Golden Gate

- Ajit Shrivastav
- Mentor - Jeff L.

1) Background

San Francisco City Government is committed to equal job opportunity (gender/ethnicity) and there is an initiative to showcase its commitment by projecting real data.
Government also needs to Analyze data in order to improve the organization on several factors. The initiative here is to help Government achieve its objectives.

Data is from San Francisco County Job that has following fields :-

* Name
* Job Title
* Base Pay
* Overtime Pay
* Total Pay

2) Requirements :-

Following are the deliverables for the Project

* Explore Base Pay distribution
* Explore Base Pay against race
* Explore Race distribution for various Groups of Job
* Explore Gender distribution for various Groups of Job
* Predict Total Pay for a given Base Pay
* Predict Base Pay for Job Groups

3) Data preparation

Following are the variables (of interest) available in the original dataset :-

* Job Title (dirty, not following a standard)
* Name (includes the full name with suffix)
* Base Pay
* Overtime Pay
* Total Pay

As a part of data activities, following are the items to be taken care of :-

* Clean up name, and create new variables for firstName and lastName. firstName and lastName would get used to determine gender/race
* Use mechanism to evaluate gender based on the First name
* Use mechanism to evaluate race based on the Last name
* Cleanup Job Title, and classify them in various Job Groups (Job types)
* For tree prediction model, add variable capturing salary range for Base Pay in 20k$ and 50k$ ranges

---

3.1) Name clean up

3.1.1) First step is to remove all the suffixes (JR. , I, II etc) (scripts/cleanNames.R)

3.1.2) Using regular expression split names as firstName and lastName (scripts/cleanNames.R)

3.2) Gender evaluation based on First Name

3.2.1) Use ‘RCurl’ and ‘rjson’ libraries to connect to WEB API and process JSON data

3.2.2) Use api.namsor.com/onomastics/api/json/gendre/ to evaluate gender based on first name (scripts/evaluateGender.R)
3.2.3) Cleanup records that has uni-sex names (such as Alex)

3.3) Race evaluation based on Last Name

3.3.1) Use data sets (http://names.mongabay.com/) to evaluate race
3.3.2) Also use ranking mechanism to determine the race if the last name belongs to multiple race/ethnicity (scripts/evaluateRace.R)

3.4) Evaluate Job Group

San Francisco HR site has listed all the Job Titles and associated Job code. Also, listed is Job Group for job codes.

Based on HR site information, we have following files :-

* job_groups.csv (Job code range -> Job Group)
* job_title_code_mapping_raw.csv (Raw job title to job code mapping)

Use job_groups.csv and job_title_code_mapping_raw.csv to define job_title_code_mapping.csv. job_title_code_mapping.csv has a mapping of each job title (job code) to Job groups (scripts/createJGMetadata.R)

Use job_title_code_mapping.csv to update Primary data sets with Job Groups (scripts/mapJobGroups.R)
Job Title in the primary data set is dirty. Around 10-15% of the data has been manually updated with Job Groups.

