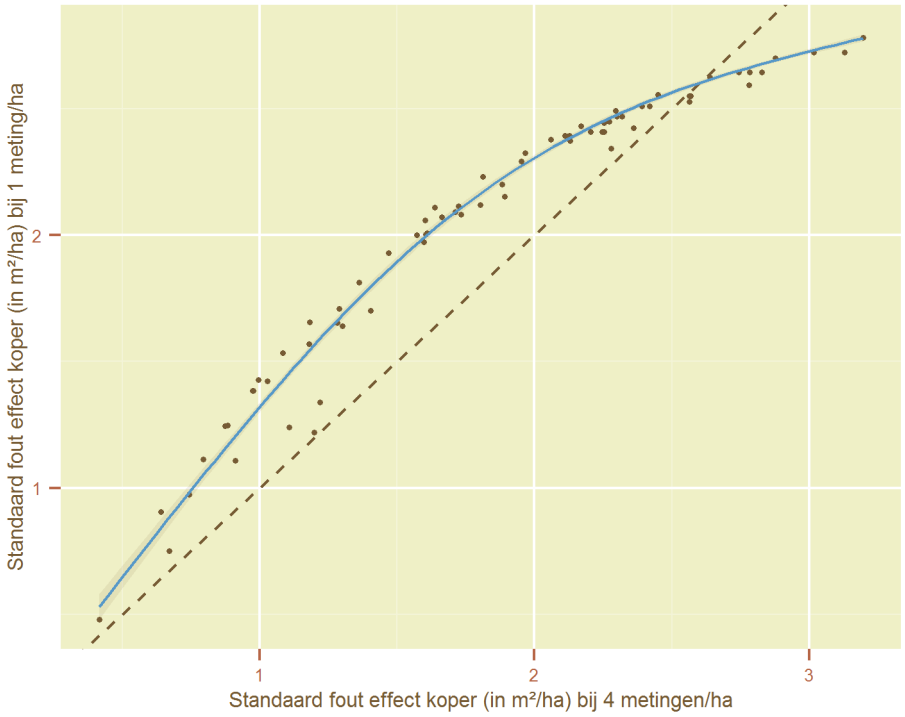
Figuur 3.13: Verband tussen de standaard fouten van het geschat effect van het lot bij K = 1 m²/ha en K = 4 m²/ha.

3.3.5 Invloed van de densiteit van de meetpunten

Bij deze metingen aan gesimuleerde loten zijn we er vanuit gegaan dat we 4 metingen per ha uitvoerden met een minimum van 4 per lot. Om een idee te krijgen van wat er gebeurt bij een lagere densiteit aan meetpunten herhalen we de analyse met een kwart van de meetpunten (1 meting per ha en minstens 1 per lot). Hierbij gebruiken we enkel de 1^e, 5^e, 9^e, 13^e, . . . meting. In figuur 3.14 geven we aan wat het verband is tussen de effecten van de kopers geschat bij 4 metingen per ha en bij 1 meting per ha. het verminderen van het aantal metingen resulteert in minder informatie. Met als gevolg dat de effecten van de kopers meer ruis vertonen en minder uitgesproken zijn. Het effect is dus vergelijkbaar als bij figuur 3.10, hoewel daar de vermindering aan informatie te wijten is aan een lagere nauwkeurigheid van de metingen. Het effect op de standaardfouten is eveneens gelijkaardig (figuur 3.15).



Figuur 3.14: Verband tussen het geschat effect van de koper in de fictieve dataset bij 4 metingen/ha en 1 meting/ha telkens bij K = 4 m²/ha.boom.



Figuur 3.15: Verband tussen de standaard fouten van het geschat effect van de koper in de fictieve dataset bij 1 meting/ha en 4 metingen/ha.

3.4 Conclusie en aanbevelingen

3.4.1 Conclusie

Met de analyse van de gesimuleerde dataset hebben we aangetoond dat de voorgestelde werkwijze bruikbaar is als een screening tool. De werklast per lot is relatief beperkt en is haalbaar om gelijktijdig uit te voeren met het aantekenen van de dunning of een snelle terreincontrole na de kap. Hoe meer metingen van een bepaald lot of een bepaalde koper, hoe beter de screening tool werkt. De kracht van de screening tool is dat gegevens van meerdere loten gebundeld kan worden waardoor patronen (bv. een koper die systematisch te weinig kapt) duidelijker worden.

