Det naturvidenskabelige fakultet
Københavns universitet

Reproduktion af hanmink

Anton Rask Lundborg Hansen

Vejleder: Anders Tolver
9. Juni 2017

Bachelorprojekt i statistik
Institut for Matematiske Fag, Københavns Universitet

Bachelor thesis in statistics
Department of Mathematical Sciences, University of Copenhagen

Abstract

This paper studies the number of kits produced in a mating season by male mink and various biological and reproductive properties of the mink. A dataset collected in a controlled study of 482 male mink is considered and trimmed with the aim of doing a complete case analysis. The study uses repeated 10-fold cross-validation to estimate prediction errors in the considered models. A number of tree-based and linear regression methods are used to construct models of the number of kits produced. A random forest model results in an estimated RMSE of 16.85 and a boosted tree model results in an estimated RMSE of 16.84. An ordinary least squares model selected with best subset selection results in 4 predictors out of seven being selected and an estimated RMSE of 16.77. Using an elastic net penalty on all seven predictors the fit is improved to result in an RMSE of 16.73. Using a model that estimates the mean results in an estimated RMSE of 17.2. Using 1000 repeats of 10-fold crossvalidation, it is reaffirmed that the linear models perform the best although the improvement from the mean model is small. All the models appear to suggest that the size of the baculum, the colour of the mink and the size of the testes are the primary predictors of interest while the physical size and asymmetry of the testes are less relevant. The lack of predictive power is assumed to have been caused by correlated predictors and a noisy relationship between predictors and response. Further studies that record properties of the female mink and/or that model non-linear relationships between predictors and response are needed to provide better models.

Indhold

1 Indledning

Mink er Danmarks tredje største animalske eksportprodukt og minkindustrien beskæftiger over 6000 personer[8] [1]. Danmark har verdens førende minkproduktion med 17.8 millioner minkskind produceret årligt, og den danske forskning på området er ligeledes på førende på internationalt plan. Forskningen er med til at sikre den bedste velfærd for dyrene, mens de er levende, og ligeledes sikre det bedst mulige produkt for industrien.

I dette projekt betragtes et datasæt bestående af data fra et reproduktionsforsøg med fokus på hanminkenes rolle i processen. Det fødte antal hvalpe for hver hanmink vil modelleres i et forsøg på først og fremmest at opnå størst prediktiv kraft, dvs. at finde en model, der kan forudsige hvilke hanmink, der er bedst at avle på, hvis man søger at maksimere antallet af hvalpe. Udover den rene prediktive kraft, vil der lægges vægt på hvilke variable hver model vælger til at være mest signifikante til at prediktere. Denne modellering vil først ske med en række træbaserede metoder i et forsøg på at maksimere prediktiv kraft. Herefter vil mere fortolkelige lineære modeller tages i brug med forskellige metoder til variabelselektion. Til sidst vil de udvalgte modeller sammenlignes både på prediktiv kraft og signifikans af variable.

2 Data

2.1 Præsentation

Det betragtede datasæt er stillet til rådighed af lektor og dyrlæge Anne Sofie Vedsted Hammer ved Institut for Veterinær sygdomsbiologi på Københavns Universitet. Datasættet stammer fra en undersøgelse af hanmink på en forsøgsfarm i Holstebro i marts 2013 og 2014. Undersøgelsen omfatter 482 hanmink, hvorom der er blevet noteret en række generelle biologiske variable og information om minkenes reproduktionsorganer. Hanminkene er alle ca. 1 år gamle på tidspunktet for notering af disse variable, da de alle er blevet aflivet efter parring.

Udover disse rent biologiske faktorer, er desuden blevet noteret resultatet af et kontrolleret reproduktionsforsøg, der er foretaget på forsøgsfarmen. Når hannerne skulle parres på forsøgsfarmen, blev de sat sammen med en tæve ad gangen, og det blev noteret om parringen blev gennemført eller ej. Denne procedure blev fortsat, indtil hannen ikke længere ville parre sig. Undersøgelsens primære formål var at afklare, om der er sammenhæng mellem størelsen på hanminkenes penisknogle og testes, hanminkenes farve og hanminkenes reproduktionsresultater. Variablene i datasættet kan ses opsummeret i Tabel 1.

Variabel	Forklaring	Antal obs.
KF	Et journalnummer for hver af de involverede mink	482
PKL	Penisknoglens længde i millimeter	459
TV1	Den ene testikels vægt i gram	480
TV2	Den anden testikels vægt i gram	467
TL1	Den ene testikels længde i millimeter	480
TL2	Den anden testikels længde i millimeter	458
PV	Parringsvillighed (0: 0 parrede tæver, 1: 1-3 2: >3)	480
F	Minkens farve (enten sort eller brun)	482
KV	Kropsvægten målt postmortem i gram	304
SV	Kropsvægten målt ved sortering i gram	191
KL	Kropslængden målt fra næse til anus i centimeter	132
FFF	Om minken var frosset før fiksering (0: nej, 1: ja)	482
K	Antallet af kuld	480
GT	Antallet af golde tæver (dvs. parrede tæver uden afkom)	480
PT	Antallet af parrede tæver	480
HF	Antallet af fødte hvalpe	480

Tabel 1: Variablene i datasættet `reproduktionsdata` med forklaringer samt hvor mange observationer af hver variabel, der er til stede i datasættet.

3 Forstudier

I dette afsnit redegøres for databehandlingen af den præsenterede data før de statistiske analyser i de følgende afsnit. Derudover inkluderes et naivt plot og formodninger, der kan have relevans for det overordnede formål.

3.1 Beskæring og behandling af data

I et forsøg på at modellere sammenhænge senere vil det være nødvendigt at beskære i datasættet. 2 af minkene i datasættet var ikke brugt til avl, så de har altså ingen observationer om reproduktionsresultaterne. For alle mink skulle gerne gælde at $\mathbf{K} = \mathbf{PT} - \mathbf{GT}$. Dette er ikke tilfældet for to af observationerne. Ingen af disse betragtes yderligere i analysen.

For nogle mink er der ikke blevet målt en penisknogle. Dette skyldtes ofte, at knoglen var knækket, og derfor ikke kunne gives et fornuftigt mål. Det er på ingen måde sikkert, at det er tilfældigt, at det netop var disse mink, der havde knækkede penisknogler, og ved at undlade dem fra analysen kunne man risikere at misse en sammenhænge mellem svage penisknogler og eventuelle responsvariable. For ikke at komplicere analysen yderligere, vil vi dog antage, at knoglerne er knækket tilfældigt, og blot udelade alle mink uden penisknoglemål fra analysen. Vi vil ved samme ræsonnement udelade alle mink, hvorom der mangler længde- eller vægtmål på begge testes. Denne udelukkelse inkluderer også testes, hvorom der er noteret en længde eller vægt på 0, da dette må betragtes som manglende måling. Antallet af målinger af **KL** er så lavt, at variablen ikke vil inkluderes i de senere analyser. Variablen **FFF** blev primært noteret af obduktionsmæssige årsager

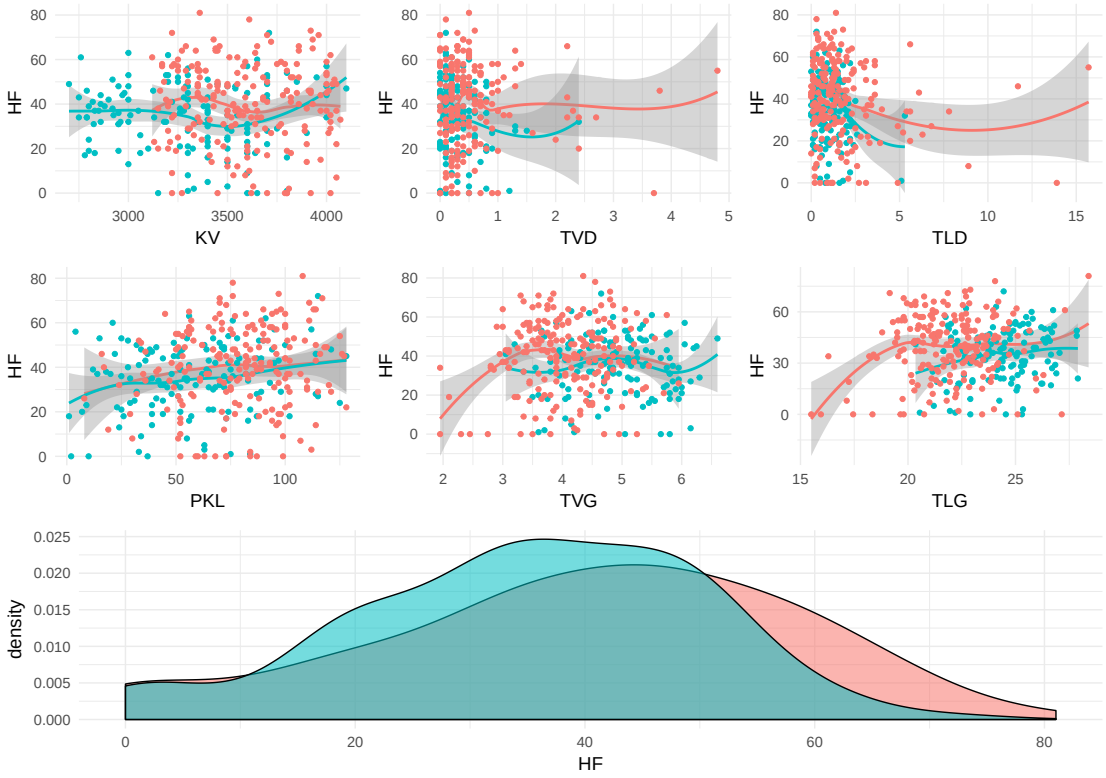
og inkluderes heller ikke i analyserne.

Ved sammenligning af **KV** og **SV** bemærkes, at kun 3 af observationerne ikke stemmer overens med hinanden. I et forsøg på at få en variabel med vægte, imputeres **KV** med sorteringsvægte alle de steder, hvor der ikke allerede er en måling. Vi vælger altså at stole mest på den seneste måling af kropsvægt. Ligesom med penisknoglen og testes, vil vi udelukke mink, hvorom der mangler data om kropsvægt (dvs. der hverken er en sorteringsvægt eller en angivet kropsvægt) Dette efterlader et datasæt på 337 hanmink, hvilket er en reduktion på ca. 30% i forhold til det komplette datasæt. Dette er indenfor rimelighedens grænser, men det er ikke uden potentielle problemer. Vi vil ikke yderligere problematisere denne proces, men blot være opmærksomme på analysens begrænsinger.

Da nummeringen af testes ikke har nogen signifikant betydning, kan det være en fordel at transformere variablene til nogle variable, der er lettere at fortolke. Vi introducerer i stedet gennemsnittet af testikelvægt, **TVG**, og gennemsnittet af testikellængde, **TLG**, for hver mink. Desuden udregnes den absolutte difference af testiklernes vægt, **TVD**, og af testiklernes længde, **TLD**. Da der udelukkende arbejdes med mink, hvorom der er data om begge testes, vil disse være veldefinerede i alle tilfælde. Ved at bruge disse variable istedet for den arbitrære nummerering, bliver det lettere at forstå påvirkningen fra de forskellige variable i modelleringsprocessen.

3.2 Indledende modelleringsovervejelser

Man kunne overveje hvilke variable, der er interessante at medtage som responsvariabel i modelleringsprocessen. Minkproducenter der søger at maksimere profit er med stor sandsynlighed ikke kun interesseret i antallet af minke, men også størrelsen og kvaliteten af pelsen. Hvis profitmaksimering er målet, er det derfor naturligt at forsøge at modellere antallet af fødte hvalpe, da vi ikke har information om andre faktorer. Modelleringsprocessen vil altså gå ud på at forklare antallet af fødte minkhvalpe udfra farven på minken, længden på penisknoglen, kropsvægten, den gennemsnitlige testikelvægt og -længde samt differencen mellem testiklernes vægt og længde. For at opnå intuition om data, vil kort reflekteres over Figur 1.



Figur 1: Plot af variable, hvor de grønne farver repræsenterer sorte mink og røde farver repræsenterer brune mink. De første to rækker er plots af de forskellige forklarende variable mod responsvariablen; antal fødte hvalpe. Den tredje række er et ikke-parametrisk densitetsplot over fødte hvalpe.

Det mest iøjenfaldende ved disse plots, er den store spredning i datasættet. Samtlige plots tyder på lav prediktiv værdi ved univariate analyser. Det lader til at farven på minken påvirker mange af de biologiske parametre. Det ser ud som om at sorte mink er mindre og har mindre penisknogler, men derimod har større testikler målt både på længde og vægt. Korrelationen mellem farve og andre variable kan forstyrre prediktionerne i mange klasser af modeller, og er derfor vigtig at være opmærksom på. Det er desuden også klart, at der er stor korrelation mellem testikelvægt og -længde, både som difference og gennemsnit. Modeller, der antager ukorrelerede prediktorer, ville potentielt have besvær med at adskille støj fra sammenhæng, hvis både længde og vægt inkluderes. Det er ikke kun de simple korrelationer mellem par af variable, men man kunne måske forestille sig at tre eller flere variable var korrellerede, hvilket øger mængden af redundant information yderligere.

4 Analyse

4.1 Metode og data

I den resterende del af rapporten vil udelukkende forsøges at modellere antallet af fødte hvalpe i det beskårne datasæt. Vi vil nøjes med at betragte compzlete cases, altså kun mink hvor der er vidne om penisknoglens længde, begge testiklers længde og vægt, farven samt kropsvægten (totalt 337 observationer). Den prediktive modelleringsproces vil tage udgangspunkt i træ-baserede metoder og brug af cross-valdiation. Resultaterne fra disse modeller vil sammenlignes med ordinære og penaliserede lineære regressionsmodeller. Til sidst vil den prediktive kræft af modellerne sammenlignes ved cross-validation.

