DSC630 - Final Project

David Suffolk

Business Problem

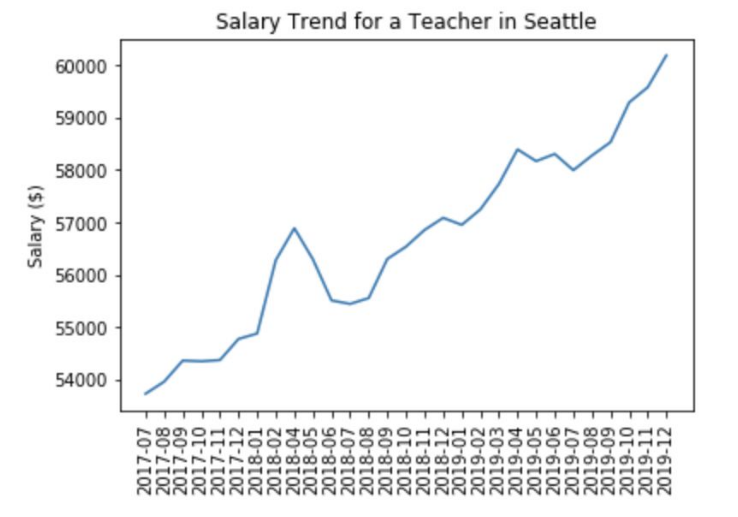
- Labor is one of the largest expenses to a business
- Adding a role or filling a vacancy is an investment by the organization
- By forecasting a salary, an organization can be prepared:
  - To speak to how much the investment will cost initially
  - To speak to how much may need to be invested over time
  - Reduce retention by making sure employee is consistently satisfied with their income
- Overall, an understanding of the salary provides a well-rounded perspective on the financial implications of onboarding an employee

Data Preparation

- Data is from Glassdoor and contains the monthly data for salaries in various metropolitan areas and the national average for several job titles
- The data also contains year over year average
- A couple of the months of data had errors and missing values and were removed for this project
- For Data Exploration, it was better to break the data down into cities and job titles to begin to understand what trends are in the data

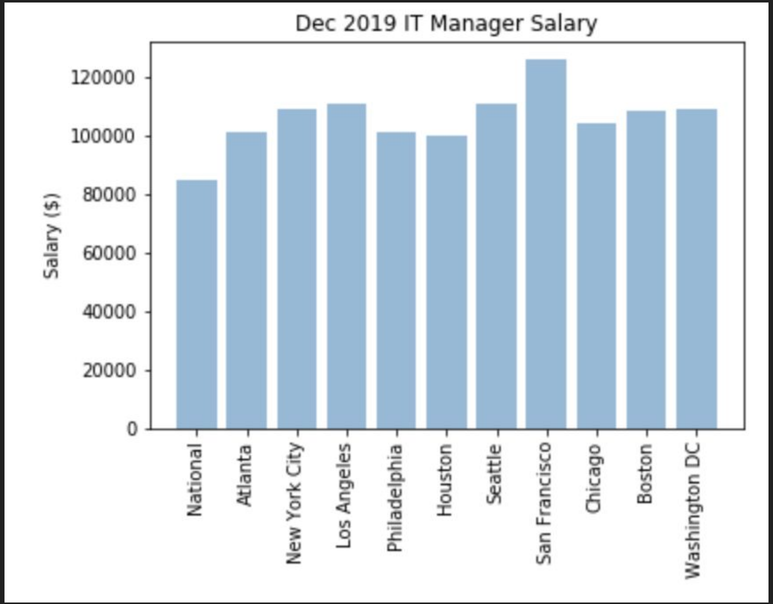
Salary Trends

- As expected many of the salaries trend higher month after month. Some trend higher at a consistent rate, such as teacher, while some trend inconsistently, such as IT Manager



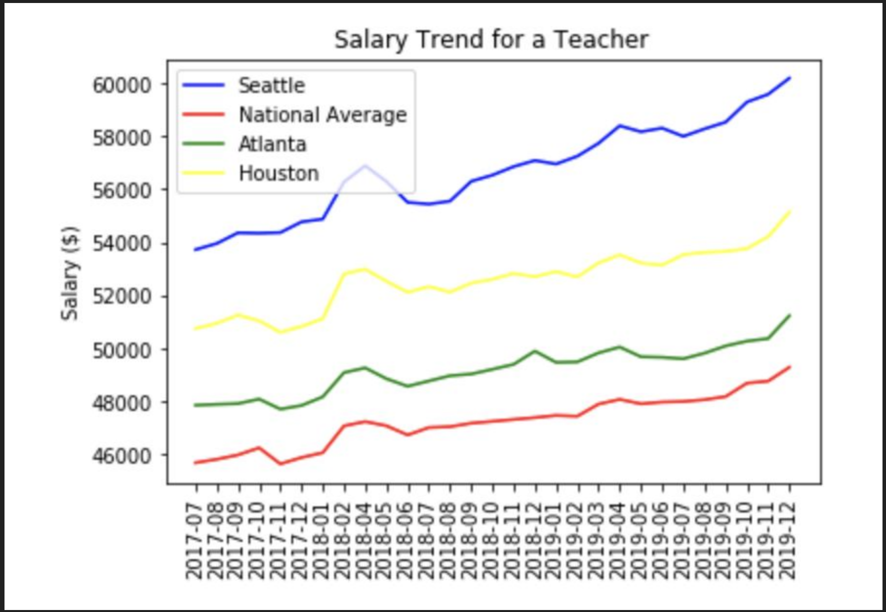
Salary Trends

- This is a one-month view of one position's salary and how it has variation from region to region.