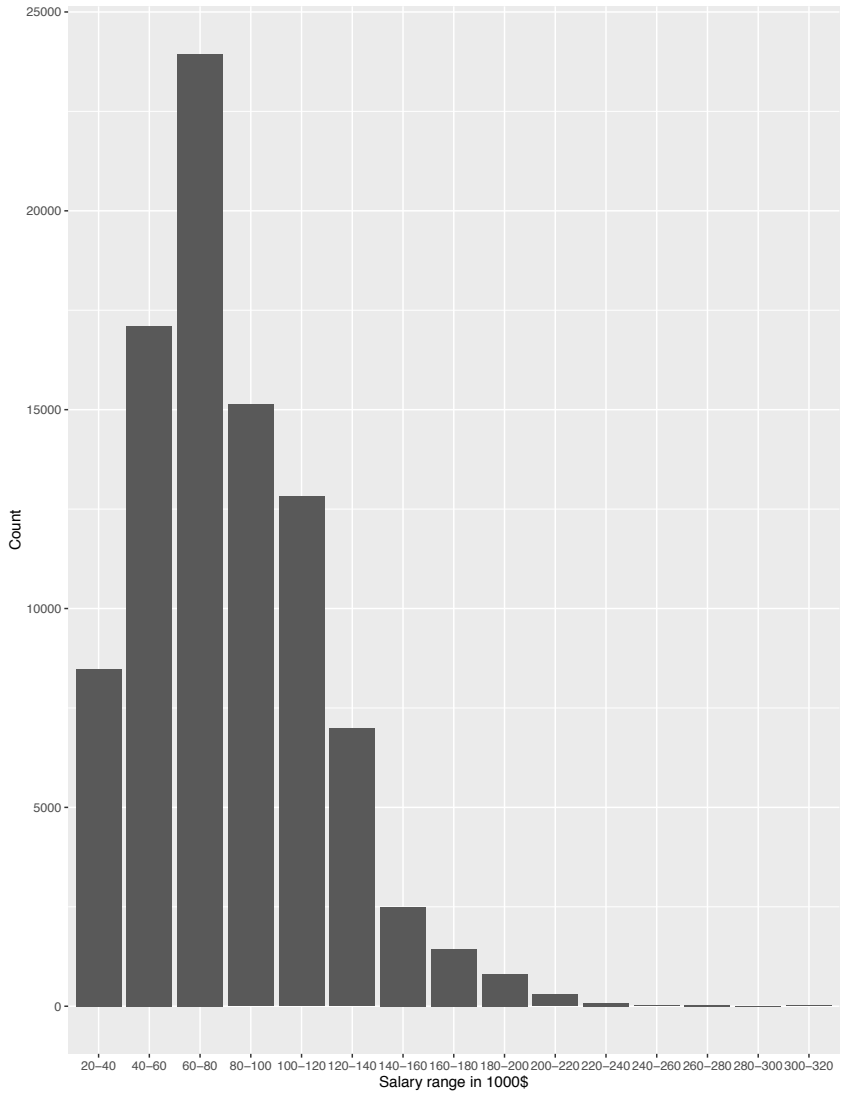
4) Explore

(ref - scripts/explore.R) This section covers data exploration. The step is done post data clean up and preparation.

4.1) Explore Base Pay distribution

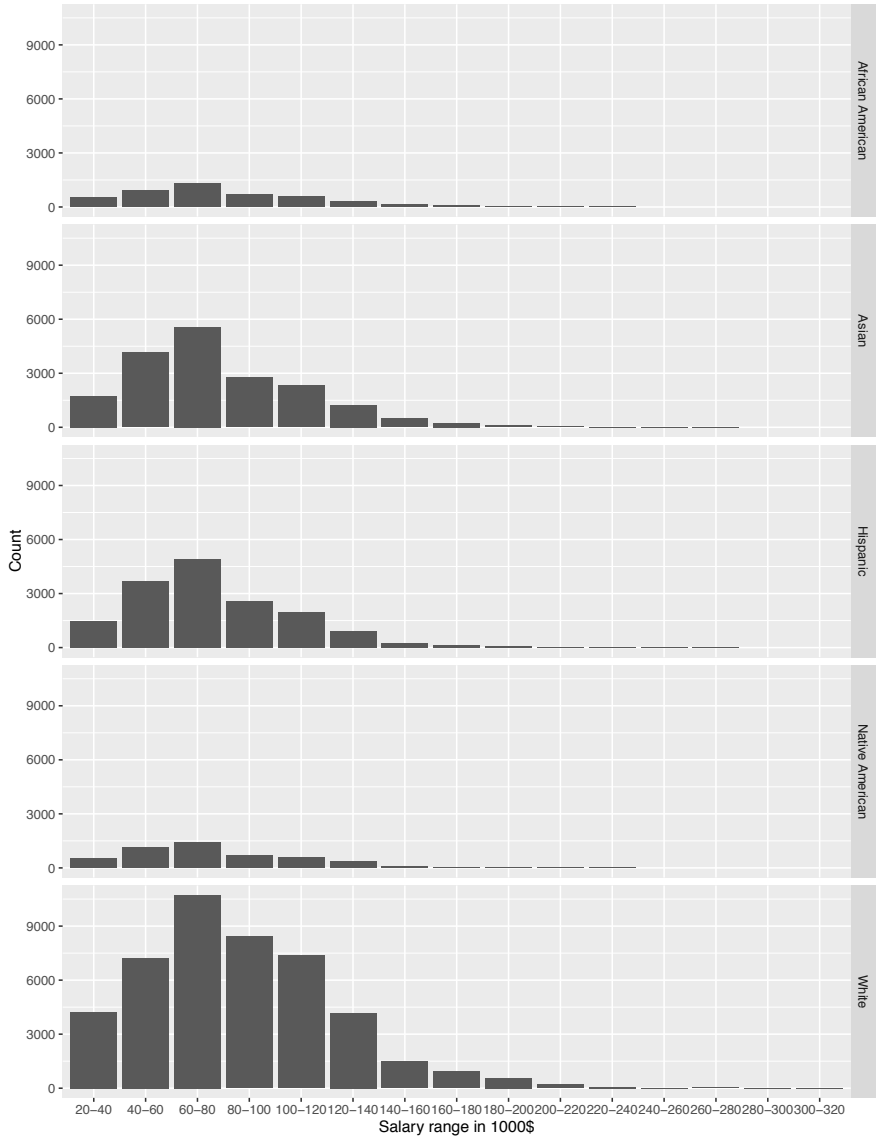
It gives organization a great insight to visualize and explore as how is the Base Pay distributed.

