San-Francisco Employee Data Prediction

Table of Contents

1. Introduction

Our application is based on San Francisco Employee compensation data which describes the various features related to employee department, organization, job profile, salary and benefits.

In a corporate structure, employees are the integral part of the organization. No matter your company size, your people are your most important asset. They are the backbone of your business. So, one of the most important aspects of running your business is keeping your employees happy by offering them high-quality employee benefits and compensation.

Employee benefits refer to all non-wage compensation or bonus provided to employees in addition to their salaries. The type of benefits your company decides to offer will vary based on the organization and job profile.

But, sometimes many companies don't realize how much time and money ineffective HR processes are costing them. Providing benefits to those type of job profile which have a very low productivity has often come into wrong consideration. So, there must be some solution in which company can know in advance about the compensation structure based on job profile and organization. This provided us the opportunity to develop model which can predict compensation and benefits based on different factors. Employers can use this model to imbibe some knowledge regarding the compensation factors and employees can use it to decide which job profiles are receiving maximum benefits

2. Data

Briefly introduce your data sets, such as which application or domain the data belongs to, where did you collect it, how large it is, how many features there are, and so forth.

- The dataset hosted by the city of San Francisco. The organization has an open data platform and they update their information according the amount of data that is brought in. The San Francisco Controller's Office maintains a database of the salary and benefits paid to City employees since fiscal year 2013.

- This dataset is updated annually. New data is added on a bi-annual basis when available for each fiscal and calendar year. It has been collected from kaggle.com (https://www.kaggle.com/san-francisco/sf-employee-compensation) and is available in csv format (170 MB). There are 8,35,308 instances(records) and 22 attributes(columns) in the dataset. Out of 22 attributes, 13 are numerical variables and 9 are categorical variables.

  Following are the attributes in this dataset:

  - Year Type: (Nominal/Categorical variable)
  - Year: (Numerical)
  - Organization Group Code: (Numerical)

- Organization Group: (Nominal/Categorical variable)
- Department Code: (Nominal/Categorical variable)
- Department: (Nominal/Categorical variable)
- Union Code: (Numerical)
- Union: (Nominal/Categorical variable)
- Job Family Code: (Nominal/Categorical variable)
- Job Family: (Nominal/Categorical variable)
- Job Code: (Nominal/Categorical variable)
- Job (Nominal/Categorical variable)
- Employee Identifier: (Numerical)
- Salaries: (Numerical)
- Overtime: (Numerical)
- Other Salaries: (Numerical)
- Total Salary: (Numerical)
- Retirement: (Numerical)
- Health/Dental: (Numerical)
- Other Benefits: (Numerical)
- Total Benefits: (Numerical)
- Total Compensation: (Numerical)

3. Problems to be Solved

List your research problems, that is, what kinds of the problems you want to solve. You cannot simply say I want to explore the data and find the patterns You should provide finer-grained research problems that can be solved by statistical techniques.

- Based on the dataset, the interested research problems are:

1. Predicting the total compensation of the employee based on various factors that will help the employers to decide what compensation should be given to employee in advance in order to keep tabs on their financial section.

2. Predicting the salaries of the employee based on benefits, compensation and job profile that will help the employees to aim for better job profiles based on high benefits.

3. Are the average salaries of all the employees same or different for various organizations or job profiles?

4. Solutions

For each problem you list above, figure out feasible solutions, and introduce your plan to perform experiments

➢ Feasible solutions:

1. We will use Multiple linear regression to predict the compensation and benefits given to the employee based on salary, organization and job profile.

2. We will use Multiple linear regression to predict the total salary given to the employee based on organization and job profile and other factors.