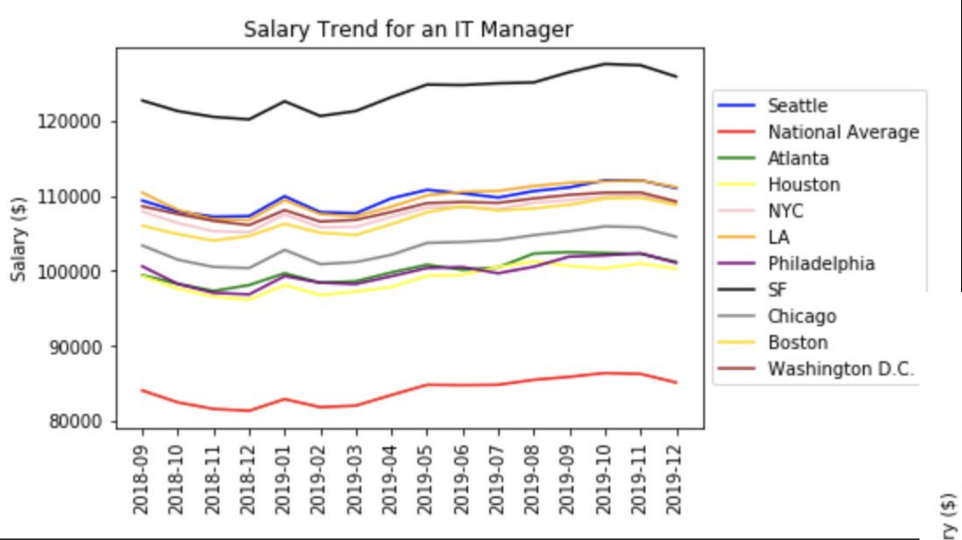


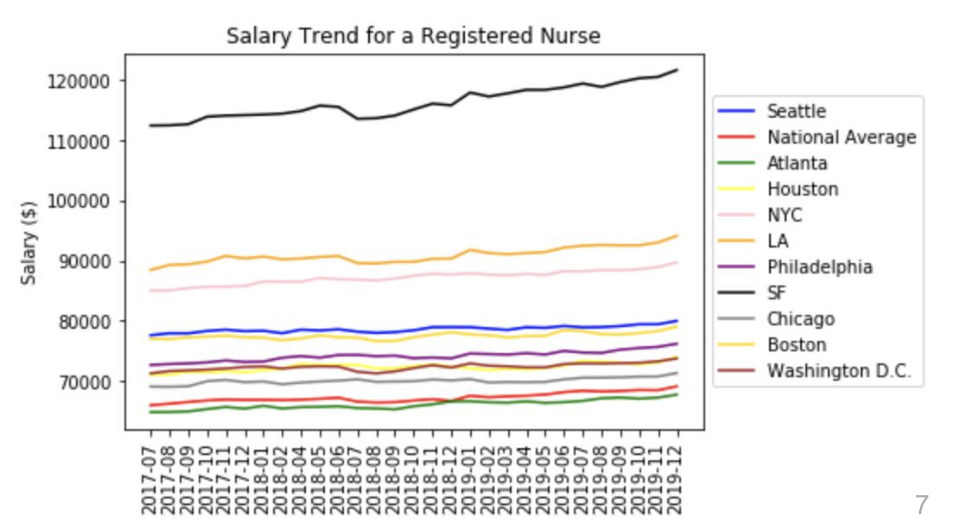
Salary Trends

- The regions in the data are metropolitan areas. In many cases, these salaries were above the national average.

