Jonothan Meyer
Linear Regression
05/07/21
Project 9

Exercise 1 (5 points): Consider the spector data set in the faraway package. Use a logistic regression model to model grade as a function of all other variables.

First a full logistic regression model was built with grade as a function of the other variables to get a baseline of AIC for the data using this response. The full model was built using the following commands in R:

```
head(spector)
names(spector)
grade.lr <- glm(grade~., data = spector, family = "binomial")
```

The AIC for the full logistic regression model was found to be 33.78.

(1) Identify the sub-model that still includes psi as a predictor and minimizes the AIC for this logistic regression model.

After using the step function on the full model the following sub-model was found to yield the lowest AIC:

```
sub.grade.lr <- glm(formula = grade ~ psi + gpa, family = "binomial", data = spector)
```

This model removes the predictor 'tuce' and has an AIC of 32.25. This is a slight improvement over the full model.

(2) Identify the sub-model that still includes psi as a predictor and minimizes the standard error associated with the estimate of the coefficient for psi.

Removing all predictors other than 'psi' was found to minimize the standard error associated with the estimate of the coefficient for psi, which was minimized to .8317. In the previous model the estimated standard error for psi was found to be 1.041. This is an improvement of .2093. The R commands below were used to build this model and interpret the results.

```
summary(glm(grade~psi, family = "binomial", data = spector))
```

(3) Interpret the psi coefficient from the sub-models from the first two parts in terms of real-world odds ratios.

The coefficient for psi in the first sub-model was found to be 2.338. Which means that when all other variables are constant that a student that was exposed to the new teaching method called "PSI" was e^(2.338) = 10.36 times more likely to have improved exam grades than a student that was not exposed to PSI. The coefficient for pis in the second sub-model was found to be 1.9. This means that a student that was exposed to the new teaching method called "PSI" was e^(1.9) = 6.67 times more likely to have improved exam grades than a student that was not exposed to PSI.

Exercise 2 (5 points): Consider the ships data set in the MASS package. Use a Poisson regression model to model incident as a function of all other variables.
(1) Identify the sub-model that minimizes the AIC for this logistic regression model.

After loading the 'MASS' package the full Poisson regression model was built with 'incident' as the response in order to understand initial baseline AIC as well as develop an understanding of significant and insignificant variables. The full model yields an AIC of 278.86, and in order for a different model to be considered it must yield a lower AIC than the full model. On first appears it seems as though the full model includes 4 predictor variables, however since 'type' is a categorical variable that contains 5 levels. This model interprets the different levels as separate coefficients, and therefore are eligible for removal if it improves the AIC. When the step() function was performed on the full model it demonstrated that the AIC would not be improved from removal of any of the predictor variables, excluding the different factor levels of the 'type' variable. Each level of the type level was filtered out prior to building the model to investigate the resulting AIC. When filtering out 'type A' the AIC = 234.88, 'type B' AIC = 143.39, 'type C' AIC = 252.93, 'type D' AIC = 248.13, 'type E' AIC = 228.71. Clearly the AIC was most improved from filtering out 'type B' from the full model. When the step function was run on this model the removal of variables 'period' and 'year' improved the AIC further. The best sub-model that minimizes the AIC for this logistic regression model can be observed in R code found below.

```
ship <- ships %>% filter(type != "B")
glm(formula = incidents ~ service + type, family = poisson, data = ship)
```

The final AIC using this sub-model was found to be exactly 140. A great improvement in AIC from the full model.

(2) Now consider all possible interactions of predictors in your Poisson regression model. Does the inclusion of the interaction term lower the AIC below the model chosen in
the previous part?

