Golden Gate

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

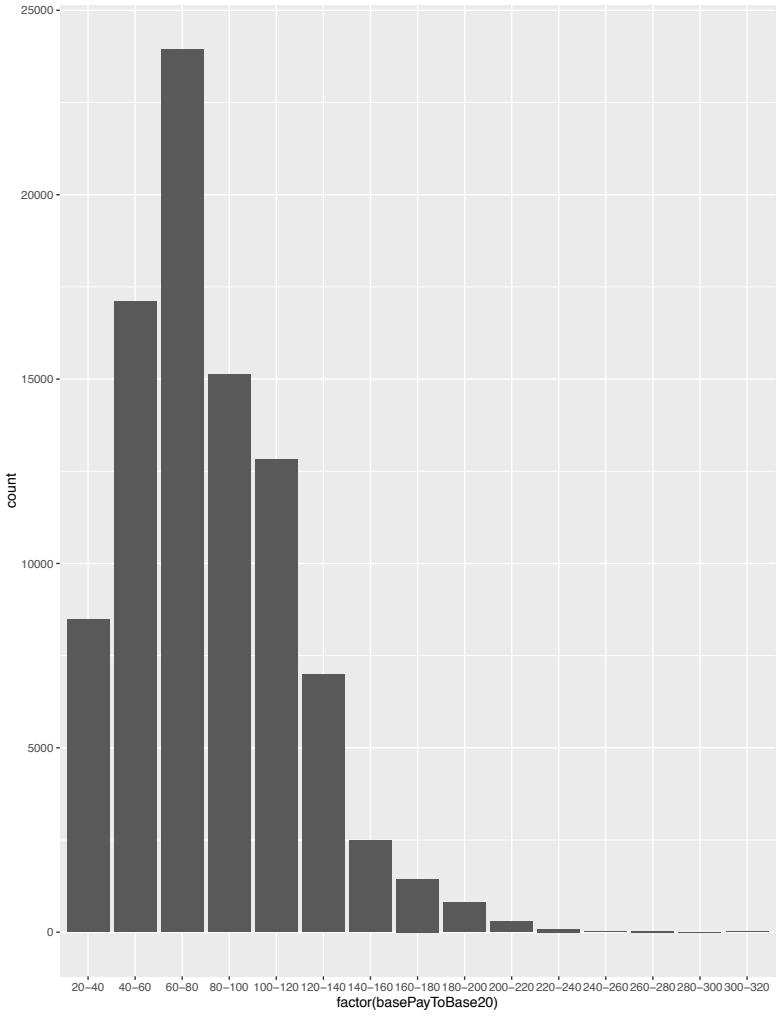
3) Explore

This section covers data exploration. The step is done post data clean up and preparation.