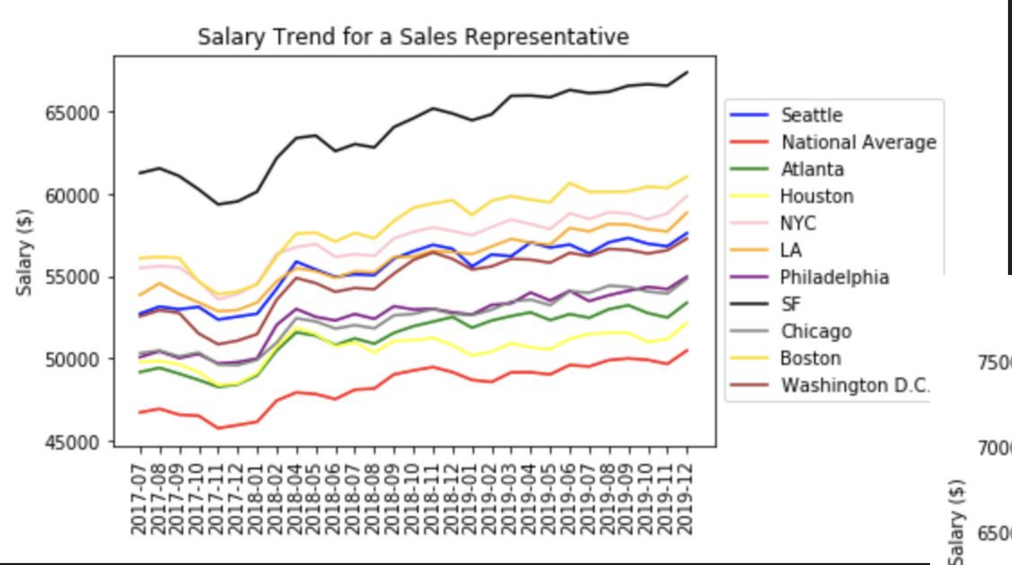


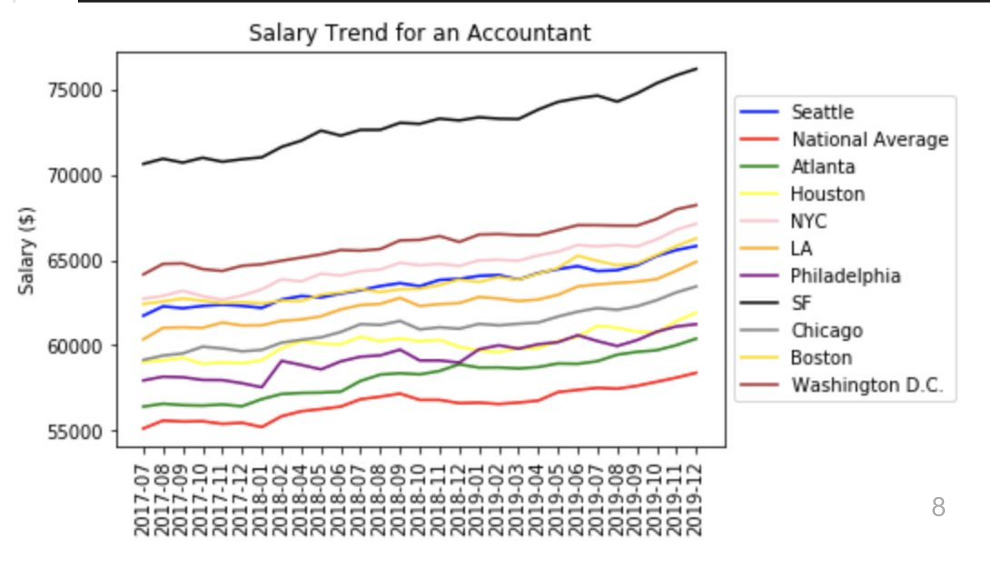
Salary Trends





Salary Trends



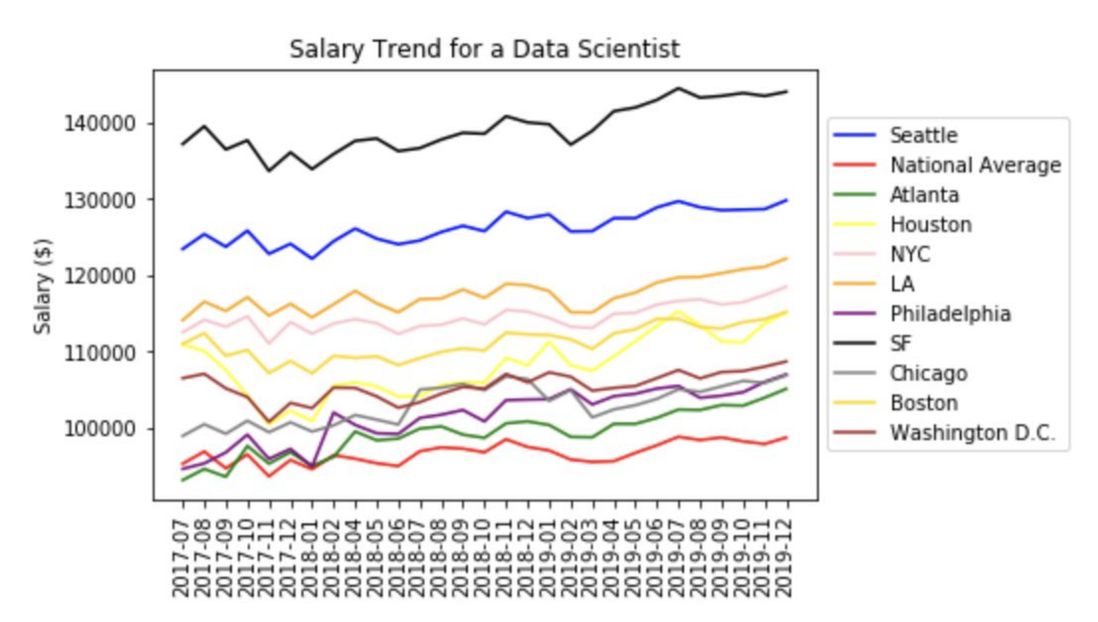


Model Building

- Five phases to building the final model for the project
  - Phase 1: Decision trees for one job title and a binary target variable
  - Phase 2: Apply techniques to more job titles
  - Phase 3: Place Target Variable Models in bins
  - Phase 4: Decision tree for multiple job titles and multiple bins for target variables
  - Phase 5: Neural Network

Model Building - Phase 1

- For my model, I decided to focus on one job title to make sure the model could be built and any initial success with accuracy.
- I chose the role of Data Scientist due to its relevance to the coursework.



