Homework 3

Alex Soupir

September 12, 2019

Packages: HSAUR3, ISLR, boot

Collaborators:

Answer all questions specified on the problem and include a discussion on how your results answered/addressed the question.

Submit your **.rmd** file with the knitted **PDF** (or knitted Word Document saved as a PDF). If you are having trouble with .rmd, let us know and we will help you, but both the .rmd and the PDF are required.

This file can be used as a skeleton document for your code/write up. Please follow the instructions found under Content for Formatting and Guidelines. No code should be in your PDF write-up unless stated otherwise.

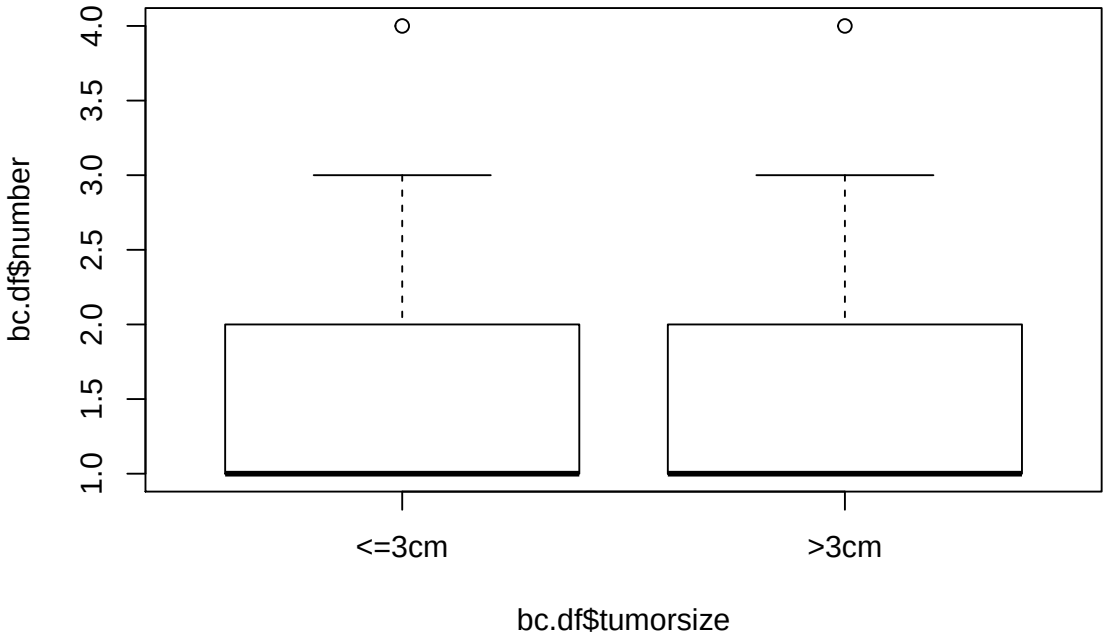
For any question asking for plots/graphs, please do as the question asks as well as do the same but using the respective commands in the GGPLOT2 library. (So if the question asks for one plot, your results should have two plots. One produced using the given R-function and one produced from the GGPLOT2 equivalent). This doesn't apply to questions that don't specifically ask for a plot, however I still would encourage you to produce both.

