Homework 1, skade2

Anna Sailegtim, Jens Fischer, Peter Garde

2024-06-10

R exercise 1

a)

```
options(scipen = 99)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v tibble  3.2.1     v dplyr   1.1.0
## v tidyr   1.3.0     v stringr 1.5.0
## v readr   2.1.3     v forcats 1.0.0
## v purrr   1.0.1
```

```
## Warning: package 'tibble' was built under R version 4.2.3
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
claim_pre = read_csv("claims.csv")
```

```
## Rows: 1529 Columns: 9
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (9): Clr, Dedr, Age, Brt, Dwt, Value, HP, Stroke, Code1
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
claim = claim_pre %>%
  mutate(Clr = Clr / 1000,
         Dedr = Dedr / 1000,
         Age=log(Age+2),
         Brt=log(Brt),
```

```r
        Dwt=log(Dwt),
        Value=log(Value),
        HP=log(HP),
        Stroke = factor(Stroke),
        Code1 = factor(Code1))

model = model.matrix(Clr ~ Age + Brt + Dwt + Value + HP + Stroke + Code1, claim)

phi = (312 - 1)^(-1)
mu = phi * 1.98

beta = c(mu, rep(0, ncol(model) - 1), phi)

log_likelihood_pareto = function(beta){
  return( -sum( log(pareto_G_optim(model, beta))
              - log(pareto_G_survival_func_optim(claim$Dedr, beta)) ) )
}

log_likelihood = function(beta, phi, x, d, modelMatrix){
  phiInverse = phi^(-1)
  return(
    length(x) * log(1 + phi)
    + sum( apply(cbind(modelMatrix, x, d),
           1,
           function(row) (1 + phiInverse)*log(
             exp(beta %*% row[1:(length(row)-2)] + phi * row[length(row)]))
          - (2 + phiInverse) * log( exp(beta %*% row[1:(length(row)-2)]
                                + phi * row[length(row) - 1]) ) ) )
    )
}

log_likelihood_optim = function(beta){
  return( -log_likelihood(beta[1:(length(beta)-1)], beta[length(beta)],
                          claim$Clr, claim$Dedr, model) )
}

optim_param = optim(beta, log_likelihood_optim)

beta = optim_param$par[1:(length(optim_param$par) - 1)]
beta
```

```
##  [1]  0.0081341817  0.0033271940 -0.0031246278  0.0001059604  0.0059856020
##  [6]  0.0041279208  0.0071957284 -0.0019530155 -0.0016895991 -0.0023760962
## [11] -0.0016644680 -0.0015452794  0.0103641029  0.0039083621  0.0033252161
```

```r
phi = optim_param$par[length(optim_param$par)]
phi
```

```
## [1] -0.02506653
```

We note that phi is negative which is quite peculiare. This might be because the r optim function has not congerged. We are not sure why this is the case. Further more phi and mu from our last was not parameterized to the G-Pareto dist. from the book. Thus these const. as starting values might be problematic.

b)

We apply backward selection where the model with the lowest AIC is chosen.

```
AIC = function(r, l){
  return(2*l - 2*r)
}
AIC(length(optim_param$par), log_likelihood_optim(optim_param$par))
```

```
## [1] 1399500
```

```
formula = Clr ~ Brt + Dwt + Value + HP + Stroke + Code1
model.m = model.matrix(formula, claim)

phi = (312 - 1)^(-1)
mu = phi * 1.98

beta = c(mu, rep(0, ncol(model) - 1), phi)

log_likelihood_optim = function(beta){
  return( -log_likelihood(beta[1:(length(beta)-1)], beta[length(beta)],
                          claim$Clr, claim$Dedr, model.m ) )
}
beta = c(mu, rep(0, ncol(model.m) - 1), phi)

optim_param = optim(beta, log_likelihood_optim)

l = log_likelihood_optim(optim_param$par)

AIC(length(optim_param$par), l)
```

```
## [1] 1399192
```

