Krysten Thompson - w271: Homework 4

Professor Jeffrey Yau

Question 18 a and b of Chapter 3 (page 192,193)

For the wheat kernel data (*wheat.csv*), consider a model to estimate the kernel condition using the density explanatory variable as a linear term.

a. Write an R function that computes the log-likelihood function for the multinomial regression model. Evaluate the function at the parameter estimates produced by multinom(), and verify that your computed value is the same as that produced by logLik() (use the object saved from multinom() within this function).

Formulas that will be used in this problem:

$$P(N_1 = n_1, ...N_J = n_J) = \frac{n!}{\prod_{j=1}^{J} n_j!} \prod_{j=1}^{J} \pi_j^{n_j}$$

$$log(\hat{\pi}_{Scab}/\hat{\pi}_{Healthy}) = Intercept - \beta_1 Density$$

$$log(\hat{\pi}_{Sprout}/\hat{\pi}_{Healthy}) = Intercept - \beta_1 Density$$

```
library(nnet)
wheat <- read.csv('wheat.csv', header=TRUE)
#head(wheat)
```

```
mod.fit <- multinom(formula = type ~ density, data=wheat)
```

```
## # weights:  9 (4 variable)
## initial  value 302.118379
## iter  10 value 229.769334
## iter  20 value 229.712304
## final  value 229.712290
## converged
```

```
summary(mod.fit)
```

```
## Call:
## multinom(formula = type ~ density, data = wheat)
##
## Coefficients:
##        (Intercept)   density
## Scab      29.37827 -24.56215
## Sprout    19.12165 -15.47633
##
## Std. Errors:
##        (Intercept)  density
## Scab      3.676892 3.017842
```

```
## Sprout    3.337092 2.691429
##
## Residual Deviance: 459.4246
## AIC: 467.4246
```

```
logL = function(beta, x, Y){

  # Find pi
  pi_healthy = 1/(1 + exp(beta[1] + beta[3]*x) + exp(beta[2] + beta[4]*x))
  pi_scab = exp(beta[1] + beta[3]*x) * pi_healthy
  pi_sprout = exp(beta[2] + beta[4]*x) * pi_healthy

  # Find the Ys
  Y_healthy = ifelse(Y == "Healthy", 1, 0)
  Y_sprout = ifelse(Y == "Sprout", 1, 0)
  Y_scab = ifelse(Y == "Scab", 1, 0)

  sum(Y_healthy*log(pi_healthy) + Y_sprout*log(pi_sprout) + Y_scab*log(pi_scab))
}

logL(coef(mod.fit), wheat$density, wheat$type)
```

```
## [1] -229.7123
```

```
logLik(mod.fit)
```

```
## 'log Lik.' -229.7123 (df=4)
```

---

b. Maximize the log-likelihood function using optim() to obtain the MLEs and the estimated covariance matrix. Compare your answers to what is obtained by multinom(). Note that to obtain starting values for optim(), one approach is to estimate separate logistic regression models for $log\left(\frac{\pi_2}{\pi_1}\right)$ and $log\left(\frac{\pi_3}{\pi_1}\right)$. These models are estimated only for those observations that have the corresponding responses (e.g., a $Y = 1$ or $Y = 2$ for $log\left(\frac{\pi_2}{\pi_1}\right)$).

```
mod.fit.optim <- optim(par = summary(mod.fit)$coefficients, fn = logL,
                       hessian = TRUE, x = wheat$density, Y = wheat$type, control = list(fnscal
```

```
print('Parmeter estimates:')
```

```
## [1] "Parmeter estimates:"
```

```
mod.fit.optim$par
```

```
##         (Intercept)   density
## Scab       29.37828 -24.56215
## Sprout     19.12165 -15.47634
```

```
cat("\n")
```

```r
print('Maximum value for log likelihood function:')
```

```
## [1] "Maximum value for log likelihood function:"
```

```r
mod.fit.optim$value
```

```
## [1] -229.7123
```

```r
cat("\n")
```

```r
print('0 means convergence achieved:')
```

```
## [1] "0 means convergence achieved:"
```

```r
mod.fit.optim$convergence
```

```
## [1] 0
```

```r
cat("\n")
```

```r
print('Estimated covariance matrix:')
```

```
## [1] "Estimated covariance matrix:"
```

```r
-solve(mod.fit.optim$hessian)
```

```
##            [,1]      [,2]       [,3]      [,4]
## [1,]  13.519523 10.325081 -11.076930 -8.272568
## [2,]  10.325081 11.136186  -8.328093 -8.970707
## [3,] -11.076930 -8.328093   9.107362  6.681949
## [4,]  -8.272568 -8.970707   6.681949  7.243799
```