If we include all possible interactions of predictors in our full Poisson regression model (including 'type B') the AIC is not lower than the previous sub-model AIC. This model was built and investigated using the following R code:

```
shipsfull.glm <- glm(incidents~year+period+service+type, data = ships, family = poisson)
stepAIC(shipsfull.glm, scope=list(upper= ~year*period*service*type, lower= ~1))
shipsfull.int.glm <- glm(glm(formula = incidents ~ year + period + service + type + year:service +
                          service:type + period:service, family = poisson, data = ships), family = poisson)
summary(shipsfull.int.glm)
```

Not only is this model far less parsimonious than any other model discussed, it also has a fairly high AIC of 206.6.

(3) Identify the sub-model of the interactions model consider in the previous part that minimizes the AIC for this logistic regression model.

If we again exclude 'type B' in a sub-model, but include interactions then a lower AIC was found than the previous sub model. To investigate the best model the step function was again used and the best model can be observed in the following R code:

```
ship <- ships %>% filter(type != "B")
ship.sub.inter <- glm(formula = incidents ~ period + service + type + period:service,
                      family = poisson, data = ship)
```

This model was found to have an AIC of 135.4. This model improves the AIC by 4.6 units over over the previously found best model.

Exercise 3 (5 points): Consider the globwarm data set. Use the response variable nhtemp and select three proxy variables as predictors.

(1) Using the data from 1856 and on, build a GLM using the three proxy variables as predictors. Use a Durban Watson test to test if ρ = 0.

Using the globwarm data set from after the year 1856 a GLM was built using 'wusa', 'jasper', and 'westgreen' as predictors. The data wrangling, construction of GLM, and Durban Watson test were performed using the following R commands:

```
red_globwarm <- globwarm %>% filter(year >= 1856)
temp.glm <- glm(nhtemp~wusa+jasper+westgreen, data = red_globwarm)
summary(temp.glm)
durbinWatsonTest(temp.glm)
```

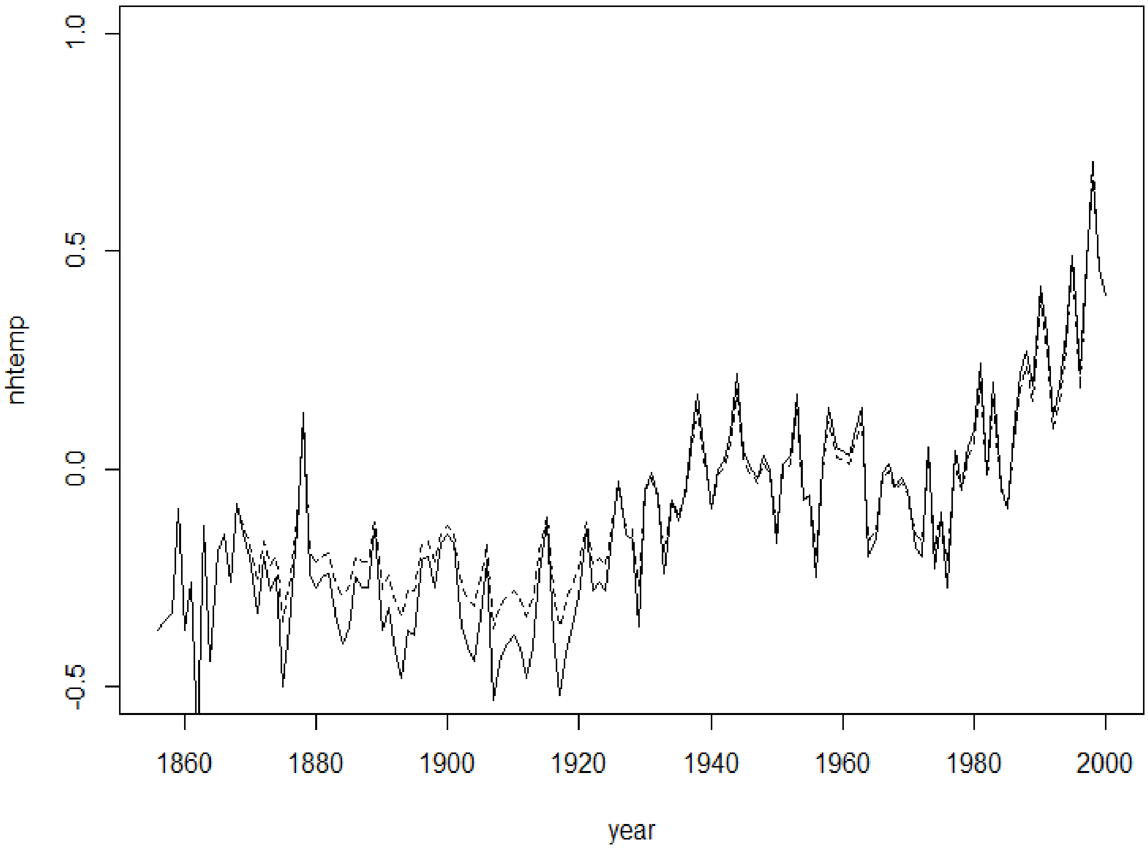
The Durban Watson test determined that ρ = 0 when using 'wusa', 'jasper', and 'westgreen' as predictors.