3. We will use ANOVA to compare average salaries of different employees based on job profiles and organization.

5. Experiments and Results

5.1. Methods and Process

1. Preprocessing:

1.1 Checking and Removal of Negative values from the following numerical variables

```
[1] 16
> subdata$salaries[subdata$salaries < 0]=mean(subdata$salaries)
> nrow(subdata[subdata$salaries<0,])
[1] 0
> nrow(subdata[subdata$overtime <0,])
[1] 14
> subdata$overtime[subdata$overtime < 0]=mean(subdata$overtime)
> nrow(subdata[subdata$overtime <0,])
[1] 0
> nrow(subdata[subdata$other_salaries <0,])
[1] 17
> subdata$other_salaries[subdata$other_salaries < 0]=mean(subdata$other_salaries)
> nrow(subdata[subdata$other_salaries <0,])
[1] 0
> nrow(subdata[subdata$total_salary <0,])
[1] 11
> subdata$total_salary[subdata$total_salary < 0]=mean(subdata$total_salary)
> nrow(subdata[subdata$total_salary <0,])
[1] 0
> nrow(subdata[subdata$retirement <0,])
[1] 82
> subdata$retirement[subdata$retirement < 0]=mean(subdata$retirement)
> nrow(subdata[subdata$retirement <0,])
[1] 0
> nrow(subdata[subdata$health_and_dental <0,])
[1] 53
> subdata$health_and_dental[subdata$health_and_dental < 0]=mean(subdata$health_and_dental)
> nrow(subdata[subdata$health_and_dental <0,])
[1] 0
> nrow(subdata[subdata$other_benefits <0,])
[1] 146
> subdata$other_benefits[subdata$other_benefits < 0]=mean(subdata$other_benefits)
> nrow(subdata[subdata$other_benefits <0,])
[1] 0
> nrow(subdata[subdata$total_benefits <0,])
[1] 101
> subdata$total_benefits[subdata$total_benefits < 0]=mean(subdata$total_benefits)
```

➢ Salaries
➢ Overtime
➢ Other Salaries
➢ Retirement
➢ Other Benefits

1.2 Replacement of missing values in the following nominal variables

➢ Department Code

The blanks were replaced by the not applicable instead of DPH. The not applicable were not replaced by DPH.

```
subdata$department_code[subdata$department_code == ""] = "__NOT_APPLICABLE__"
```

- Union

1.3 Removal of unnecessary columns from the dataset

- Employee Identifier
- Department
- Job family code
- Union code
- Organization group code
- Job code

Normalization is not performed as per the changes told.

The dataset was being used without normalizing the numerical variables.

2. Resolving issues while loading the dataset

```
Console   Terminal ×   Jobs ×
C:/Users/raina/Downloads/527-Data Analytics/
 $ Employee.Identifier     : int  8540990 8540990 8540990 8577148 8603109 8547213 8544058 8504938 8559329 850
6973 ...
 $ Salaries                : num  674 674 674 124709 155489 ...
 $ Overtime                : num  0e+00 0e+00 0e+00 1e+05 0e+00 ...
 $ Other.Salaries          : num  5.76 5.76 5.76 5501.78 1500 ...
 $ Total.Salary            : num  680 680 680 230710 156989 ...
 $ Retirement              : num  131 131 131 23272 29240 ...
 $ Health.and.Dental       : num  0 0 0 14294 14308 ...
 $ Other.Benefits          : num  53.9 53.9 53.9 3934 11100.6 ...
 $ Total.Benefits          : num  185 185 185 55976 69327 ...
 $ Total.Compensation      : num  865 865 865 286686 226316 ...
> lm(data$Salaries ~ .,data=data)
Error: cannot allocate vector of size 18.2 Gb
> 
```

Following solutions were performed in order to reduce the size of the dataset :

2.1 Grouping was performed on the following columns

- Job
- Job Family
- Union

```
> levels(ndata$Job)[levels(ndata$Job)=="Planner 1"] = "Planners"
> levels(ndata$Job)[levels(ndata$Job)=="Planner 3"] = "Planners"
> levels(ndata$Job)[levels(ndata$Job)=="Planner V"] = "Planners"
> levels(ndata$Job)[levels(ndata$Job)=="Planner 2"] = "Planners"
> levels(ndata$Job)[levels(ndata$Job)=="Planner IV"] = "Planners"
> levels(ndata$Job)[levels(ndata$Job)=="Planner 5"] = "Planners"
> levels(ndata$Job)[levels(ndata$Job)=="Planner 4"] = "Planners"
```