As seen below, it follows a normal distribution with mean around 60K-80K $. Also, looks like there are few outliers for employees having Base Pay greater than 220K $



4.2) Explore Base Pay/Job Type population for race

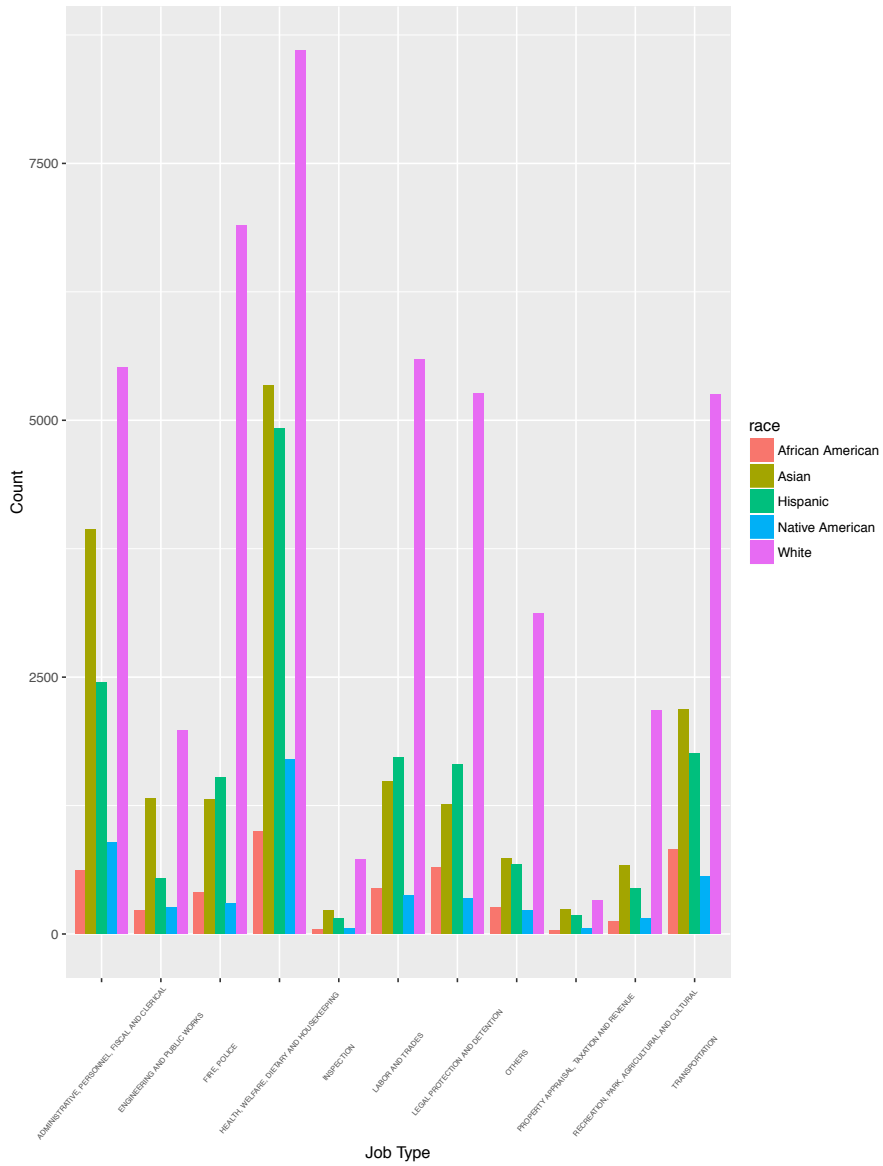
Exploring general Base Pay population distribution against a race would also give Organization a fair idea as how employees are distributed/populated across different race and how their Base Pay compares to each other.



Visuals above gives a clear picture of Base Pay distribution across different races. Again, for each race, it is quite visible that Base Pay follows a Normal distribution (on Base Pay scale).

From the visual it is clear that “White”’s are more populous than other categories. “African American” and “Native American” are least populous.

Let’s plot another visual to see how different race employees are distributed across various Job types



From the visual above, it is clear that “White” is more populous across each job types. It is also interesting to see how employees for certain race tend to take up Job Type.

In “FIRE, POLICE”, “White” tend to take more jobs in comparison to other race. “Asian” and “Hispanic” tend to take more jobs in “HEALTH, WELFARE, DIETARY AND HOUSEKEEPING”. “Asian” also tend to take more jobs in “ADMINISTRATIVE, PERSONNEL, FISCAL AND CLERICAL”.