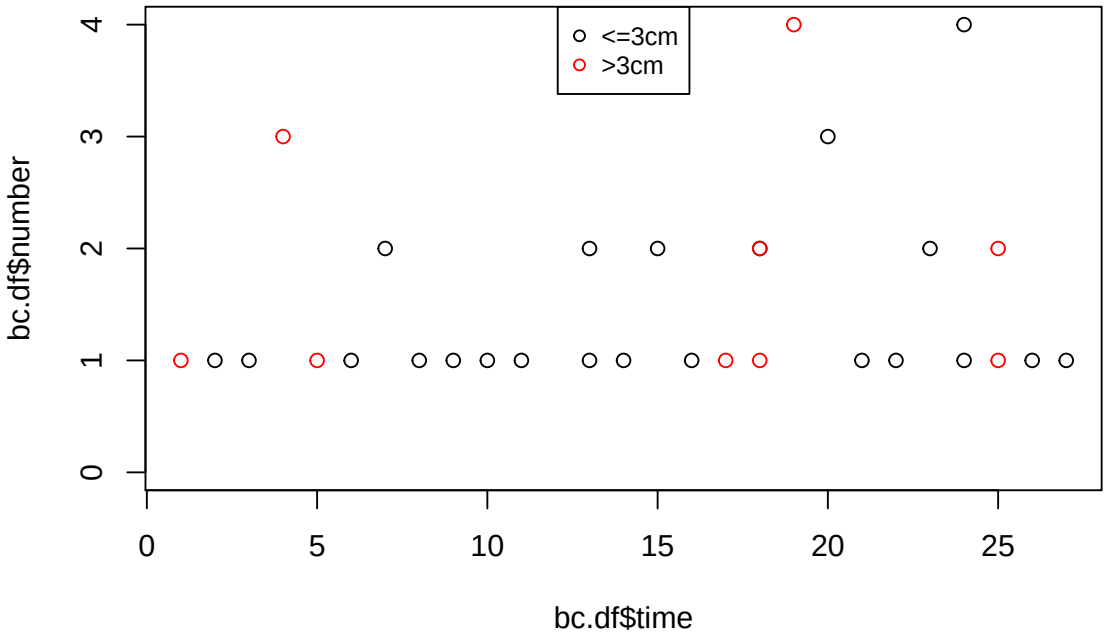
You do not need to include the above statements.

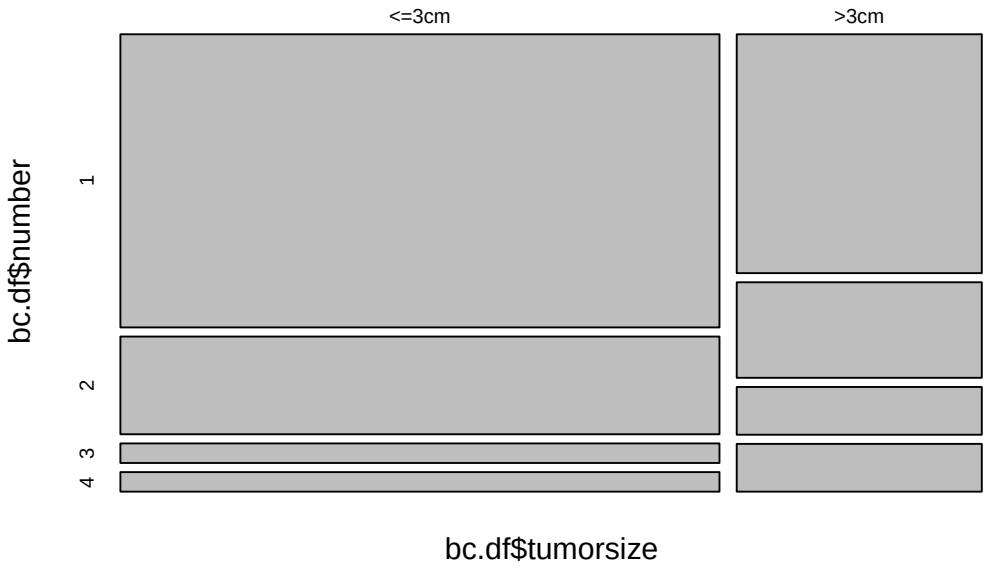
Please do the following problems from the text book R Handbook and stated.

1. Use the **bladdercancer** data from the **HSAUR3** library to answer the following questions

    a) Construct graphical and numerical summaries that will show the relationship between tumor size and the number of recurrent tumors. Discuss your discovery. (Hint: mosaic plot may be a great way to assess this)


bc.df$number
1.0
1.5
2.0
2.5
3.0
3.5
4.0
<=3cm
>3cm
bc.df$tumorsize


bc.df$number
0
1
2
3
4
<=3cm
>3cm
0
5
10
15
20
25
bc.df$time



A boxplot of the tumour size against the number of recurrent tumours shows that the distribution of is similar between the number of recurrent tumours and the size of the primary tumour. The median number of recurrent tumours was 1 for both size categories for the primary tumour.

Looking at a scatter plot which is plotting the number of tumours against the time after the removal of the primary tumour (split by size of the primary tumour) shows that none of the recorded patients had shown 0 recurrent tumours. It can also be seen that there are much less recorded observations following a patient with a tumour size greater than 3 cm.

