Neeraj Asthana
nasthan2
CS 498: Applied Machine Learning
HW6 Report

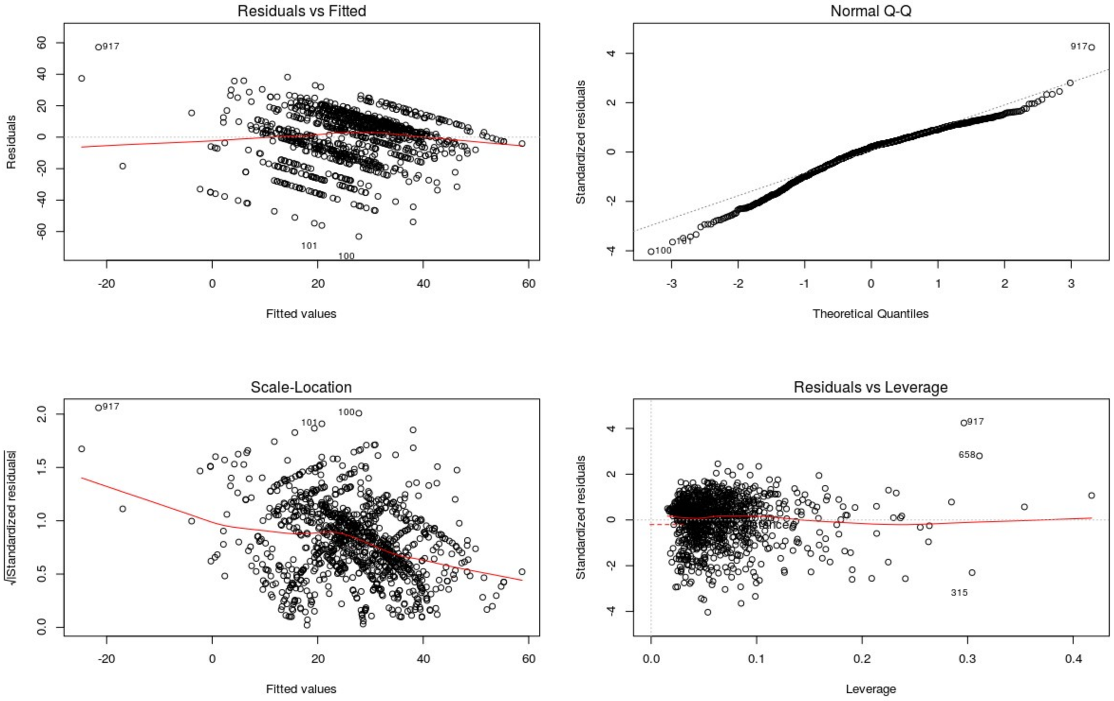
Problem 1:

I used the “default_features_1059_tracks.txt” dataset to solve these problems. This dataset has 70 features or predictors and 1059 examples. All latitude and longitude values have been appropriately transformed to ensure that no negative values are predicted (scaled by 90 each). I will use Mean Squared Error (MSE) to compare different models. In order to properly compare the unregularized linear regression model with the regularized models with cross validation, I created another simple linear model (separate from the the first part of Problem 1) using glmnet, however I set my lambda parameter to be very close to zero to negate the effects of regularization (1e-8). The cross validated mean squared error (MSE) of this model for comparison was:

- Latitude Simple Linear Regression MSE: `286.4461`
- Longitude Simple Linear Regression MSE: `2025.137`

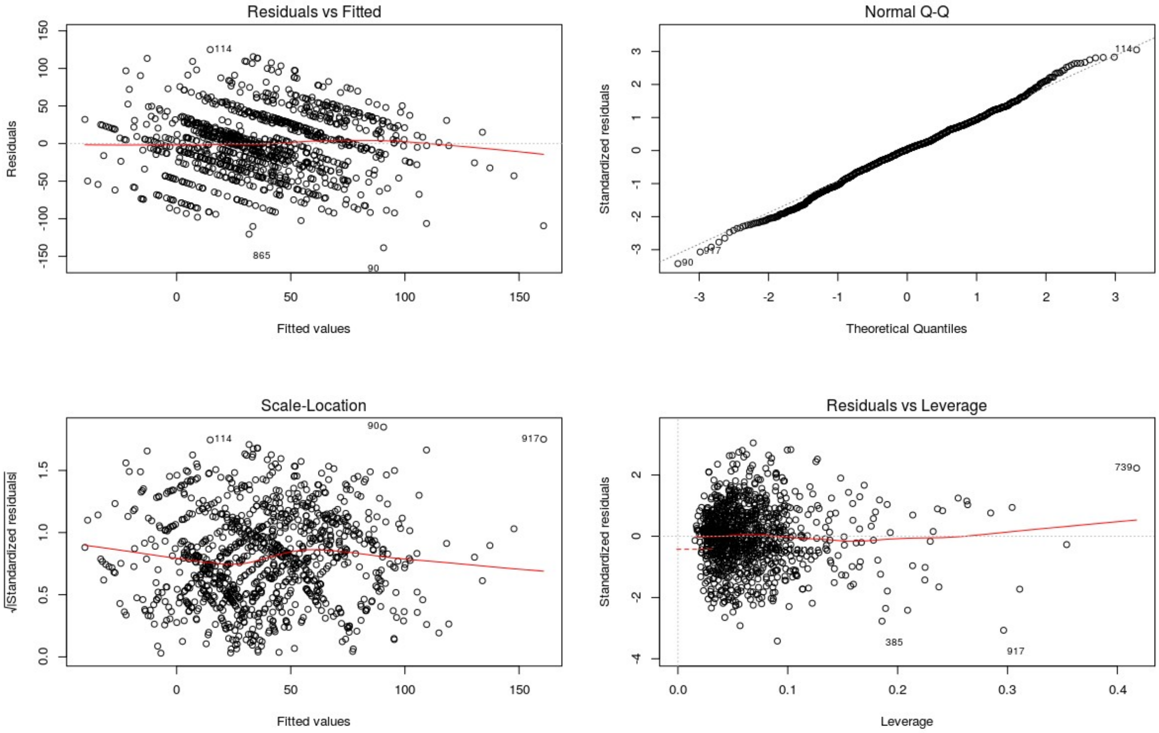
1. Straightforward Linear Regression:

- Latitude Fit
  - R-squared: `0.29`
  - MSE (without cross validation): `241.6968`
  - Diagnostic Plots:
    From the Residuals vs. Fitted Plot, the residuals appear to be randomly dispersed around for each fitted value and there appears to be no pattern to this plot. Additionally, the plot is centered around a mean value of 0. Lastly, the Normal Q-Q plot slighly deviates from a straight line so some transformation may be better for this data. It appears from the diagnostic plots that the simple regression model works decently.



