STAT 534 HW3

1.

(a)

```
library(RMark)

cormorant <- import.chdata('cormorant.txt', field.types = c('n'))

ft <- list(formula = ~time)
cormt1 <- mark(cormorant, model = "CJS",
               model.parameters = list(Phi = ft, p = ft), output = FALSE)
cormt1$results$real
```

```
##                     estimate        se       lcl       ucl fixed note
## Phi g1 c1 a0 t1 0.6313619 0.0615072 0.5050237 0.7419316
## Phi g1 c1 a1 t2 0.7521545 0.0533588 0.6339318 0.8417292
## Phi g1 c1 a2 t3 0.7452033 0.0455821 0.6462663 0.8240042
## Phi g1 c1 a3 t4 0.7060546 0.0448671 0.6112521 0.7858382
## Phi g1 c1 a4 t5 0.5661777 0.0377102 0.4913391 0.6381151
## Phi g1 c1 a5 t6 0.6135211 0.0419093 0.5288955 0.6918034
## Phi g1 c1 a6 t7 0.5082264 0.0411453 0.4280632 0.5879688
## Phi g1 c1 a7 t8 0.4795735 0.0509108 0.3818840 0.5788504
## Phi g1 c1 a8 t9 0.4973530 0.0000000 0.4973530 0.4973530
## p g1 c1 a1 t2   0.4237126 0.0591575 0.3138029 0.5417271
## p g1 c1 a2 t3   0.4721773 0.0435517 0.3884312 0.5575191
## p g1 c1 a3 t4   0.4718003 0.0351037 0.4039557 0.5407029
## p g1 c1 a4 t5   0.3686770 0.0283483 0.3150499 0.4257584
## p g1 c1 a5 t6   0.4060984 0.0292146 0.3503424 0.4643849
## p g1 c1 a6 t7   0.4481761 0.0322042 0.3862115 0.5117929
## p g1 c1 a7 t8   0.4511359 0.0374953 0.3792154 0.5251575
## p g1 c1 a8 t9   0.2836780 0.0334469 0.2228906 0.3535023
## p g1 c1 a9 t10  0.5327890 0.0000000 0.5327890 0.5327890
```

(b)

The estimated survival probability for 1985-1986 is **0.7522**, and its se is **0.05336**.

(c)

The estimated survival probability for 1985-1986 is **0.7863**, and its se is **0.04906**.

```
f0 <- list(formula = ~1)

cormt2 <- mark(cormorant, model= "CJS",
               model.parameters = list(Phi = ft, p = f0), output = FALSE)

cormt2$results$real[2,]
```

```
##                     estimate        se       lcl       ucl fixed note
## Phi g1 c1 a1 t2 0.7862613 0.0490564 0.6748921 0.8669981
```

(d)

The estimated model-averaged survival probability for 1985-1986 is **0.7540**, and its se is **0.05369**.

```r
run.cjs <- function(data) {
  f0 <- list(formula = ~1)
  ft <- list(formula = ~time)
  cormt1 <- mark(data, model = "CJS",
              model.parameters = list(Phi = ft, p = ft), output = FALSE)
  cormt2 <- mark(data, model="CJS",
              model.parameters = list(Phi = ft, p = f0), output = FALSE)

  collect.models()
}

cjs.models <- run.cjs(cormorant)
model.average(cjs.models, indices = 2)
```

```
##   par.index  estimate         se
## 1         2 0.7539888 0.05369041
```

(e)

```r
fT <- list(formula = ~Time)
cormt3 <- mark(cormorant, model = "CJS",
              model.parameters = list(Phi = fT, p = ft), output = FALSE)

cormt3$results$beta[1:2,]
```

```
##                     estimate        se        lcl        ucl
## Phi:(Intercept)  1.216523 0.1210688  0.9792281  1.4538178
## Phi:Time        -0.181923 0.0278111 -0.2364328 -0.1274132
```

(f)

```r
run.cjs3 <- function(data) {
  f0 <- list(formula = ~1)
  ft <- list(formula = ~time)
  fT <- list(formula = ~Time)
  cormt1 <- mark(data, model = "CJS",
              model.parameters = list(Phi = ft, p = ft), output = FALSE)
  cormt2 <- mark(data, model = "CJS",
              model.parameters = list(Phi = ft, p = f0), output = FALSE)
  cormt3 <- mark(data, model = "CJS",
              model.parameters = list(Phi = fT, p = ft), output = FALSE)
  collect.models()
}

cjs.models3 <- run.cjs3(cormorant)
print(model.table(cjs.models3, model.name = F), digits = 4)
```

```
##     Phi     p  model npar AICc DeltaAICc   weight Deviance
```

