

Stat 534 Homework 1

1.

(a)

```
oakD <- read.csv("oakD.csv")
oakM <- read.csv("oakM.csv")
both <- read.csv("bothoak.csv")

lnlNBO <- function(param, y, trace = F) {
  # compute lnl for Negative binomial distribution
  # for a single sample of observations
  # no precautions for mu < 0 or r < 0

  mu <- param[1]
  r <- param[2]

  ratio <- r/(r+mu)

  lnli <- lgamma(r+y) - lgamma(y+1) - lgamma(r) + r*log(ratio) + y*log(1-ratio)
  # lnL contribution for each observation

  s = sum(lnli)
  if (trace) {
    print(c(mu, r, s))
  }
  s
}

fitdry1 <- optim(c(20, 5), lnlNBO , y = oakD$dry, method = 'BFGS',
  control = list(fnscale = -1), hessian = T, trace = T)
```

```
## [1] 20.0000 5.0000 -115.4639
## [1] 20.0010 5.0000 -115.4675
## [1] 19.9990 5.0000 -115.4603
## [1] 20.0000 5.0010 -115.4728
## [1] 20.0000 4.9990 -115.4549

## Warning in log(ratio): NaNs produced

## [1] 16.43000 -3.94904 NaN
## [1] 19.286000 3.210192 -95.852946
## [1] 19.287000 3.210192 -95.855472
## [1] 19.285000 3.210192 -95.850421
## [1] 19.286000 3.211192 -95.863520
## [1] 19.286000 3.209192 -95.842372

## Warning in log(ratio): NaNs produced
```

```
## [1] 16.760741 -7.363933      NaN
## [1] 18.780948  1.095367 -70.964543
## [1] 18.781948  1.095367 -70.965512
## [1] 18.779948  1.095367 -70.963574
## [1] 18.780948  1.096367 -70.975222
## [1] 18.780948  1.094367 -70.953870
## [1] -26.34651 -462.03627  747.70123
## [1] -26.34551 -462.03627  747.67744
## [1] -26.34751 -462.03627  747.72503
## [1] -26.34651 -462.03527  747.70119
## [1] -26.34651 -462.03727  747.70128
## [1] -46.38388 -607.95234 1212.94356
## [1] -46.38288 -607.95234 1212.92146
## [1] -46.38488 -607.95234 1212.96566
## [1] -46.38388 -607.95134 1212.94349
## [1] -46.38388 -607.95334 1212.94363
## [1] -345.8251 -2794.0700  7565.8590
## [1] -345.8241 -2794.0700  7565.8392
## [1] -345.8261 -2794.0700  7565.8788
## [1] -345.8251 -2794.0690  7565.8588
## [1] -345.8251 -2794.0710  7565.8591
## [1] -3948.656 -29111.145  82056.986
## [1] -3948.655 -29111.145  82056.966
## [1] -3948.657 -29111.145  82057.005
## [1] -3948.656 -29111.144  82056.986
## [1] -3948.656 -29111.146  82056.986
## [1] -3968.047 -29111.316  82432.909
## [1] -3968.046 -29111.316  82432.889
## [1] -3968.048 -29111.316  82432.928
## [1] -3968.047 -29111.315  82432.909
## [1] -3968.047 -29111.317  82432.909
## [1] -36848.15 -29404.06  526214.33
## [1] -36848.15 -29404.06  526214.32
## [1] -36848.15 -29404.06  526214.34
## [1] -36848.15 -29404.06  526214.32
## [1] -36848.15 -29404.06  526214.33
## [1] -70592.40 -29715.75  796071.36
## [1] -70592.40 -29715.75  796071.36
## [1] -70592.40 -29715.75  796071.37
## [1] -70592.40 -29715.75  796071.35
## [1] -70592.40 -29715.75  796071.37
## [1] -140567.97 -30385.08 1155500.93
## [1] -140567.97 -30385.08 1155500.92
## [1] -140567.97 -30385.08 1155500.93
## [1] -140567.97 -30385.08 1155500.91
## [1] -140567.97 -30385.08 1155500.95
## [1] -254341.29 -31507.28 1529354.48
## [1] -254341.29 -31507.28 1529354.48
## [1] -254341.29 -31507.28 1529354.48
## [1] -254341.29 -31507.28 1529354.45
## [1] -254341.29 -31507.28 1529354.51
## [1] -254343.71 -31536.22 1530197.81
## [1] -254343.71 -31536.22 1530197.80
## [1] -254343.72 -31536.22 1530197.81
```