3.1) Explore Base Pay distribution

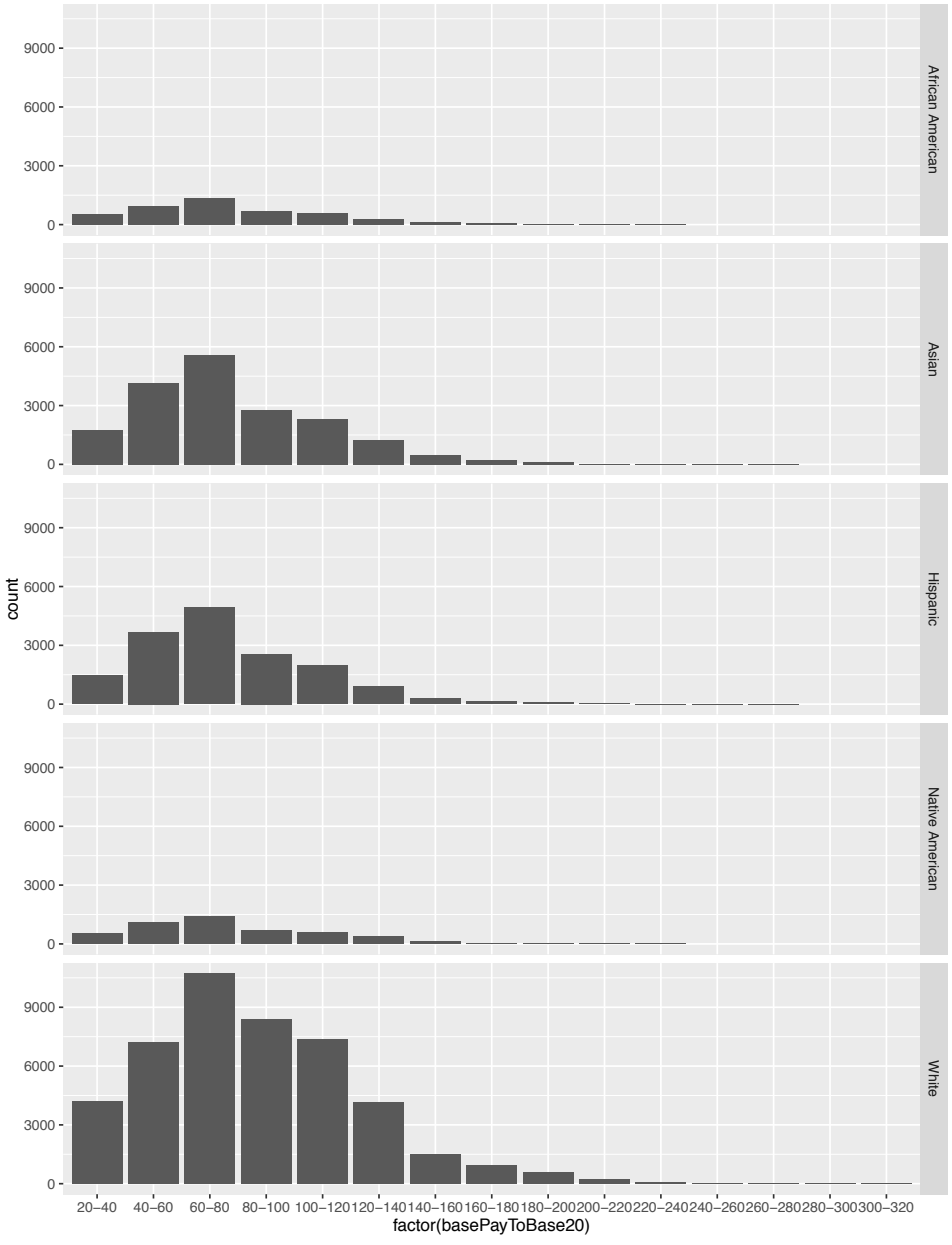
It gives organization a great insight to visualize and explore as how is the Base Pay distributed.

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 $



3.2) Explore Base Pay against race