Full: 1399500 Age: 1399192

Age removed.

```
formula = Clr ~ Brt + Dwt + Value + HP + Stroke
model.m = model.matrix(formula, claim)

log_likelihood_optim = function(beta){
  return( -log_likelihood(beta[1:(length(beta)-1)], beta[length(beta)],
                          claim$Clr, claim$Dedr, model.m ) )
}
beta = c(mu, rep(0, ncol(model.m) - 1), phi)

optim_param = optim(beta, log_likelihood_optim)

l = log_likelihood_optim(optim_param$par)

AIC(length(optim_param$par), l)
```

```
## [1] 24425.14
```

Brt: 1399301 Dwt: 1399301 Value: 1399307 HP: 1399307 Stroke: 1399417 Code1: 24425.14

Remove Code1

```
formula = Clr ~ Dwt + Value + HP + Stroke
model.m = model.matrix(formula, claim)

log_likelihood_optim = function(beta){
  return( -log_likelihood(beta[1:(length(beta)-1)], beta[length(beta)],
                          claim$Clr, claim$Dedr, model.m ) )
}
beta = c(mu, rep(0, ncol(model.m) - 1), phi)

optim_param = optim(beta, log_likelihood_optim)

l = log_likelihood_optim(optim_param$par)

AIC(length(optim_param$par), l)
```

```
## [1] -326245.6
```

Brt: -326245.6

```
formula = Clr ~ Dwt + Value + HP + Stroke
model.m = model.matrix(formula, claim)

log_likelihood_optim = function(beta){
  return( -log_likelihood(beta[1:(length(beta)-1)], beta[length(beta)],
                          claim$Clr, claim$Dedr, model.m ) )
}
beta = c(mu, rep(0, ncol(model.m) - 1), phi)

optim_param = optim(beta, log_likelihood_optim)

l = log_likelihood_optim(optim_param$par)

AIC(length(optim_param$par), l)
```

```
## [1] -326245.6
```

Dwt: -264989.7 Value: -98277.82 HP: 99326.02 Stroke: 748955.1

We have thus found the best model by applying backward selection

```
beta0 = optim_param$par[1:(length(optim_param$par)- 1)]
phi0 = optim_param$par[length(optim_param$par)]

beta0
```

```
## [1]  20.2160000   8.4451378 -35.3516642 -12.7302477   0.1522923
```

```
phi0
```

```
## [1] 0.01089538
```

c)

```
claim_pre = read_csv("claims.csv")
```

```
## Rows: 1529 Columns: 9
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (9): Clr, Dedr, Age, Brt, Dwt, Value, HP, Stroke, Code1
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
claim = claim_pre %>%
  mutate(Clr = Clr / 1000000,
         Dedr = Dedr / 1000000,
         Age=log(Age+2),
         Brt=log(Brt),
         Dwt=log(Dwt),
         Value=log(Value),
         HP=log(HP),
         Stroke = factor(Stroke),
         Code1 = factor(Code1))

phi = (312 - 1)^(-1)
mu = phi * 1.98

beta = c(mu, rep(0, ncol(model) - 1), phi)
```

```
formula = Clr ~ Brt + Dwt + Value + HP + Stroke + Code1
model.m = model.matrix(formula, claim)

log_likelihood_optim = function(beta){
  return( -log_likelihood(beta[1:(length(beta)-1)], beta[length(beta)],
                          claim$Clr, claim$Dedr, model.m ) )
}
beta = c(mu, rep(0, ncol(model.m) - 1), phi)

optim_param = optim(beta, log_likelihood_optim)

l = log_likelihood_optim(optim_param$par)

AIC(length(optim_param$par), l)
```

```
## [1] -231795.3
```