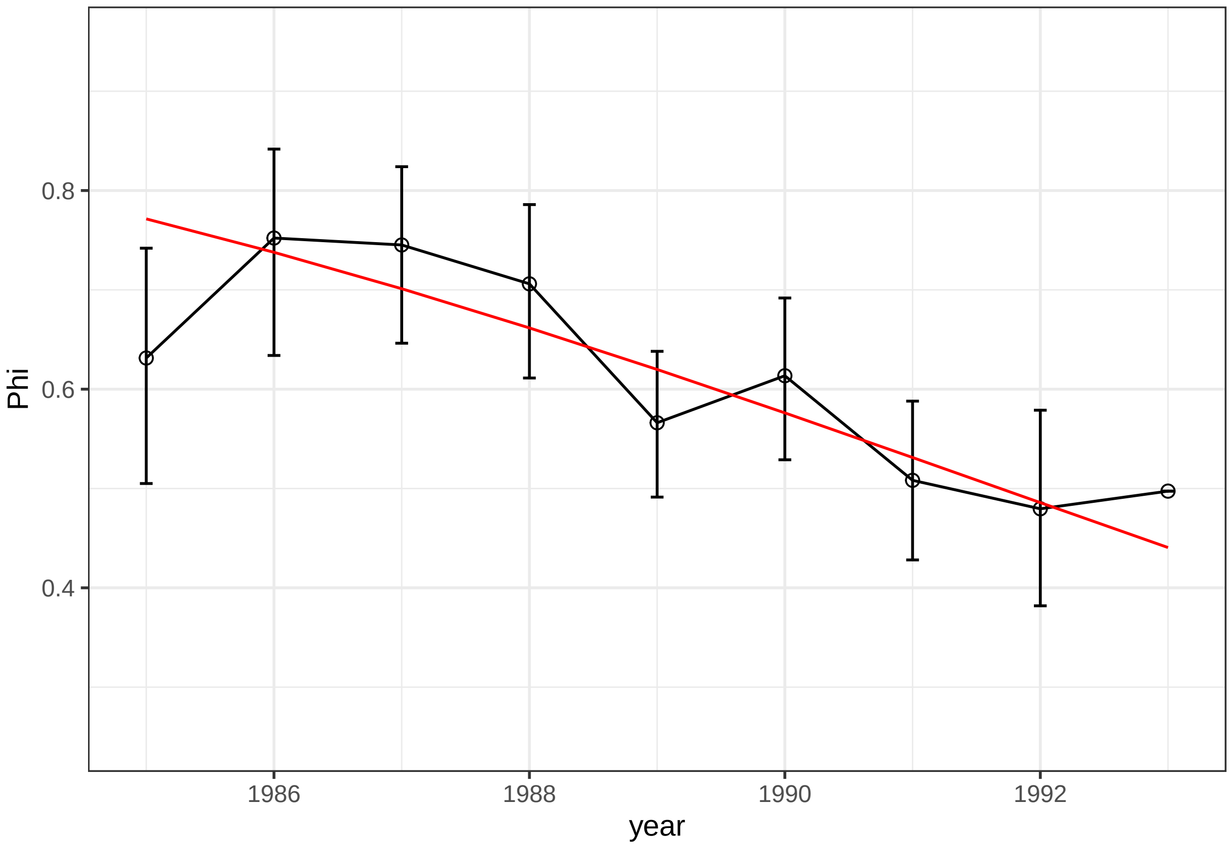
```
## 3 ~Time ~time cormt3   11 8358     0.000 0.931541     1659
## 1 ~time ~time cormt1   18 8363     5.332 0.064777     1650
## 2 ~time    ~1 cormt2   10 8369    11.067 0.003682     1672
```

(g)

```
library(ggplot2)
# using line for 95% CI
#year <- 1985:1993
#Phi <- cormt1$results$real$estimate[1:9]
#se <- cormt1$results$real$se[1:9]
#lower <- Phi - qnorm(0.975)*se
#upper <- Phi + qnorm(0.975)*se
#plot(year, Phi, pch = 16, col = "blue", ylim = c(0.3, 1))
#lines(year[1:8], lower[1:8], lty = 'dashed', col = 'red')
#lines(year[1:8], upper[1:8], lty = 'dashed', col = 'red')
#curve(1/(1 + exp(-(1.2165 - 0.1819*(x - 1985)))), add = TRUE)

# using error bars for 95% CI
d <- data.frame(year = 1985:1993, Phi=cormt1$results$real$estimate[1:9],
                lcl = cormt1$results$real$lcl[1:9],
                ucl = cormt1$results$real$ucl[1:9],
                fitted = cormt3$results$real$estimate[1:9])

ggplot(d, aes(x = year, y = Phi)) + geom_point(shape = 1,size=2) + geom_line()+
  geom_errorbar(aes(ymin = lcl, ymax = ucl), width = 0.1)+
  geom_line(aes(x = year, y = fitted), colour = "red")+
  ylim(c(0.25, 0.95)) + theme_bw()
```




```
#ggsave("figure_1g.pdf",height=5, width=7)
```

2.