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



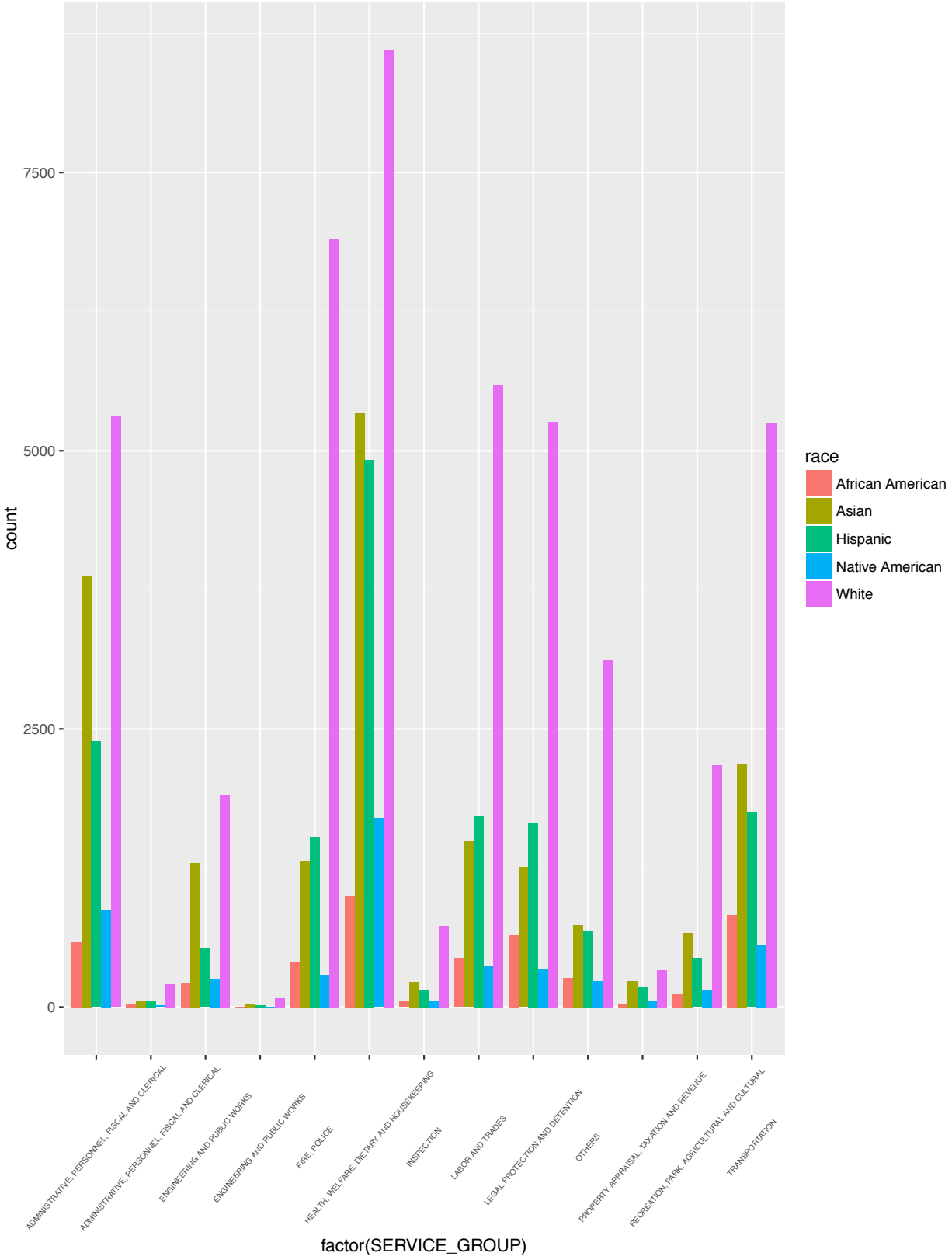
Visuals above gives a clear picture of Base Pay distribution across different races. Again, for each race, it is quite visible that Base Pay follows a Normal distribution (on Base Pay scale).