3.4.2 Aanbevelingen

- Hoe meer gegevens, hoe beter de screening tool zal werken. Gegevens van 50 loten verdeelt over 10 exploitanten is een minimum. Merk op de gegevens van meerdere jaren gecombineerd mag worden. De aantallen slaan op de gecombineerde dataset.
- Gebruik een geschikte grondvlakconversiefactor K in functie van het lot. Gebruik een hoge K bij weinig overzichtelijke loten of bij loten met zeer dikke bomen in de omliggende loten.
- Gebruik binnen een lot steeds dezelfde K waarde, zowel voor als na de kapping. Tussen de loten mag de K waarde verschillen.
- Blijf op minstens 20 m van de bestandsrand bij de plaatsen van een Bitterlich proefvlak. Dit impliceert dat de methode minder bruikbaar is voor de hele kleine loten (< 0.2 ha).
- Streef naar minstens 4 metingen per lot bij een dichtheid van minstens 4 metingen per ha.
- De methode wordt best toegepast op zoveel mogelijk (alle) loten van dunningen die een voldoende oppervlakte (> 0.2 ha) beslaan. Voor eindkappen is de methode weinig relevant.

3.5 `R` code van een voorbeeldanalyse

Hieronder geven we de code van een voorbeeldanalyse. De code veronderstelt een dataset met een rij per Bitterlich meting en vijf variabelen:

- `Grondvlak`: het gemeten grondvlak met de Bitterlich methode
- `Geschalmd`: het geschalmd grondvlak indien de meting na de kap is, anders 0 m²/ha.
- `ExtraKap`: een indicatorvariabele: 1 voor de kap, 0 na de kap
- `Lot`: een unieke code per lot
- `Koper`: een unieke code per koper