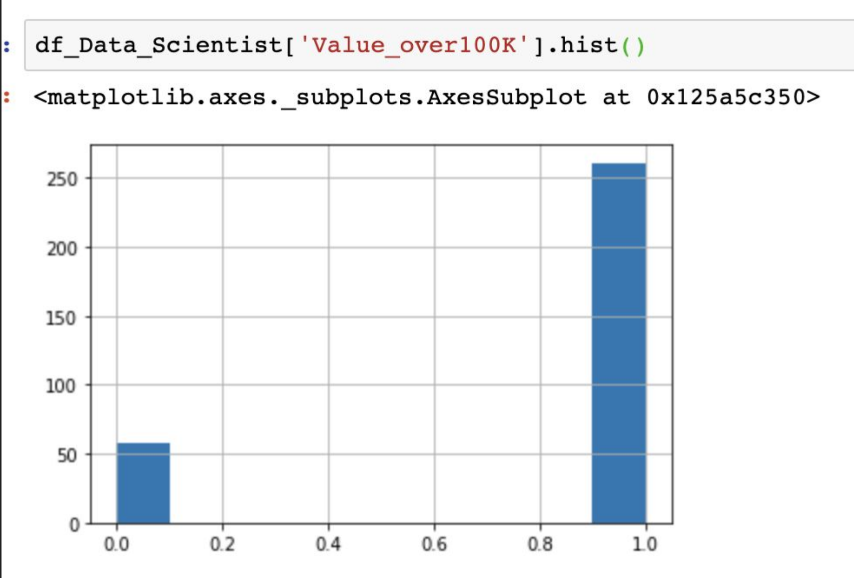
Model Building - Phase 1

- For the region variable, I did one-hot encoding to turn them into binary variables

	Atlanta	Boston	Chicago	Houston	Los Angeles	National	New York City	Philadelphia	San Francisco	Seattle	Washington DC
89	0	0	0	0	0	1	0	0	0	0	0
173	1	0	0	0	0	0	0	0	0	0	0
257	0	0	0	0	0	0	1	0	0	0	0
341	0	0	0	0	1	0	0	0	0	0	0
425	0	0	0	0	0	0	0	1	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...
565	0	0	0	0	0	0	0	0	0	1	0
649	0	0	0	0	0	0	0	0	1	0	0
733	0	0	1	0	0	0	0	0	0	0	0
817	0	1	0	0	0	0	0	0	0	0	0
901	0	0	0	0	0	0	0	0	0	0	1

Model Building - Phase 1

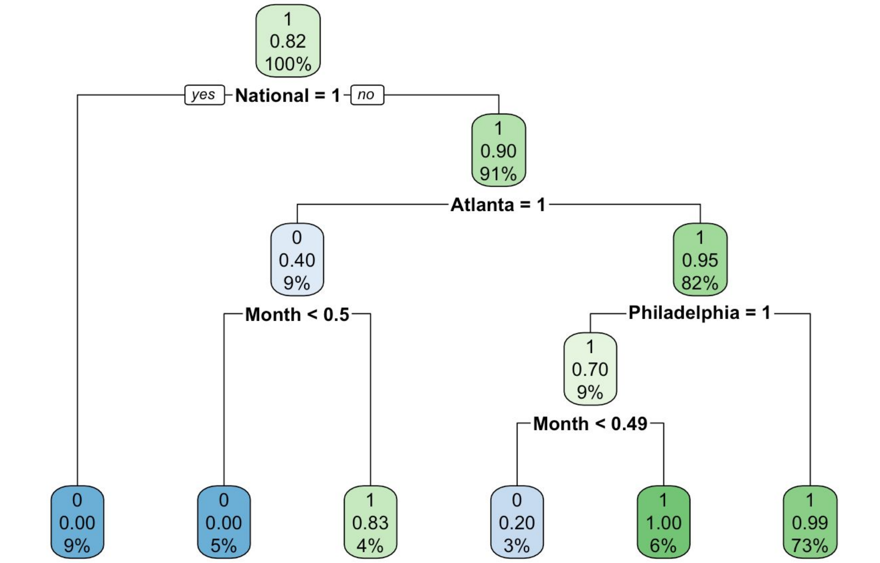
- For the target variable, I converted it into a binary variable of over $100K and under $100K. As I develop the model, I hope to create several bins for the target variable across all job titles. However, at this stage, I kept it to this option.