This also raises a question on population of Employees. From the graph it is quite clear that 'White' race dominates in terms of population. Hispanic and Asian race pretty much are comparable in terms of population. Native American / African American are the least populous. This can come out as an Action item for HR department to introspect if there is a bias in terms of employment.

Next section would explore Employee distribution across each Job Groups.

3.3) Explore Race distribution for various Groups of Job

We have seen above how is the race population distributed. Over here, we will take it one step further to see the race population distribution for various Job Groups

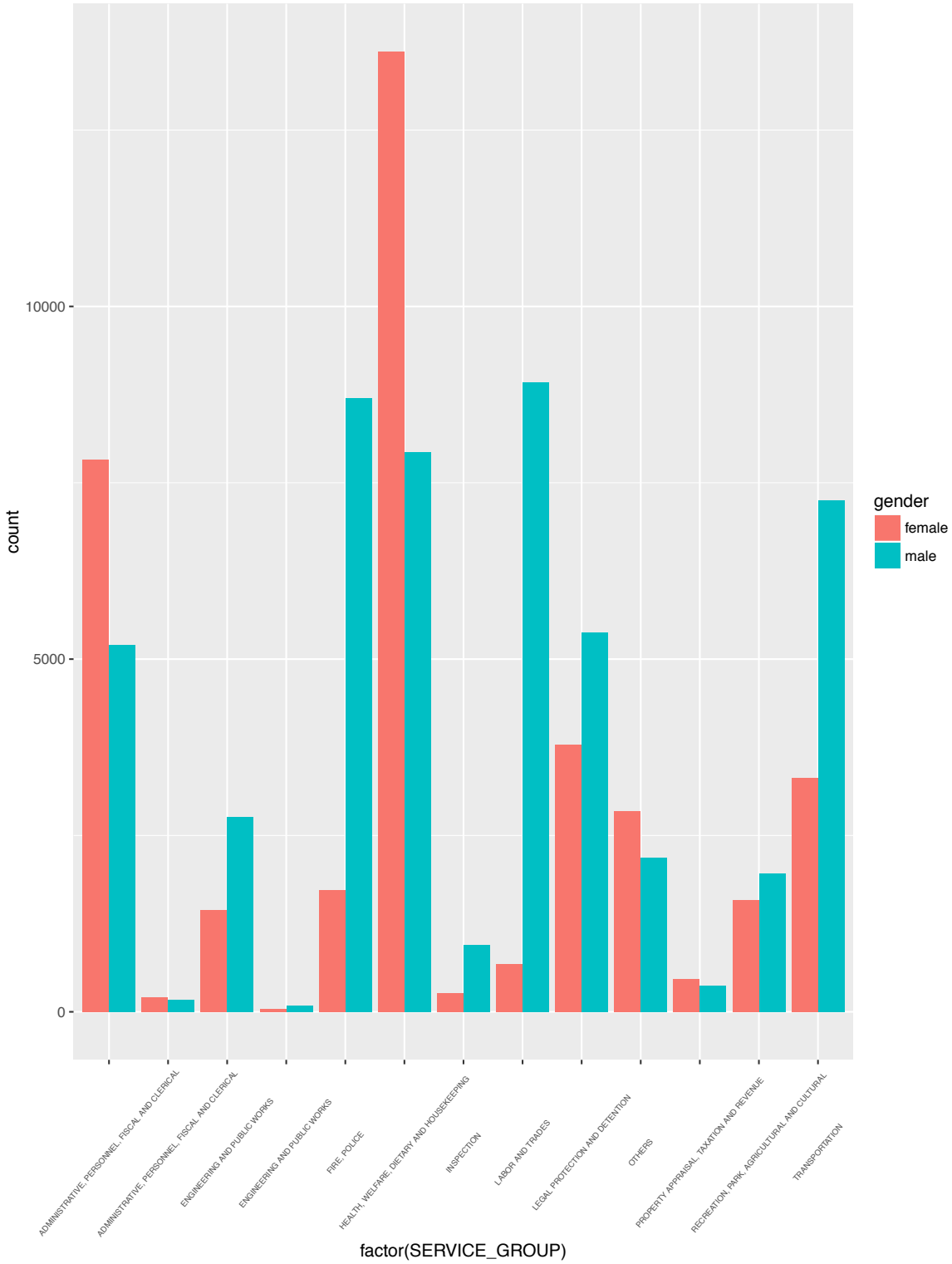


From the above visual, it is clear that White race is dominating across all the Job Groups. “White”s are more dominating in areas such as “FIRE/POLICE”, “LOBOR AND TRADES” and “LEGAL, PROTECTION AND DETENTION”. “Asian”s are faring well in areas such as “ADMINISTRATIVE,

PERSONNEL, FISCAL AND CLERICAL", "HEALTH, WELFARE, DIETARY AND HOUSEKEEPING".

3.4) Explore Gender distribution for various Groups of Job

Over here, we will see how gender population is distributed across various Job Groups.



The above visual makes a lot of sense and gives us a good insight on male/female distribution across various Job Groups.

For instance, in “LABOR AND TRADES” and “FIRE, POLICE, we see “male” dominating “female” with a huge margin. “ADMINISTRATIVE, PERSONNEL, FISCAL and CLERICAL” jobs shows “female”s dominating “male”s. Also, for “HEALTH, WELFARE, DIETARY AND HOUSEKEEPING” shows “female”s dominating “male”s by a good margin. The distribution makes a lot of sense and doesn’t raise any question on gender discrimination at a high level.

4) Predict

This section would focus on building Prediction Model with the available data.