In het onderstaande voorbeeld zijn de unieke code voor `Lot` en `Koper`, doch dat is geen vereiste. De codes mogen uit letters bestaan (bv. de naam van de koper), op voorwaarde dat ze consistent en uniek zijn. M.a.w. indien een koper meerdere loten koopt, moet de koper ID telkens dezelfde zijn. En elke koper moet een andere code krijgen.

```
#fictieve gegevens inlezen van een R data file
load("dataset.rda")
#overzicht van de dataset
summary(dataset)
```

```
##    Grondvlak      Geschalmd        ExtraKap        Lot           Koper
##  Min.   : 0.0   Min.   : 0.00   Min.   :0.0   Min.   :   1   Min.   :  1
##  1st Qu.:12.0   1st Qu.: 0.00   1st Qu.:0.0   1st Qu.: 258   1st Qu.: 85
##  Median :20.0   Median : 0.00   Median :0.5   Median : 496   Median :115
##  Mean   :22.7   Mean   : 3.78   Mean   :0.5   Mean   : 504   Mean   :107
##  3rd Qu.:32.0   3rd Qu.: 6.58   3rd Qu.:1.0   3rd Qu.: 749   3rd Qu.:127
##  Max.   :84.0   Max.   :21.01   Max.   :1.0   Max.   :1000   Max.   :211
##       Jaar
##  Min.   :2015
##  1st Qu.:2016
##  Median :2017
##  Mean   :2017
##  3rd Qu.:2018
##  Max.   :2019
```

```r
#lme4 is het package dat nodig is om het model te fitten
library(lme4)
#fit het model aan de dataset
model <- lmer(
  Grondvlak ~ offset(Geschalmd) + ExtraKap + (0 + ExtraKap|Koper) + (ExtraKap|Lot),
  data = dataset
)
# betrouwbaarheidsintervallen van parameters
# (Intercept): grondvlak na de kap van het gemiddeld lot
# ExtraKap: gemiddeld extra gekapt grondvlak
confint(model, method = "Wald")
```

```
##              2.5 % 97.5 %
## (Intercept) 15.117 16.079
## ExtraKap     4.708  6.166
```

```r
#haal de effecten van de koper uit het model
effect.koper <- ranef(model, condVar = TRUE)$Koper
colnames(effect.koper) <- "Effect"
#de ID van de koper zit in de rijnamen. Voeg deze toe aan de effecten
effect.koper$Koper <- rownames(effect.koper)
#bereken de standaard fout
effect.koper$SE <- sqrt(attr(effect.koper, "postVar")[1, 1, ])
#effecten sorteren volgens effectgrootte
effect.koper <- effect.koper[order(effect.koper$Effect), c("Koper", "Effect", "SE")]
#bereken de onder- en bovengrens van het betrouwbaarheidsinterval van het effect
effect.koper$LCL <- qnorm(0.025, mean = effect.koper$Effect, sd = effect.koper$SE)
effect.koper$UCL <- qnorm(0.975, mean = effect.koper$Effect, sd = effect.koper$SE)
#selecteer de kopers met de 20 laagste effecten
head(effect.koper, 20)
```

```
##     Koper Effect     SE     LCL     UCL
## 130   130 -6.106 1.5963  -9.235 -2.9775
## 73     73 -5.756 1.3011  -8.306 -3.2060
## 67     67 -5.500 1.6041  -8.644 -2.3561
## 145   145 -5.254 1.1981  -7.602 -2.9052
## 110   110 -5.088 0.6408  -6.343 -3.8317
## 33     33 -4.986 2.6375 -10.156  0.1830
## 167   167 -4.488 2.6375  -9.657  0.6817
## 102   102 -4.481 0.7945  -6.038 -2.9233
## 95     95 -4.063 3.1963 -10.327  2.2020
## 122   122 -3.984 0.7428  -5.439 -2.5277
## 181   181 -3.797 2.6375  -8.967  1.3719
## 53     53 -3.692 2.6375  -8.862  1.4770
## 109   109 -3.677 2.6375  -8.846  1.4925
## 135   135 -3.595 1.2828  -6.109 -1.0805
## 77     77 -3.590 2.8770  -9.228  2.0491
## 155   155 -3.140 1.4038  -5.891 -0.3884
## 182   182 -3.116 2.2964  -7.617  1.3845
## 132   132 -3.040 2.4493  -7.841  1.7601
## 5       5 -3.012 2.8770  -8.651  2.6266
## 8       8 -2.994 2.8280  -8.537  2.5488
```