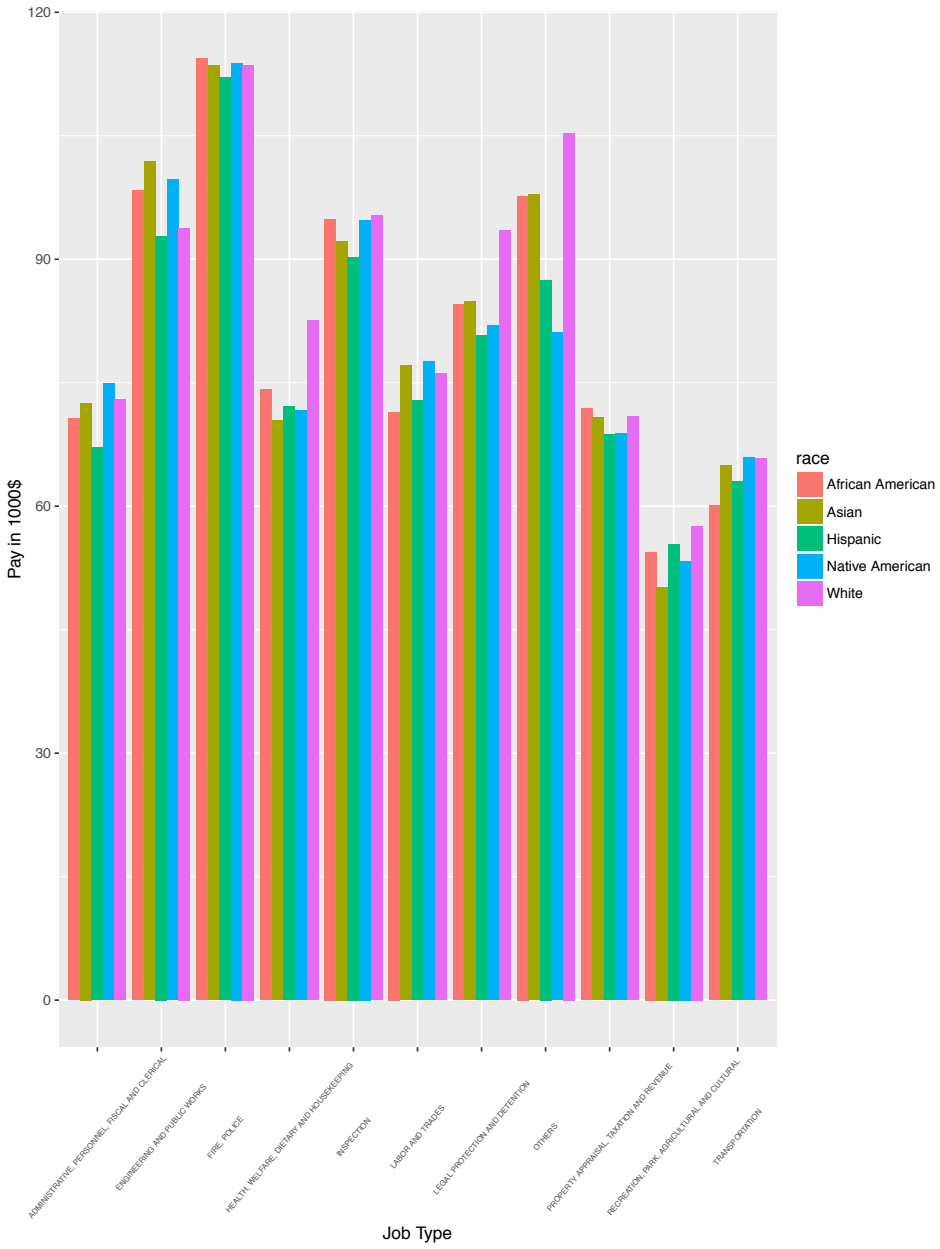
Three conclusions we can draw from the graph above are :-

* Whites tend to take more jobs in each Job Types (some places with a huge margin)
* Asian / Hispanic tend to take jobs more in Administrative, Health and Housekeeping categories
* African American / Native American are the least populous among each category but that doesn't imply any strong bias towards a Job Type

Next section would explore Racial / Gender discrimination across each Job Groups.

4.3) Explore discrimination for Race

We have seen above how is the race population distributed. Over here, we will take it one step further to see how Base Pay of a Race compares to others in certain Job categories. This will help in understanding if there is any bias/discrimination



We understand that population of a race would not help us inferring any bias / discrimination. Big difference in average Pay for different Race in the same Job Group should raise a question. But from the visual, it doesn't looks like that's the case. There is a marginal difference but that sounds reasonable and acceptable.

To take it a level further, to prove there is no discrimination of BasePay with respect to race, we should consider the hypothesis "BasePay has a relation to Race" and try to prove NULL Hypothesis is true in that case.

Let's try to build 2 models to prove the hypothesis "BasePay has a relation to Race"

4.3.1) Linear model :-

```
lm_basePay_race <- lm(BasePay ~ race, data = salary)
summary(lm_basePay_race)
```

```
lm(formula = BasePay ~ race, data = salary)

Residuals:
    Min      1Q  Median      3Q     Max
-64.614 -23.614  -6.601  22.386 233.386

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)         78.1036     0.5180 150.767  < 2e-16 ***
raceAsian           -0.5022     0.5785  -0.868 0.385315
raceHispanic        -1.9563     0.5880  -3.327 0.000878 ***
raceNative American -1.2562     0.7220  -1.740 0.081857 .
raceWhite            7.5107     0.5437  13.814  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 35.2 on 89653 degrees of freedom
Multiple R-squared:  0.01482,   Adjusted R-squared:  0.01477
F-statistic: 337.1 on 4 and 89653 DF,  p-value: < 2.2e-16
```

From the summary above, R-squared value implies, the model accounts for not more than 14.8% which is a very low value to prove the hypothesis “BasePay has a relation to Race” with linear model.