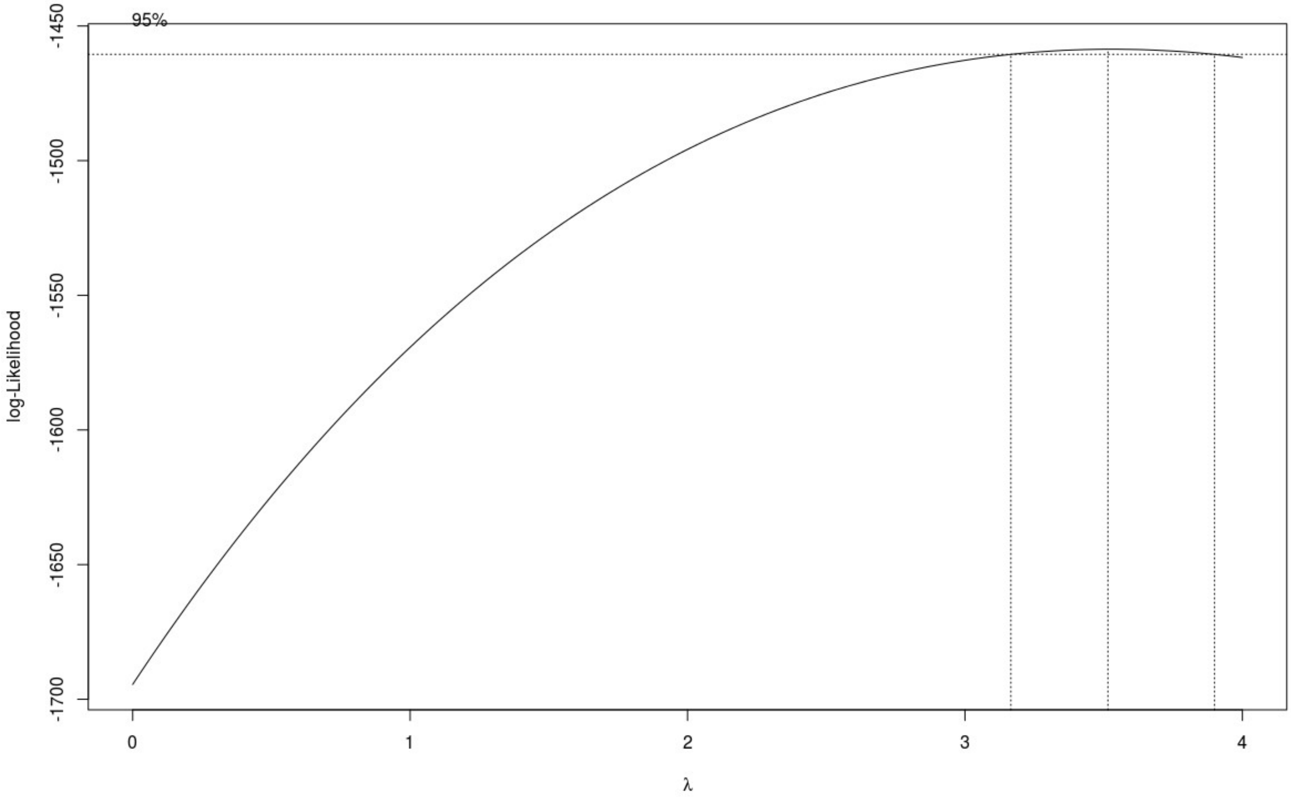
- Longitude Fit
  - R-squared: `0.3355`
  - MSE (without cross validation): `1687.574`
  - Diagnostic Plots:

    From the Residuals vs. Fitted Plot, the residuals appear to be randomly dispersed around for each fitted value and there appears to be no pattern to this plot. Additionally, the plot is centered around a mean value of 0. Lastly, from the Normal Q-Q plot roughly follows a straight line so it appears that the residuals are normally distributed. Therefore, from the Diagnostic plots, it appears that the simple regression model works decently.



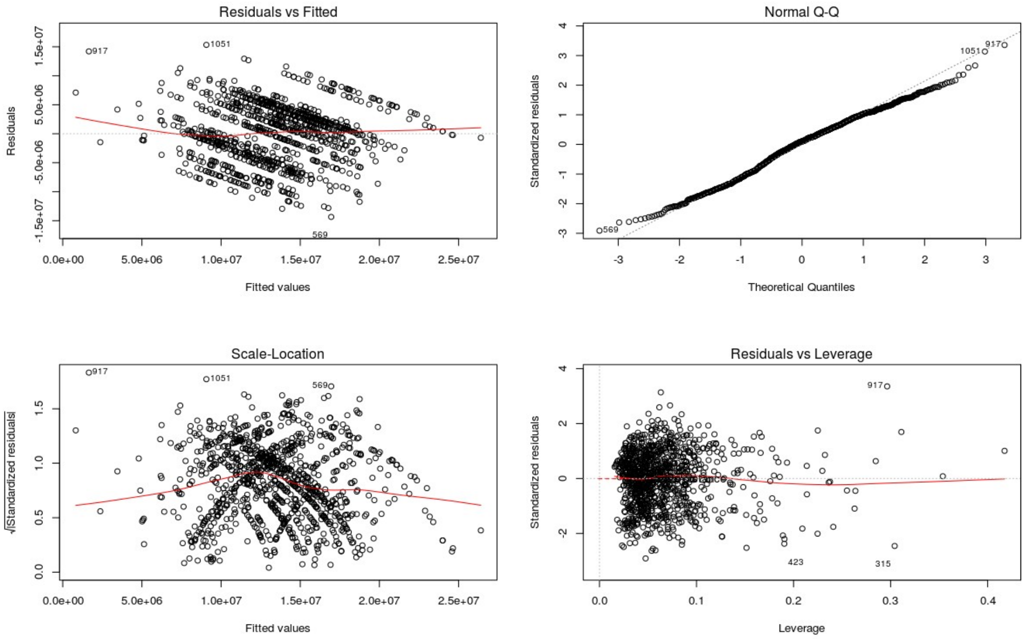
2. Box Cox Transformation

- Box Cox transformation for Latitude Fit:
  - The box cox plot suggests that a transformation of the latitude variable is necessary as the lambda value of 1 is not contained in the 95 % confidence interval. The plot below suggests a transformation of 3.5. Therefore I will create a new regression with the response variable being latitude$^{3.5}$ (results and diagnostic plots for this new regresison are included below).
  - Plot:

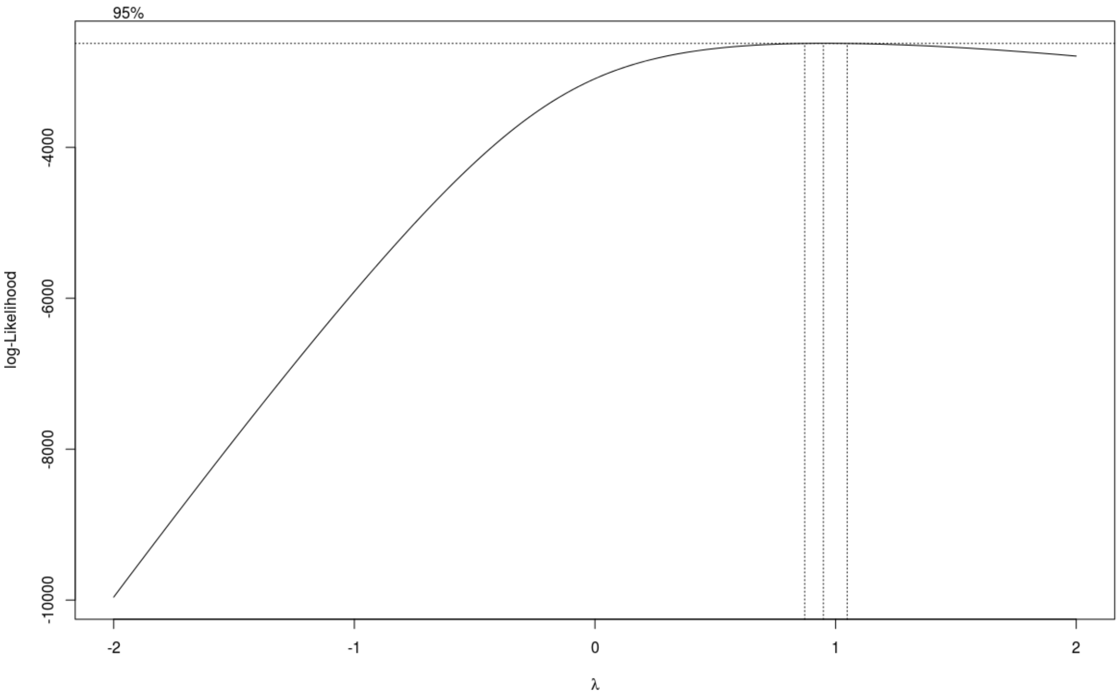


- Transformed Latitude Fit Regression:
  - Lambda: 3.43
  - R-squared: 0.3185
  - MSE: 2.383951e+13
  - Diagnostic Plots:

    From the Residuals vs. Fitted Plot, the residuals do not appear to be randomly dispersed around for each fitted value as there appears to be a slight parabolic pattern to this plot. Residuals for small and large fitted values tend to be larger than middle fitted values. Lastly, from the Normal Q-Q plot roughly follows a straight line so it appears that the residuals are normally distributed. Therefore, from the Diagnostic plots, it appears that the simple regression transformation model works decently, however, it is not perfect. I will elect not to use the Box Cox transformation even though there is a slight rise in the R-squared value. I will instead use my previous model as it will be easier to compare to the unregularized models since I will not be using extremely high mean squared error values.