```r
#selecteer de kopers met de 20 hoogste effecten
tail(effect.koper, 20)
```

```
##     Koper Effect     SE      LCL    UCL
## 39     39  3.119 0.9118  1.33211  4.906
## 29     29  3.276 1.2885  0.75059  5.801
## 178   178  3.491 2.8770 -2.14807  9.129
```

```
## 158   158  3.582 2.6375 -1.58738  8.751
## 63     63  3.703 2.6375 -1.46642  8.872
## 25     25  3.744 2.8770 -1.89462  9.383
## 47     47  3.900 2.1304 -0.27601  8.075
## 13     13  3.913 2.8770 -1.72561  9.552
## 177   177  4.603 2.0606  0.56425  8.642
## 169   169  4.701 2.6375 -0.46828  9.870
## 65     65  5.008 2.4493  0.20766  9.809
## 170   170  5.103 2.3612  0.47485  9.731
## 35     35  5.122 3.1963 -1.14303 11.386
## 91     91  5.688 2.8770  0.04901 11.326
## 12     12  5.895 1.4681  3.01733  8.772
## 42     42  6.154 2.1101  2.01806 10.289
## 1       1  6.681 2.2796  2.21355 11.149
## 90     90  6.908 1.7109  3.55452 10.261
## 48     48  7.605 1.2183  5.21738  9.993
## 105   105  9.468 0.9737  7.55969 11.376
```

```
#fit het model aan de gegevens van 2015
model.2015 <- lmer(
  Grondvlak ~ offset(Geschalmd) + ExtraKap + (0 + ExtraKap|Koper) + (ExtraKap|Lot),
  data = subset(dataset, Jaar == 2015)
)
# betrouwbaarheidsintervallen van parameters
# (Intercept): grondvlak na de kap van het gemiddeld lot
# ExtraKap: gemiddeld extra gekapt grondvlak
confint(model.2015, method = "Wald")
```

```
##              2.5 % 97.5 %
## (Intercept) 14.449 16.520
## ExtraKap     4.865  7.032
```

```
#haal de effecten van de koper uit het model
effect.koper.2015 <- ranef(model.2015, condVar = TRUE)$Koper
colnames(effect.koper.2015) <- "Effect"
#de ID van de koper zit in de rijnamen. Voeg deze toe aan de effecten
effect.koper.2015$Koper <- rownames(effect.koper.2015)
#bereken de standaard fout
effect.koper.2015$SE <- sqrt(attr(effect.koper.2015, "postVar")[1, 1, ])
#effecten sorteren volgens effectgrootte
effect.koper.2015 <- effect.koper.2015[order(effect.koper.2015$Effect), c("Koper", "Effect", "SE")]
#bereken de onder- en bovengrens van het betrouwbaarheidsinterval van het effect
effect.koper.2015$LCL <- qnorm(0.025, mean = effect.koper.2015$Effect, sd = effect.koper.2015$SE)
effect.koper.2015$UCL <- qnorm(0.975, mean = effect.koper.2015$Effect, sd = effect.koper.2015$SE)
#selecteer de kopers met de 20 laagste effecten
head(effect.koper.2015, 20)
```

```
##     Koper Effect    SE    LCL     UCL
## 135   135 -3.740 2.077 -7.812  0.3318
## 130   130 -3.461 1.941 -7.265  0.3427
## 145   145 -3.416 2.072 -7.477  0.6449
## 110   110 -3.251 1.231 -5.663 -0.8381
## 157   157 -2.894 2.209 -7.224  1.4364
## 73     73 -2.812 2.172 -7.068  1.4447
## 67     67 -2.470 1.934 -6.260  1.3196
## 53     53 -2.394 2.648 -7.584  2.7961
## 203   203 -2.130 2.648 -7.320  3.0601
## 109   109 -1.998 2.648 -7.188  3.1923
## 124   124 -1.963 2.648 -7.153  3.2268
## 89     89 -1.933 2.461 -6.757  2.8907
```