[1] -254343.71 -31536.22 1530197.78
[1] -254343.71 -31536.22 1530197.84
[1] -258860.14 -85368.11 2619529.71
[1] -258860.13 -85368.11 2619529.71
[1] -258860.14 -85368.11 2619529.72
[1] -258860.14 -85368.11 2619529.70
[1] -258860.14 -85368.12 2619529.73
[1] -263400.5 -139383.6 3254881.7
[1] -263400.5 -139383.6 3254881.7
[1] -263400.5 -139383.6 3254881.7
[1] -263400.5 -139383.6 3254881.7
[1] -263400.5 -139383.6 3254881.7
[1] -272144.6 -243267.9 4019118.3
[1] -272144.6 -243267.9 4019118.2
[1] -272144.6 -243267.9 4019118.3
[1] -272144.6 -243267.9 4019118.2
[1] -272144.6 -243267.9 4019118.3
[1] -285365.2 -400161.9 4740037.7
[1] -285365.2 -400161.9 4740037.7
[1] -285365.2 -400161.9 4740037.7
[1] -285365.2 -400161.9 4740037.7
[1] -285365.2 -400161.9 4740037.7
[1] -285378.0 -400164.6 4740209.9
[1] -285378.0 -400164.6 4740209.9
[1] -285378.0 -400164.6 4740209.9
[1] -285378.0 -400164.6 4740209.9
[1] -285378.0 -400164.6 4740209.9
[1] -1278016.1 -607726.6 15140519.5
[1] -1278016.1 -607726.6 15140519.5
[1] -1278016.1 -607726.6 15140519.5
[1] -1278016.1 -607726.6 15140519.5
[1] -1278016.1 -607726.6 15140519.5
[1] -3436707 -1059139 33687330
[1] -3436707 -1059139 33687330
[1] -3436707 -1059139 33687330
[1] -3436707 -1059139 33687330
[1] -3436707 -1059139 33687330
[1] -24204092 -5401956 202179033
[1] -24204092 -5401956 202179033
[1] -24204092 -5401956 202179033
[1] -24204092 -5401956 202179033
[1] -24204092 -5401956 202179033
[1] -730686164 -153139661 5905681363
[1] -730686164 -153139661 5905681363
[1] -730686164 -153139661 5905681363
[1] -730686164 -153139661 5905681363
[1] -730686164 -153139661 5905681363
[1] -730686168 -153139681 5905681793
[1] -730686168 -153139681 5905681793
[1] -730686168 -153139681 5905681793
[1] -730686168 -153139681 5905681793
[1] -730686168 -153139681 5905681793
[1] -730686168 -153139681 5905681793
[1] -730686172 -153139701 5905682223
[1] -730686172 -153139701 5905682223

```
## [1] -730686172 -153139701 5905682223
## [1] -730686172 -153139701 5905682223
## [1] -730686172 -153139701 5905682223
## [1] -730705304 -153241944 5907837966
## [1] -730705304 -153241944 5907837966
## [1] -730705304 -153241944 5907837966
## [1] -730705304 -153241944 5907837966
## [1] -730705304 -153241944 5907837966
## [1] -850450044 -793138562 12714065116
## [1] -850450044 -793138562 12714065116
## [1] -850450044 -793138562 12714065116
## [1] -850450044 -793138562 12714065116
## [1] -850450044 -793138562 12714065116
## [1] -905597765 -1087839032 14495101897
## [1] -905597765 -1087839032 14495101897
## [1] -905597765 -1087839032 14495101897
## [1] -905597765 -1087839032 14495101897
## [1] -905597765 -1087839032 14495101897
## [1] -1182730357 -2568790518 21403100875
## [1] -1182730357 -2568790518 21403100875
## [1] -1182730357 -2568790518 21403100875
## [1] -1182730357 -2568790518 21403100875
## [1] -1182730357 -2568790518 21403100875
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
## [1] -1182730372 -2568790519 21403101104
```

(b)

```
# a more reasonable starting value
mean(oakD$dry)
```

```
## [1] 3.772727
```

```
fitdry2 <- optim(c(4, 5), ln1NB0 , y = oakD$dry, method = 'BFGS',
                control = list(fnscale = -1), hessian = T)
fitdry2$par
```

```
## [1] 3.772698 3.179152
```