4.3.2) Random Forest Model :-

Lets build a Random Forest model (200 trees) predicting Base Pay for Race to prove the hypothesis “BasePay has a relation to Race”

```
basePay_race.rf <- randomForest(as.factor(basePayToBase50) ~ race,
                    train_set,
                    ntree=200)
predictForest <- predict(basePay_race.rf, newdata = test_set)
confusion_matrix <- table(test_set$basePayToBase50, predictForest)
```

Confusion matrix :-

```
> confusion_matrix
          predictForest
            0-50 100-150 150-200 200-250 250-300 300-350 50-100
  0-50         0       0       0       0       0       0   4307
  100-150      0       0       0       0       0       0   6421
  150-200      0       0       0       0       0       0    941
  200-250      0       0       0       0       0       0    117
  250-300      0       0       0       0       0       0     16
  300-350      0       0       0       0       0       0      4
  50-100       0       0       0       0       0       0  15092
```

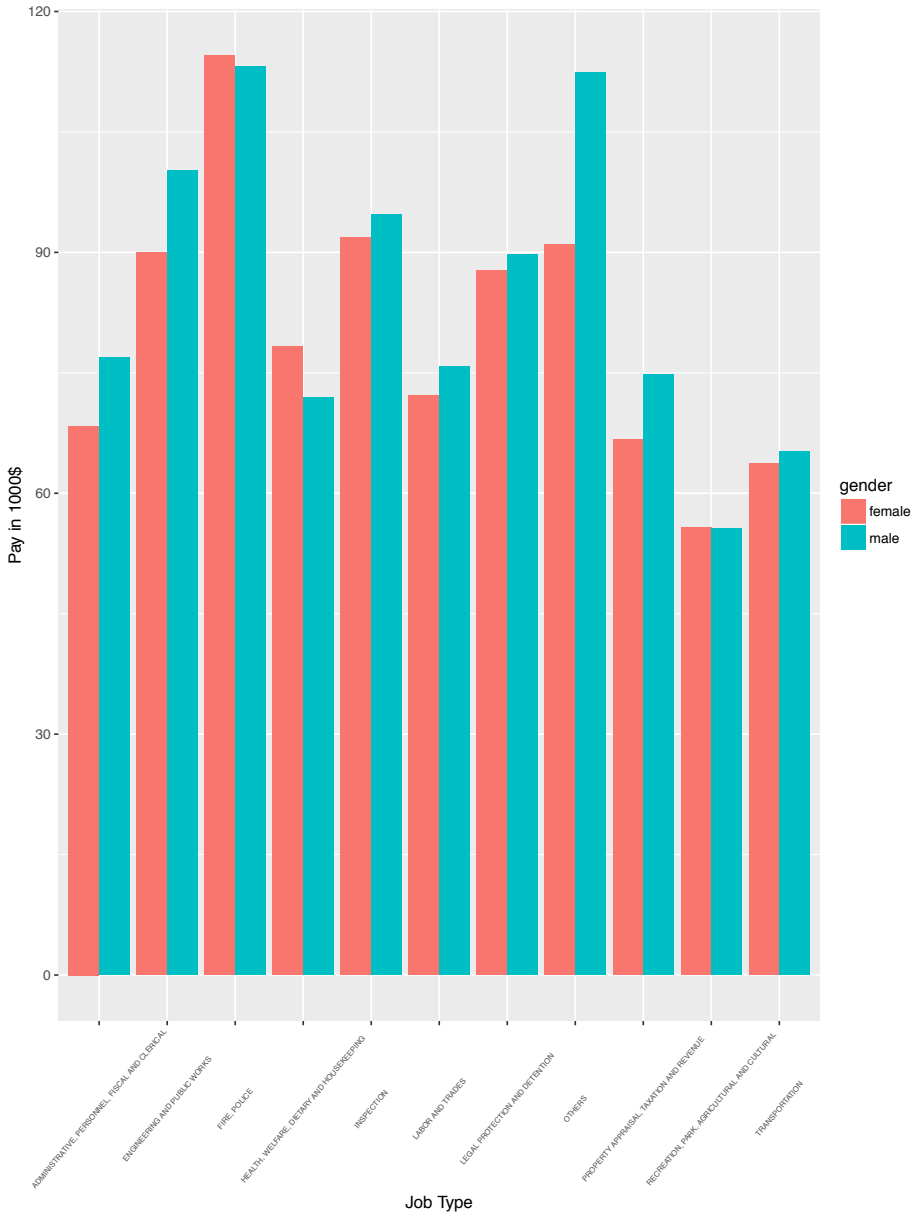
Even though the accuracy for the model is around 56.1%, based on the confusion matrix, it is quite evident that base pay predicted is always 50-100K. Hence, Base Pay prediction is always constant irrespective of the race.

From both the models (Linear & Random Forest), we can easily conclude NULL hypothesis is true for “BasePay

has a relation to Race”. Hence implying that there is a discrimination on Pay based on Race/Ethnicity would be wrong.

4.4) Explore discrimination for Gender

Lets analyze and explore Pay distribution for Males/ Females across different Job groups.
Average Base Pay for Males and Females across different Job groups is plotted.