Model Building - Phase 1

- For the initial model, I have been working with Decision Trees. Due to the binary target variable and limited number of features, I thought this would be a great place to start with the model building.
- There are a couple of limitations to this model
  - Binary target variable (over or under $100K)
  - Job Title is not a variable
- The model was built in R and had 93.75% accuracy

Model Building - Phase 1




```
p <- predict(dtm, data_test, type="class")
confMat <- table(data_test$Value_over100K,p)
accuracy <- sum(diag(confMat))/sum(confMat)
accuracy*100
```

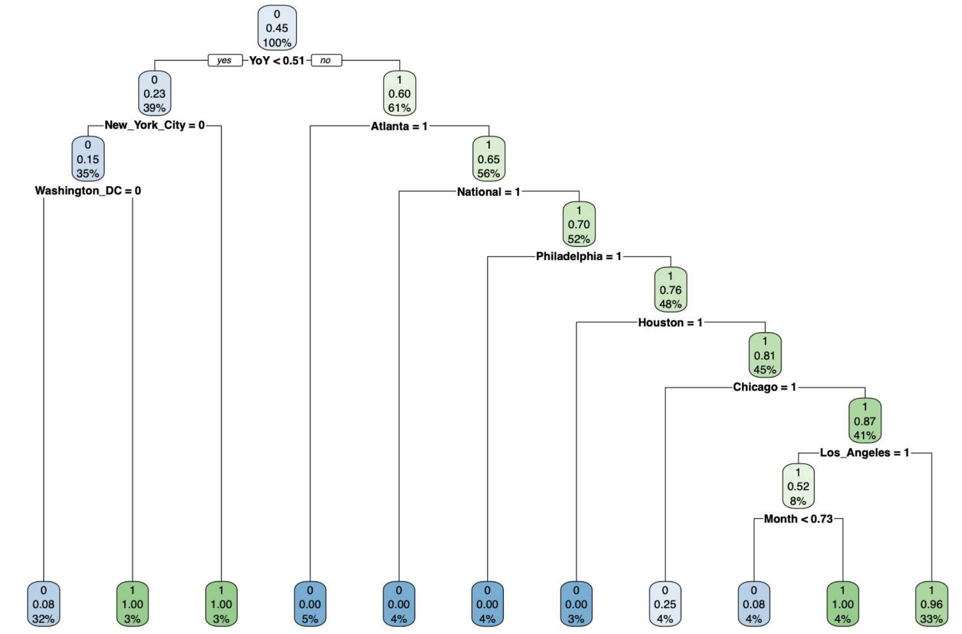
```
## [1] 93.75
```

Model Building - Phase 2

- Expanded the process used for Data Scientist to many more job titles
- To make the process more efficient:
    - Built functions to prepare data in Python
    - Built functions to build decision tree model in R
    - Set the target variable to be above or below the mean
- Each job title showed differences in how the data was split in the Decision Tree to reach the target variable
- Each model was able to obtain a high accuracy