Finally a mosaic plot was created (as recommended) and against the number of observations where primary tumours were less than or equal to 3 cm in size is more than those with tumours greater than 3 cm in diameter. The mosaic plot also shows the distribution within each better than the boxplot since the mosaic plot breaks up the plot by the number of tumours. The number of individuals that have 1 recurrent tumour is about half for both groups of tumour classification.

```
##       time       tumorsize      number
##  Min.   : 2.00   <=3cm:22   Min.   :1.000
##  1st Qu.: 9.25   >3cm : 0   1st Qu.:1.000
##  Median :14.50              Median :1.000
##  Mean   :15.09              Mean   :1.455
##  3rd Qu.:21.75              3rd Qu.:2.000
##  Max.   :27.00              Max.   :4.000
```

```
##       time       tumorsize     number
##  Min.   : 1.00   <=3cm:0   Min.   :1.000
##  1st Qu.: 5.00   >3cm :9   1st Qu.:1.000
##  Median :18.00             Median :1.000
```

```
##  Mean   :14.67             Mean   :1.778
##  3rd Qu.:19.00             3rd Qu.:2.000
##  Max.   :25.00             Max.   :4.000
```

The range in summary statistics, the range of both primary tumour categories is from 1 to 4 recurrent tomours. The average number of recurrent tumours in patients with a primary tumour greater than 3 cm is 1.778 whereas the average number of recurrent tumours in patients with a primary tumour less than 3 cm is 1.455. However, the median number of tumours is 1 for both categories, just like the boxplot shows.

b) Build a Poisson regression that estimates the effect of size of tumor on the number of recurrent tumors. Discuss your results.

```
##
## Call:
## glm(formula = number ~ tumorsize, family = poisson(), data = bc.df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.6363  -0.3996  -0.3996   0.4277   1.7326
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)     0.3747     0.1768   2.120    0.034 *
## tumorsize>3cm   0.2007     0.3062   0.655    0.512
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 12.80  on 30  degrees of freedom
## Residual deviance: 12.38  on 29  degrees of freedom
## AIC: 87.191
##
## Number of Fisher Scoring iterations: 4
```

```
##
## Call:
## glm(formula = number ~ tumorsize + time, family = poisson(),
##     data = bc.df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.8183  -0.4753  -0.2923   0.3319   1.5446
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.14568    0.34766   0.419    0.675
## tumorsize>3cm  0.20511    0.30620   0.670    0.503
## time           0.01478    0.01883   0.785    0.433
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 12.800  on 30  degrees of freedom
## Residual deviance: 11.757  on 28  degrees of freedom
## AIC: 88.568
```

```
##
## Number of Fisher Scoring iterations: 4
```

```
##
## Call:
## glm(formula = number ~ tumorsize + time + tumorsize * time, family = poisson(),
##     data = bc.df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.6943  -0.5581  -0.2413   0.2932   1.4644
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)          0.03957    0.43088   0.092    0.927
## tumorsize>3cm        0.46717    0.66713   0.700    0.484
## time                 0.02138    0.02418   0.884    0.377
## tumorsize>3cm:time -0.01676    0.03821  -0.439    0.661
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 12.800  on 30  degrees of freedom
## Residual deviance: 11.566  on 27  degrees of freedom
## AIC: 90.377
##
## Number of Fisher Scoring iterations: 4
```