4.1) Predict Total Pay for a given Base Pay

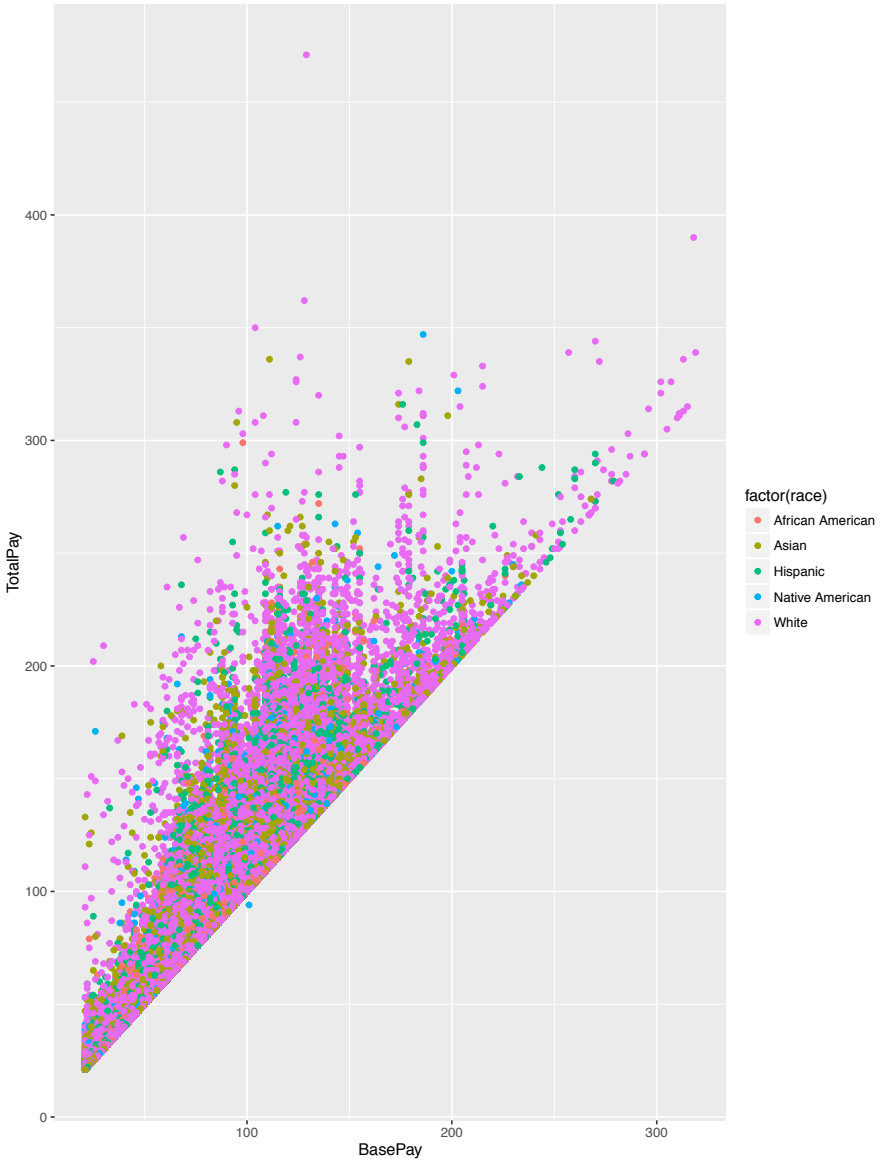
For a Person joining as an Employee has Base Pay and Total Pay. Total Pay is Base Pay + Overtime Pay.

Overtime Pay is a component which is not fixed. It becomes very important for an individual to determine what would be the Total Pay for a given Base Pay.

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

Plotting code :-

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


TotalPay
400
300
200
100
0
100
200
300
BasePay
factor(race)
African American
Asian
Hispanic
Native American
White

Looking at the above plot, it is quite clear that Linear Model could be a very strong for predicting Total Pay against Base Pay.

Linear model code :-

```
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
```

Looks like with R-squared as 0.8502, it is a strong model for prediction.

Formula :-

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

4.2) 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.

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 to a bucket of 20K. (ex - 20K-40K $, 40K-60K $

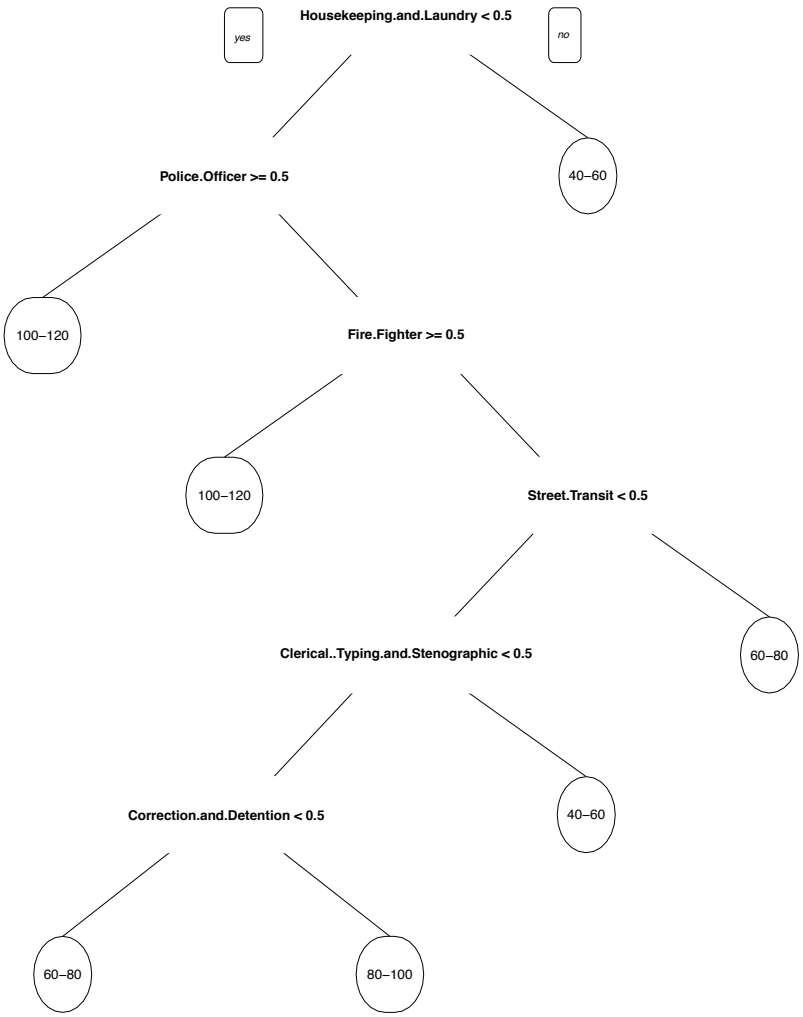
-> Job Group (which is evaluated based on available data) is spread across the data set as columns with binary values