Model Building - Phase 2

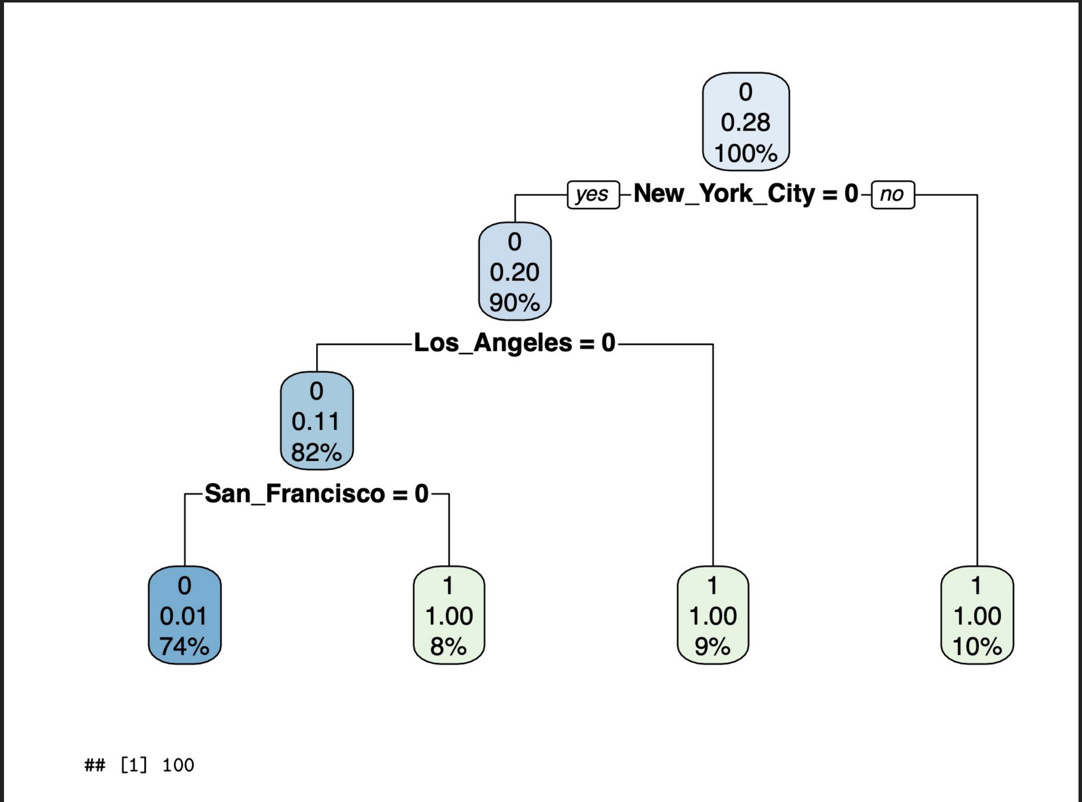
Accountant




```
## [1] 93.75
```

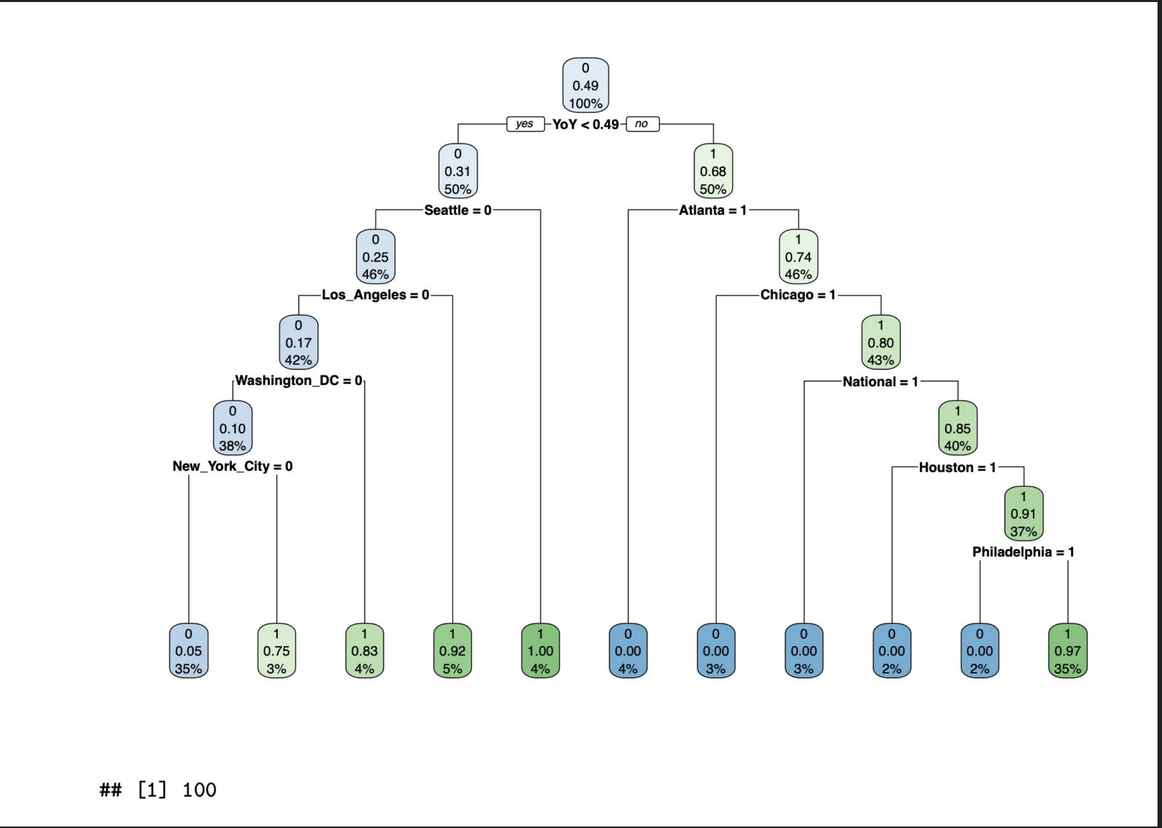
Model Building - Phase 2

Registered Nurse



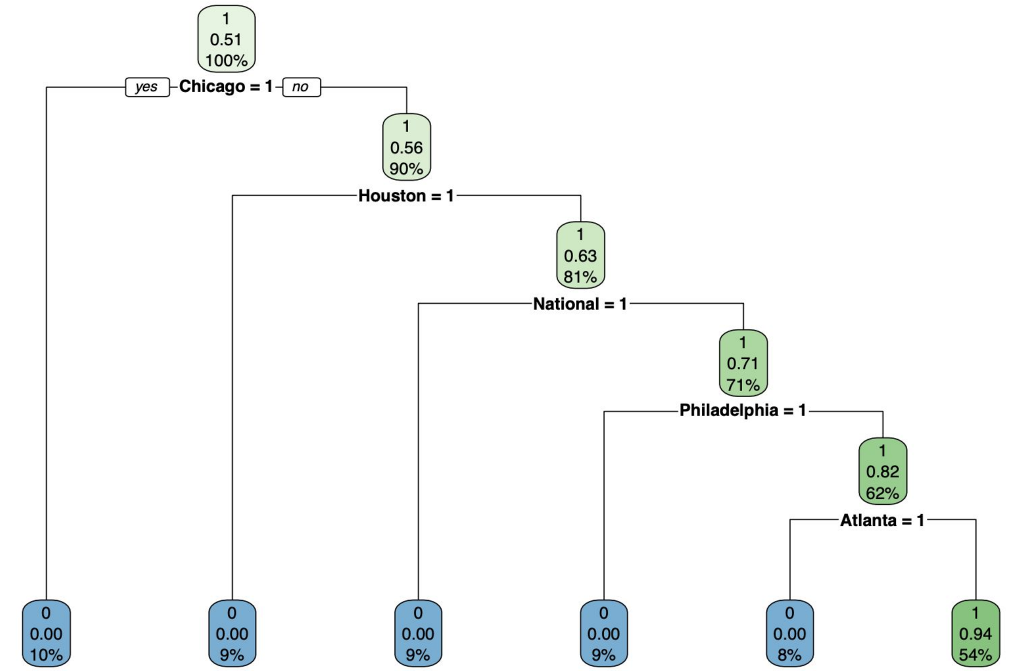
Model Building - Phase 2

Data Analyst



Model Building - Phase 2

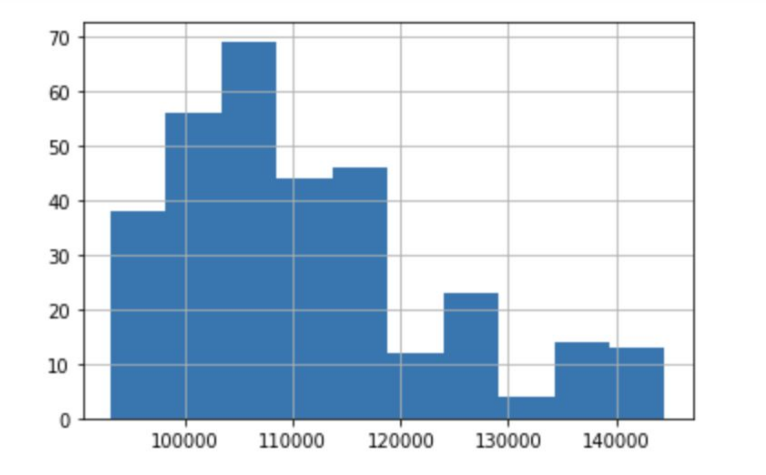
Graphic Designer




```
## [1] 90.625
```