4.1.1 Cross-validation

I dette projekt vil cross-validation anvendes flere gange, da dette givet estimat for prediktionsfejlen på fremtidig data, og dermed giver en måde at udføre modelselektion. I statistisk kontekst skelner man mellem fejlen på træningsdata og på testdata. Fejlen på træningsdata er den fejl man har på det datasæt man fitter sin model på, hvorimod fejlen på testdata er fejlen på en ny observation. For kvantitative data vil fejlen typisk opgøres som root-mean-squared-error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}$$

Enkelte foretrækker at opgive MSE, altså RMSE^2, hvor minimering af de to oplagt er ækvivalent. Selvom fejlen på træningsdata ikke er uvæsentlig, vil dens minimering ikke nødvendigvis føre til en bedre model, da man risikere at modellere den iboende støj i datasættet. Et langt bedre mål for en models prediktive kraft ville være minimering af fejlen på testdata, men desværre har man sjældent nye observationer til rådighed til at estimere denne fejl.

En måde at estimere fejlen på testdata er ved Leave-One-Out-Cross-Validation (LOOCV). For hver observation fittes den betragtede model på de resterende data og RMSE på observationen noteres. Gennemsnittet af dette for alle observationer er estimatet for RMSE. For store n og komplicerede modeller, kan dette være svært at udføre i praksis. Denne metode kan generaliseres til, i stedet for at inddele datasættet i n dele, at inddele datasættet i $k < n$ dele og gennemføre proceduren fra før. Dette kaldes k-fold cross-validation, hvor LOOCV er specialtilfældet $k = n$. Data inddeles i k ca. lige store folder tilfældigt. For hver fold, fittes den betragtede model på de resterende folder og RMSE for den fold, der er holdt udenfor, estimeres. Gennemsnittet af de k estimater for RMSE'en bliver procedurens RMSE estimat.

For $k < n$ er metoden altså ikke længere deterministisk, og der introduceres variation i estimatet for fejlen på testdata. Ved at udregne RMSE estimatet for forskellige inddelinger af datasættet vha. k-fold cross-validation og tage gennemsnittet af dem, fås et mere robust estimat. Denne procedure kaldes repeated k-fold cross-validation. Det er klart, at for større værdier af k vil metoden have mindre bias, da hver model bruger en større mængde data. Dog vil store værdier af k resultere i større varians, da de fittede

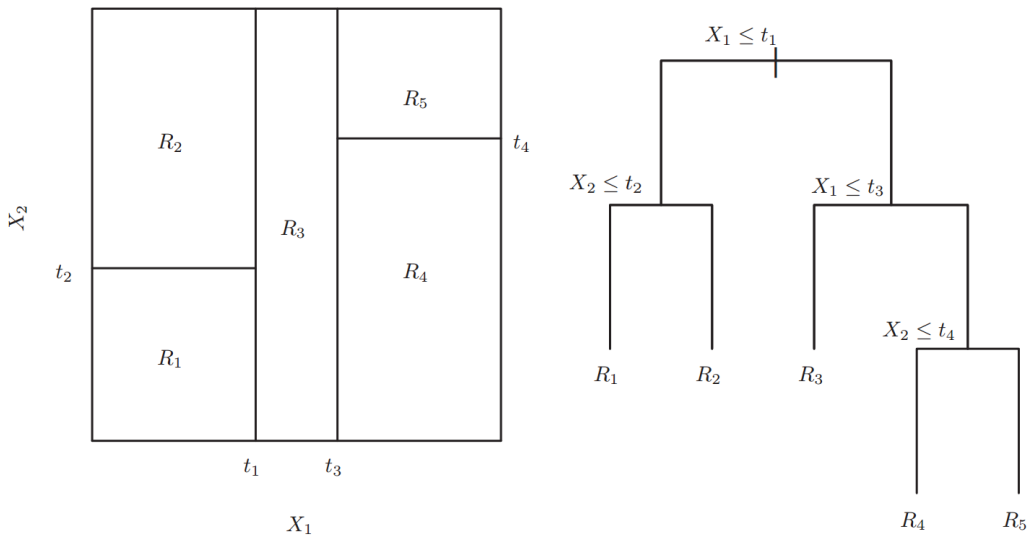
modeller vil være korrelerede, da de fittes på samme data punkter. Derfor bruges typisk $k = 5$ eller $k = 10$ da empiriske undersøgelser har vist at disse er optimale for samspillet mellem bias og varians [5].

Da cross-validation har et stokastisk element i form af inddelingen af data i folder, vil hver fold i hver gentagelse af repeated k-fold cross-validation have et forskelligt estimat for RMSE'en. Det er naturligt at overveje stabiliteten af dette estimat ved at udregne den empiriske spredning af estimaterne. Man kan ydermere udregne standard erroren på sædvanligvis ved at dividere med kvadratroden af antallet af fittede modeller (dvs. k gange antallet af repeats). I et forsøg på at undgå overfitting, defineres "one-standard-error" reglen. Når modelselektion gennemføres ved cross-validation, vælges den simpleste model, der ligger indenfor en standard error af den bedste model. Simpel kan både betyde nemmest at fitte komputationelt eller simplest rent matematisk f.eks. ved at inkludere færre parametre. Dette princip søger at sikre, at vi får den simpleste model samtidigt med, at vi får stærk prediktiv kraft og undgår overfitting [7].

4.2 Random Forest model

4.2.1 Træbaseret regression og random forest

En måde at foretage regression på et datasæt er ved hjælp af regressionstræer. Ideen bag et regressionstræ er at inddele prediktorrummet (hvis vi har prediktorerne $X_1, \ldots, X_p$ er prediktorrummet $X_1 \times \cdots \times X_p$) i J disjunkte mængder $R_1, \ldots, R_J$ således, at vi for hver observation i R_j blot predikterer gennemsnittet af de observerede værdier i R_j.



Figur 2: Til venstre ses en inddeling af et prediktorrum med to kontinuerte prediktorer X_1 og X_2. Til højre ses en træbaseret visualisering af inddelingen [4].

Når et regressionstræ konstrueres vil man kun betragte rektangulære mængder R_j i et forsøg på at give tilpas simple fortolkninger og beregninger. Man forsøger altså at bestemme kasser $R_1, \ldots, R_J$, der minimerer RMSE

$$\text{RMSE} = \sqrt{\sum_{j=i}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2}$$

I praksis kan det ikke lade sig gøre at vurdere RMSE for alle rektangulære inddelinger af prediktorrummet, så istedet bruges en algoritme kaldet recursive binary splitting [4]. Ideen bag denne algoritme er for hver prediktor at inddele de mulige værdier denne kan antage i to, således at RSS minimeres. Dette gøres for samtlige prediktorer og den prediktor der minimerer RMSE mest muligt udvælges som det først split. Dette gentages så for hver af de to fundne inddelinger, og en af dem deles i to. Processen fortsætter på de tre inddelinger, og gentages indtil et stopkriterie er opfyldt. Stopkriteriet er som regel, at hver inddeling har et minimum eller maksimum af observationer eller at RMSE ikke kan reduceres yderligere. For regressionstræer vælges som regel at hver kasse skal indeholde 5 observationer. Antallet af inddelinger kaldes træets dybde. Denne måde at danne træer på giver modeller, der er lette at fortolke og forklare til folk uden kendskab til statistik, men deres prediktive kraft er sjældent stor. Der er stort potentiale for overfitting og træerne er meget sensitive overfor ændringer i træningsdata [4] [7]. Random forest modeller bruger en kombination af bootstrapping og regressiontræer til at opnå langt kraftigere modeller.

Når en random forest model fittes til data, genereres først B datasæt fra det oprindelige datasæt ved at sample med tilbagelægning (alle af samme størrelse som det oprindelige datasæt). For hvert datasæt udvælges et antal, m, af prediktorne $X_1, \ldots, X_p$ tilfældigt (typisk $p/3$). Et regressiontræ fittes til hver af de B datasæt, dog kun med de udvalgte prediktorer, og modellens prediktion for en observation er blot gennemsnittet af prediktionerne i hver af de B træer. Ved kun at fitte på en delmængde af prediktorerne, kan man undgå, at en stærk prediktor overskygger mindre stærke prediktorer, da en række af træerne vil fittes uden den stærke prediktor [4]. Man kan bruge cross-validation til at optimere på m, således at man får den model med størst prediktiv kraft. Når B bliver tilpas stor vil RMSE ikke øges yderligere, men der er sjældent fare for at B bliver for stor.

Der findes en meget naturlig måde at vurdere prediktionsfejlen i random forest modeller ved at betragte out-of-bag erroren (OOB). For hvert af de B datasæt genereret ved bootstrapping, vil man forvente at ca. 2/3 af observationerne anvendes. OOB-fejlen for en observation beregnes ved at tage gennemsnittet af prediktionsfejlen for obserationen i de modeller genereret fra et datasæt uden den betragtede observation. Prediktionsfejlen på fremtidig data kan dermed estimeres ved at tage gennemsnittet af OOB-fejlene for hver observation. I dette projekt vil dog primært arbejdes med repeated 10-fold cross-validation, da dette gør det muligt at sammenligne prediktionsfejl på tværs af modelklasser. Desuden viser nyere simulationsstudier at repeated 10-fold cross-validation er et bedre estimat for den sande RMSE end OOB estimatet [6]. Vi vælger i stedet at betragte out-of-fold (OOF) prediktionen for en given observation. OOF-prediktionen er blot gennemsnittet af den betragtede observations estimat i de modeller, hvor observationen ikke er brugt til at træne modellen. For repeated 10-fold cross-validation med r repeats, vil der være r modeller, hvor en observation ikke er brugt til at træne modellen. Man kan ligeledes

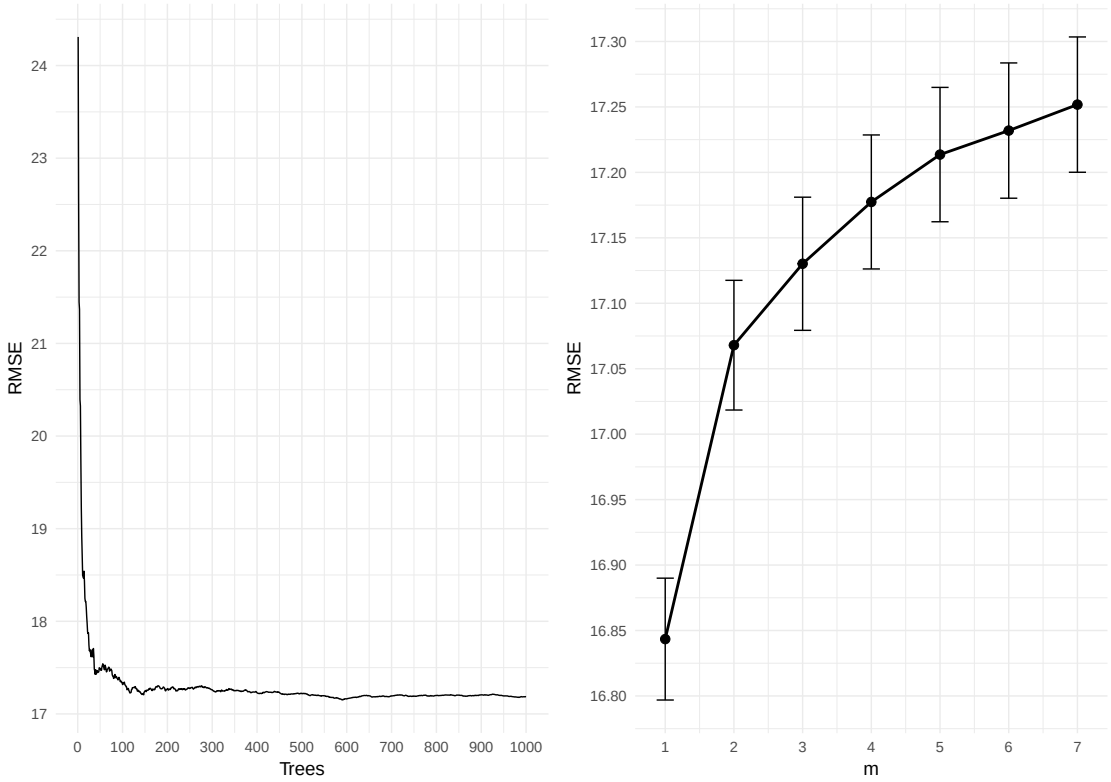
udregne spredningen af prediktionerne på tværs af gentagelserne og udregne standard erroren ved at dividere spredningen med kvadratroden af antallet af repetitioner. Hvis modellen havde stor prediktiv kraft, ville man forvente at OOF prediktionerne var tæt på de observerede værdier, så et plot af observerede værdier mod OOF-prediktioner, vil give et indblik i modellens styrke. Ved at inkludere errorbars fås desuden et indblik i modellens sensitivitet overfor ny data.

Random forest modeller kan imodsætning til regressionstræer ikke repræsenteres med et enkelt træ, men det er stadig muligt at afgøre hvilke prediktorer, der har størst betydning for modellen. En måde at vurdere dette er ved at permutere samtlige værdier af en prediktor tilfældigt og se på ændringen i RMSE før og efter permutation. Det er klart, at hvis permutationen af prediktoren ikke har stor effekt på den prediktive værdi, må prediktoren være uvæsentlig og vice versa. En anden måde at vurdere vigtigheden af prediktorer er ved at betragte faldet i RMSE hver gang en prediktor bruges til at inddele prediktorrummet i et træ. Ved at tage gennemsnittet af dem over de træer hvor prediktoren indgår, kan prediktorens vigtighed vurderes. Dette mål har dog en tendens til at være biased, særligt for kategoriske variable, så vi nøjes med at anvende det første [7]. Da målene afhænger af responsvariablens måleskala, vil målet skaleres så den mindst relevante prediktor får vigtighed 0 og den mest relevante får vigtighed 100.