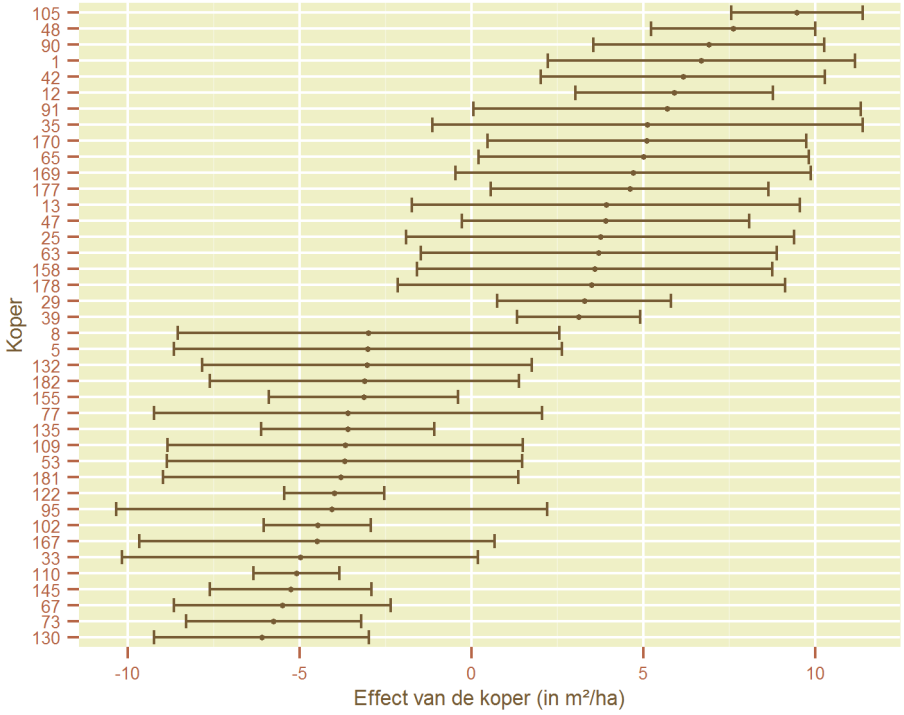
```
## 85     85 -1.920 1.503 -4.866  1.0261
## 21     21 -1.775 2.430 -6.538  2.9874
## 155   155 -1.764 2.324 -6.319  2.7903
## 76     76 -1.740 2.430 -6.503  3.0228
## 152   152 -1.719 2.199 -6.030  2.5918
## 182   182 -1.656 2.648 -6.847  3.5337
## 102   102 -1.467 1.979 -5.346  2.4124
## 29     29 -1.317 2.359 -5.942  3.3070
```

```
#selecteer de kopers met de 20 hoogste effecten
tail(effect.koper.2015, 20)
```

```
##     Koper Effect    SE     LCL   UCL
## 202   202  1.264 2.648 -3.9258 6.455
## 86     86  1.378 2.648 -3.8120 6.568
## 92     92  1.382 2.209 -2.9480 5.712
## 49     49  1.467 2.648 -3.7229 6.657
## 4       4  1.527 2.648 -3.6631 6.717
## 52     52  1.679 2.369 -2.9646 6.323
## 17     17  1.713 2.648 -3.4770 6.903
## 100   100  1.793 1.656 -1.4523 5.038
## 45     45  1.803 1.926 -1.9707 5.577
## 39     39  1.841 1.662 -1.4167 5.098
## 62     62  1.948 2.648 -3.2425 7.138
## 158   158  1.950 2.648 -3.2397 7.141
## 177   177  1.962 2.258 -2.4635 6.389
## 136   136  2.112 2.258 -2.3139 6.538
## 44     44  2.132 2.606 -2.9766 7.240
## 169   169  3.018 2.648 -2.1722 8.208
## 42     42  3.398 2.258 -1.0285 7.824
## 48     48  3.958 1.727  0.5725 7.343
## 12     12  4.368 2.040  0.3704 8.366
## 105   105  5.945 1.704  2.6047 9.286
```

```
#selecteer de kopers met de 20 laagste en 20 hoogste effecten
extreem.koper <- rbind(
  head(effect.koper, 20),
  tail(effect.koper, 20)
)
#zet de koper ID om naar een factor waarvan de levels in volgorde van de effectgrootte staan
extreem.koper$Koper <- factor(
  extreem.koper$Koper,
  levels = extreem.koper$Koper[order(extreem.koper$Effect)]
)
#ggplot2 is het package om grafieken te maken
library(ggplot2)
#basisdefinities van de plot
#dataset + welke variabele op welke as moet komen
ggplot(extreem.koper, aes(x = Effect, y = Koper, xmin = LCL, xmax = UCL)) +
#voeg horizontale foutvlaggen toe (xmin, xmax vs y)
  geom_errorbarh() +
#voeg punten toe (x vs y)
  geom_point() +
  xlab("Effect van de koper (in m²/ha)")


#haal de effecten van het lot uit het model
effect.lot <- ranef(model, condVar = TRUE)$Lot
colnames(effect.lot) <- c("Basis", "Effect")
#de ID van het lot zit in de rijnamen. Voeg deze toe aan de effecten
```