(c)

```
vc <- solve(-fitdry2$hessian)
se <- sqrt(diag(vc))[1]
se

## [1] 0.6123596
# the 95% CI is
fitdry2$par[1] + c(-1, 1)*se*qnorm(0.975)

## [1] 2.572495 4.972901
```

(d)

```
lnlNB1 <- function(param, y, trace=F) {
  # compute lnL for Negative binomial distribution
  # for a single sample of observations
  # using log(mu) and r as parameters

  mu <- exp(param[1])
  r <- param[2]

  ratio <- r/(r+mu)

  lnli <- lgamma(r+y) - lgamma(y+1) - lgamma(r) + r*log(ratio) + y*log(1-ratio)
  # lnL contribution for each observation

  s = sum(lnli)
  if (trace) {
    print(c(mu, r, s))
  }
  s
}

fitdry3 <- optim(c(log(4), 5), lnlNB1, y = oakD$dry, method = 'BFGS',
               control = list(fnscale = -1), hessian = T)

# the mle's
fitdry3$par

## [1] 1.327835 3.178792
fitdry2$value

## [1] -51.55232
fitdry3$value

## [1] -51.55232
```

(e)

Two ways:

i

```
fitdry3$par[1]

## [1] 1.327835
vc3 <- solve(-fitdry3$hessian)
se3 <- sqrt(diag(vc3))[1]
se3

## [1] 0.1623194
# the 95% CI for the log mean is
fitdry3$par[1] + c(-1, 1)*se3*qnorm(0.975)

## [1] 1.009695 1.645975
# backtransform, the 95% CI for the mean is
exp(fitdry3$par[1] + c(-1, 1)*se3*qnorm(0.975))

## [1] 2.744764 5.186066
```

ii

use glm.nb (The default link function for a Negative Binomial distribution is log)

```
library(MASS)
fitdry4 <- glm.nb(dry ~ 1, data = oakD)
summary(fitdry4)

##
## Call:
## glm.nb(formula = dry ~ 1, data = oakD, init.theta = 3.179911507,
##        link = log)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.3278      0.1623   8.181 2.82e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(3.1799) family taken to be 1)
##
## Null deviance: 26.084 on 21 degrees of freedom
## Residual deviance: 26.084 on 21 degrees of freedom
## AIC: 107.1
##
## Number of Fisher Scoring iterations: 1
##
##
##              Theta:  3.18
##            Std. Err.:  1.94
##
## 2 x log-likelihood: -103.105
```

```
# the 95% CI for the log mean is
fitdry4$coefficients + c(-1, 1)*0.1623*qnorm(0.975)
```

```
## [1] 1.009696 1.645900
```

```
# backtransform, the 95% CI for the mean is
exp(fitdry4$coefficients + c(-1, 1)*0.1623*qnorm(0.975))
```

```
## [1] 2.744766 5.185677
```

(f)

```
bothfit.m0 <- glm.nb(oak ~ 1, data = both)
bothfit.m1 <- glm.nb(oak ~ woodland, data = both)

tes <- -2*(logLik(bothfit.m0) - logLik(bothfit.m1))
tes
```

```
## 'log Lik.' 2.652844 (df=2)
```

```
pchisq(tes, df = 1, lower.tail = F)
```

```
## 'log Lik.' 0.1033648 (df=2)
```

(g)

```
summary(bothfit.m1)
```

```
##
## Call:
## glm.nb(formula = oak ~ woodland, data = both, init.theta = 1.871455084,
## link = log)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.3278    0.1906   6.966 3.27e-12 ***
## woodlandmoist  0.4547    0.2745   1.656  0.0977 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(1.8715) family taken to be 1)
##
## Null deviance: 48.871 on 39 degrees of freedom
## Residual deviance: 46.112 on 38 degrees of freedom
## AIC: 213.22
##
## Number of Fisher Scoring iterations: 1
##
##
##             Theta:  1.871
##             Std. Err.:  0.637
##
## 2 x log-likelihood: -207.221
```

```
-0.4547 + c(-1, 1)*0.2745*qnorm(0.975)
```

```
## [1] -0.99271011 0.08331011
```

Or using emmeans()

```
library(emmeans)
```

```
emm.m1 <- emmeans(bothfit.m1, 'woodland', infer = c(T, T))
```

```
pairs(emm.m1, infer = c(T, T))
```

```
## contrast estimate SE df asymp.LCL asymp.UCL z.ratio p.value
```

```
## dry - moist -0.455 0.275 Inf -0.993 0.0834 -1.656 0.0977
```

```
##
```

```
## Results are given on the log (not the response) scale.
```

```
## Confidence level used: 0.95
```