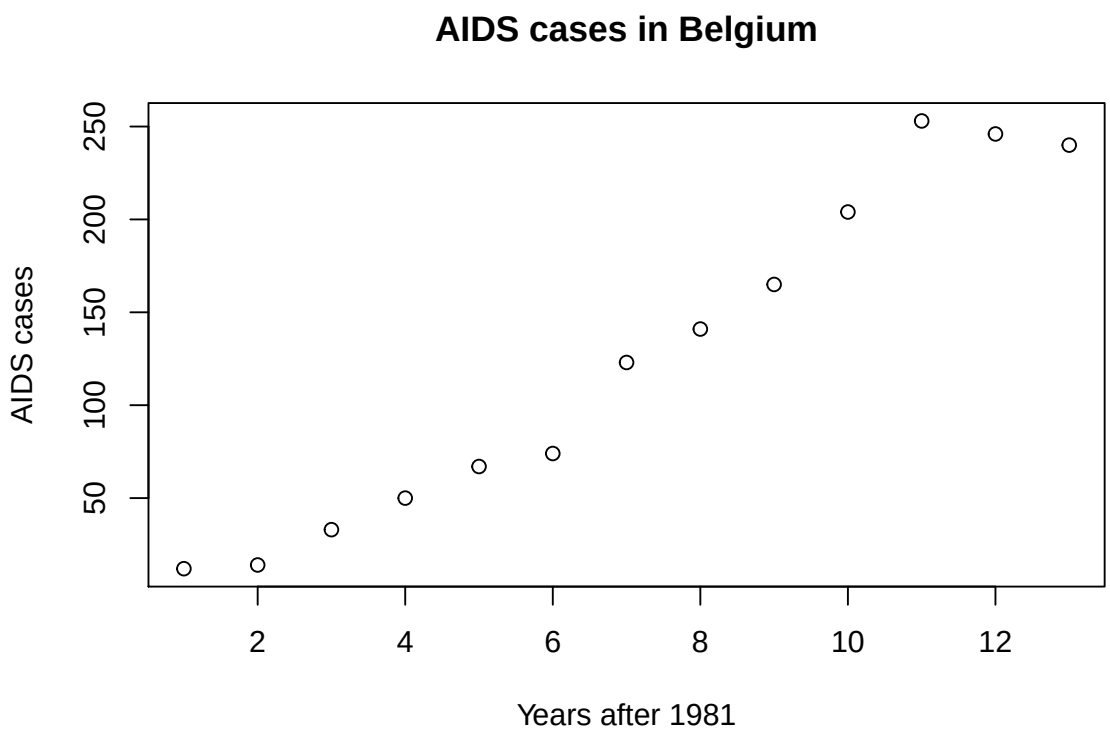
The models created were done using different combinations of the tumour size and time after the primary tumour was removed to determine how many recurrent tumours there will be. The tumour size coefficient wasn't significantly different than zero for any of the models, nor was time. The Residual deviance was about 2.5x lower than the degrees of freedom for all 3 models, which is better than the lecture example but it is actually lower than the degrees of freedom. The AIC is lowest for the model using both *time* and *tumorsize*.

2. The following data is the number of new AIDS cases in Belgium between the years 1981-1993. Let t denote time

```
y = c(12, 14, 33, 50, 67, 74, 123, 141, 165, 204, 253, 246, 240)
t = 1:13
```