Figuur 3.16: Grafische voorstelling van de extreemste koper effecten voor de fictieve dataset na correctie van het globale effect en het effect per lot.

```
effect.lot$Lot <- rownames(effect.lot)
#bereken de standaard fout
effect.lot$SE <- sqrt(attr(effect.lot, "postVar")[2, 2, ])
#jaartal per lot
jaartal <- unique(dataset[, c("Lot", "Jaar")])
#voeg het jaartal toe aan de effecten
effect.lot <- merge(
  effect.lot,
  jaartal
)
#effecten sorteren volgens effect
effect.lot <- effect.lot[order(effect.lot$Effect), c("Lot", "Jaar", "Effect", "SE")]
#bereken de onder- en bovengrens van het betrouwbaarheidsinterval van het effect
effect.lot$LCL <- qnorm(0.025, mean = effect.lot$Effect, sd = effect.lot$SE)
effect.lot$UCL <- qnorm(0.975, mean = effect.lot$Effect, sd = effect.lot$SE)
#selecteer de loten met de 20 laagste effecten
head(effect.lot, 20)
```

```
##     Lot Jaar Effect    SE    LCL      UCL
## 618 705 2019 -4.807 1.022 -6.811 -2.80360
## 553 643 2017 -4.099 1.764 -7.556 -0.64238
## 318 410 2019 -3.978 1.407 -6.735 -1.22108
## 579 667 2016 -3.867 1.193 -6.205 -1.52926
## 403 492 2016 -3.462 1.340 -6.089 -0.83454
## 768  85 2019 -3.410 1.197 -5.757 -1.06335
## 700 787 2016 -3.245 2.019 -7.202  0.71342
## 150 246 2015 -3.235 1.559 -6.291 -0.17985
## 509  60 2019 -3.231 1.124 -5.434 -1.02790
```

```
## 517  61 2015 -3.092 1.241 -5.525 -0.65934
## 268 360 2019 -2.997 1.502 -5.941 -0.05283
## 660 749 2018 -2.950 1.155 -5.213 -0.68623
## 702 789 2018 -2.866 1.786 -6.367  0.63503
## 238  33 2017 -2.680 1.417 -5.458  0.09804
## 184 279 2018 -2.674 1.944 -6.484  1.13625
## 281 373 2017 -2.524 1.509 -5.482  0.43429
## 266 359 2018 -2.450 1.790 -5.959  1.05941
## 181 276 2015 -2.416 1.127 -4.625 -0.20674
## 812 890 2019 -2.391 1.151 -4.648 -0.13435
## 445 535 2019 -2.315 1.503 -5.260  0.63065
```

```
#selecteer de loten met de 20 hoogste effecten
tail(effect.lot, 20)
```