(h)

```
exp(-0.4547)
```

```
## [1] 0.6346383
```

```
exp(-0.4547 + c(-1, 1)*0.2745*qnorm(0.975))
```

```
## [1] 0.370571 1.086879
```

```
# or
```

```
pairs(emm.m1, infer = c(T, T), type = 'response')
```

```
## contrast ratio SE df asymp.LCL asymp.UCL null z.ratio p.value
```

```
## dry / moist 0.635 0.174 Inf 0.371 1.09 1 -1.656 0.0977
```

```
##
```

```
## Confidence level used: 0.95
```

```
## Intervals are back-transformed from the log scale
```

```
## Tests are performed on the log scale
```

2.

(a)

```
library(mvabund) # for manyglm() and related functions
```

```
library(dplyr) # for data manipulation operations
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:MASS':
```

```
##
```

```
## select
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
## filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
## intersect, setdiff, setequal, union
```



```
worms <- read.csv("worms.csv")

worms.mva <- mvabund(
  worms %>% select(-treatment, -time, -group))

worms <- worms %>% mutate(group.f = factor(group))

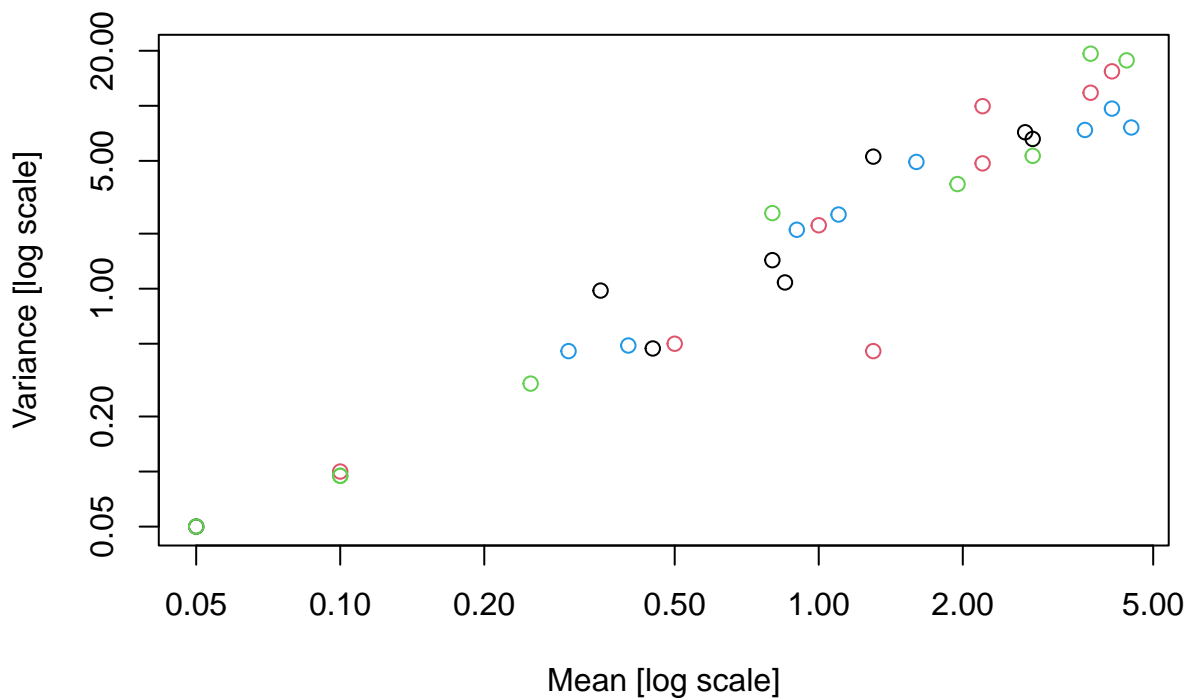
meanvar.plot(worms.mva ~ worms$group.f)
```

```
## START SECTION 2
## Plotting if overlay is TRUE

## using grouping variable worms$group.f 3 mean values were 0 and could
##                                     not be included in the log-plot

## using grouping variable worms$group.f 3 variance values were 0 and could not
##                                     be included in the log-plot
```