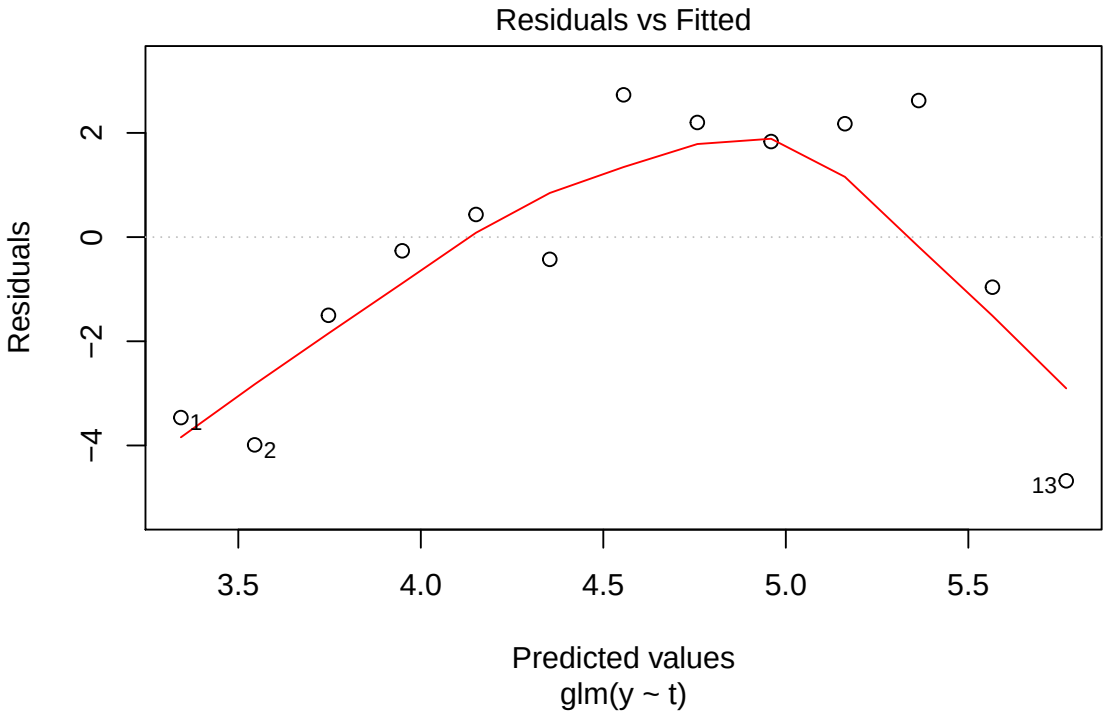
Do the following

a) Plot the relationship between AIDS cases against time. Comment on the plot



b) Fit a Poisson regression model $log(\mu_i) = \beta_0 + \beta_1 t_i$. Comment on the model parameters and residuals (deviance) vs Fitted plot.

```
##
## Call:
## glm(formula = y ~ t, family = poisson())
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.6784  -1.5013  -0.2636   2.1760   2.7306
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 3.140590   0.078247   40.14   <2e-16 ***
## t           0.202121   0.007771   26.01   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 872.206  on 12  degrees of freedom
## Residual deviance:  80.686  on 11  degrees of freedom
## AIC: 166.37
##
## Number of Fisher Scoring iterations: 4
```

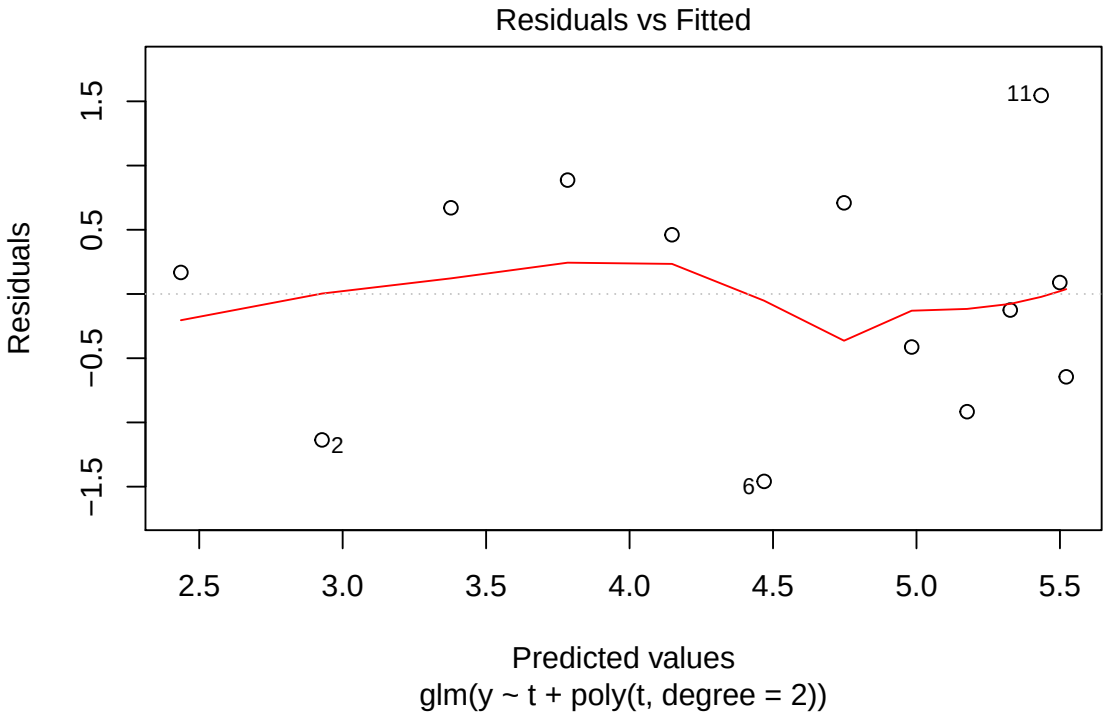


The time metric seems to be significantly different than zero (p < 2x10^-16) when predicting the number of AIDS cases. The residual deviance is incredibly high, though, at 80.7 on 11 degrees of freedom. In lecture, this was stated to be do to over dispersion of the poisson distribution. The plot shows the residual vs fitted for the model and since the points don't 'bounce randomly' around the residual of zero, the assumption that the relationship is linear is not reasonable.