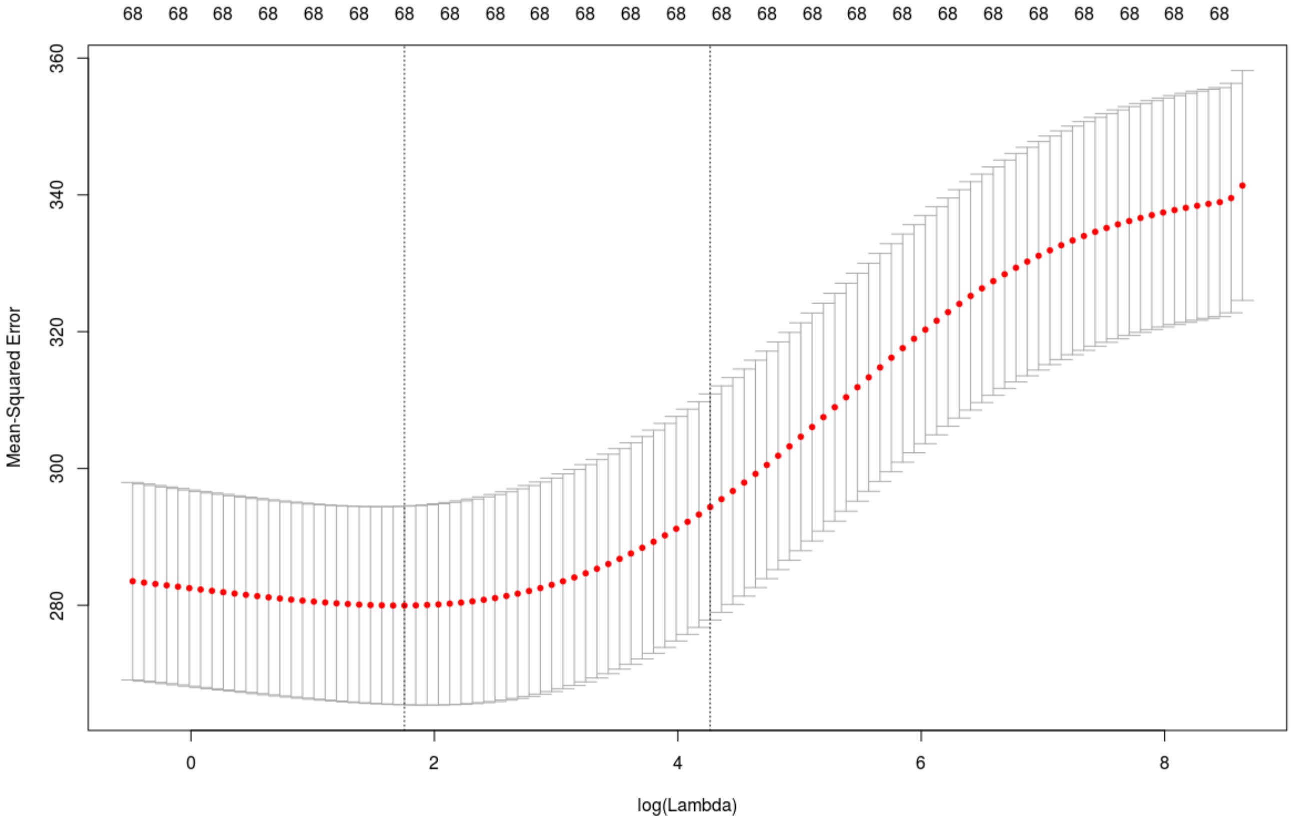
- Box Cox Transformation for Longitude Fit:
  - The box cox plot suggests that no transformation of the longitude variable is necessary as the lambda value of 1 is in the 95 % confidence interval. Therefore I will maintain to original linear regression model from before.
  - Plot:



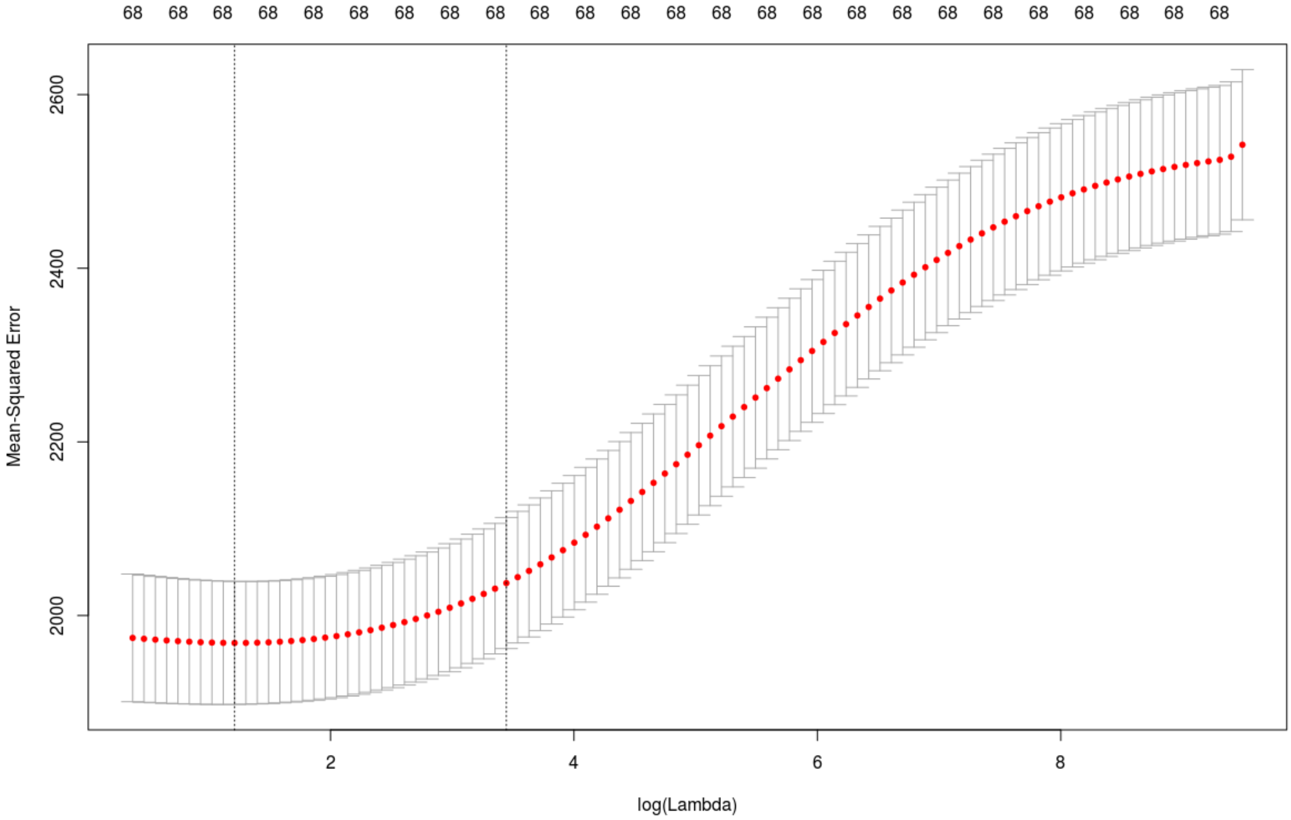
3. Does a Box Cox transformation improve the regressions?

- Box Cox transformations finds the best way to transform a response variable in order to normalize its values so that the newly produced models better aligns with the assumptions of least squares regression. Box Cox helps produce a better regression line to describe the data with a single line as variance among residuals for different fitted values is generally reduced. Although it generally improves performance, Box Cox can sometimes have negative effects on performance and will not always improve regressions. The results of a Box Cox transformation must be validated against simpler models and tested for the assumptions of linear regression.