Full: -81573.54 Age: -231795.3

We remove Age

```
formula = Clr ~ Dwt + Value + HP + Stroke + Code1
model.m = model.matrix(formula, claim)

log_likelihood_optim = function(beta){
  return( -log_likelihood(beta[1:(length(beta)-1)], beta[length(beta)],
                          claim$Clr, claim$Dedr, model.m ) )
}
beta = c(mu, rep(0, ncol(model.m) - 1), phi)

optim_param = optim(beta, log_likelihood_optim)

l = log_likelihood_optim(optim_param$par)

AIC(length(optim_param$par), l)
```

```
## [1] -856889.5
```

Brt: -856889.5

We remove Brt

```
formula = Clr ~ Value + HP + Stroke + Code1
model.m = model.matrix(formula, claim)

log_likelihood_optim = function(beta){
  return( -log_likelihood(beta[1:(length(beta)-1)], beta[length(beta)],
                          claim$Clr, claim$Dedr, model.m ) )
}
beta = c(mu, rep(0, ncol(model.m) - 1), phi)

optim_param = optim(beta, log_likelihood_optim)

l = log_likelihood_optim(optim_param$par)

AIC(length(optim_param$par), l)
```

```
## [1] -1705258
```

Dwt: -1705258

We remove Dwt

```
formula = Clr ~ Value + Stroke + HP
model.m = model.matrix(formula, claim)

log_likelihood_optim = function(beta){
  return( -log_likelihood(beta[1:(length(beta)-1)], beta[length(beta)],
                          claim$Clr, claim$Dedr, model.m ) )
}
beta = c(mu, rep(0, ncol(model.m) - 1), phi)

optim_param = optim(beta, log_likelihood_optim)
```

```r
l = log_likelihood_optim(optim_param$par)

AIC(length(optim_param$par), l)
```

```
## [1] -1894167
```

Value: -464538.3 HP: -829095.9 Code1: -1894167

We remove Code1

```r
formula = Clr ~ Value +  HP
model.m = model.matrix(formula, claim)

log_likelihood_optim = function(beta){
  return( -log_likelihood(beta[1:(length(beta)-1)], beta[length(beta)],
                          claim$Clr, claim$Dedr, model.m ) )
}
beta = c(mu, rep(0, ncol(model.m) - 1), phi)

optim_param = optim(beta, log_likelihood_optim)

l = log_likelihood_optim(optim_param$par)

AIC(length(optim_param$par), l)
```

```
## [1] -1900667
```

Value: -1858249 Stroke: -1900667

We remove Stroke

```r
formula = Clr ~ Value
model.m = model.matrix(formula, claim)

log_likelihood_optim = function(beta){
  return( -log_likelihood(beta[1:(length(beta)-1)], beta[length(beta)],
                          claim$Clr, claim$Dedr, model.m ) )
}
beta = c(mu, rep(0, ncol(model.m) - 1), phi)

optim_param = optim(beta, log_likelihood_optim)

l = log_likelihood_optim(optim_param$par)

AIC(length(optim_param$par), l)
```

```
## [1] -1903760
```

Value: -1815618
HP: -1903760

We remove HP

```r
formula = Clr ~ 1
model.m = model.matrix(formula, claim)

log_likelihood_optim = function(beta){
  return( -log_likelihood(beta[1:(length(beta)-1)], beta[length(beta)],
                          claim$Clr, claim$Dedr, model.m ) )
}
beta = c(mu, rep(0, ncol(model.m) - 1), phi)

optim_param = optim(beta, log_likelihood_optim)

l = log_likelihood_optim(optim_param$par)

AIC(length(optim_param$par), l)
```

```
## [1] -2212313
```

Value: -2212313

We remove Value. Only the intercept is kept in the final model.

```r
beta0 = optim_param$par[1]
phi0 = optim_param$par[2]

beta0
```

```
## [1] -745.3276
```

```r
phi0
```

```
## [1] 22.27594
```

Compared to b) we get a totally different result. Here none of the covariate holds enougth information to be kept in the model. We suspect that that the exp(.) in the log-likelihood of G-Pareto might be troublesome. As the claims and deductables gets very small. Its not certain that this is the root of the change.