(2) Using the data from 1856 and on, build a lag 1 autoregressive model using only previous year's nhtemp as a predictor. Complete this using code similar to Faraway's code on page 55. Use a Durban-Watson test to test if ρ = 0.

Using the previous filtered data a lag 1 autoregressive model was built using only previous year's nhtemp as a predictor with the following R commands:

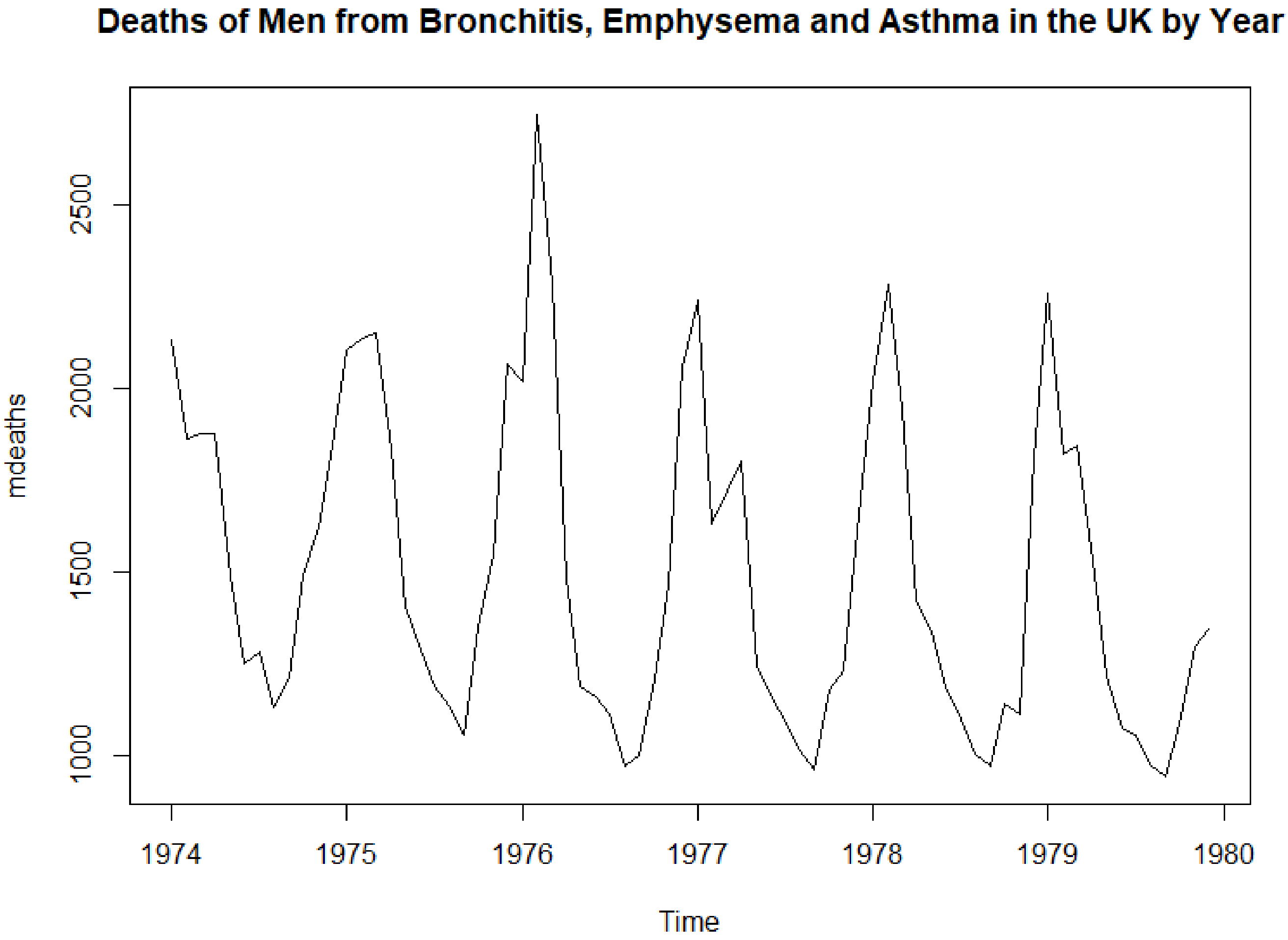
```
lagdf <- embed(red_globwarm$nhtemp,14)
colnames(lagdf) <- c("y",paste0("lag",1:13))
lagdf <- data.frame(lagdf)
globmod <- lm(y ~ lag1, data.frame(lagdf))
summary(globmod)
plot(nhtemp~year, red_globwarm, type="l", ylim=c(-.5, 2))
ypred <- exp(predict(globmod)) - 1
lines(red_globwarm$year[13:144], ypred, lty=2)
durbinWatsonTest(globmod)
```