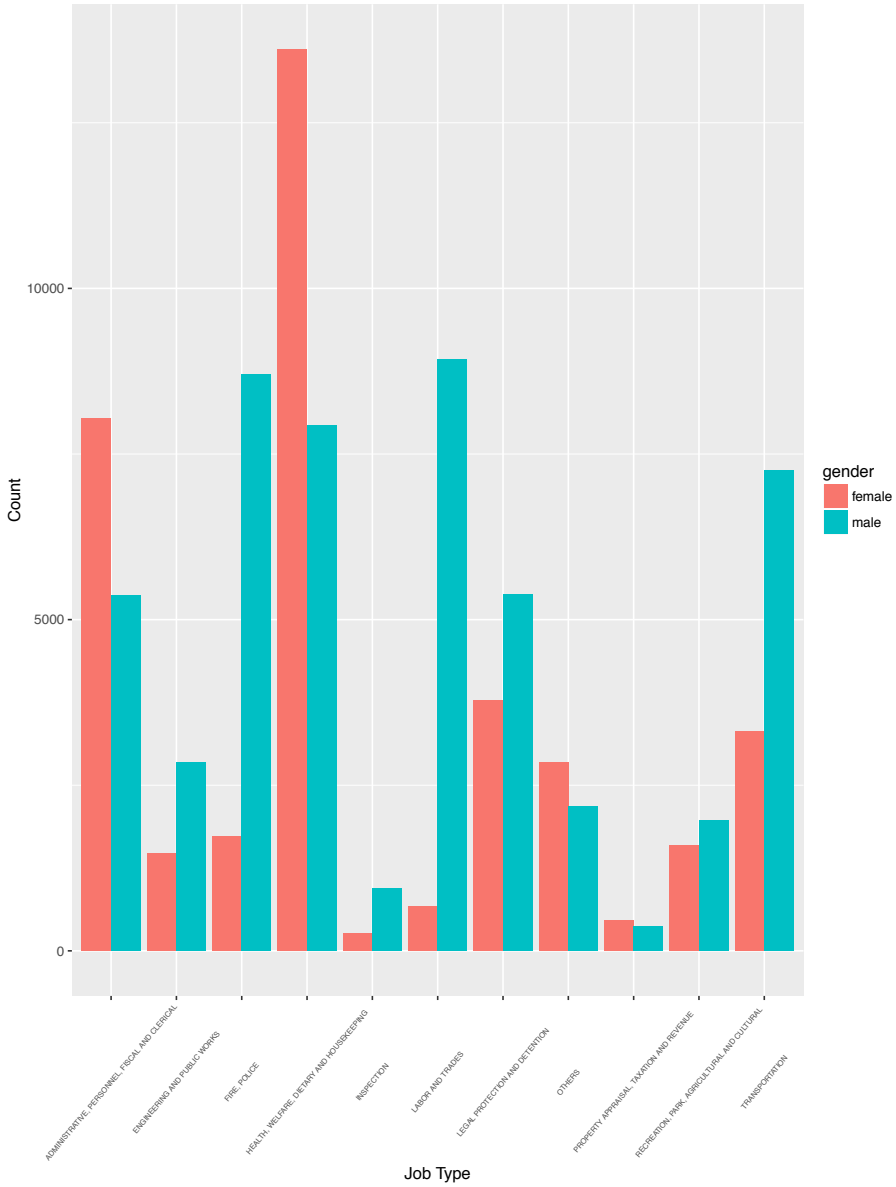


The above visual makes a lot of sense and gives us a good insight on Males/Females average Pay distribution across various Job Groups.

From the above plot, the Pay distribution for both genders doesn't seem to have any bias. The pay difference seems to be in acceptable limits.

We should also explore Gender bias in terms of Jobs in various Job Groups.

Below is how the distribution looks like for Job distribution for male/female



What is quite evident from the visual above is, there is some amount of disparity between genders for different types of Jobs. But we also need to consider the type / nature of the job before concluding that there is a bias.

If we look at the visual, “Males” dominate in certain job types such as “FIRE, POLICE”, “LABOR AND TRADES” whereas “Females” dominate in certain job types such as “ADMINISTRATIVE, PERSONNEL, FISCAL AND CLERICAL”, “HEALTH, WELFARE, DIETARY AND HOUSEKEEPING”. Analyzing the visual, it doesn’t raise an alarm with respect to disparity once the nature/type of job is considered.

To conclude, we can say that “Males” tend to take certain job types such as “FIRE, POLICE”, “LABOR AND TRADES” and “Females” tend to take certain job types such as “ADMINISTRATIVE, PERSONNEL, FISCAL AND CLERICAL”, “HEALTH, WELFARE, DIETARY AND HOUSEKEEPING”. But that doesn’t amount to a bias once nature/type of job is considered.

5) Predict

(Ref - scripts/predict.R) This section would focus on building Prediction Model for 2 use cases.