4. Ridge Regression:
   - Latitude Ridge Regression
     - Lambda (Although the BoxCox transformation had me change transform the latitude variable, I do not think the fit was better so I am retaining the original fit): 1
     - Regularization Coefficient: `9.200195`
     - Minimum MSE: `282.2629`
     - Plot:



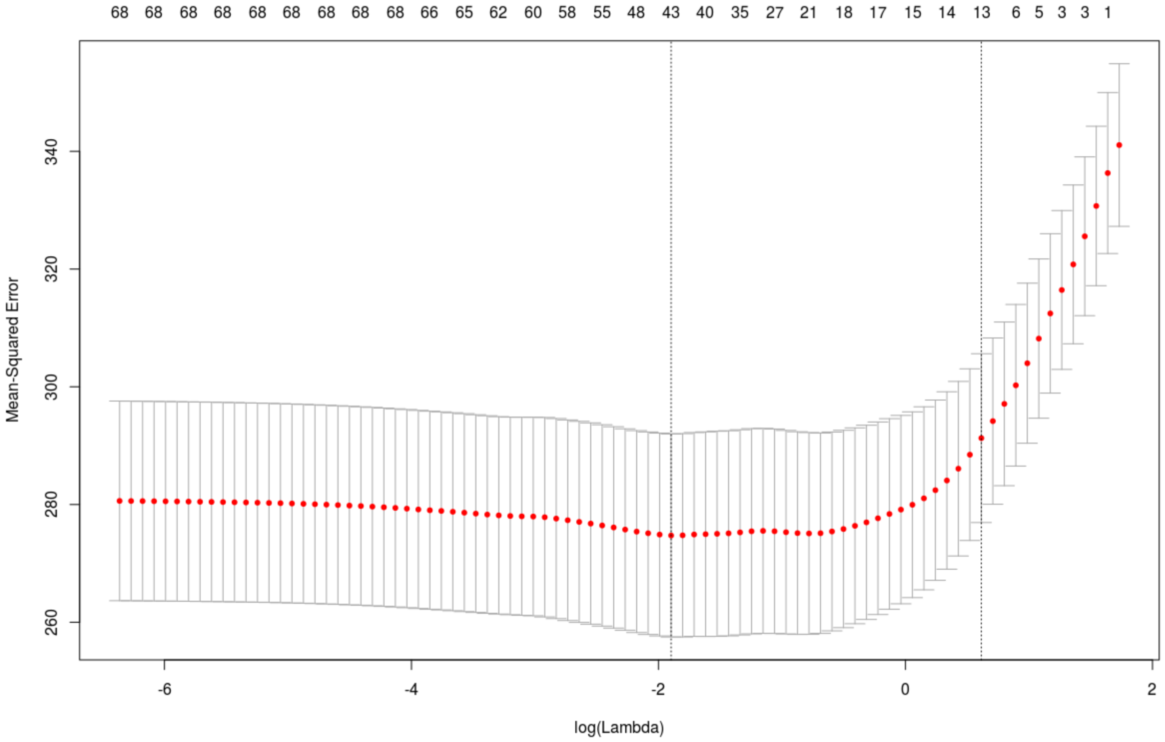
   - Longitude Ridge Regression
     - Lambda (From the BoxCox transformation before): 1
     - Regularization Coefficient: `3.360084`
     - Minimum MSE: `1968.479`
     - Plot:



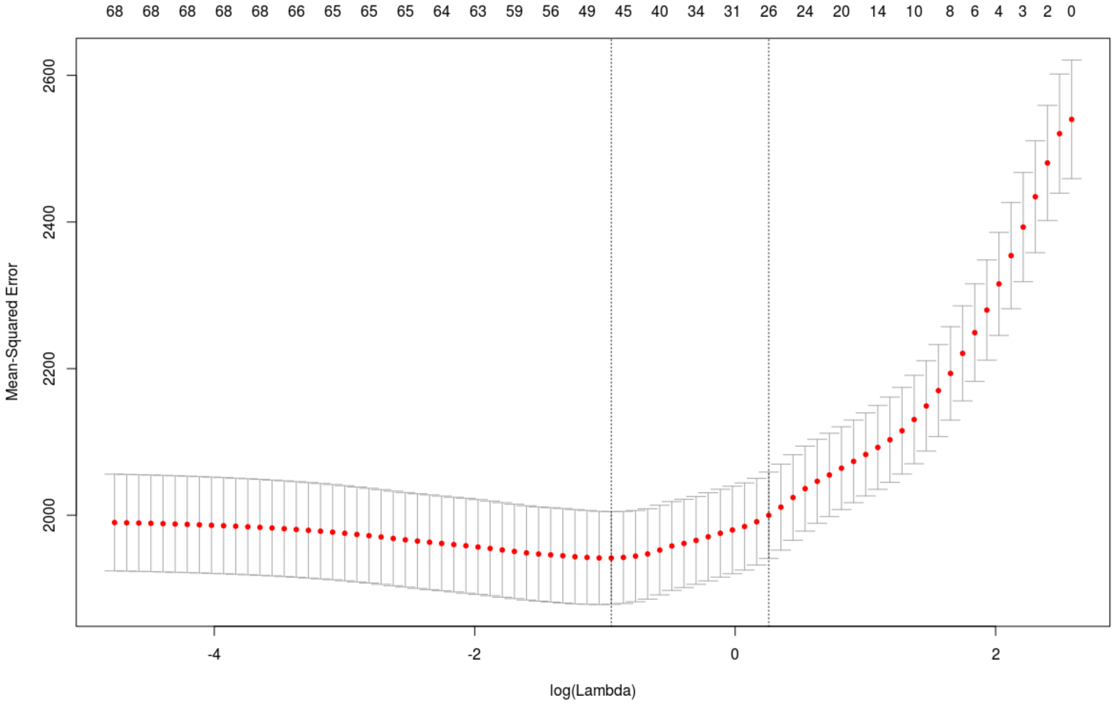
- Performance Comparison:

  In order to properly compare the unregularized linear regression model with the regularized ridge models, I created another simple linear model (separate from the the first part of Problem 1) using glmnet (in order to receive a cross validated mean squared error), however I set my lambda parameter to be very close to zero to negate the effects of regularization (1e-8). The ridge model is much better for both the latitude and the longitude regressions when compared to simple linear regression based on the cross validated mean squared errors. For the latitude regression, the mean squared error for the ridge (regularization parameter= 9.200195) is 282.2629 while the mean squared error for simple linear regression is 286.4461. For the longitude regression, the mean squared error for the ridge (regularization parameter= 3.360084) is 1968.479 while the mean squared error for simple linear regression is 2025.137.

5. Lasso Regression:
   - Latitude Lasso Regression
     - Lambda (Although the BoxCox transformation had me change transform the latitude variable, I do not think the fit was better so I am retaining the original fit): 1
     - Regularization Coefficient: `0.1499402`
     - Minimum MSE: `274.7402`
     - Plot:



- Longitude Lasso Regression
  - Lambda (From the BoxCox transformation before): 1
  - Regularization Coefficient: `0.386328`
  - Minimum MSE: `1941.661`
  - Plot:

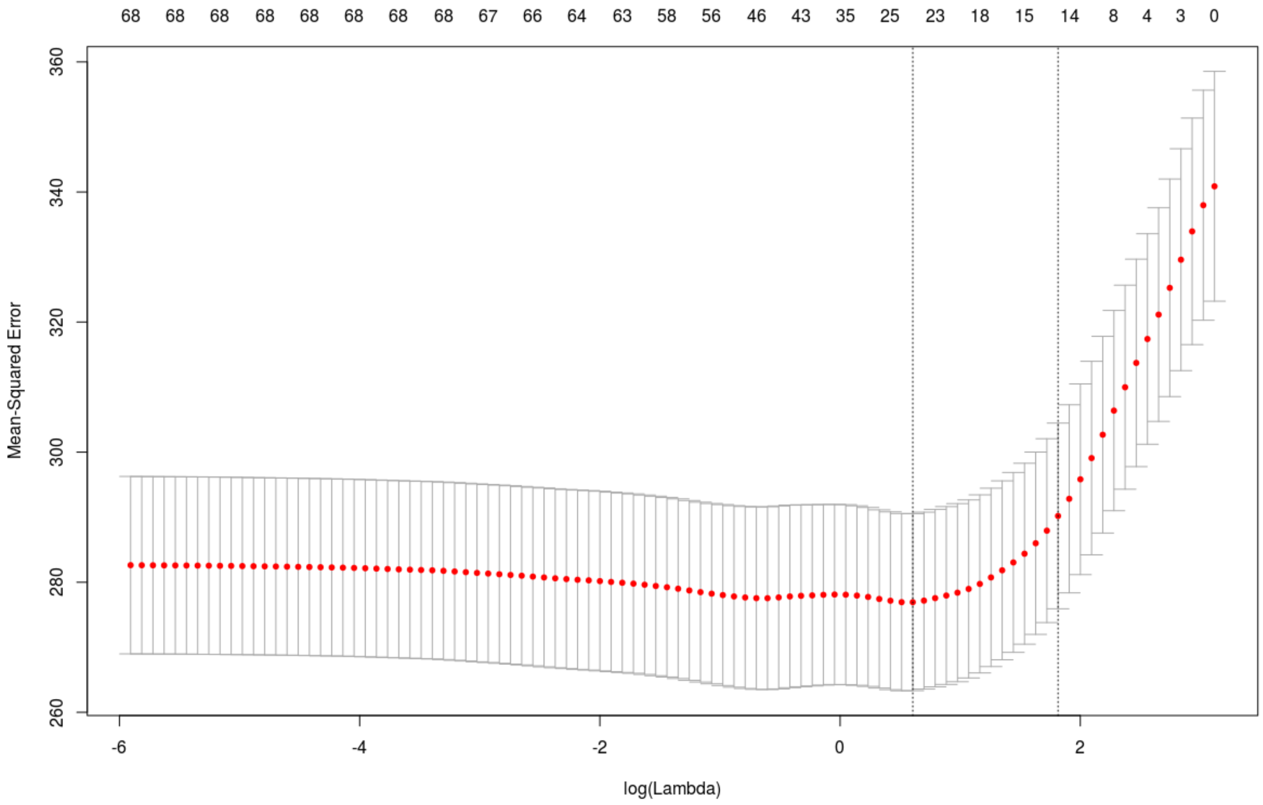


- Performance Comparison:
  In order to properly compare the unregularized linear regression model with the regularized lasso models, I created another simple linear model (separate from the the first part of Problem 1) using glmnet (in order to receive a cross validated mean squared error), however I set my lambda parameter to be very close to zero to negate the effects of regularization (1e-8). The lasso model is much better for both the latitude and the longitude regressions when compared to simple linear regression based on the cross validated mean squared errors. For the latitude regression, the mean squared error for the lasso (regularization parameter= 0.1499402) is 274.7402 while the mean squared error for simple linear regression is 286.4461. For the longitude regression, the mean squared error for the lasso (regularization parameter= 0.386328) is 1941.661 while the mean squared error for simple linear regression is 2025.137.

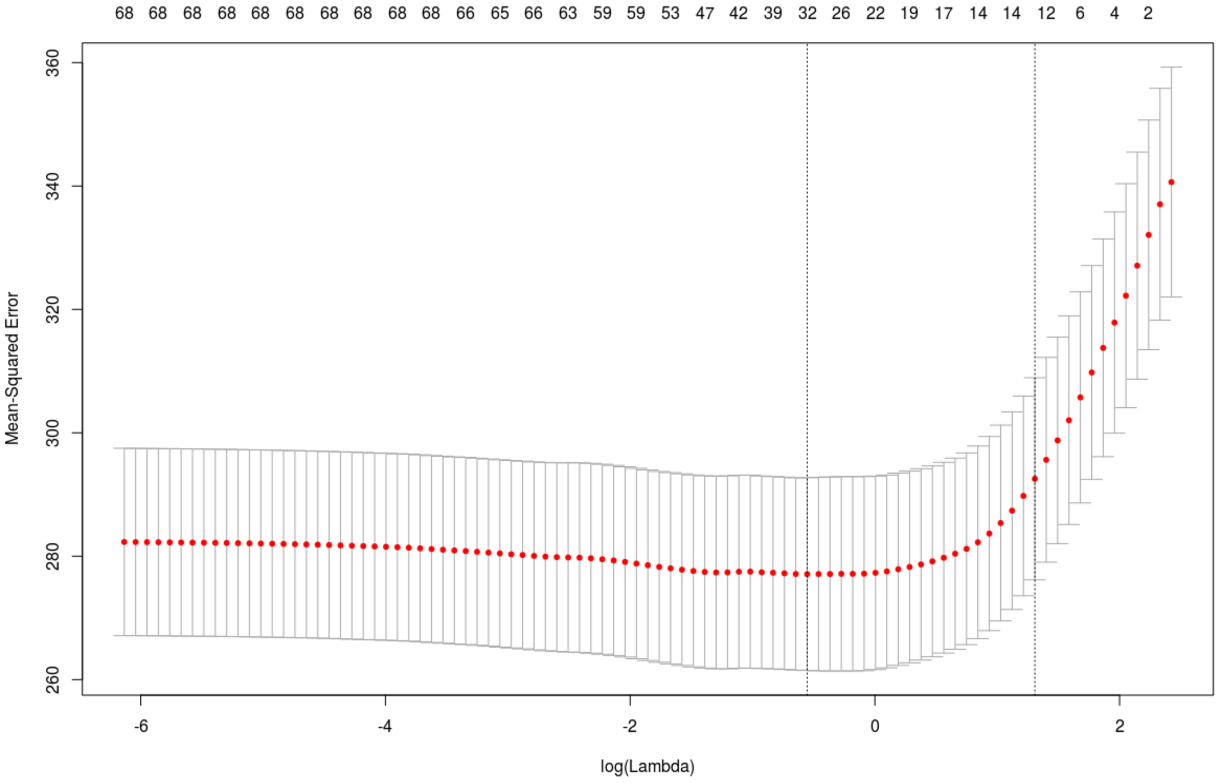
6. Elastic Net:

- I evaluated 3 different alpha values (.25,.5,.75) in order to see which one of these had the best fit for the elastic net regression. I compared each of the elastic net's cross validated mean squared errors to choose the best value for alpha.
- Latitude Elastic Net Regression:
  - Lambda (Although the BoxCox transformation had me change transform the latitude variable, I do not think the fit was better so I am retaining the original fit): 1
  - Best Alpha coefficient: .25
  - Regularization Coefficient: 1.8315834
  - Minimum MSE: 276.9361
  - Plot (order of plots: alpha=(.25,.5,.75)):