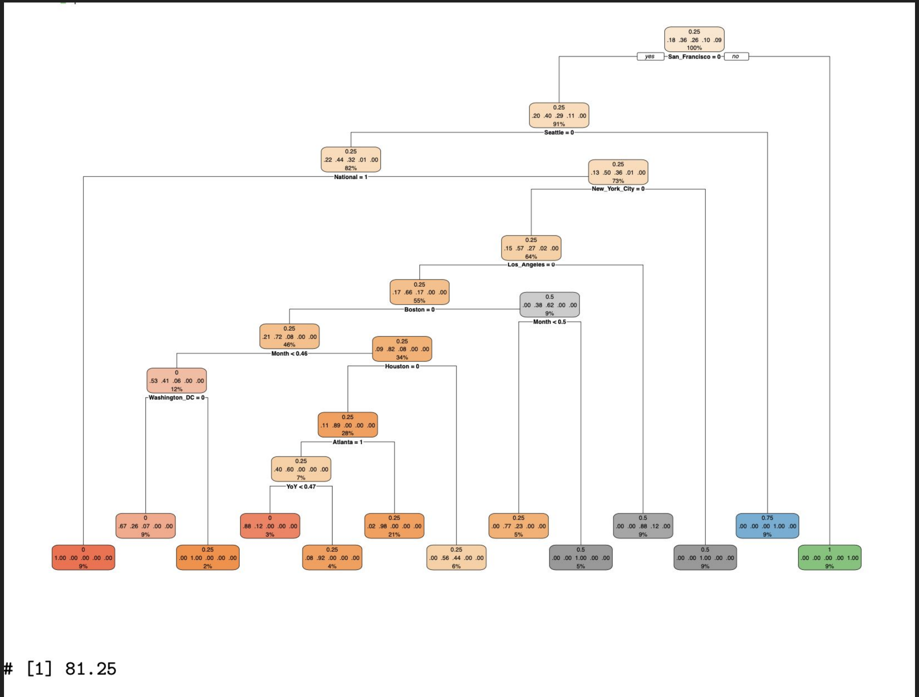
Model Building - Phase 3

- One limitation in the previous models was that the target variable was binary
- I continued working with the Data Scientist dataset and created 5 bins for the Target Variable
- After assessing the distribution of values, I set the bins to be less than $100K, between $100-110K, $110-120K, $120-130K, and above $130K.