The Durban Watson test on this lag 1 autoregressive model yeilded a ρ != 0 that was consistent across several test iterations. Typically the ρ value was between (.06, .1). The plot below illustrates how well this model fits the data where the solid line is the actual data, and the dotted line is our model.



Exercise 4 (5 points): Complete Textbook Exercise 4 from Chapter 4 (page 57), stated as follows: The dataset mdeaths reports the number of deaths from lung diseases for men in the UK from 1974 to 1979.

(1) Make an appropriate plot of the data. At what time of year are deaths most likely to occur?



Based on this plot it seems as though most deaths occurred around New Years. December and January appear to be the most deadly months.

(2) Fit an autoregressive model of the same form used for the airline data in Section 4.3. Are all the predictors statistically significant?

The following R commands were used to fit an autoregressive model using the mdeaths data set:

```
lagdf <- embed(log(mdeaths), 14)
colnames(lagdf) <- c("y", paste0("lag", 1:13))
dmod <- lm(y ~ lag1+lag2+lag3+lag4+lag5+lag6+lag7+lag8+lag9+lag10+lag11+lag12+lag13,
data.frame(lagdf))
summary(dmod)
```

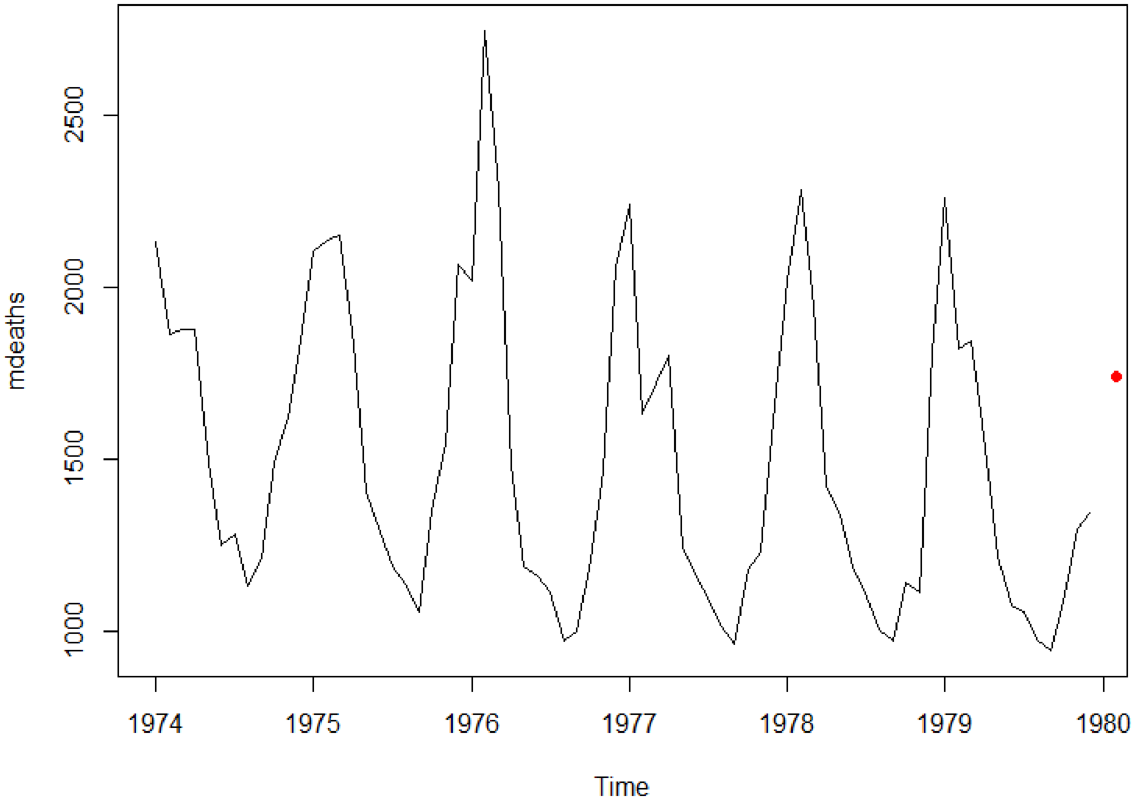
All predictors yielded a p-value > 0.5 other than 'lag1' and 'lag13' which were found to have p-values of 0.0137 and 0.023 respectively. According to this model only the 'lag1' and lag13' predictors appear to be statistically significant. After initially looking at the plot, this result is not unexpected and supports my initial inclination that most deaths occur right at the end and the beginning of the year.