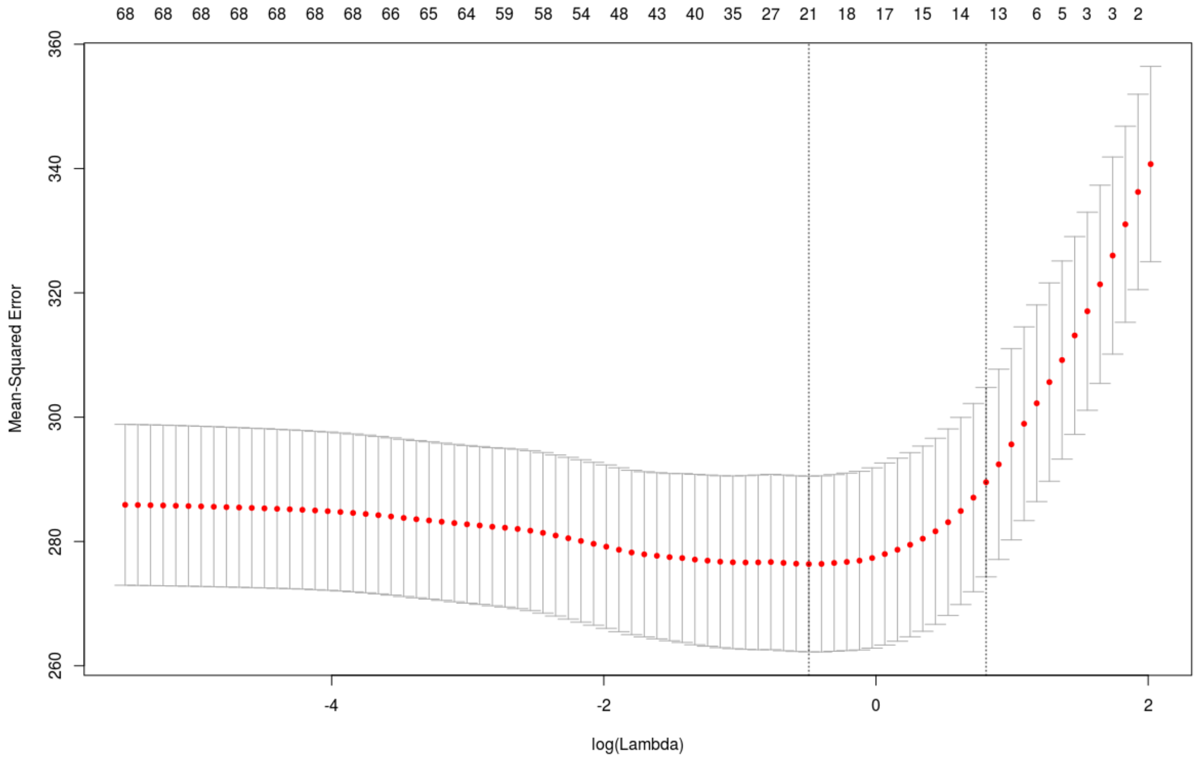
alpha = .25



alpha = .5

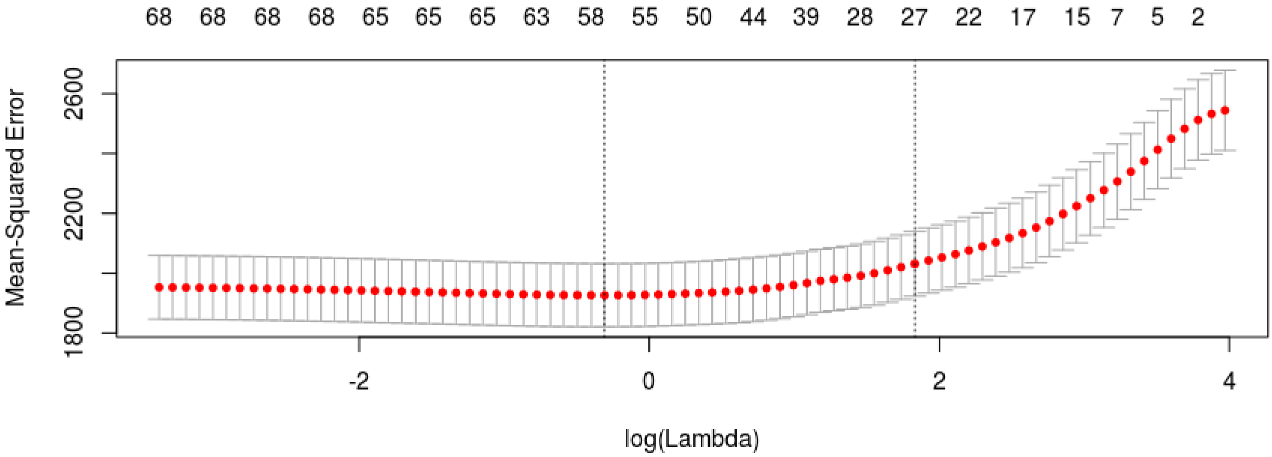


alpha = .75

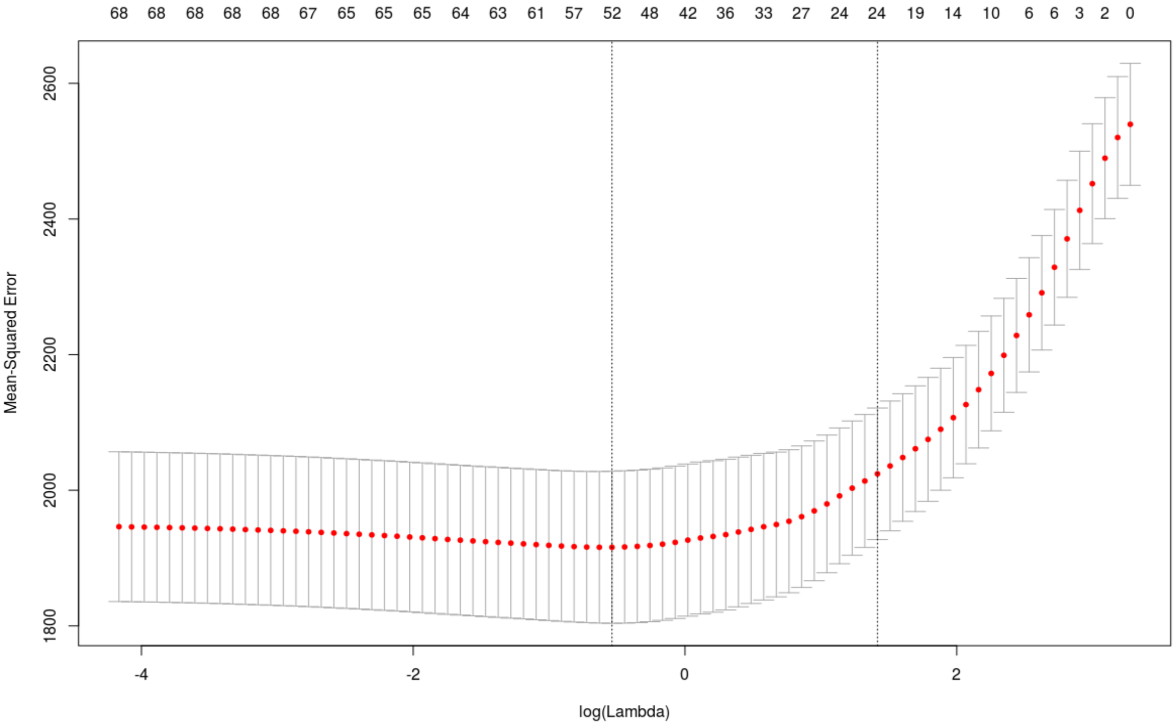


- Longitude Elastic Net Regression:
  - Lambda ((From the BoxCox transformation before)): 1
  - Best Alpha coefficient: .5
  - Regularization Coefficient: `0.5844859`
  - Minimum MSE: `1915.930`
  - Plot:

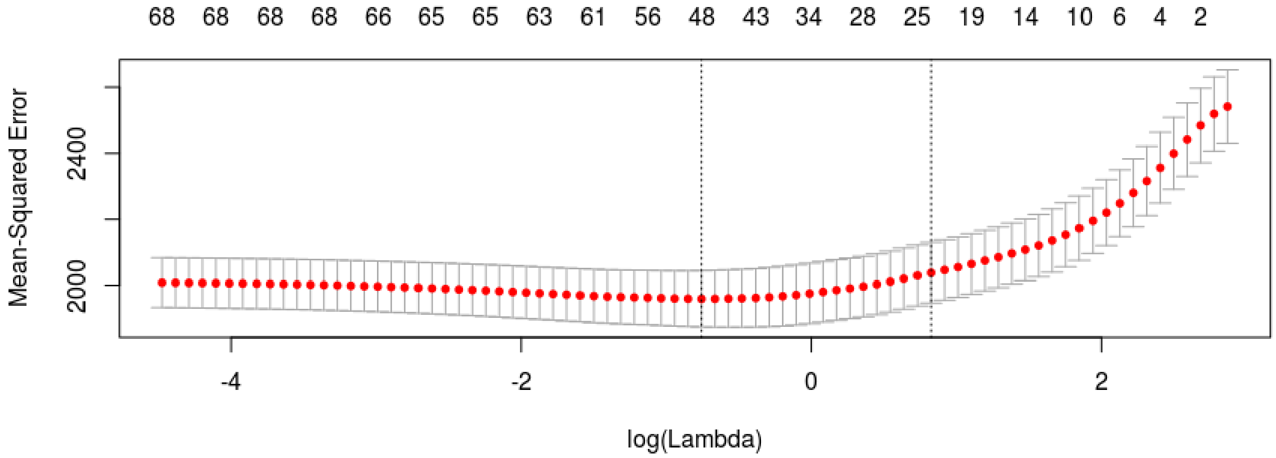
alpha = .25



alpha = .5



alpha = .75

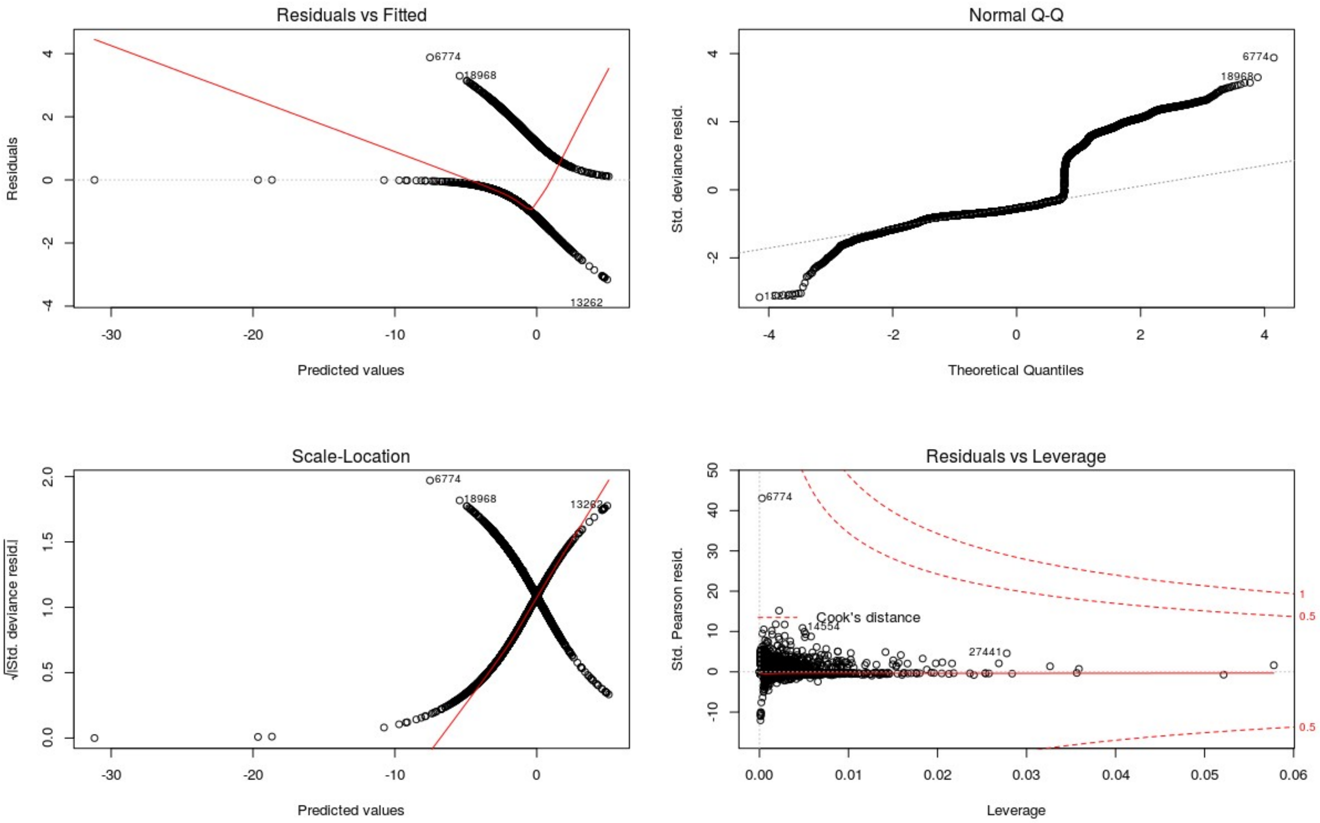


- Performance Comparison:

  In order to properly compare the unregularized linear regression model with the elastic net models, I created another simple linear model (separate from the the first part of Problem 1) using glmnet (in order to receive a cross validated mean squared error), however I set my lambda parameter to be very close to zero to negate the effects of regularization (1e-8). The elastic net model is much better for both the latitude and the longitude regressions when compared to simple linear regression based on the cross validated mean squared errors. For the latitude regression, the mean squared error for the elastic net (alpha=.25, regularization parameter=1.83) is 276.9361 while the mean squared error for simple linear regression is 286.4461. For the longitude regression, the mean squared error for the elastic net (alpha=.5, regularization parameter= 0.584485) is 1915.930 while the mean squared error for simple linear regression is 2025.137.

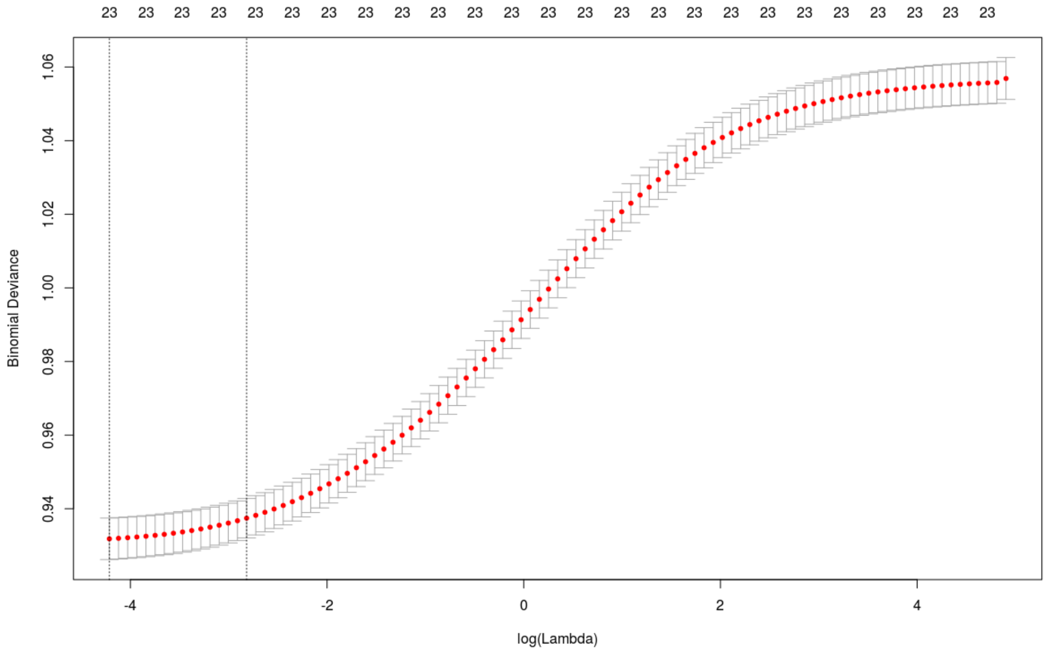
Problem 2 (Logistic Regression):