4.2.2 Random forest model for fødte hvalpe

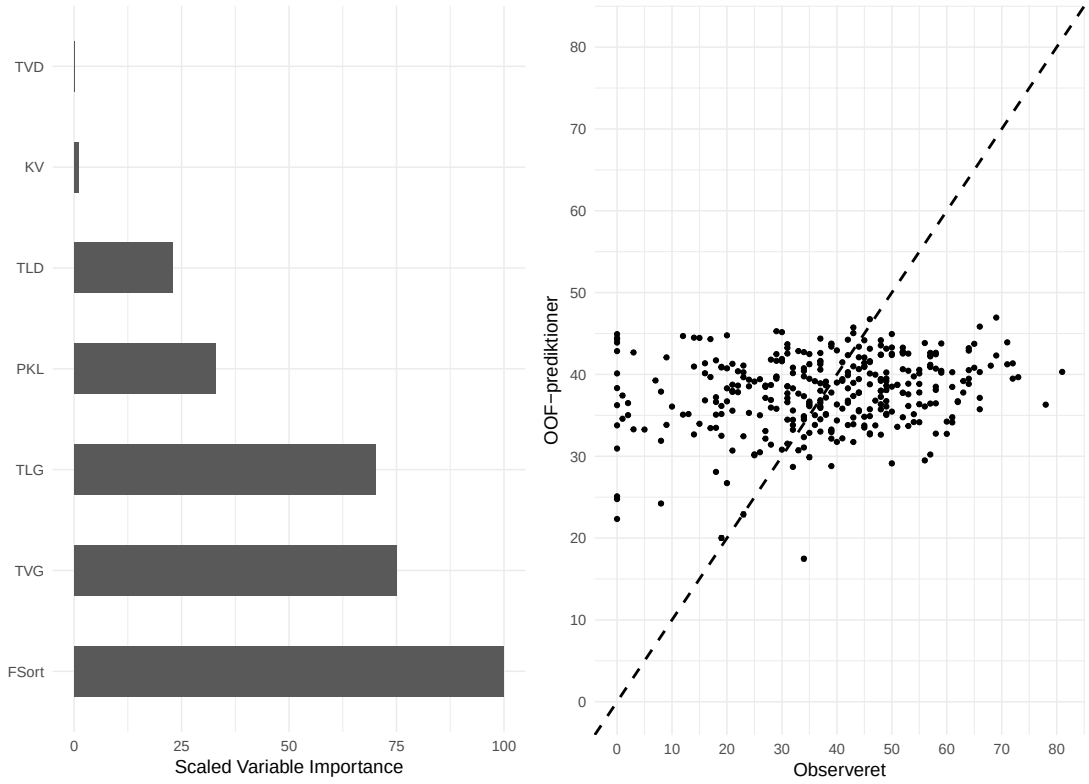
Random forest modeller er implementeret i R-pakken `randomForest`. Først fittes en random forest model til data med et stort antal bootstrap samples for at estimere hvor mange træer er nødvendige for at opnå bedst mulige prediktioner. Ved at plotte udviklingen af RMSE som funktion af antallet af træer, kan antallet af træer bestemmes. Da der er 7 prediktorer, sættes $m = 2 \approx p/3$.

Det ses på figur 3, at RMSE'en er stabil efter 300 træer. For at bestemme den bedste værdi af m i random forest algoritmen bruges `caret` pakken i R, til at foretage repeated 10-fold cross-validation med 100 repeats. Cross-validation bruges altså til at give et estimat for RMSE'en for nye datapunkter for random forest modeller for data med m mellem 1 og 7. Hver random forest model bruger 300 bootstrap samples til at generere træer. One-standard-error-reglen bruges til model selektion.



Figur 3: Venstre: Plot over RMSE som funktion af antallet af træer i en random forest model for data med $m = 2$.
Højre: Plot over RMSE i random forest modellen som funktion af m med errorbars af længde to standard error, estimeret ved repeated 10-fold cross-validation med 100 repeats.

På figur 3 ses, at $m = 1$ giver den model med stærkest prediktiv kraft med en RMSE på 16.85. Det ses ydermere, at det er den simpleste model og den eneste der ligger indenfor en standard error, så denne udvælges som den bedste model. En RMSE på ca. 17 er enormt stor når antallet af hvalpe er mellem 0 og ca. 80. Dette er altså på ingen måde en stærk model. Problemet illustreres yderligere ved at plotte out-of-fold prediktionerne mod de observerede værdier som set på figur 4. Out-of-fold prediktionerne har errorbars af længde 2 standard error, så prediktionerne er ganske stabile. Dette bekræftes yderligere af at standard erroren i gennemsnit er 0.124 og den største standard error er 0.279. Det ses dog også at modellen predikterer ekstremt ringe langt fra middelværdien af antal fødte hvalpe.



Figur 4: Venstre: Mål for vigtigheden af prediktorerne i random forest modellen med $m = 1$ og 300 træer.
Højre: Out-of-fold prediktioner for 100 repeats af 10-fold cross-validation plottet mod observerede værdier med error bars af længde 2 standard error.

Vi kan stadig få en ide om hvilke prediktorer, der bidrager mest til modellens præcision ved at plotte vigtigheden af variablene i random forest modellen med $m = 1$ og 300 træer. Det ses på figur 4, at farven på minken er den bedste prediktor, hvilket ikke er overraskende, da farven indeholder information om mange af de resterende prediktorer. Random forest modeller er designet til at kunne håndtere korrelerede prediktorer, så det er ikke overraskende at testikelvægt og -længde bidrager i lignende omfang både når det kommer til gennemsnitlig størrelse. Om den ringe prediktive kraft skyldes en begrænsning i modellen eller data, vil først kunne afgøres efter yderligere analyser.

4.3 Gradient tree boosting

4.3.1 Boosting af træer

Et alternativ til random forest modeller, der også forsøger at forbedre modelleringsevnerne ved regressiontræer er boostede træer. Boosting er en generel teknik, der blev udviklet til at kombinere et stort antal modeller med svag prediktiv kraft(weak learners) til en stor model med (potentielt) stærk prediktiv kraft (strong learner). Regressionstræer er en oplagt kandidat til at blive brugt som weak learner, da det er muligt at begrænse træets kompleksitet ved blot at begrænse antallet af splits i træet. Det er desuden nemt at kombinere prediktionerne fra flere træer og at generere træerne komputationelt. Et simpelt eksempel på en model vha. boostede træer, er Simple Gradient Boosting [7]. I denne algoritme vælges først et fast antal af totale træer, K, og en maksimal træ dybde, D og

et minimum antal observationer i hvert træ split, M. Hver observations prediktion initialiseres til gennemsnittet af responsvariablen og residualerne beregnes. Et regressiontræ med dybde D (under restriktionen at hvert split skal indeholde mindst M observationer) fittes til residualerne og prediktionerne opdateres ved at lægge træets prediktion til observationens prediktion. Residualerne opdateres med den nye prediktion og proceduren gentages K gange.

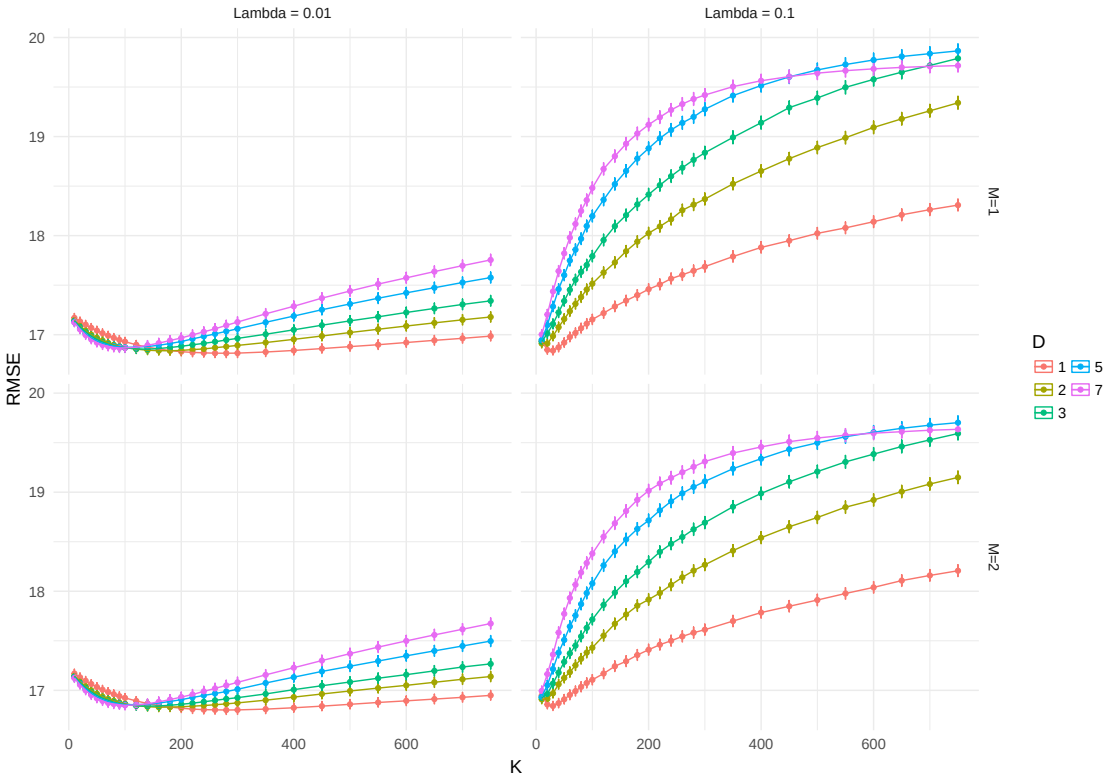
Typiske parametervalg er $M = 10$ og $D = 1$, da disse sikrer at hvert træ har tilpas svag prediktivkraft. Der er intet fast fornuftigt valg af K og den bestemmes ved cross-validation. I praksis kan man også bruge cross-validation til at vælge optimale værdier af M og D. Simple Gradient Boosting vælger den optimale reduktion i residualerne i hver iteration af proceduren; algoritmen er altså greedy. Dette kunne føre til overfitting af data, selv med tilpas lille valg af D. Algoritmen modificeres ved i stedet for at lægge den predikterede værdi fra det fittede træ i hvert trin til, lægges λ gange prediktionen til i hvert trin, hvor $\lambda \in (0, 1)$ og kaldes indlæringsraten. Dette sikrer at algoritmen lærer tilpas langsomt til at begrænse risikoen for overfitting. Typiske valg af indlæringsraten er 0.1 eller 0.01, dog vil lavere værdier af λ lede til et behov for større værdier af K og derfor længere komputationstider. Cross-validation er endnu engang et godt bud på at vælge den bedst mulig værdi af λ.

En sidste modifikation fører til Stochastic Gradient Boosting. I stedet for at betragte hele datasættet når hvert træ fittes, betragtes kun en delmængde af datasættet, samplet med tilbagelægning. Empiriske studier viser at sampling af et datasæt af halvt så stor størrelse som det oprindelige giver optimale resultater [7]. Da hvert enkelt træ er meget sensitivt overfor de værdier det fittes på, vil man ved kun at betragte en delmængde af data mange gange, kunne reducere støjen i modelleringsprocessen og få bedre estimater. Selvom både boostede træer og random forest modeller er baseret på en kombination af regressionstræer, er metoderne ganske forskellige. I random forest modeller er træerne uafhængige og bidrager allesammen ligeligt til modellens prediktion hvorimod i boostede modeller er træerne afhængige og bidrager forskelligt til modellens endelige prediktion.

Ligesom for random forest modeller, findes mål for variables vigtighed i boostede træer. Reduktionen i RMSE summeres for hver variabel over samtlige fittede træer. Dette mål skaleres så den mindst brugte prediktor får værdien 0 og den mest brugte får værdien 100, da ændring i RMSE er skalaafhængigt og ofte ufortolkeligt alene. Det er vigtigt at bemærke, at dette mål i større grad vil fokusere på få stærke prediktorer end målet for random forests, da disse dekorellerer prediktorerne ved kun at fitte på en delmængde af dem ad gangen.