c) Now add a quadratic term in time (*i.e.*, $log(\mu_i) = \beta_0 + \beta_1 t_i + \beta_2 t_i^2$) and fit the model. Comment on the model parameters and assess the residual plots.

```
##
## Call:
## glm(formula = y ~ t + poly(t, degree = 2), family = poisson())
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.45903  -0.64491   0.08927   0.67117   1.54596
##
## Coefficients: (1 not defined because of singularities)
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)           2.64858    0.11323  23.390  < 2e-16 ***
## t                     0.25716    0.01167  22.038  < 2e-16 ***
## poly(t, degree = 2)1       NA         NA      NA       NA
## poly(t, degree = 2)2 -0.95511    0.11896  -8.029 9.82e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
```

```
##     Null deviance: 872.2058  on 12  degrees of freedom
## Residual deviance:   9.2402  on 10  degrees of freedom
## AIC: 96.924
##
## Number of Fisher Scoring iterations: 4
```



This model shows that the time coefficient, both linear term and quadratic term, are significantly different from zero. The residual deviance is also much closer to that of the degrees of freedom (9.24 on 10 DOF). This low difference means that there is less overdispersion. Looking at the plot of residual vs fitted is much more balanced around 0 which indicates less error in the model.

d) Compare the two models using AIC. Which model is better?

The first model without the quadratic term had an AIC of 166.37, while the second model with a 2 order polynomial had an AIC of 96.924. This value also shows that the second model fits the data better just like the residual deviance and plot had shown.

e) Use *anova()*-function to perform χ^2 test for model selection. Did adding the quadratic term improve model?

The chi-squared test indicates that the 2 models are significantly different from one-another and that adding the quadratic term did in fact improve the model. The chi-sq also shows the residual deviances and difference which is handy.

3. Load the **Default** dataset from **ISLR** library. The dataset contains information on ten thousand customers. The aim here is to predict which customers will default on their credit card debt. It is a 4 dimensional dataset with 10000 observations. You had developed a logistic regression model on HW #2. Now consider the following two models

- Model1 → Default = Student + balance

- Model2 → Default = Balance

For the two competing models do the following

a) With the whole data compare the two models (Use AIC and/or error rate)

```
## [1] 0.02130176
```

```
## [1] 0.02170579
```

```
##
## Call:
## glm(formula = default ~ student + balance, family = binomial(),
##     data = default.df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4578  -0.1422  -0.0559  -0.0203   3.7435
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.075e+01  3.692e-01 -29.116  < 2e-16 ***
## studentYes  -7.149e-01  1.475e-01  -4.846 1.26e-06 ***
## balance      5.738e-03  2.318e-04  24.750  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2920.6  on 9999  degrees of freedom
## Residual deviance: 1571.7  on 9997  degrees of freedom
## AIC: 1577.7
##
## Number of Fisher Scoring iterations: 8
```

```
##
## Call:
## glm(formula = default ~ balance, family = binomial(), data = default.df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2697  -0.1465  -0.0589  -0.0221   3.7589
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.065e+01  3.612e-01  -29.49   <2e-16 ***
## balance      5.499e-03  2.204e-04   24.95   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2920.6  on 9999  degrees of freedom
## Residual deviance: 1596.5  on 9998  degrees of freedom
## AIC: 1600.5
##
## Number of Fisher Scoring iterations: 8
```

The mean squared error of the model including both student and balance as predictors had a slightly lower (0.0213 v 0.0217) than the model with just balance as the preditor. Looking at AIC values from both models, the model with both student and balance was lower (AIC = 1577.7) than the model with just the balance (AIC = 1600.5). The residual deviance of both models was close but the student+balance model did acheive a lower value (1571.7 on 9997 DOF vs 1596.5 on 9998 DOF)