It is understood that Base Pay would be primarily driven with kind of Job (Job Group). Also, Decision Tree fits here very well as based on Job Groups, decisions are taken to reach to the final outcomes.

Decision tree model code :-

```
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 $.

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 bucket.

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.

6) Implementation

6.1) Data cleanup and preparation

6.1.1) First step is to remove all the suffixes (JR. , I, II etc) (pass1_gender.R)
6.1.2) Using regular expression split names as firstName and lastName (pass1_gender.R)
6.1.3) Use data set from to evaluate gender at first pass (pass1_gender.R)

6.2) Gender evaluation based on First Name

6.2.1) Use 'RCurl' and 'rjson' libraries to connect to WEB API and process JSON data
6.2.2) Use http://api.namsor.com/onomastics/api/json/gendre to evaluate gender based on first name (get_gender_from_web.R)
6.2.3) Cleanup records that has uni-sex names (such as Alex)

6.3) Race evaluation based on Last Name

6.3.1) Use data sets (http://names.mongabay.com/) to evaluate race
6.3.2) Also use ranking mechanism to determine the race if the last name belongs to multiple race (eval_race.R)

6.4) Base Pay / Total Pay preparation

6.4.1) Round up Base Pay and Total Pay to base of 1000$

```
salary <- salary  %>% mutate (BasePay = as.integer(BasePay/1000))
salary <- salary  %>% mutate (TotalPay = as.integer(TotalPay/1000))
```

6.4.2) Put salary in buck of 20K (base to 20K$). Example 27K would be 20K-40K $, 60K would map to 40K-60K$

```
salary <- salary  %>% mutate (basePayToBase20 =
paste(as.character(as.integer(BasePay/20)*20), as.character(as.integer(BasePay/20)*20 +
20),sep="-"))
```

6.5) Job Groups evaluation and preparation

6.5.1) Create a mapping file for job code to Job Title (https://www.jobaps.com/SF/auditor/ClassSpecs.asp)

6.5.2) Create a mapping file mapping job codes to Job Groups

6.5.3) In the data set, assign Job Group to each record based on the mapping files (map_jg_sg_primary_data.R)

6.5.4) Tidy the data on Job Groups for Prediction :-

```
salary <- salary %>%
  mutate(yesno = 1) %>%
  distinct %>%
  spread(JOB_GROUP, yesno, fill = 0)
```

6.6) Explore :-

6.6.1) (plot_salary.R)

6.7) Predict :-

6.6.1) Linear regression model for Base Pay (independent variable) to Total Pay (dependent variable) (plot_salary.R)

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

6.6.2) Decision Tree and Random Forest prediction model for Job Group (independent variable) to Base Pay (dependent variable) (plot_salary.R)

```
#### Decision Tree model
sal_tree <- rpart(basePayToBase20 ~ . , data = train_set)
prp(sal_tree, varlen=40, tweak=0.75)

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

pred = predict(sal_tree, type="class")
table(pred)
table(pred, test_set$basePayToBase20)


#### Random Forest model
varNames <- names(train_set)
varNames <- varNames[!varNames %in% c("basePayToBase50")]
varNames1 <- paste(varNames, collapse = "+")
rf.form <- as.formula(paste("basePayToBase50", varNames1, sep = " ~ "))
salary.rf <- randomForest(rf.form,
                          train_set,
                          ntree=200)
predictForest <- predict(salary.rf, newdata = test_set)
confusion_matrix <- table(test_set$basePayToBase50, predictForest)
```