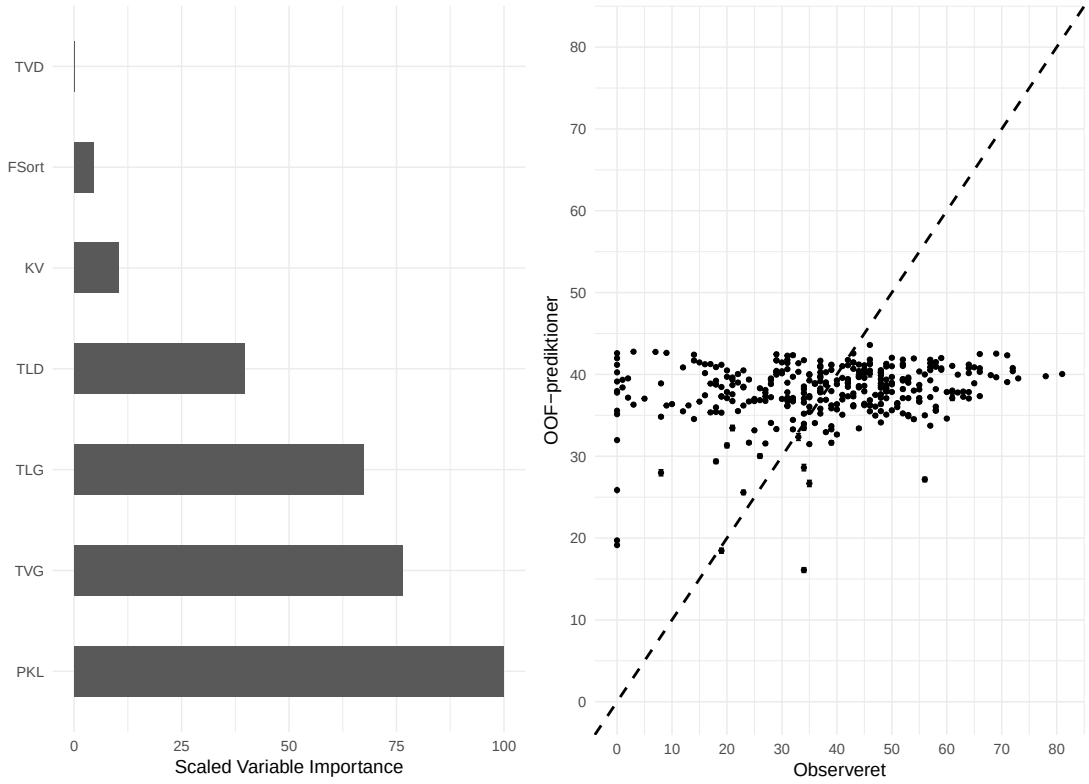
4.3.2 Boosted tree model for fødte hvalpe

Stochastic Gradient Boosting er implementeret i R-pakken `gbm`. Ved at bruge `caret`-pakken i R, foretages 10-fold cross-validation med 100 repeats i et forsøg på at vælge de optimale værdier af K, M, D og λ. K optimeres fra 20 til 750 træer, M optimeres fra 1 til 10 minimum observationer, D optimeres fra 1 til 7 splits og λ udvælges blandt 0.1 og 0.01. Modelselektion sker med one-standard-error reglen, dvs. den selekterede model, er den simpleste model indenfor en standard error af den model med lavest RMSE. Herunder på figur 5 ses en afbildning af RMSE'en for værdierne $M = 1$ og $M = 2$, de resterende plots kan ses i appendix A.



Figur 5: Plot over RMSE som funktion af antal træer (K) for $M = 1$ og $M = 2$, samt forskellige værdier af D og λ med error bars af længde to standard error, estimeret ved repeated 10-fold cross-validation med 100 repeats.

Der ses en generel tendens til at lavere værdier af D giver en bedre model på tværs af alle værdier af λ og M. Generelt lader en lavere værdi af λ til at have fladere kurver, men ca. samme minimum som den større værdi af λ. Modellen med lavest RMSE har $\lambda = 0.01$, $K = 280$, $M = 2$ og $D = 1$ med en RMSE på 16.801 og en standard error på 0.045. Den simpleste model indenfor en standard error af den bedste har $\lambda = 0.1$, $K = 30$, $M = 1$ og $D = 1$ med en RMSE på 16.840. Ligesom med random forest modellerne, er den prediktive kraft ikke specielt imponerende for nogen af modellerne. Modellen udvælges som den bedste model og vi kan plotte den skalerede variabelvigtighed samt OOF-prediktionerne for modellen med tilhørende error bars af længde 2 standard error som set på figur 6.



Figur 6: Venstre: Mål for vigtigheden af prediktorerne i boosted tree modellen med $\lambda = 0.1$, $M = 1$, $K = 30$ og $D = 1$.
Højre: Out-of-fold prediktioner for 100 repeats af 10-fold cross-validation plottet mod observerede værdier med error bars af længde 2 standard error.

Prediktionerne ligner utrolig meget dem fra random forest modellen, idet at den prediktive kraft er ekstremt ringe langt fra middelværdien. Prediktionerne er dog stabile, errorbarene er små og den gennemsnitlige standard error er på 0.137 med en maskimalværdi på 0.419. Vi kan se at penisknoglelængden vurderes som den vigtigste prediktor sammen med gennemsnitlig længde og vægt på testes hvorimod kropsvægt, testikelasymmetri og farven udvælges i markant mindre grad.

Selvom størrelsen på testes var med i både random forest modellen og denne model, er det interessant at penisknoglelængde og farven indgår i så forskelligt et omfang. Da random forest modeller dekorrellerer variablene der indgår, kunne man overveje om det var et udtryk for korrelation mellem penisknoglelængden og farven på minkene. Dette kunne forklare hvorfor at de boostede træer ville udføre splits på penisknoglelængden langt oftere, da denne er numerisk og derfor lettere at inddele, hvis farve og penisknoglelængde indeholder redundant information. Random forest modellen har slet ikke mulighed for at gennemføre splits på penisknoglelængden i mange tilfælde og farven bliver derfor brugt oftere, da den fælles information tydeligvis bidrager til prediktionsevnen. Vi vil senere redegøre klarere for omfanget af korrelationen mellem prediktorerne.

Det er efterhånden klart at de træbaserede metoder ikke kan prediktere effektivt på dette datasæt. Da disse modeller ofte anvendes grundet deres store fleksibilitet, er det efterhånden sandsynligt at data simpelthen ikke tillader bedre prediktioner.

4.4 Lineære regressionsmodeller

4.4.1 OLS linear regression

En af de ældste og mest grundlæggende statistiske metoder er lineær regression. Vi forestiller os en situation med en N-dimensionel vektor Y af responsvariable og p N-dimensionelle prediktorvektorer $X_1, \ldots X_p$. Prediktorerne placeres som søjler i en $N \times (p+1)$ matrix, X, hvor den første søjle fyldes med 1-taller for at repræsentere en intercept parameter. Vi søger en estimator, f, der vha. X kan give et estimat for Y og i konteksten er lineær regression vil f have formen

$$f(X) = X\beta$$

for en $p+1$-dimensionel vektor af parametre $\beta = (\beta_0, \beta_1, \ldots, \beta_p)$. Den kanoniske estimationsprocedure forsøger at minimere de kvadrerede residualer (residual sum of squares) altså

$$\mathrm{RSS}(\beta) = (Y - X\beta)^T(Y - X\beta)$$

Denne metode kaldes ordinary least squares, da de kvadrerede residualer kan betragtes som summen af arealet af en række kvadrater i tilfældet $p = 1$ [3]. RSS er en kvadratisk funktion i de p parametre og under antagelsen om at X har fuld rang, vil minimaet være entydigt givet ved

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

I tilfældet hvor X ikke har fuld rang, vil $\hat{\beta}$ ikke længere være entydig. Ofte findes måder at droppe en eller flere søjler i X så repræsentationen bliver entydig, hvilket ofte er implementeret i software, der fitter lineære modeller. I praksis kan dette komme til udtryk hvis en eller flere prediktorer er korrellerede, hvilket ofte resulterer i modeller med ringe fortolkning. Hvis man ønsker at udlede mere generelle resultater om opførslen af estimatoren $\hat{\beta}$ og de estimerede data $\hat{Y}$, vil man være nødsaget til at gøre flere antagelser om den sande fordeling af data. Hvis man antager at den sande sammenhæng mellem prediktorerne og responsen kan beskrives ved

$$Y = f(X) + \varepsilon = X\beta + \varepsilon$$

hvor $\mathrm{E}(\varepsilon \mid X) = 0$ og at X har rang $p+1$, kan man vise at $\hat{\beta}$ er unbiased. Der gælder at

$$\begin{aligned}\hat{\beta} &= (X^T X)^{-1} X^T Y = (X^T X)^{-1} X^T (X\beta + \varepsilon)\varepsilon \\ &= (X^T X)^{-1} X^T X\beta + (X^T X)^{-1} X^T = \beta + (X^T X)^{-1} X^T \varepsilon\end{aligned}$$

Hvis vi tager expectationen af begge sider fås

$$\mathrm{E}(\hat{\beta}) = \mathrm{E}(\beta + (X^T X)^{-1} X^T \varepsilon) = \beta$$

da vi har antaget at $\mathrm{E}(\varepsilon \mid X) = 0$. Hvis man ydermere antager at $\mathrm{Var}(\varepsilon \mid X) = \sigma^2 I$, hvor I er $N \times N$ identitetsmatricen og $\sigma^2 > 0$ kan det vises at

$$\mathrm{Var}(\hat{\beta}) = (X^T X)^{-1}\sigma^2$$

Under disse antagelser gælder Gauss-Markov sætningen, der siger at OLS estimatoren er BLUE (best linear unbiased estimator) [3]. Dette lyder umiddelbart som en god ting, men lad os se hvad det betyder for fejlen på en ny observation ved at bestemme den forventede MSE. Vi har samme antagelser som i Gauss-Markov, dog kan den sande sammenhæng mellem Y og X, f, have en vilkårlig form. Situationen er altså at vi har estimeret denne form $\hat{f}$ og ser på fejlen for en ny observation y. Vi får

$$\mathrm{E}(y - \hat{f})^2 = \mathrm{E}(y^2) + \mathrm{E}(\hat{f}^2) - 2\mathrm{E}(\hat{f}y)$$

Per definition af variansen af en stokastisk variabel og da vi har antaget at $\mathrm{E}(y) = \mathrm{E}(f + \varepsilon) = f$ fås

$$\mathrm{E}(y - \hat{f})^2 = \mathrm{E}(y)^2 + \mathrm{Var}(y) + \mathrm{E}(\hat{f})^2 + \mathrm{Var}(\hat{f}) - 2f\mathrm{E}(\hat{f})$$

da f er deterministisk vil $\mathrm{Var}(y) = \mathrm{Var}(f + \varepsilon) = \mathrm{Var}(\varepsilon) = \sigma^2$. Vi får altså

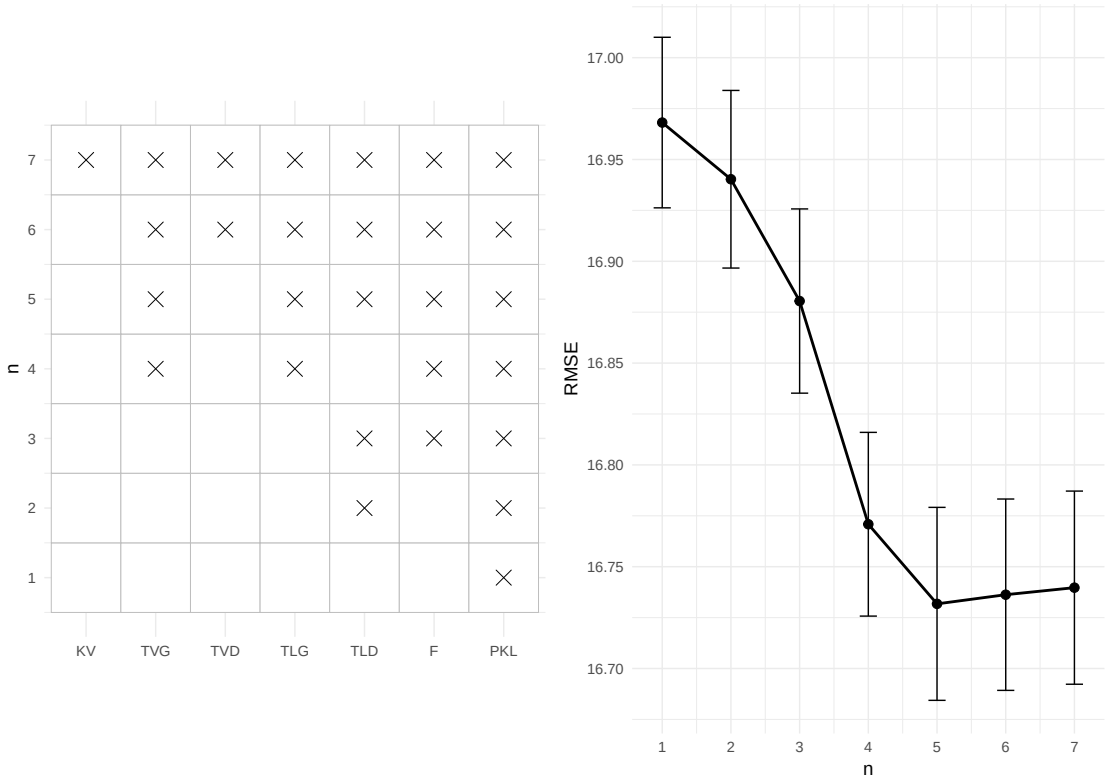
$$\mathrm{E}(y - \hat{f})^2 = \sigma^2 + \mathrm{Var}(\hat{f}) + \mathrm{E}(\hat{f})^2 + f^2 - 2f\mathrm{E}(\hat{f}) = \sigma^2 + \mathrm{Var}(\hat{f}) + \mathrm{E}(f - \hat{f})^2$$

Der gælder dermed at under disse tilpas milde antagelser vil fejlraten på en ny observation kunne inddeles i en irreducibel støj ,σ^2, en varians ved estimation ,$\mathrm{Var}(\hat{f})$, og biasen ved estimation i anden, $\mathrm{E}(f - \hat{f})^2$. Når vi har afgjort at OLS estimatorerne er BLUE, er det dermed ikke ensbetydende med den bedste prediktive model! Vi skal senere udforske estimatorer med en smule bias, der potentielt kan have langt mindre varians og dermed overordnet en bedre prediktiv kraft.

Hvis man søger en stærkt fortolkelig model hvor man er villig til at lave yderligere antagelser om den sande sammenhæng mellem respons og prediktorer, kan man udlede en række asymptotiske fordelingsegenskaber om estimaterne i den lineære regression. Man kan teste signifikansen af enkelte eller grupper af prediktorer og dermed reducere kompleksiteten af modellen og nedsætte risikoen for overfitting. I en mere prediktiv og praktisk kontekst, vil det være langt mere naturligt at udnytte cross-validation til at udføre modelselektion. Imodsætning til de træbaserede modeller, er der ikke på samme måde parametre at tune på i den lineære regression, man vil i stedet forsøge at tune på antallet af inkluderede prediktorer. Der er flere metoder at tilgå denne proces afhængig af antallet af prediktorer. Hvis antallet af prediktorer er tilpas lille, kan man undgå greedy algoritmer og gennemføre best subset selection. For hvert $n \in \{1, \ldots, p\}$ fittes de $\binom{p}{n}$ mulige lineære regressions modeller og deres performance evalueres f.eks. med cross-validation. Cross-validation med one-standard-error reglen bruges herefter til at selektere hvilket n, der performer bedst indenfor en standard error af den model med lavest RMSE.