5.1) Predict Total Pay for a given Base Pay

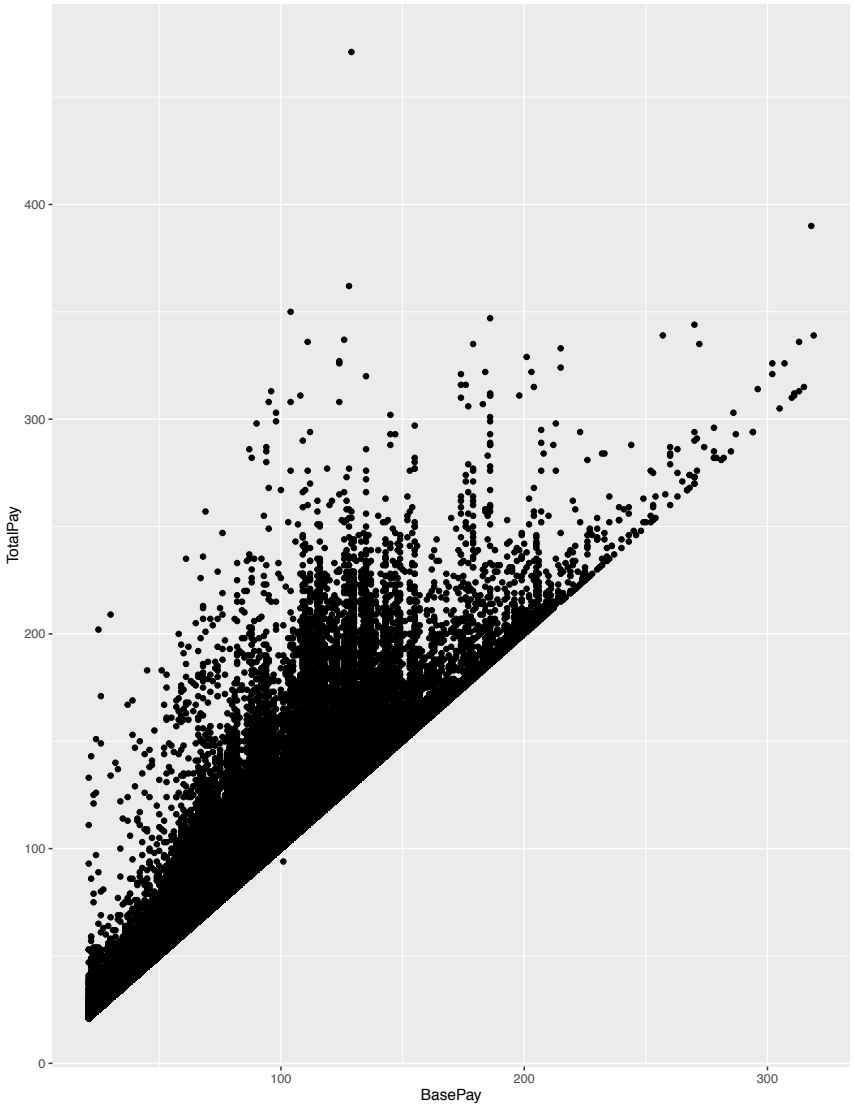
Use Case :- John Smith is joining as an Employee and he has been offered a BasePay of 110K$. John has been told that there is overtime hours work expected from the job and he would be paid appropriately as per the number of hours spent overtime.
He is also been told that overtime hours is a variable component as per the need basis, hence they cannot tell them in advance what would be his over time pay for the next whole year.
So now, John is trying to estimate what would be his TotalPay approximately for the BasePay he has been offered. Can he really come up with an estimate? Let's look at the available data and see if a model can be derived that would help John to estimate TotalPay.

From the available data, we have BasePay and TotalPay available to us.

So, first thing we should do is look at the data and see if there is a relation between BasePay and TotalPay. Let's plot a graph with Base Pay on X-axis and Total Pay on Y-axis

Plotting :-

```
plot <- ggplot(data=salary, aes(BasePay, TotalPay))+geom_point()
plot
```



Looking at the above plot, it is quite clear that TotalPay increases linearly with BasePay. Let’s try building Linear model here.

Linear model:-

```
lm_totalPay_basePay <- lm(TotalPay ~ BasePay, data = salary)
summary(lm_totalPay_basePay)
```