mean-var plot, worms\$group.f



```
## FINISHED SECTION 2
```

(b)

```
worms.glm.nb <- manyglm(worms.mva ~ group.f, data = worms)
worms.glm.poi <- manyglm(worms.mva ~ group.f, family = "poisson", data = worms)

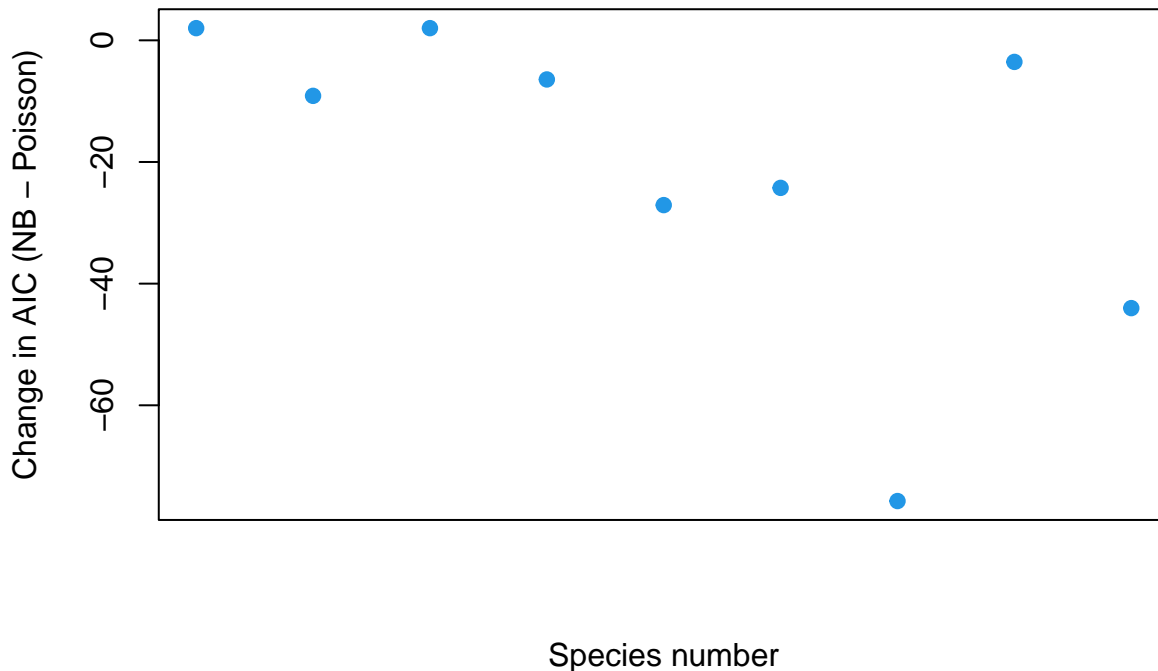
sum(AIC(worms.glm.nb))

## [1] 1487.723

sum(AIC(worms.glm.poi))
```

```
## [1] 1673.949
```

```
plot(AIC(worms.glm.nb) - AIC(worms.glm.poi), pch = 19, col = 4,  
     xlab = 'Species number', ylab = 'Change in AIC (NB - Poisson)',  
     xaxt = 'n')
```



(c)

```
worms <- worms %>% mutate(time.f = factor(time),  
                          treatment.f = factor(treatment))
```

```
worms.glm <- manyglm(worms.mva ~ time.f * treatment.f, data = worms)  
anova(worms.glm)
```

```
## Time elapsed: 0 hr 0 min 10 sec
```

```
## Analysis of Deviance Table
```

```
##
```

```
## Model: worms.mva ~ time.f * treatment.f
```

```
##
```

```
## Multivariate test:
```

```
##
```

	Res.Df	Df.diff	Dev	Pr(>Dev)
(Intercept)	59			
time.f	58	1	31.64	0.006 **
treatment.f	57	1	21.38	0.034 *
time.f:treatment.f	56	1	7.41	0.597

```
## (Intercept)
```

```
## time.f
```

```
## treatment.f
```

```
## time.f:treatment.f
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Arguments:
```

```
## Test statistics calculated assuming uncorrelated response (for faster computation)
```

```
## P-value calculated using 999 iterations via PIT-trap resampling.
```

(e)

```
worms.glm2 <- manyglm(worms.mva ~ time.f, data = worms)
```

```
temp2 <- anova(worms.glm2, p.uni='adjusted')
```

```
## Time elapsed: 0 hr 0 min 3 sec
```

```
which(temp2$uni.p[2,] < 0.05)
```

```
## Ali
```

```
## 8
```