4.4.2 Best subset lineær regression for fødte hvalpe

Best subset selection er implementeret i R-pakken `bestglm`. Ligesom tidligere bruges 10-fold cross-validation med 100 repeats til at estimere RMSE'en i hver model. De udvalgte prediktorer for de bedste modeller for hvert antal af parametre n mellem 1 og 7 kan ses på figur 7 herunder sammen med den estimerede RMSE. Den bedste model har 5 parametre med en RMSE på 16.738 og en standard error på 0.048. Den simpleste model indenfor en standard error af den bedste, er modellen med 4 parametre, der har en RMSE på 16.770 og denne selekteres som den bedste lineære model.



Figur 7: Venstre: De inkluderede prediktorer i den bedste model for hvert n. Højre: Plot over RMSE i de lineære modeller som funktion af n med errorbars af længde to standard error, estimeret ved repeated 10-fold cross-validation med 100 repeats.

Den udvalgte lineære regressionsmodel performer ikke signifikant bedre end de tidligere anvendte træbaserede modeller. Dette illustreres yderlige ved at betragte OOF-prediktionerne i den lineære model på figur 8, hvor det er svært at se forskel på denne model og de tidligere. Prediktionerne er dog endnu mere stabile med en gennemsnitlig standard error på 0.061 og en maksimal standard error på 0.195. Den udvalgte lineære model er på formen

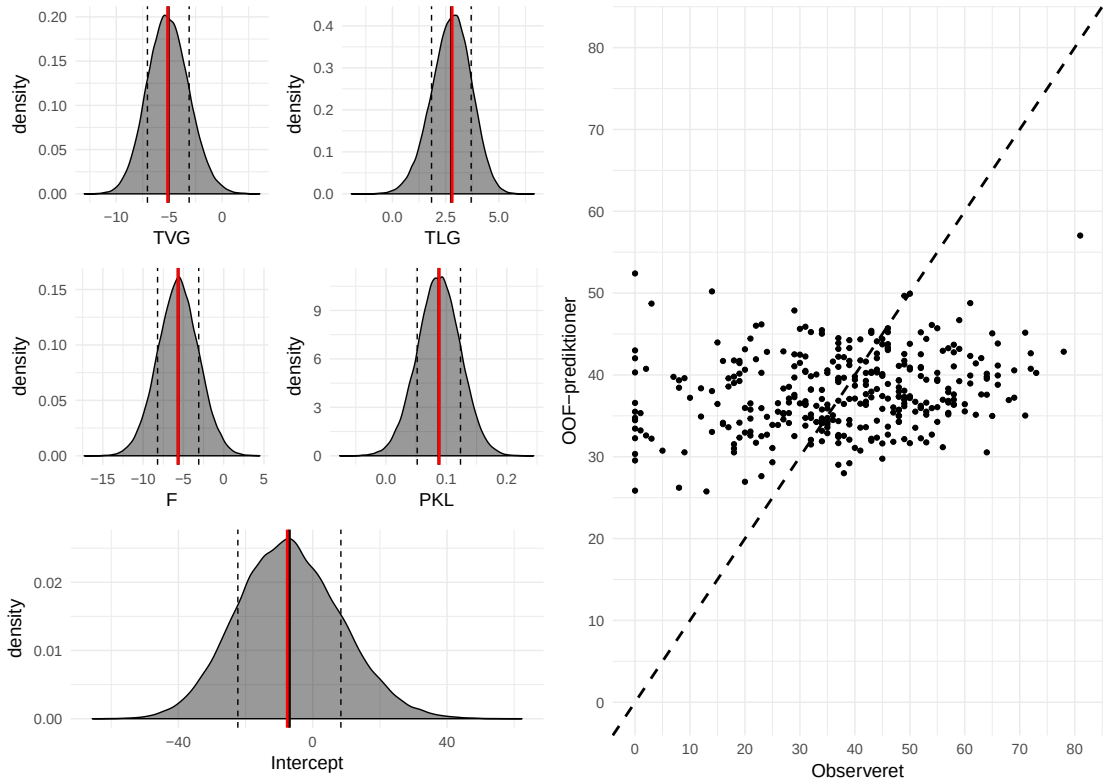
$$\hat{y} = \beta_0 + \beta_{\mathbf{TVG}} \cdot \mathbf{TVG} + \beta_{\mathbf{TLG}} \cdot \mathbf{TLG} + \beta_{\mathbf{PKL}} \cdot \mathbf{PKL} + \beta_{\mathbf{F}} \cdot 1_{\mathbf{F}=\text{sort}}$$

hvor $1_{\mathbf{F}=\text{sort}}$ er en indikatorfunktion, der er 1 hvis minken er sort. Estimaterne kan aflæses af nedenstående tabel.

Parameter	β_0	$\beta_{\mathbf{TVG}}$	$\beta_{\mathbf{TLG}}$	$\beta_{\mathbf{PKL}}$	$\beta_{\mathbf{F}}$
Estimat	-7.579	-5.150	2.805	0.088	-5.658

Tabel 2: Parameterestimaterne for den bedste lineære model med 4 prediktorer.

Disse estimater er bekymrende, idet vi tidligere har set at **TVG** og **TLG** er stærkt positivt korrelerede, så det virker ikke fornuftigt at større testikelængde øger antallet af fødtehvalpe mens tungere testes reducerer antallet af fødtehvalpe. Ukorrellerede prediktorer er en forudsætning for god performance i en lineær model og en måde hvorpå korrellerede prediktorer kommer til udtryk, er gennem en stor usikkerhed på parameterestimaterne. I et forsøg på at kvantificere problemet foretages bootstrapping; der genereres B datasæt (af samme længde som det oprindelige) fra det oprindelige ved at sample med tilbagelægning, den lineære model fittes og estimaterne gemmes. Disse estimater er plottet herunder på figur 8 for $B = 10^5$.



Figur 8: Venstre: 10^5 Bootstrappede parameterestimater for den bedste lineære model med 4 prediktorer med ikke-parametrisk afglattet densitetsestimat. Den solide sorte linje er gennemsnittet af de bootstrapede estimater og den røde er estimaterne fra fittet på hele datasættet. De stiplede linjer er placeret i gennemsnitet $\pm$ standard afvigelsen af de bootstrappede estimater.
Højre: Out-of-fold prediktioner for 100 repeats af 10-fold cross-validation plottet mod observerede værdier med error bars af længde 2 standard error.

Det ses at enkelte af de estimerede parameter varierer voldsomt, særligt interceptet har enormt stort spredning. En af de største fordele ved at anvende en lineær model; parameternes simple fortolkning, er naturligvis undergravet hvis vi ikke kan være sikre på de estimater vi har fået. I et forsøg på at kvantificere kollineariteten i de 4 prediktorer

plottes de kontinuerte prediktorer mod hinanden i et scatter plot og histogrammer over fordelingen af prediktorerne betinget på farven laves herunder.

Figur 9: Øvre trekant: Korrelationen mellem variable både ubetinget og betinget på farven.
Diagonalen: Ikke-parametrisk densitetsestimat for hver variabel betinget på farven.
Nedre trekant: Scatterplot over samspillet mellem hver variabel farvet efter farven på minken.

Det ses, at ikke nok med at der er stor kollinearitet mellem testikelvægt og -længde, har farven en stor indflydelse på både testikler og penisknogle. Man kan bemærke at korrelationen mellem testiklernesvægt og -længde og penisknoglelængden er meget tæt på 0 hvis man betragter data naivt, men hvis man betinger på farven, er der en større korrelation på ca. 0.2. Dette er i den lave ende og burde ikke påvirke synderligt, men det illustrerer yderligere at mange af disse variable indeholder redundant information og sammen med den store korrelation mellem testikelvægt og -længde kan man slå tvivl om den lineære models forudsætninger er opfyldt. I et forsøg på at løse dette problem vil vi i det næste afsnit betragte en modificeret version af lineær regression, der kan håndtere korrellerede prediktorer i større grad end ordinær lineær regression.

4.4.3 Penaliseret lineær regression

Vi så tidligere at OLS estimaterne var den bedste lineære unbiased estimator. Vi så også, at prediktionsfejl på en ny observation kunne inddeles i irreducibel støj, kvadreret bias og varians ved estimation. Det lyder umiddelbart som en fordel at have unbiased estimatorer, men hvis vi kan finde en biased estimator med lavere varians, er det muligt at vores model

er bedre til at prediktere fremtidig data. En måde at introducere bias, er ved penaliseret lineær regression, hvor det der penaliseres er størrelsen af ens parameterestimater. Et eksempel på dette er ridge regression, hvor man i stedet for at minimere RSS vil bestemme $\hat{\beta}_{\text{ridge}}$ som

$$\hat{\beta}_{\text{ridge}} = \arg\min_{\beta} \left(\text{RSS}(\beta) + \lambda \sum_{i=1}^{p} \beta_i^2 \right)$$

Vi penaliserer altså summen af størrelsen af de kvadrerede parameterestimater med en fast kompleksitetsparameter $\lambda \geq 0$. Det ses at for $\lambda = 0$ får vi blot OLS estimaterne, hvorimod for $\lambda \to \infty$ vil estimaterne gå mod 0. λ vil typisk selekteres vha. cross-validation. Bemærk, at vi ikke begrænser størrelsen på parameterestimatet for β_0, da interceptet blot er udtrykket for den predikterede værdi når prediktorerne antager værdien 0. Derudover er det klart at estimaterne ikke er invariante overfor skalering af prediktorerne, da vi minimerer summen af de kvadrererede parameterestimater. Før man fitter en ridge regression model, vil man derfor skalere hver prediktor ved at dividere med estimatet for standardafvigelsen for hver prediktor. Dette sikrer at alle prediktorerne har en standardafvigelse på 1 [3].

En af de største fordele ved ridge regression er muligheden for at håndtere korrellerede prediktorer. Som vi også så tidligere, kan to stærkt positivt korrellerede prediktorer have voldsomt store parameterestimater med modsat fortegn, hvilket intuitivt ikke giver specielt god mening og desuden ofte resulterer i stor varians i estimaterne. Ved at begrænse størrelsen på de kvadrerede parameterestimater, vil man, ved passende valg af λ, sikre at estimaternes størrelse forbliver lille og derfor også med samme fortegn. Variabelselektion i ridge regression kunne f.eks. gøres vha. best subset som beskrevet tidligere.

Det er ikke oplagt, at det netop skal være summen af kvadraterne på parameterestimaterne, der skulle være den bedste penalisering. Penaliseringen i ridge regression kaldes L_2 penalisering, da vi penaliserer L_2-normen af β. Penalisering ved L_1-normen kaldes lasso regression og estimatet for parameterne i denne model er givet ved

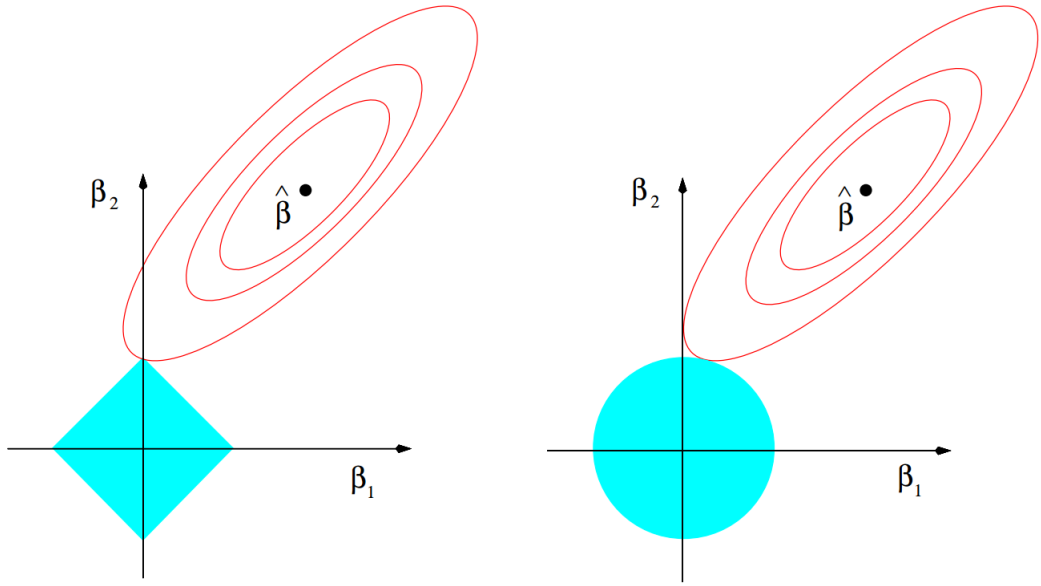
$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} \left(\text{RSS}(\beta) + \lambda \sum_{i=1}^{p} |\beta_i| \right)$$

Fuldstændig som i ridge regression bliver vi nødt til at skalere prediktorerne før modellen fittes. Lasso regression kan ligesom ridge regression håndtere korrellerede prediktorer, dog ikke helt i samme grad som ridge regression, men der er andre fordele ved lasso regression. Imodsætning til ridge regression vil lasso regression faktisk sætte koefficienter til 0 for passende værdier af λ. Lassoen udfører altså en form for kontinuert variabel selektion. Der gælder ligesom i ridge regression at for $\lambda = 0$ returneres blot OLS estimaterne og i grænsen $\lambda \to \infty$ vil $\beta = 0$. Man kan vise at ridge og lasso regressions problemerne er ækvivalente med

$$\hat{\beta}_{\text{ridge}} = \arg\min_{\beta} \left(\text{RSS}(\beta) \right) \quad \text{hvor} \sum_{i=1}^{p} \beta_i^2 \leq t$$

$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta} \left(\text{RSS}(\beta) \right) \quad \text{hvor} \sum_{i=1}^{p} |\beta_i| \leq t$$

Det vil sige, at for hvert λ i den oprindelige definition, findes et t i disse definitioner og vice versa. Vi kan nu illustrere hvorfor lasso regression kan selektere variable ved at betragte figur 10.