b) Use validation set approach and choose the best model. Be aware that we have few people who defaulted in the data.

```
## [1] 0.02435605
```

```
## [1] 0.02512413
```

```
##
## Call:
## glm(formula = default ~ student + balance, family = binomial(),
##     data = train.default)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5024  -0.1327  -0.0502  -0.0176   3.4058
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.116e+01  4.338e-01 -25.714  < 2e-16 ***
## studentYes  -6.764e-01  1.676e-01  -4.035 5.45e-05 ***
## balance      5.956e-03  2.703e-04  22.035  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2323.7  on 8050  degrees of freedom
## Residual deviance: 1204.3  on 8048  degrees of freedom
## AIC: 1210.3
##
## Number of Fisher Scoring iterations: 8
```

```
##
## Call:
## glm(formula = default ~ balance, family = binomial(), data = train.default)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3235  -0.1382  -0.0532  -0.0191   3.3320
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.105e+01  4.241e-01  -26.05   <2e-16 ***
## balance      5.720e-03  2.567e-04   22.28   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 2323.7  on 8050  degrees of freedom
## Residual deviance: 1221.4  on 8049  degrees of freedom
## AIC: 1225.4
##
## Number of Fisher Scoring iterations: 8
```

Using a train/test split of 80/20, the summaries show that the model with student and balance has a slightly lower residual deviance (1204.3 vs 1221.4) than the model that only has balance. The mean squared error for the student + balance model is also slightly lower (0.024 vs 0.025) than the balance model. This difference is incredibly minor, however. All coefficients are significantly different than zero at p<0.05.

c) Use LOOCV approach and choose the best model

```
## [1] 0.0267
```

```
## [1] 0.0275
```

Using the test split from the previous part because it is taking a great deal of time.

The model with student had shown a LOOCV error of 0.0267 while without student as input the LOOCV error was 0.0275. The better of the models here includes the student parameter in the model. These values are slightly further apart than the train/test split method, which is interesting.

d) Use 10-fold cross-validation approach and choose the best model

```
## [1] 0.0266
```

```
## [1] 0.0276
```

The better model of these 2 10-fold cross-validations is again the model with both student and balance as predictors. Similarly to the other approaches, the difference between having student and not having student as an input seems to make little difference.

Report validation misclassification (error) rate for both models in each of the three assessment methods. Discuss your results.

Out of all of the models, the lowest error was in train/test split models. This was this was almost twice as low as the leave one out cross-validation. The highest error came from the model without student as an input in the k-fold cross-validation but it was only 0.0002 higher than the same model in LOO cross-validation.

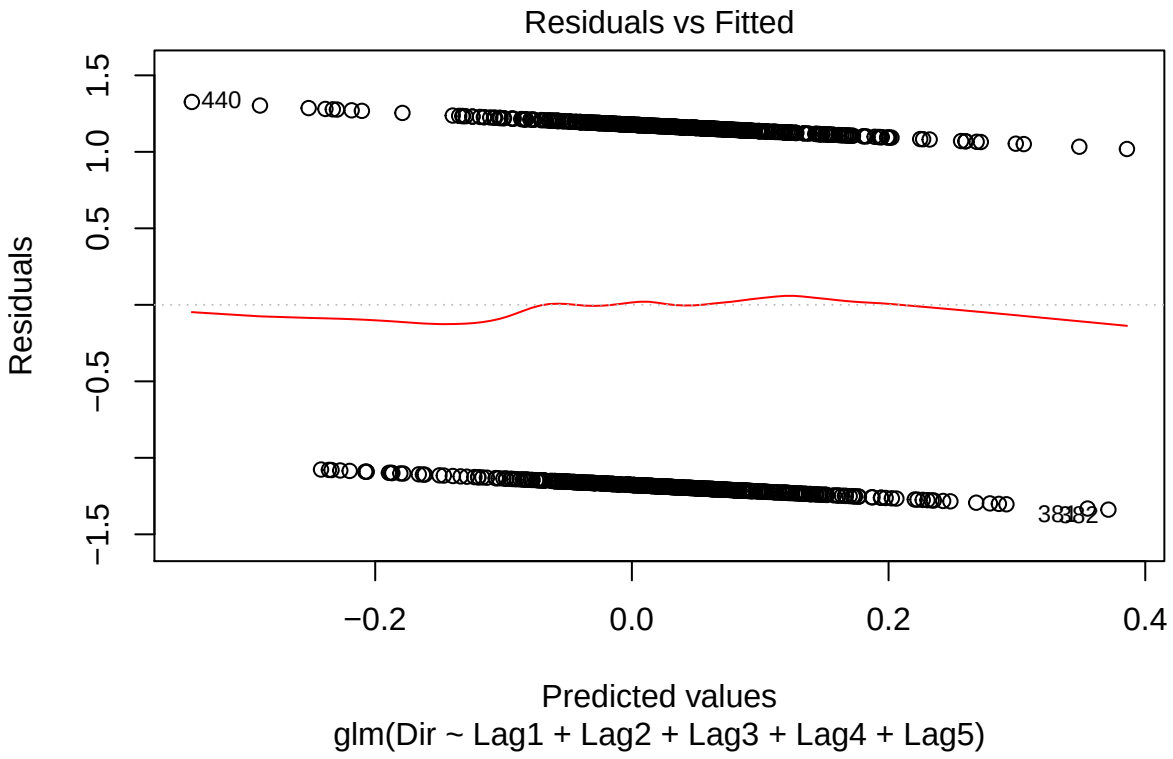
I had an issue with the LOOCV. I was trying to do the loop for the polynomial degree as my CV here, which was completely incorrect and instead changed to the *cv.glm()* as explained later in lecture.