(a)

```
huggins <- import.chdata('huggins.txt', field.types = c('f','n','f','n'))

run.huggins <- function(data) {
  fh <- list(formula = ~ home.range, share = TRUE)
  fs <- list(formula = ~sex, share = TRUE)
  fw <- list(formula = ~weight, share = TRUE)
  ftt <- list(formula = ~trap.type, share = TRUE)

  huggins1 <- mark(data, model = "Huggins",
              model.parameters = list(p = fh), output = FALSE)
  huggins2 <- mark(data, model = "Huggins",
              model.parameters = list(p = fs), groups = c('sex'), output = FALSE)
  huggins3 <- mark(data, model = "Huggins",
              model.parameters = list(p = fw), output = FALSE)
  huggins4 <- mark(data, model = "Huggins",
              model.parameters = list(p = ftt), groups = c('trap.type'), output = FALSE)

  collect.models()
}
```

```r
fit.huggins <- run.huggins(huggins)
print(model.table(fit.huggins, model.name = F), digits = 4)
```

```
##              p c    model npar  AICc DeltaAICc    weight Deviance
## 1 ~home.range   huggins1    2 635.0      0.00 1.000e+00    631.0
## 3     ~weight   huggins3    2 669.5     34.44 3.321e-08    665.4
## 4  ~trap.type   huggins4    3 673.7     38.74 3.879e-09   1092.5
## 2        ~sex   huggins2    2 674.7     39.71 2.387e-09   1222.2
```

(b)

```r
fit.huggins[['huggins1']]$results$derived
```

```
## $`N Population Size`
##   estimate       se      lcl      ucl
## 1 219.3206 12.38151 200.6794 250.5159
```

(c)

```r
f0 <- list(formula = ~1, share = TRUE)
huggins0 <- mark(huggins, model = 'Closed', model.parameters = list(p = f0),
                 output = FALSE)
huggins0$results$derived
```

```
## $`N Population Size`
##   estimate       se      lcl      ucl
## 1 202.6915 8.150204 190.5068 223.3568
```

(e)

```r
run.pledger <- function(data) {
  fms <- list(formula = ~mixture, share = TRUE)
  fmts <- list(formula = ~mixture + time, share = TRUE)

  mh <- mark(data, model = "FullHet", model.parameters = list(p = fms))
  mth <- mark(data, model = "FullHet", model.parameters = list(p = fmts))

  collect.models()
}

fit.pledger <- run.pledger(huggins)
```

```
##
## Output summary for FullHet model
## Name : pi(~1)p(~mixture)c()f0(~1)
##
## Npar :  4  (unadjusted=3)
## -2lnL:  -769.7785
## AICc :  -761.7007  (unadjusted=-763.73194)
##
## Beta
##                   estimate         se         lcl        ucl
## pi:(Intercept)  -2.882150  0.5873160   -4.033290  -1.731011
```

```
## p:(Intercept)   15.670872 79.5382490 -140.224100 171.565840
## p:mixture2     -15.998144 79.5382410 -171.893100 139.896810
## f0:(Intercept)   3.663009  0.3018644    3.071355   4.254663
##
##
## Real Parameter pi
##
##
## mixture:1 0.053043
##
##
## Real Parameter p
##
##                   1         2         3
## mixture:1 0.9999998 0.9999998 0.9999998
## mixture:2 0.4189044 0.4189044 0.4189044
##
##
## Real Parameter c
##
##                   2         3
## mixture:1 0.9999998 0.9999998
## mixture:2 0.4189044 0.4189044
##
##
## Real Parameter f0
##
##         1
##  38.97846
##
## Output summary for FullHet model
## Name : pi(~1)p(~mixture + time)c()f0(~1)
##
## Npar :  6  (unadjusted=5)
## -2lnL:  -770.8552
## AICc :  -758.6911  (unadjusted=-760.7382)
##
## Beta
##                   estimate         se          lcl         ucl
## pi:(Intercept)  -2.8766795  0.5841304   -4.0215752  -1.7317839
## p:(Intercept)   15.5037060 80.8360170 -142.9348900 173.9423000
## p:mixture2     -15.9197550 80.8360060 -174.3583300 142.5188200
## p:time2          0.0620882  0.2034861   -0.3367446   0.4609210
## p:time3          0.2045816  0.2026399   -0.1925926   0.6017559
## f0:(Intercept)   3.6591944  0.3021277    3.0670242   4.2513647
##
##
## Real Parameter pi
##
##
## mixture:1 0.0533185
##
##
## Real Parameter p
```

```
##
##                    1         2         3
## mixture:1 0.9999998 0.9999998 0.9999998
## mixture:2 0.3974625 0.4124222 0.4473292
##
##
## Real Parameter c
##
##                    2         3
## mixture:1 0.9999998 0.9999998
## mixture:2 0.4124222 0.4473292
##
##
## Real Parameter f0
##
##          1
##  38.83005
```

```
print(model.table(fit.pledger, model.name = F), digits = 4)
```

```
##   pi                p c f0 model npar   AICc DeltaAICc weight Deviance
## 1 ~1        ~mixture   ~1    mh    4 -761.7      0.00 0.8183    6.810
## 2 ~1 ~mixture + time   ~1   mth    6 -758.7      3.01 0.1817    5.734
```

```
cleanup(ask = F)
```