Figur 10: Venstre: Estimation i lasso regression med 2 parametre.
Højre: Estimation i ridge regression med 2 parametre.
De blå områder er områderne der opfylder kriteriet $\|\beta\| \leq t$ og de røde ellipser er niveaukurver for RSS[3].

Figuren illustrerer estimation i tilfældet $p = 2$ for ridge og lasso regression. Estimatet i hver model er det første sted de røde niveaukurver, rammer de blå områder. Parameterområdet for lassoen har hjørner, så hvis RSS kurven rammer et hjørne vil en parameter blive sat til 0. Når p øges vil antallet af hjørner øges så sandsynligheden for at ramme et eller flere hjørner øges drastisk og dermed vil flere parametre sættes til 0.

Man kunne nu begynde at udvide de betragtede L_1 og L_2 penaliseringer til L_q penaliseringer for $q \geq 0$. Værdier af $q \in (1, 2)$ kunne være et kompromis mellem lasso og ridge regression, men desværre vil ingen af disse dele lassoens egenskab om variabel selektion. Af denne grund og af komputationelle årsager introduceres elastic net regression hvor estimatet findes ved

$$\hat{\beta}_{\text{elastic net}} = \arg\min_{\beta} \left(\text{RSS}(\beta) + \lambda \sum_{i=1}^{p} ((1-\alpha)\beta_i^2 + \alpha|\beta_i|) \right)$$

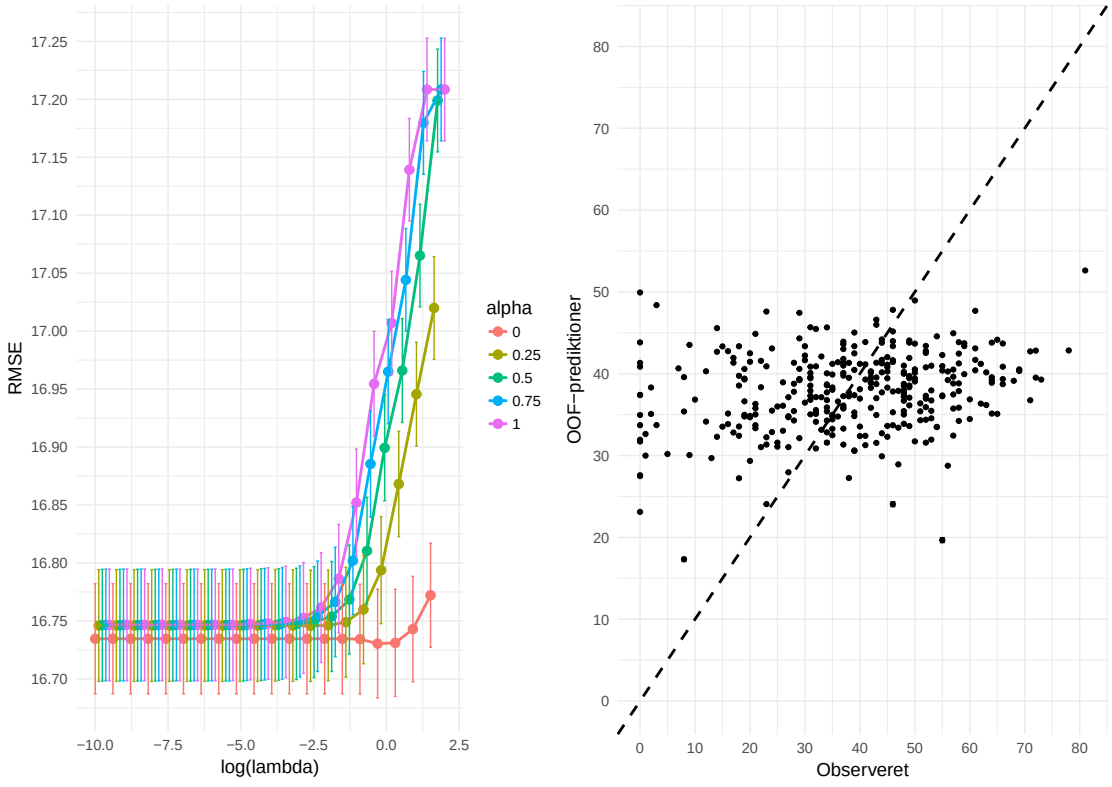
Dette er et kompromis mellem lasso og ridge regression, der forsøger at bevare variabelselektionen fra lassoen og håndteringen af korrellerede prediktorer fra ridge regression. α er en parameter mellem 0 og 1 og for $\alpha = 1$ er det lasso regression og for $\alpha = 0$ er det ridge regression. Både α og λ kan udvælges vha. cross-validation i et forsøg på at finde den bedste balance mellem ridge og lasso regression.

I en klassisk lineær regressionsmodel, vil man ofte være interesseret i usikkerheder på parameterestimater, særligt hvis man er interesseret i en fortolkelig model. Dette skyldes den centrale natur af estimatorne, da man altså kan være sikker på i gennemsnit at ramme

rigtigt under passende betingelser. Man kan derfor koncentrere sig om hvordan ens estimat bevæger sig omkring den centrale værdi. I penaliserede regressionsmodeller har vi med vilje introduceret bias i et forsøg på at reducere variansen og dermed formulere en stærkere prediktiv model. Et forsøg på at kvantificere variansen af koefficienterne i penaliseret regression, vil derfor ikke have den store fortolkningsmæssige interesse, da estimatoren ikke er central og spredningen altså ikke sker omkring nogen sand parameterværdi [2]. Vi vil derfor ikke yderligere interessere os for spredningen af disse estimater.

4.4.4 Elastic net model for fødte hvalpe

Elastic net penalisering er implementeret i R-pakken `glmnet`. Ligesom tidligere bruges 10-fold cross-validation med 100 repeats til at estimere RMSE'en i hver model vha. `caret`-pakken. Imodsætning til tidligere vil vi ikke bruge one-standard-error reglen her, da α og λ ikke påvirker kompleksiteten af modellen synderligt. Der er ikke komputationelle fordele og modellen bliver ikke simplere for tilpas valg af α eller λ. Vi vil derfor blot udvælge den model med lavest RMSE. Vi udfører cross-validation på α mellem 0 og 1 i skridt på 0.05 og λ uniformt på log-skala mellem -10 og 2 med 100 værdier i alt. Udvalgte α kan ses på figur 11 og resten kan ses i appendix B. Den bedste model har $\alpha = 0$ og $\lambda = 0.941$ med en RMSE på 16.730 og en standard error på 0.047. Dette er altså en ren ridge regression model og denne selekteres som den bedste elastic net model.



Figur 11: Venstre: Plot over RMSE som funktion af λ for forskellige værdier af α med error bars af længde to standard error, estimeret ved repeated 10-fold cross-validation med 100 repeats.
Højre: Out-of-fold prediktioner for 100 repeats af 10-fold cross-validation plottet mod observerede værdier med error bars af længde 2 standard error.

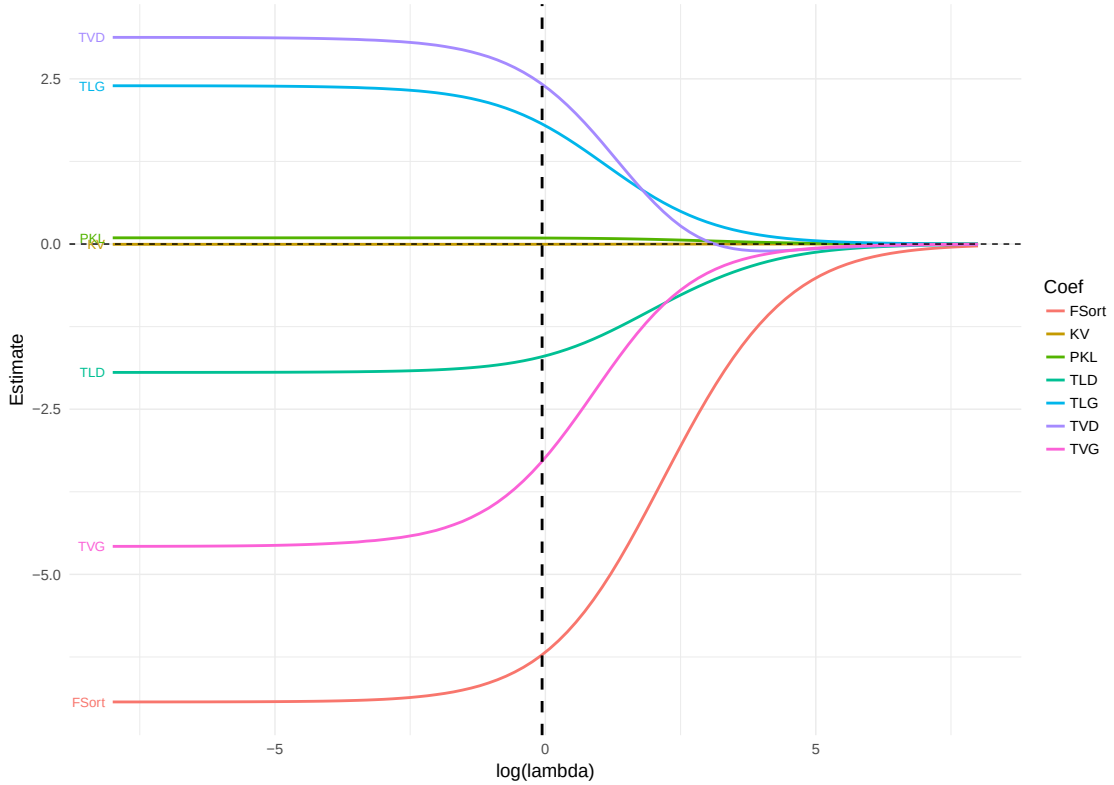
Vi betragter endnu en gang OOF-prediktionerne for den udvalgte model og bliver mødt med endnu et lignende billede af ringe prediktion. Modellens stabilitet minder meget om den lineære model fra før, da den gennemsnitlige standard error er på 0.072 og den maksimale er 0.30. Det lader ikke til at penalisering har forbedret prediktionerne markant og vi står stadig tilbage med en stor RMSE.

Parameterestimaterne i den udvalgte model og for OLS modellen med alle prediktorer kan aflæses herunder i tabel 3

Parameter	β_0	$\beta_{\mathbf{TVG}}$	$\beta_{\mathbf{TLG}}$	$\beta_{\mathbf{PKL}}$	$\beta_{\mathbf{F}}$	$\beta_{\mathbf{TLD}}$	$\beta_{\mathbf{TVD}}$	$\beta_{\mathbf{KV}}$
OLS estimat	15.7	−4.58	2.40	0.094	−6.93	−1.94	3.13	−0.004
Elastic net estimat	21.7	−3.29	1.82	0.091	−6.22	−1.71	2.42	−0.004

Tabel 3: Parameterestimaterne i elastic net modellen med $\alpha = 0$ og $\lambda = 0.941$ samt OLS estimaterne.

Præcis som konstrueret kan vi bemærke at parameterestimaterne i elastic net modellen er mindre end i OLS modellen. Vi lader dog ikke til at have fikset problemet med de korrellerede prediktorer helt, da testikelvægt og -længde stadig er til stede med modsat fortegn på trods af den stærke korrelation mellem dem. Det er muligt, at der simpelthen er for meget støj til at proceduren kan optimeres yderligere. Herunder på figur 12 ses et plot af parameterestimaterne som funktion af $\log(\lambda)$.



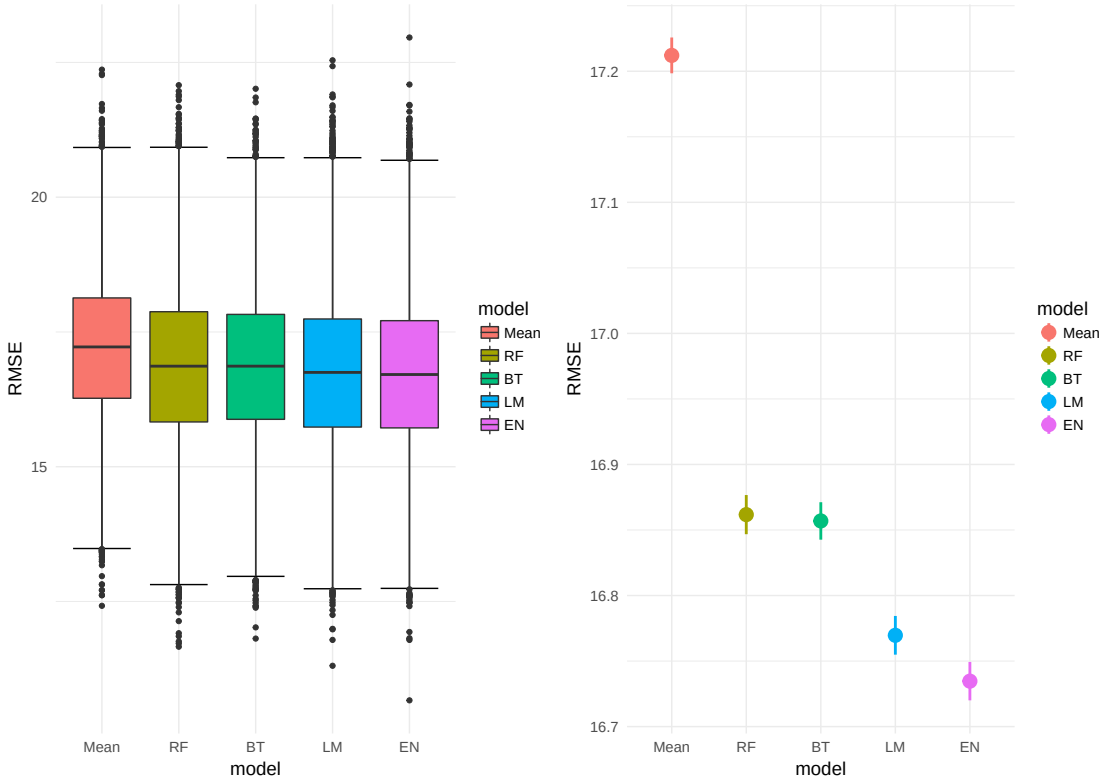
Figur 12: Parameterestimaterne i elastic net modellen med $\alpha = 0$ som funktion af $\log(\lambda)$. Den lodrette stiplede linje repræsenterer modellen med lavest RMSE vurderet vha. cross-validation.