Model summary :-

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.589039   0.140088   4.205 2.62e-05 ***
BasePay     1.125776   0.001578 713.431  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 16.75 on 89656 degrees of freedom
Multiple R-squared:  0.8502,    Adjusted R-squared:  0.8502
F-statistic: 5.09e+05 on 1 and 89656 DF,  p-value: < 2.2e-16
```

R-squared as 0.8502, signifies that it is a strong model for prediction.

Formula :-

$$(Total\ Pay) = 0.589 + (1.126)*(Base\ Pay)$$

Using the linear model, John predicts his TotalPay :-

TotalPay = 0.589 +(1.126)*(110) = 124.45K ~ 125K $

5.2) Predict Base Pay for Job Groups

From the data and nature/domain, it is obvious that Linear model would not fit to predict Base Pay for a given Job Type.

Before Predicting, there is 2 cleanup activity :-
-> BasePay is rounded in K (1000$). Also, we would be predicting for a categorical / factor BasePay. Hence, Base Pay is added as a range of 20K. (ex - 20K-40K $, 40K-60K $

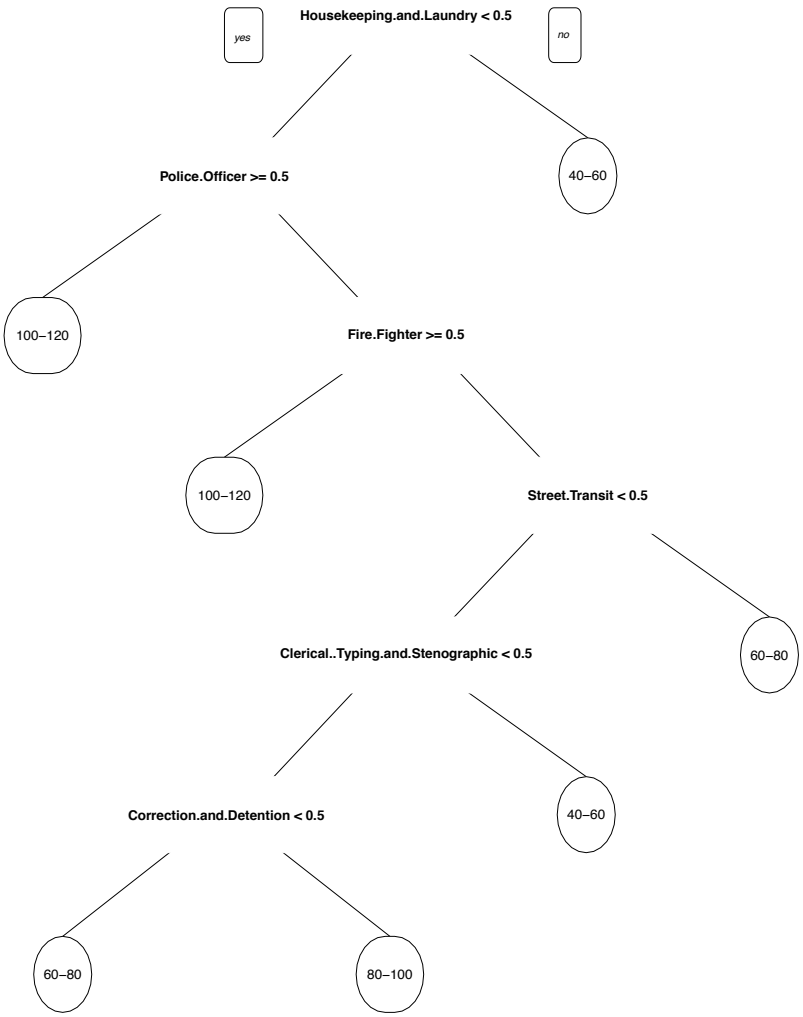
-> Job Group (which is evaluated based on available data) is spread across the data set as columns with binary values

It is understood that Base Pay would be primarily driven with kind of Job (Job Type). Also, Decision Tree may fit here very well based on Job Groups.

Decision tree model code (70:30 - Training:Test):-

```
salary_tree_model <- rpart(basePayToBase20 ~ . , data = train_set)
prp(sal_tree, varlen=40, tweak=0.75)
```

The above model gives us the following decision tree :-



Above model gives a good prediction model to determine Base Pay salary bucket for various Job Groups.

Example :- For a “Fire Fighter” and “Police Officer”, average Base Pay prediction comes to be the same between 100k-120k $. For “Housekeeping and Laundry” Base Pay can be predicted to fall in 40K-60K $.

There are certain take aways from the Decision Tree model :-

* Housekeeping, Laundry, Clerical are the least paying jobs
* Fire, Police are good paying jobs
* Street Transit, Correction Detention are moderate paying jobs.

However, Decision tree model gives an accuracy that is not acceptable.

Random Forest is a model that may improve the accuracy. Based on the data available, BasePay is divided in 50K Base Pay range.

Available data is divided in training (70%) and test (30%) data set.

Random forest model is built with 150/200/300/500 trees. Based on the model generated, there is no improvement after 200 trees. Hence 200 trees model is built on training set.

To see the accuracy, Confusion Matrix is created from the Predicted model on training data set :-

```
> confusion_matrix
         predictForest
           0-50 100-150 150-200 200-250 250-300 300-350 50-100
  0-50      657    2237       0       0       0       0   7156
  100-150     0   11372       0       0       0       0   3611
  150-200     0    1215       0       0       0       0    979
  200-250     3     167       0       0       0       0    102
  250-300     0      30       0       0       0       0      7
  300-350     1       5       0       0       0       0      2
  50-100    137    5120       0       0       0       0  29959
> sum(diag(confusion_matrix))
[1] 41988
> sum(confusion_matrix)
[1] 62760
> 41988/62760
[1] 0.6690249
>
```

On training data set, there is an accuracy of 66.9%.

The same model is applied on test data set. Confusion Matrix from Predicted test model :-

```
> confusion_matrix
         predictForest
           0-50 100-150 150-200 200-250 250-300 300-350 50-100
  0-50      302     896       0       0       0       0   3109
  100-150     0    4888       0       0       0       0   1533
  150-200     0     524       0       0       0       0    417
  200-250     1      71       0       0       0       0     45
  250-300     0      11       0       0       0       0      5
  300-350     1       3       0       0       0       0      0
  50-100     60    2270       0       0       0       0  12762
> sum(diag(confusion_matrix))
[1] 17952
> sum(confusion_matrix)
[1] 26898
> 17952/26898
[1] 0.6674102
>
```

Test model predicts 66.7% of accuracy for Random Forest Model.

The RandomForest model could be taken as a valid model to predict BasePay for certain Job Types.