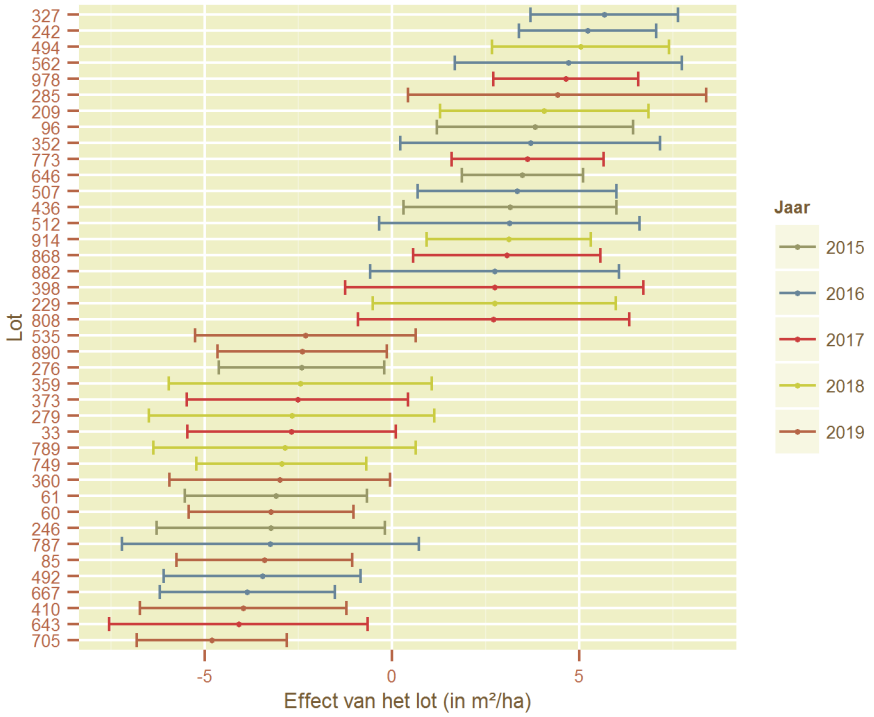
```
##     Lot Jaar Effect     SE     LCL   UCL
## 724 808 2017  2.713 1.8509 -0.9144 6.341
## 132 229 2018  2.733 1.6524 -0.5060 5.971
## 305 398 2017  2.735 2.0304 -1.2450 6.714
## 803 882 2016  2.738 1.6939 -0.5816 6.059
## 788 868 2017  3.071 1.2756  0.5712 5.571
## 836 914 2018  3.113 1.1163  0.9249 5.301
## 420 512 2016  3.136 1.7717 -0.3369 6.608
## 344 436 2015  3.152 1.4493  0.3117 5.993
## 415 507 2016  3.338 1.3545  0.6836 5.993
## 556 646 2015  3.478 0.8246  1.8617 5.094
## 686 773 2017  3.615 1.0340  1.5883 5.642
## 259 352 2016  3.693 1.7678  0.2280 7.158
## 881  96 2015  3.820 1.3381  1.1972 6.442
## 112 209 2018  4.061 1.4185  1.2807 6.841
## 190 285 2019  4.413 2.0304  0.4334 8.392
## 898 978 2017  4.643 0.9856  2.7112 6.575
## 473 562 2016  4.704 1.5455  1.6744 7.733
## 405 494 2018  5.030 1.2057  2.6671 7.394
## 147 242 2016  5.222 0.9333  3.3930 7.052
## 235 327 2016  5.668 1.0031  3.7021 7.634
```

```
#selecteer de loten met de 20 hoogste effecten van een bepaald jaar
tail(
  subset(
    effect.lot,
    Jaar == 2015
  ),
  20
)
```

```
##     Lot Jaar Effect     SE      LCL   UCL
## 888 966 2015  1.281 1.9929 -2.62501 5.187
## 567 656 2015  1.402 1.8855 -2.29354 5.097
## 524 616 2015  1.500 2.0187 -2.45656 5.456
## 70  166 2015  1.590 1.5276 -1.40399 4.584
## 744 826 2015  1.613 2.0109 -2.32797 5.555
## 497 586 2015  1.623 2.0240 -2.34426 5.590
## 486 576 2015  1.695 2.0240 -2.27210 5.662
## 750 831 2015  1.721 0.8933 -0.03008 3.472
## 726  81 2015  1.785 2.0088 -2.15258 5.722
## 589 676 2015  1.835 2.0194 -2.12244 5.793
## 124 221 2015  1.837 2.0194 -2.12119 5.795
## 135 231 2015  1.873 1.9953 -2.03825 5.783
## 381 471 2015  1.893 1.8356 -1.70483 5.490
```

```
## 317  41 2015  2.016 1.9422 -1.79025 5.823
## 280 371 2015  2.045 1.9948 -1.86497 5.955
## 694 781 2015  2.226 1.7739 -1.25126 5.702
## 339 431 2015  2.508 1.8394 -1.09758 6.113
## 344 436 2015  3.152 1.4493  0.31167 5.993
## 556 646 2015  3.478 0.8246  1.86166 5.094
## 881  96 2015  3.820 1.3381  1.19724 6.442
```

```
#selecteer de loten met de 20 laagste en 20 hoogste effecten
extreem.lot <- rbind(
  head(effect.lot, 20),
  tail(effect.lot, 20)
)
#zet het lot ID om naar een factor waarvan de levels in volgorde van de effectgrootte staan
extreem.lot$Lot <- factor(
  extreem.lot$Lot,
  levels = extreem.lot$Lot[order(extreem.lot$Effect)]
)
#zet het jaartal om naar een factor
extreem.lot$Jaar <- factor(extreem.lot$Jaar)
#basisdefinities van de plot
#dataset + welke variabele op welke as moet komen
ggplot(extreem.lot, aes(x = Effect, y = Lot, xmin = LCL, xmax = UCL, colour = Jaar)) +
#voeg horizontale foutvlaggen toe (xmin en xmax vs y, kleur volgens jaartal)
  geom_errorbarh() +
#voeg punten toe (x vs y, kleur volgens jaartal)
  geom_point() +
  xlab("Effect van het lot (in m²/ha)")
```



Figuur 3.17: Grafische voorstelling van de extreemste lot effecten uit de fictieve dataset na correctie van het globale effect en het effect per koper.

4 Referenties

Agresti, Alan. 2002. *Categorical Data Analysis*. Hoboken, New Jersey: John Wiley; Sons.
Fay, M. P. 2010. "Two-sided Exact Tests and Matching Confidence Intervals for Discrete Data." *R Journal* 2 (1): 53–58.
R Core Team. 2013. "R: A Language and Environment for Statistical Computing." Vienna, Austria: R Foundation for Statistical Computing. http://www.r-project.org/.
Zhang, Ed, Vicky Qian Wu, Shein-Chung Chow, and Harry G. Zhang. 2013. "TrialSize: R Functions in Chapter 3,4,6,7,9,10,11,12,14,15."