Man kan bemærke at da $\alpha = 0$ er dette ridge regression og derfor går estimaterne mod 0 når λ vokser, men ingen af dem bliver sat til 0 før i grænsen. Det er interessant, at på trods af de stærkt korrellerede prediktorer, udvælges en relativt lav værdi af af λ som den der resulterer i den stærkeste model. Man kunne måske have forventet at en større begrænsning af estimaternes størrelse ville have modvirket korrelationen. Det er ligeledes også interessant at større α generelt giver dårligere modeller, da en lasso effekt potentielt kunne have sat enkelte af de korrellerede prediktorer til 0 samtidigt med at L_2 begrænsningen havde mindsket størrelsen.

Alt i alt vil man ikke ligefrem kalde nogen af de betragtede lineære modeller specielt stærke og i det næste afsnit vil vi opsummere analysens resultater.

4.5 Vurdering af modeller

I løbet af modelleringsprocessen har vi betragtet fire forskellige modeller og forsøgt at skabe en stærk prediktiv model. For endegyldigt at vurdere hvilken modelklasse, der virker bedst, vil 10-fold cross-validation med 1000 repeats gennemføres og modellerne sammenlignes med modellen der blot gætter på gennemsnittet.



Figur 13: Venstre: Boxplot over RMSE'en i hver fold for hver model i 10-fold cross-validation med 1000 repeats.
Højre: Den gennemsnitlige RMSE for hver model med errorbars af længde to standard error for 10-fold cross-validation med 1000 repeats.

På figur 13 kan det tydelig ses, at RMSE'en i hver fold har enorm stor spredning i alle modellerne, hvilket nok nærmere er en konsekvens af cross-validation end af selve modellen. Alle fire modeller performer bedre end at gætte på gennemsnittet, men det er kun

en forbedring på mellem 0.35 og 0.45 RMSE, dvs. en forbedring på ca. 2.5%. Det er ikke ligefrem opmuntrende resultater, men efter at fire forskellige modeller allesammen resulterer i ringe prediktiv kraft, er det nærmere end konsekvens af data end en konsekvens af modelvalg. Generelt performer de lineære modeller bedre end de træbaserede modeller, hvilket er en smule overraskende, da træbaserede modeller burde være mere fleksible. Hvis man betragter OOF-prediktionerne for hver af de fire modeller (se appendix C for en figur med alle fire modeller samtidigt), er det svært at se forskel på modellernes prediktioner. Det kunne måske se ud som at de lineære modeller er en smule bedre end de træbaserede, men ingen af dem ville klassificeres som tilfredsstillende modeller.

Der var generel enighed på tværs af modellerne omkring at penisknoglelængden, farven på minken og testikelstørrelsen var de vigtigste prediktorer i modelleringsprocessen. Testikelasymmetrien indgik i varierende grad og kropsvægten blev som udgangspunkt ikke vurderet til at være væsentlig. Det er svært at være præcis omkring omfanget af påvirkningen fra prediktorerne grundet den store kollinearitet i data. Det er muligt, at farven kun er signifikant fordi den indeholder information omkring størrelsen på minkens reproduktionsorganer, men det kunne ligeledes være en ren genetisk ting for sorte mink.

Man kunne kritisere den datadrevne natur af modelleringsstrategien. Det er meget muligt at det totale antal hvalpe er den mest interessante responsvariabel, men den indeholder både et adfærdsaspekt, dvs. hanminkens parringsvillighed, en succesrate for parring og et hvalpepotentiale, dvs. et mål for hvor mange hvalpe man kan forvente pr. succesfuld parring. Det er sandsynligt at man ville få større succes med at prediktere en af disse aspekter med de givne prediktorer, men den rene datadrevne strategi blev valgt, da den kræver så få antagelser som overhovedet muligt. Hvis man havde forsøgt sig med en mere teoriladet modelleringsstrategi, vil man skulle retfærdiggøre uafhængigheden af de førnævnte delelementer og den er på ingen måde oplagt.

Det er vigtigt at pointere, at de målte prediktorer sagtens kan være brugbare prediktorer, men evt. manglende prediktorers støj, kan skjule deres prediktive kraft. Hvis man f.eks. havde målt på de parrede hunners størrelse og reproduktionsorganer, kunne man opdage at efter at have kontrolleret for de nye prediktorer, vil hanners egenskaber pludselig være langt bedre prediktorer. Da der er tale om biologisk modellering, vil man skulle forvente en hel del støj i målingerne og prediktorerne, men en model der kun er marginalt bedre end at gætte på gennemsnittet burde alligevel kunne forbedres med andre prediktorer og flere observationer.

De eneste robuste konklusioner fra analysen må være, at hvis man som minkfarmer står og skal udvælge hanmink til parring, vil større penisknogler være korrellerede med flere hvalpe og brune mink vil som udgangspunkt få flere hvalpe end sorte. De lineære modeller virker enige om disse konklusioner. Effekten af testikelstørrelsen og -asymmetrien er mere uklar. Hvis sammenhængen skulle afklares yderligere burde man betragte ikke-lineære modeller f.eks. generalized additive models eller metoder der yderligere kan reducere kollineariteten i data, som f.eks. principal component regression.

5 Konklusion

I dette projekt er opstillet en række modeller, der forsøger at predikterer antallet af fødte minkhvalpe udfra biologiske egenskaber ved hanminken, særligt dens reproduktionsorganer. Modellernes prediktionsfejl blev vurderet ved repeated 10-fold cross-validation med 100 repeats. En random forest model blev opstillet med 300 træer og $m = 1$ blev udvalgt som den bedste parameter med en RMSE på 16.85. En boosted tree model blev opstillet med 30 træer, $D = 1$, $M = 1$ og $\lambda = 0.1$ hvilket resulterede i en RMSE på 16.84. En lineær regressionsmodel blev udvalgt med best subset selection og inkluderede 4 prediktorer; penisknoglelængden, farven på minken og testikelvægt og -længde. Denne blev vurderet til at have en RMSE på 16.77: En elastic net lineær regressionsmodel blev udvalgt med $\alpha = 0$ og $\lambda = 0.941$ hvilket resulterede i en RMSE på 16.73. De udvalgte modeller blev sammenlignet med en model der gættede på gennemsnittet, der resulterede i en RMSE på 17.2. Ingen af modellerne lod altså til at performe signifikant bedre end en model der blot gætter på gennemsnittet. Modellerne var dog enige om at farven på minken, penisknoglelængden og størrelsen på testes var blandt de vigtigste prediktorer mens kropsvægten og testikelasymmetrien var blandt de mindst væsentlige.

Referencer

[1] Københavns Universitet Det Sundhedsvidenskabelige Fakultet: *Nyt center styrker forskning i mink*. 2015. URL: `http://sund.ku.dk/nyheder/nyheder2015/nyt-center-styrker-minkforskningen/`.

[2] J. J. Goeman: "L1 penalized estimation in the Cox proportional hazards model". I: *Biometrics* 52 (2010).

[3] Trevor Hastie, Robert Tibshirani og Jerome Friedman: *Elements of Statistical Learning, Second Edition*. Springer, 2009.

[4] Gareth James m.fl.: *An Introduction to Statistical Learning*. Springer, 2013.

[5] Max Kuhn: *Comparing Different Species of Cross-Validation*. 2014. URL: `http://appliedpredictivemodeling.com/blog/2014/11/27/vpuig01pqbklmi72b8lcl3ij5hj2qm`.

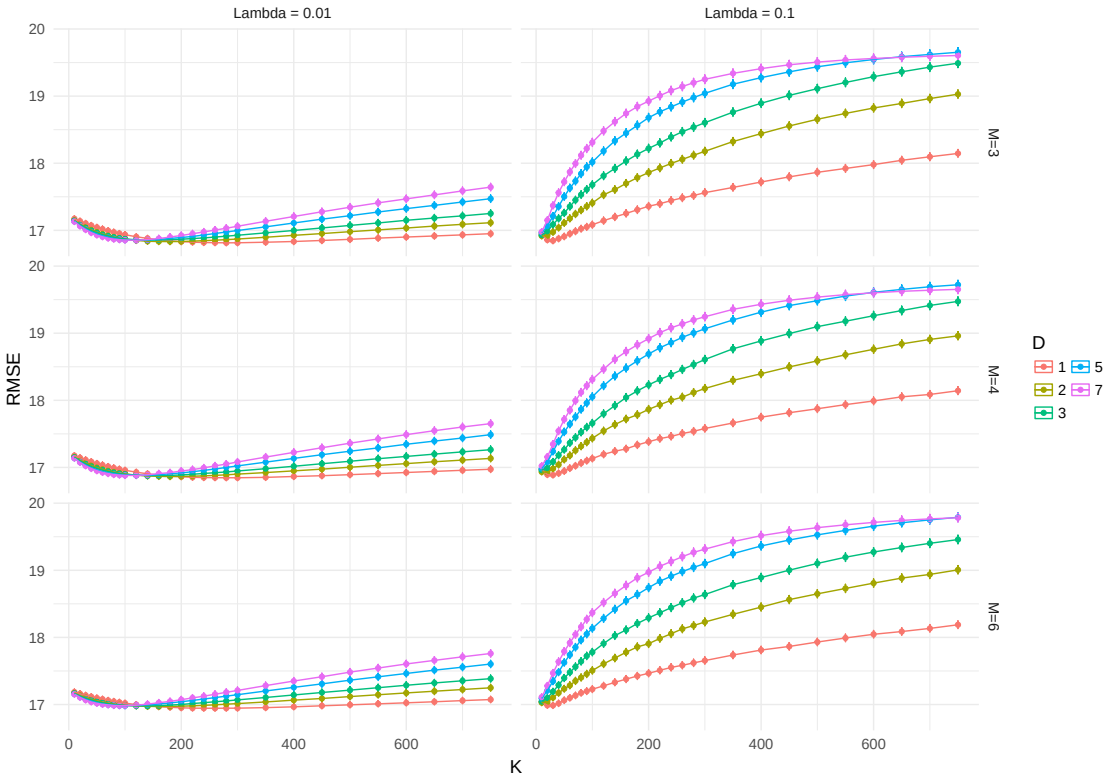
[6] Max Kuhn: *Comparing the Bootstrap and Cross-Validation*. 2014. URL: `http://appliedpredictivemodeling.com/blog/2014/11/27/08ks7leh0zof45zpf5vqe56d1sahb0`.

[7] Max Kuhn og Kjell Johnson: *Applied Predictive Modelling*. Springer, 2013.

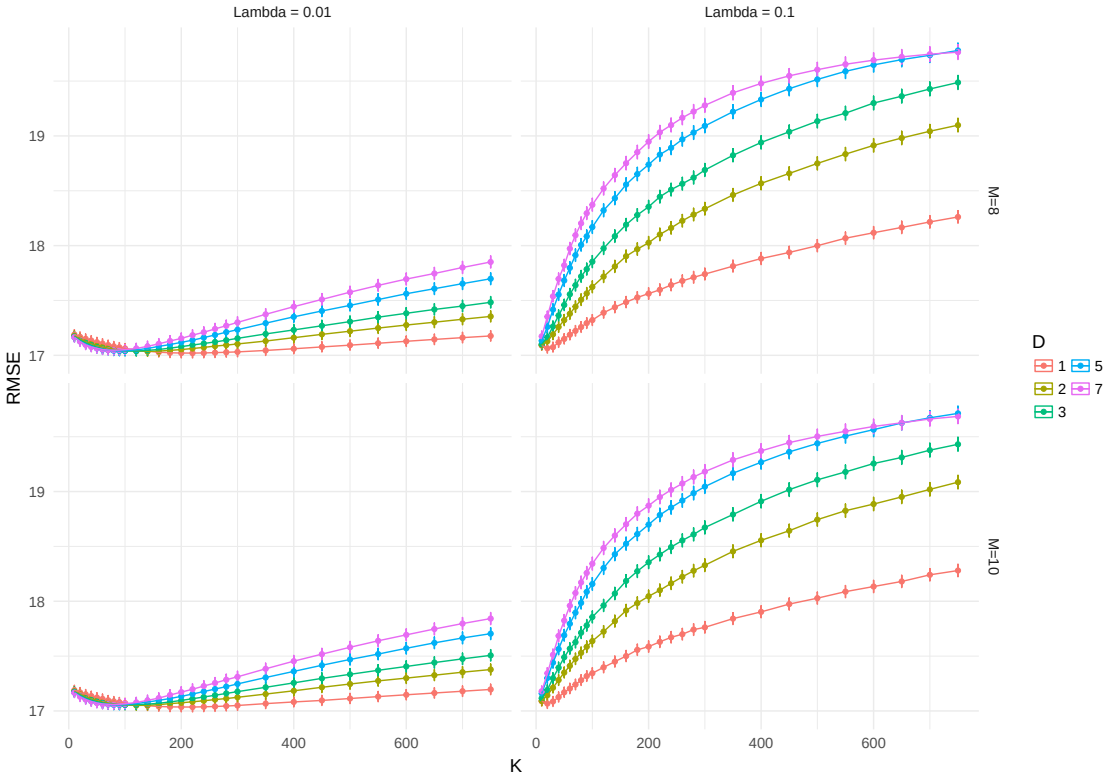
[8] Danske Minkavlere: *Fakta om minkavl*. 2012. URL: `http://www.danskeminkavlere.dk/fakta-om-minkavl/`.