Model Building - Phase 3

- The Decision Tree delivered an accuracy of 81.25%. A decrease from the initial models



Model Building - Phase 4

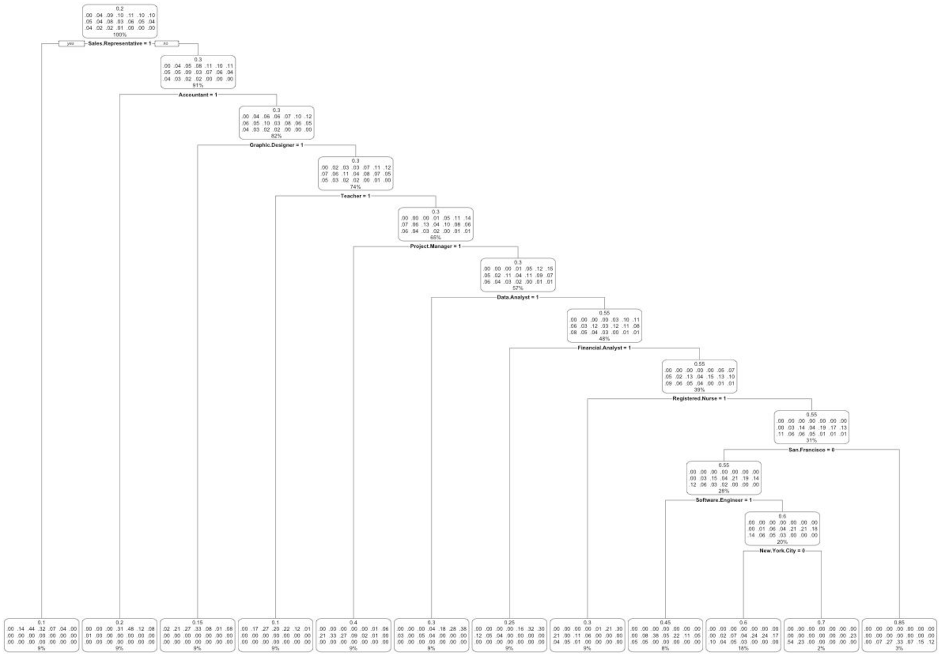
- Build a Decision Tree Model with the Job Title as the variable
- One-Hot encoded 12 job titles. These are the same ones that were included for the previous Decision Tree models

```
array(['Software Engineer', 'Project Manager', 'Financial Analyst',
       'Accountant', 'Sales Representative', 'Professor',
       'Registered Nurse', 'Teacher', 'Graphic Designer',
       'Data Scientist', 'Data Analyst', 'IT Manager'], dtype=object)
```

Model Building - Phase 4

- For the Target Variable, there were 28 bins. They started at 20,000 and increased by 5,000 to 150,000. There were bins for less than 20,000 and more than 150,000
- The Decision Tree model (next slide) was not successful with this dataset. The accuracy was 14.63%
- This makes sense since there are so many clusters for the Target Variable and this is not optimal for the Decision Tree algorithm

Model Building - Phase 4




```
## [1] 14.63415
```

Model Building - Phase 5

- The Neural Network in Python was split on an 80/20 training and test split
- The Accuracy Score was much higher than the Decision Trees at 85.48%

	precision	recall	f1-score	support
5.0	1.00	1.00	1.00	1
6.0	0.94	0.97	0.95	31
7.0	0.93	0.91	0.92	57
8.0	0.87	0.90	0.88	72
9.0	0.92	0.85	0.88	78
10.0	0.84	0.95	0.89	62
11.0	0.91	0.90	0.90	89
12.0	0.90	0.83	0.86	46
13.0	0.84	0.76	0.80	34
14.0	0.82	0.92	0.87	65
15.0	0.73	0.50	0.59	16
16.0	0.84	0.80	0.82	51
17.0	0.76	0.79	0.77	39
18.0	0.81	0.74	0.77	23
19.0	0.77	0.86	0.81	28
20.0	0.76	0.81	0.79	16
21.0	0.46	0.75	0.57	8
22.0	0.89	0.53	0.67	15
23.0	0.00	0.00	0.00	1
24.0	1.00	1.00	1.00	4
25.0	1.00	1.00	1.00	1
accuracy			0.85	737
macro avg	0.81	0.80	0.80	737
weighted avg	0.86	0.85	0.85	737

Model Building - Phase 5

```
[[ 1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0 30  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  2 52  3  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  3 65  4  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  7 66  5  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  2 59  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  6 80  3  0  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  7 38  1  0  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  1 26  7  0  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  4 60  1  0  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  6  8  2  0  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  2 41  8  0  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  6 31  2  0  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  2 17  4  0  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  2 24  2  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  0  3 13  0  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  2  6  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  7  8  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  4  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  1]]
```

```
Model Accuracy: 85.48%
```

Final Takeaways

- I was able to demonstrate that a single model can be built to predict salary for a specific job title in multiple regions and for multiple months and years
- Opportunities for Improvement
    - Increase the number of job titles being used. 12 were in the latest model but there are over 100 in the dataset.
    - Build an automated process for new monthly reports to be included for testing and training
    - Make it more accessible. How can the model take inputs?