James Yang

DATA 558

Hw4

1. Giving g(x) = e^x, we can give plot a graph of each.
   1. ĝ(x) = 0

Histogram

Description automatically generated

* 1. ĝ(x) =

= 2.256

Graphical user interface

Description automatically generated with low confidence

* 1. ĝ(x) = ax+b

After running lm(y ~ x), we get the coefficients to be:

(Intercept) x

2.194379 3.105231

Chart, line chart

Description automatically generated

* 1. ĝ(x) =

After running a lm with y ~ x + x^2, the equation is:   
1.58870890\*x^2 + 2.5476450\*x + 0.08390203

Graphical user interface, histogram

Description automatically generated

Histogram

Description automatically generated with medium confidence

Using this code:

Text

Description automatically generated

1. We will plot it out after calculating x = [-2,6]:

|  |  |
| --- | --- |
| X | Y |
| -2 | 2 |
| -1 | 2 |
| 0 | 5 |
| 1 | -1 |
| 2 | -4 |
| 3 | -6 |
| 4 | -10 |
| 5 | 2 |
| 6 | 2 |

Using this code:

Diagram, schematic, scatter chart

Description automatically generated

Chart, scatter chart

Description automatically generated

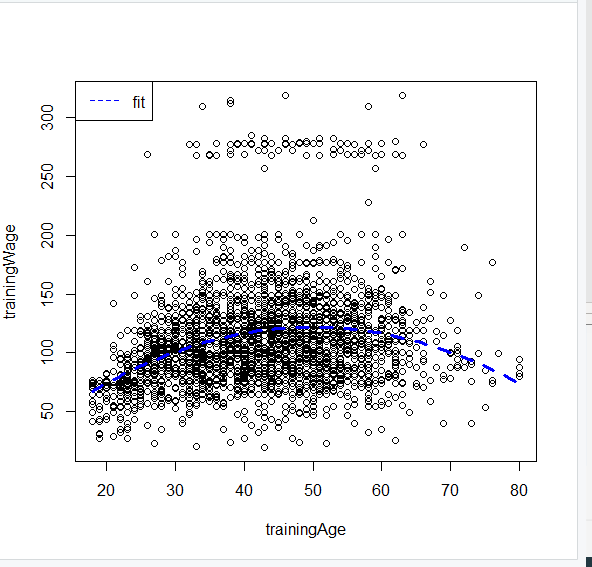
It appears that the curve is constant from -2 and -1 where y = 2. From 0 to 1, y is linear with y = -6x + 5. From 1 to 2, it is linear with y = -3x + 2. From 2 to 3 it is linear with y = -2x. From 3 to 4 it is linear with y = -4x + 6. From 4 to 5 it is linear with y = 12x – 58. From 5 to 6 it is constant with y = 2.

A piece of paper with writing

Description automatically generated with medium confidence

1. The MSEs will be labeled at the bottom of the problem.
   1. After splitting data in 2800 training and 200 testing, polynomial function ended up being:

Y = -0.05321429\*x^2 + 5.31585967\*x -10.87610589



Producing a graph like this with the blue dashed line being the fitted values.

Chart, scatter chart

Description automatically generated

Chart, scatter chart

Description automatically generated

Chart, scatter chart

Description automatically generated

Chart, scatter chart

Description automatically generated

After analyzing all of the errors:

MSE 4a: 1609.01

MSE 4b: 1731.71

MSE 4c: 1532.32

MSE 4d: 1590.30

The best approach for this problem according to MSE was the piecewise polynomial. This is interesting but it could be because the dataset is fitting the curvature of step function in a polynomial manner. However, after reevaluating the MSE values, they do fluctuate a bit but keep a consistent lowest value for 4c.

* 1. When fitting a regression tree without pruning, we get these results. **Its accuracy predicts 85.4%**

Diagram, schematic

Description automatically generated

Text

Description automatically generated

Chart, histogram

Description automatically generated

Diagram, schematic

Description automatically generated

When pruning with a best of 4, we get a slightly higher accuracy of 89.58%

Text

Description automatically generated

A picture containing diagram

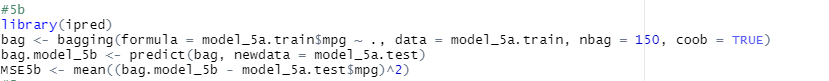
Description automatically generated

When pruning with a best of 4, it gives a higher accuracy rating than when we simply created a tree regression. This means the pruning process has produced a more interpretable tree, but also has slightly improved classification accuracy. This makes sense because as the value of best increases, we obtain a larger pruned tree that can give us lower classification accuracy but higher interpretability.

After calculating the MSE, 

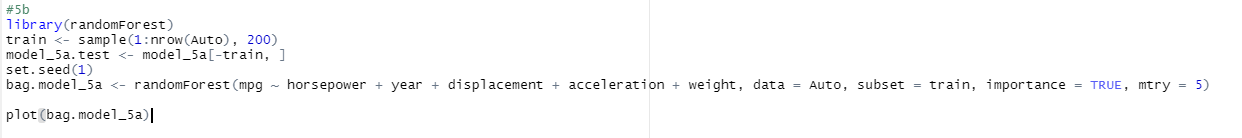
We get that the MSE is about **20.31**, meaning the model does not fit the data very well with regression trees.

* 1. Using this code:



We get that the MSE is about **17.64** which means the bagging did improve the model accuracy a bit. The tuning parameters used were the number of bags = 150 and that the coob = TRUE.

* 1. When fitting a random forest model:



With an mtry of 5:

Graphical user interface, chart

Description automatically generated

**Mtry = 5**

Obviously as the ntrees progress, we see that the error decreases. When the trees increase though, it makes the model less interpretable which kind of defeats the purpose of having a regression tree.

Text

Description automatically generated with low confidence

When calculating the MSE, we get **8.425** with an mtry of 5.

* 1. Using this code:

A picture containing text

Description automatically generated

|  |  |
| --- | --- |
| Histogram  Description automatically generated with medium confidence |  |
|  |  |
|  |  |
|  |  |
|  |  |



After calculating the MSE on the GAM model, we find that the MSE is **7.504** which is less than the regression tree, random forest, and bagging. So it is the best model so far. Also, when doing summary(gam.model\_5a), we get that the p-values were less than 0.05 which indicates that they are signficant in estimating mpg.

A screenshot of a computer

Description automatically generated with low confidence

* 1. Considering both accuracy and interpretability of the fitted models, I prefer the random forest model. Even though it didn’t produce the best MSE, the error to tree graph made the most sense as it became clear when the optimal point of number trees would arise with the greatest reduction of error. With regression trees in this model, it becomes highly complex and increasingly less interpretable and its important to find when that sweet spot occurs which is what the bagged regression tree model tells us. The GAM model was the best model for accuracy, but I don’t think it was as interpretable as the random forests model.

There is an article written by Arrieta, Rodriguez, Ser about the concepts, taxonomies, opportunities and challenges of AI for interpretability. Table

Description automatically generated

They have assembled this idea that as the models go down the list, they become more difficult to interpret to a human. Decision trees and regression trees are some of the easiest models for humans to interpret, and the trade off that the GAM model had in this exercise did not seem worth it. They divide the ideas into simulatability, decomposability, and algorithmic transparency. GAM algorithmic transparency drives at an exponential slope when it comes to interpretability for humans, therefore making it less interpretable and more difficult as more predictors come into play. The same goes for regression trees, but they are way more interpretable at the beginning states of tree nodes (especially when pruning). That’s why I believe random forests are somewhat in the middle of interpretability and they fit this problem quite well.