Appendix

A Cross-validation af boosted tree model for fødte hvalpe

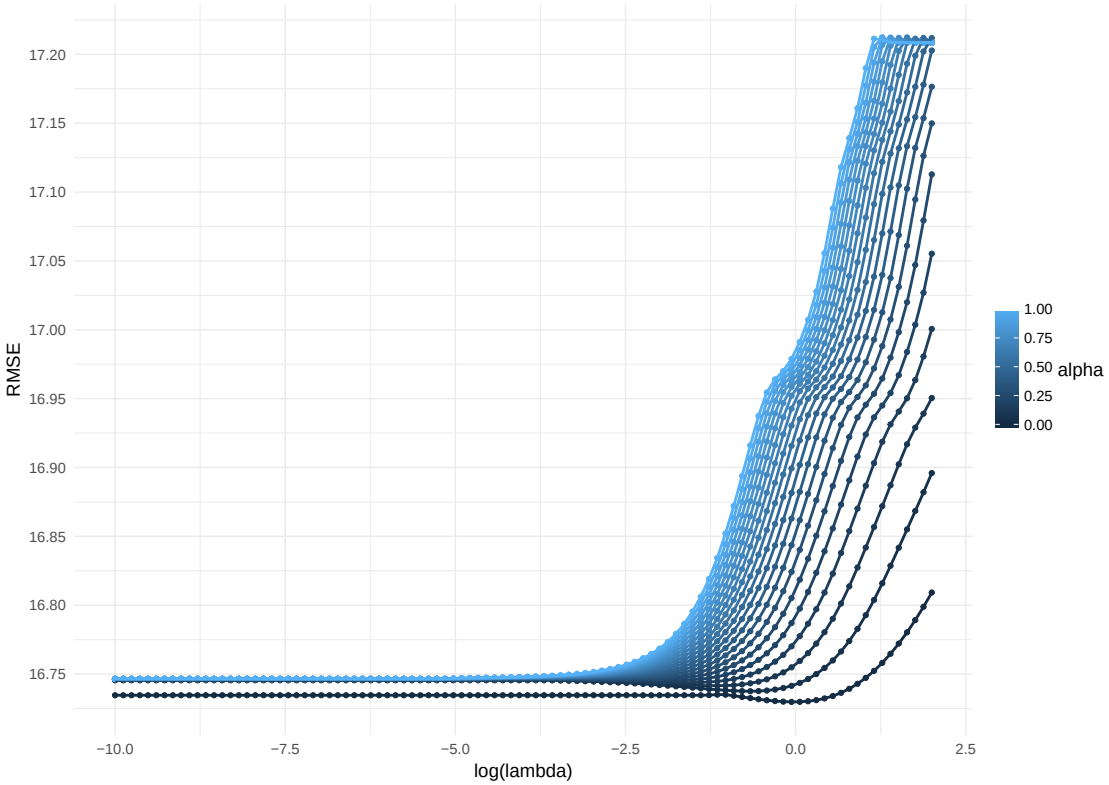


Figur 14: Plot over RMSE som funktion af antal træer (K) for $M = 3$, $M = 4$ og $M = 6$, samt forskellige værdier af D og λ med error bars af længde to standard error, estimeret ved repeated 10-fold cross-validation med 100 repeats.



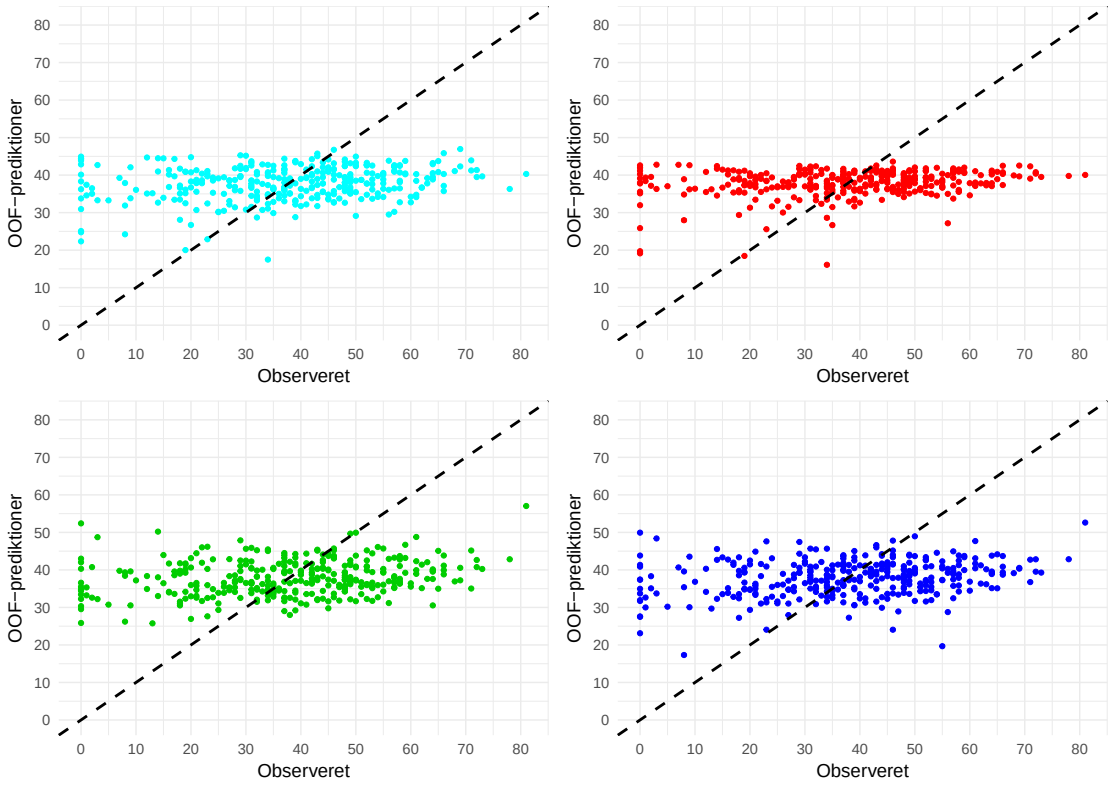
Figur 15: Plot over RMSE som funktion af antal træer (K) for $M = 8$ og $M = 10$, samt forskellige værdier af D og λ med error bars af længde to standard error, estimeret ved repeated 10-fold cross-validation med 100 repeats.

B Cross-validation af elastic net model for fødte hvalpe



Figur 16: Plot over RMSE som funktion af λ for forskellige værdier af α, estimeret ved repeated 10-fold cross-validation med 100 repeats.

C OOF-prediktioner for selekterede modeller



Figur 17: OOF prediktioner for de selekterede modeller.
Øverst venstre (cyan): Random forest
Øverst højre (rød): Boosted træer
Nederst venstre (grøn): OLS
Nederst højre (blå): Elastic net

D R-kode til fit af modeller

Herunder ses en delmængde af den anvendte R-kode. Den inkluderede kode er primært brugen af cross-validation og fitting af modeller.

```
library(tidyverse)
library(randomForest)
library(caret)
library(gbm)
library(bestglm)
library(glmnet)

###FarveCC er et dataframe med 8 variable;
####HF, TVG, TLG, PKL, F, KV, TLD, TVD

###Fit af simpel random forest model
rf.rf <- randomForest(HF ~ ., data=FarveCC, ntree=1000,mtry=2)

###Cross-validation af random forest model
rf.control <- trainControl(method="repeatedcv",number=10,
        repeats=1,search="grid",savePredictions="final",
        selectionFunction="oneSE")
rf.caret <- train(HF ~ .,data=FarveCC, method = "rf",
        tuneGrid=expand.grid(.mtry=seq(1,7)),ntree=300,
        importance=TRUE,trControl=rf.control)

###OOF for random forest
rf.preds <- rf.caret$pred
rf.preds <- dplyr::summarise(dplyr::group_by(rf.preds,rowIndex,obs),
        meanpred=mean(pred),SE=sd(pred)/10)


###Cross-validation af boosted tree model
bt.control <- trainControl(method="repeatedcv",number=10,repeats=100,
        search="grid",savePredictions="final",selectionFunction="oneSE")
bt.grid <- expand.grid(.interaction.depth = c(1,2,3,5,7),
        .n.trees=c(1:9*10,seq(100,300,by=20),seq(350,750,by=50)),
        .shrinkage=c(0.1,0.01),.n.minobsinnode=c(1,2,3,4,6,8,10))
bt.caret <- train(HF ~ .,data=FarveCC, method="gbm",
        verbose=FALSE,trControl=bt.control,tuneGrid=bt.grid)

###OOF for boosted tree
bt.preds <- bt.caret$pred
bt.preds <- dplyr::summarise(dplyr::group_by(bt.preds,rowIndex,obs),
        meanpred=mean(pred),SE=sd(pred)/10)
```

```
###Best subset selection
lm.best <- bestglm(FarveCC,family = gaussian,IC = "CV",
        CVArgs=list(Method="HTF",K=10,REP=100))

###Cross-validation for de 7 udvalgte OLS modeller med caret
lm.control <-  trainControl(method="repeatedcv",number=10,repeats=100,
        savePredictions="final",selectionFunction="oneSE")
lm.caret7 <- train(HF  ~ KV + TVG + TVD + TLG + TLD  + F + PKL,
         data=FarveCC,method="lm",trControl=lm.control)
RepFolds <- lm.caret7$control$index
lm.control <-  trainControl(method="repeatedcv",number=10,repeats=100,
        savePredictions="final",selectionFunction="oneSE",index=RepFolds)
lm.caret6 <- train(HF  ~ KV + TVG + TLG + TLD  + F + PKL,
        data=FarveCC,method="lm",trControl=lm.control)
lm.caret5 <- train(HF  ~ TVG + TLG + TLD  + F + PKL,
        data=FarveCC,method="lm",trControl=lm.control)
lm.caret4 <- train(HF  ~ TVG + TLG  + F + PKL,
        data=FarveCC,method="lm",trControl=lm.control)
lm.caret3 <- train(HF  ~ TLD  + F + PKL,
        data=FarveCC,method="lm",trControl=lm.control)
lm.caret2 <- train(HF  ~ TLD + PKL,
        data=FarveCC,method="lm",trControl=lm.control)
lm.caret1 <- train(HF  ~ PKL,
        data=FarveCC,method="lm",trControl=lm.control)

###Bootstrap af estimater fra OLS model
R <- 100000

lm.boot <- matrix(rep(NA,5*R),ncol=5)
colnames(lm.boot) <- c("Intercept","TVG","TLG","F","PKL")
for(i in 1:R)
{
  tempdata <- sample_n(FarveCC,dim(FarveCC)[1],replace=TRUE)
  lm.boot[i,]<- coef(lm(HF  ~ TVG + TLG  + F + PKL ,
        data=tempdata),correlation =T)
}

###OOF for OLS model
lm.preds <- lm.caret4$pred
lm.preds <- dplyr::summarise(dplyr::group_by(lm.preds,rowIndex,obs),
        meanpred=mean(pred),SE=sd(pred)/10)

###Cross-validation for elastic net
glmnet.control <- trainControl(method="repeatedcv",number=10,repeats=100,
        search="grid",savePredictions="final",selectionFunction="oneSE")
glmnet.grid <- expand.grid(alpha = seq(from = 0, to=1, by=0.05),
        lambda = exp(seq(from=-10,to=2,length.out=100)))
```

```
glmnet.caret <-  train(HF  ~. ,data=FarveCC,method="glmnet",
        trControl=glmnet.control,tuneGrid=glmnet.grid)

###OOF for elastic net
glmnet.preds <- glmnet.caret$pred
glmnet.preds <- dplyr::summarise(dplyr::group_by(glmnet.preds,
        rowIndex,obs),meanpred=mean(pred),SE=sd(pred)/10)


####Model sammenligning med cross-validation
comp.control <- trainControl(method="repeatedcv",number=10,repeats=1000,
        savePredictions="final")
rf.comp <- train(HF ~ .,data=FarveCC,method="rf",trControl=comp.control,
        ntree=300,tuneGrid = expand.grid(.mtry=1))
comp.control <- trainControl(method="repeatedcv",number=10,repeats=1000,
        savePredictions="final",index = rf.comp$control$index)
bt.comp <- train(HF ~ .,data=FarveCC,method="gbm",verbose=FALSE,
        trControl=comp.control, tuneGrid =
         expand.grid(.interaction.depth=1,.n.trees=30,
        .shrinkage=0.1,.n.minobsinnode=1))
lm.comp <- train(HF ~ TVG+TLG+PKL+F,data=FarveCC,
        method="lm",trControl=comp.control)
glmnet.comp <- train(HF ~ .,data=FarveCC,method="glmnet",
trControl=comp.control, tuneGrid = expand.grid(alpha=0,
lambda=glmnet.caret$bestTune$lambda))

  ####Mean model
L <- length(rf.comp$control$index)
mean_rmse <- rep(NA,L)
for (i in 1:L)
{
  xbar <- mean(FarveCC$HF[unname(unlist(rf.comp$control$index[i]))])
  mean_rmse[i] <- sqrt(sum((FarveCC$HF[-unname(unlist(
  rf.comp$control$index[i]))]-xbar)^2)/length(
  FarveCC$HF[-unname(unlist(rf.comp$control$index[i]))]))
}
mean.comp <- data.frame(RMSE = mean(RMSE),RMSESE = sd(RMSE)/sqrt(L))
```