4. In the **ISLR** library load the **Smarket** dataset. This contains Daily percentage returns for the S&P 500 stock index between 2001 and 2005. There are 1250 observations and 9 variables. The variable of interest is Direction which is a factor with levels Down and Up indicating whether the market had a positive or negative return on a given day. Since the goal is to predict the direction of the stock market in the future, here it would make sense to use the data from years 2001 - 2004 as training and 2005 as validation. According to this, create a training set and testing set. Perform logistic regression and assess the error rate.

I'm going to make the assumption that the return of today has less predictive power than the lag values and the volume of shares traded because what the market has done seems more valuable and there are a lot of different combinations that can be made with the other values in the data frame.

Model 7 and model 8 are statistically significant using a chi-sq at p<0.1, with model 7 being slightly more significant. I'll use model 7 since it is the most significant

```
##
## Call:
## glm(formula = Dir ~ Lag1 * Lag2 + Lag2 * Lag3 + Lag3 * Lag4 +
##     Lag4 * Lag5, family = binomial(), data = train.df)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.411  -1.188   1.030   1.163   1.448
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.0329182  0.0634873   0.519    0.604
## Lag1        -0.0522398  0.0525263  -0.995    0.320
## Lag2        -0.0418927  0.0522250  -0.802    0.422
## Lag3         0.0030935  0.0521812   0.059    0.953
## Lag4         0.0128109  0.0522987   0.245    0.806
## Lag5        -0.0030206  0.0515501  -0.059    0.953
## Lag1:Lag2    0.0001411  0.0355783   0.004    0.997
## Lag2:Lag3    0.0145373  0.0348189   0.418    0.676
## Lag3:Lag4   -0.0276171  0.0348350  -0.793    0.428
## Lag4:Lag5    0.0292119  0.0353687   0.826    0.409
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1383.3  on 997  degrees of freedom
## Residual deviance: 1379.8  on 988  degrees of freedom
## AIC: 1399.8
##
## Number of Fisher Scoring iterations: 3
```



This model has each lag day interacting with the next day, e.g. day 1 with day 2 and day 2 with day 3 and so on. Even though its p=0.058, none fo the term coefficients are statistically significant from zero. Looking at the residual vs fitted plot, there appears to be 2 distinct groups distributed fairly equally about Residuals=0.

Train test split error

```
## [1] 0.2480854
```

Leave out one cross-validation error

```
## [1] 0.0275
```

K-fold validation error

```
## [1] 0.5
```

The best model for predicting whether the market will move up or down uses the previous 5 days percentage return. The lowest error came from the train/test split MSE (0.2480854) with LOO cross-validation being only slightly higher 0.0275. The k-fold of 10 has a cross-validation error of 0.5 on the test set, which is interestingly high compared to the other methods.

An interesting thing that I noticed while running the LOO and K-fold cross-validations was that there were a fair amount of errors that get thrown if the same data frame isn't used for testing and training the model, except for problem number 4 when that would return NA but the train and test data frames would produce a value (with errors or warnings). I don't know why this is an issue or how to fix it.

Resources:

- newonlinecourses.science.psu.edu

- StackExchange