(3) Use the model to predict the number of deaths in January 1980 along with a 95% prediction interval.

First a new autoregressive model was built that only used the predictors that were found to be significant. Then by plugging this model in the predict function in conjunction with shifted coefficients from the model we are able to get the transformed fit prediction value inside a 95% prediction interval, which was found to be 7.32. By then plugging this value, newly shifted coefficient values, and the model into the predict function we are able to predict the number of deaths in January 1980. Before this number is interpreted it must be transformed back to its original form. The following R commands were used to predict the number of deaths in January 1980:

```
dmod <- lm(y ~ lag1+lag13, data.frame(lagdf)) #Building model with significant predictors
lagdf[nrow(lagdf),] #Finding needed coefficients to build appropriate prediction interval
pred_int <- predict(dmod, data.frame(lag1=7.201171, lag13=7.502168),
          interval="prediction") #Finding 95% prediction interval
exp(pred_int) #Converting prediction interval back to real world values
jan1980 <- predict(dmod, data.frame(lag1=7.317722, lag13= 7.724447)) #Estimating death in Jan 1980
exp(jan1980) #Converting prediction back to real world value
```

The 95% prediction interval for the number of estimated deaths in the UK using our model for January 1980 is (1059, 2143). We estimate that in the month of January 1980 that 1740 deaths will occur. The red dot below illustrates this estimation.


mdeaths
Time