1. Simple Logistic Regression
   - Deviance: `27877.2 (0.9316288)`
   - Diagnostic Plots:

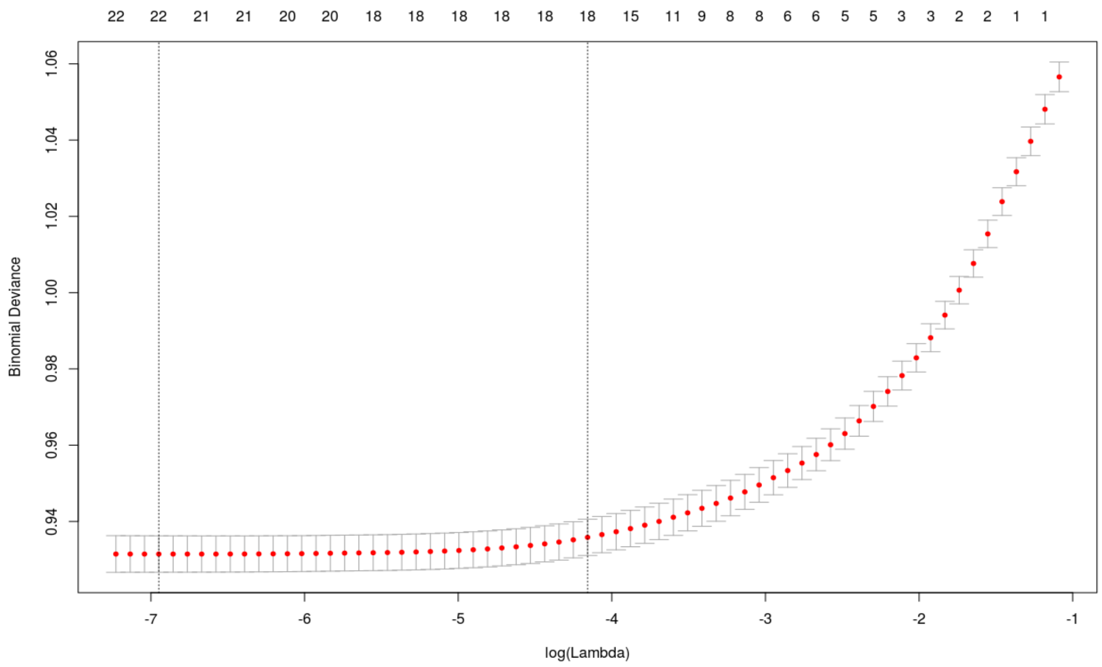


2. Other models tried:

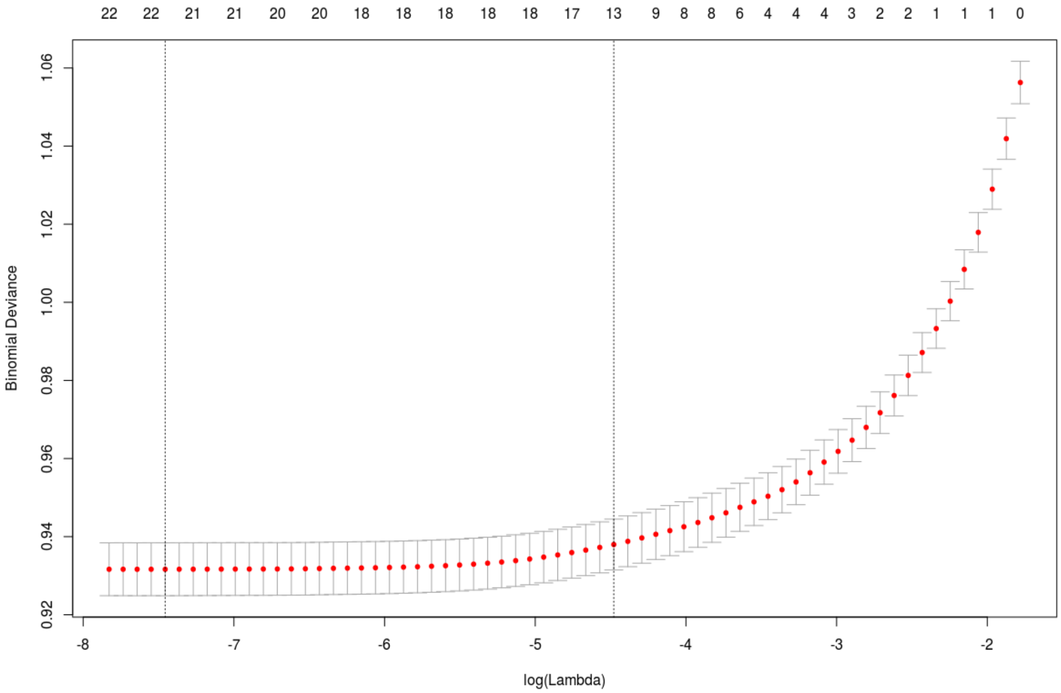
- Ridge model (alpha = 0)
  - Regularization Parameter: 0.01479508
  - Deviance: 0.9318325
  - Plot:



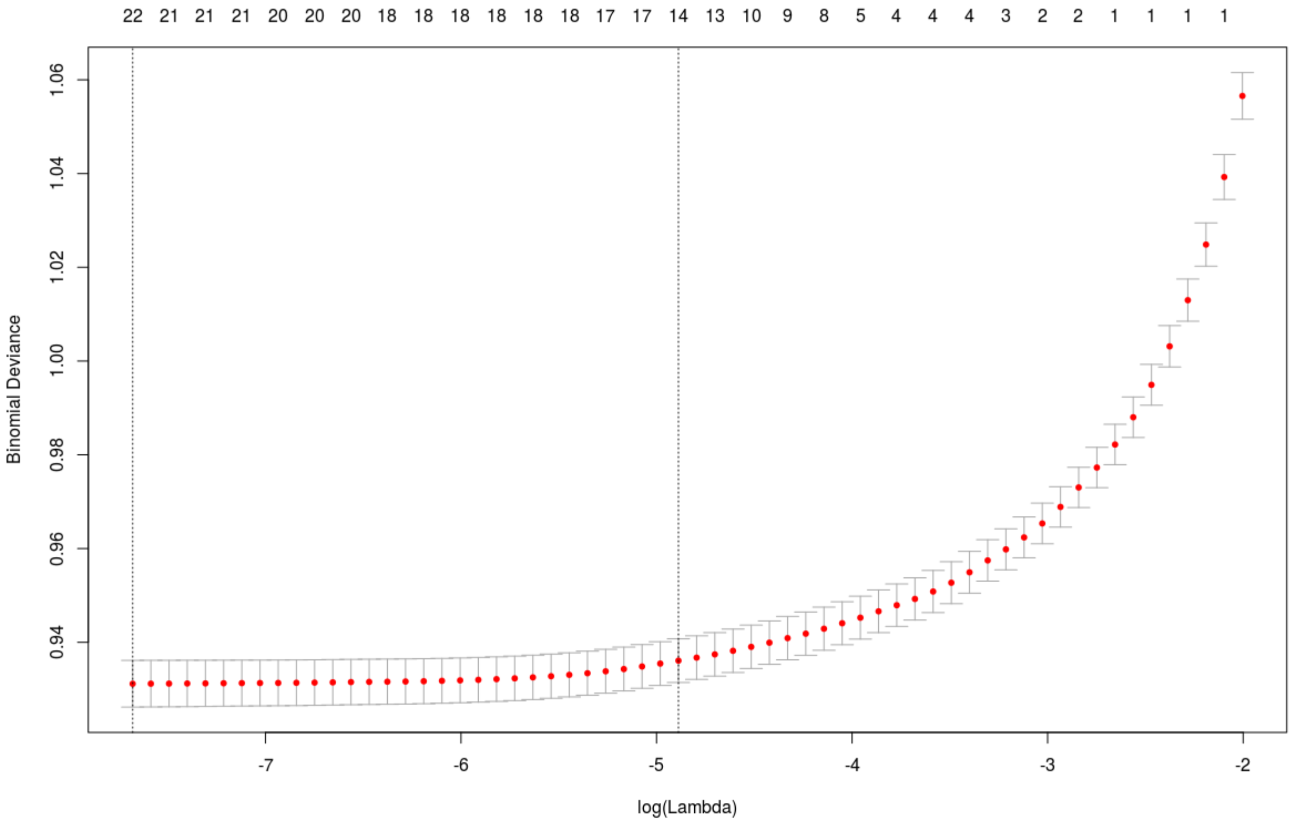
- ElasticNet model (alpha = .4)
  - Regularization Parameter: 0.0009598395
  - Deviance: 0.9314511
  - Plot:



- ElasticNet model (alpha = .8)
  - Regularization Parameter: 0.0005780651
  - Deviance: 0.9316493
  - Plot:



- Lasso model (alpha = 1)
  - Regularization Parameter: `0.01479508`
  - Deviance: `0.9311578`
  - Plot:



It appears as though the Lasso Model is the best model to use as it produces the lowest cross-validated deviance among all of the different regularized models I tried.