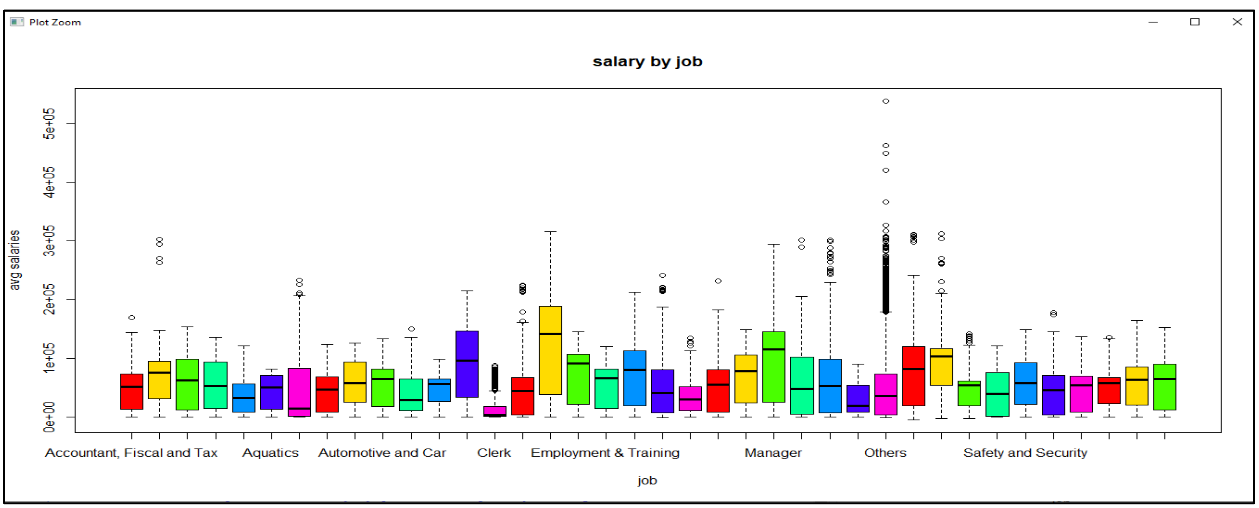
2.2 Sampling the dataset to 150000

```
> # sampling
> set.seed(5)
> sample_size=150000
> sdata = sample(1:nrow(data),sample_size,replace=F)
>
```

3 ANOVA and Hypothesis Testing for Job

ANOVA is used to compare average salaries of different employees based on job profiles.

- Boxplot for Salaries vs Job



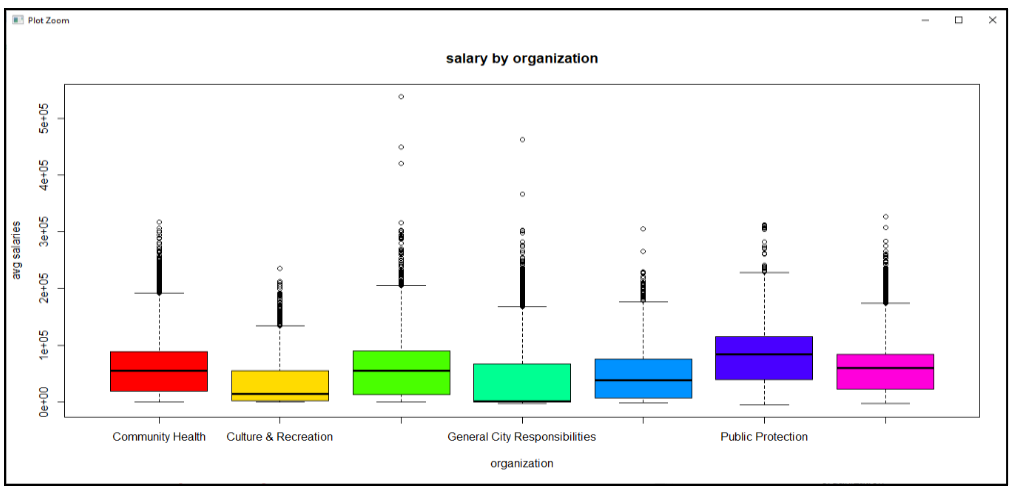
- ✓ Null Hypothesis : All the average salaries for jobs are equal
- ✓ Alternate hypothesis : Not all the average salaries for jobs are equal

```
> anova(anov)
Analysis of Variance Table

Response: y
              Df     Sum Sq    Mean Sq F value    Pr(>F)
j             37 3.3370e+13 9.0188e+11  434.53 < 2.2e-16 ***
Residuals 149962 3.1125e+14 2.0755e+09
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>
```

- ✓ At 95% confidence level, p-value is less than 0.05, we can reject null- hypothesis. Hence, the avg salaries are not equal for all jobs.

4 Mean comparison for Organization_group

- Boxplot for Salaries vs Organization_group
  Boxplot is used to compare average salaries of different employees based on the organization group.



5 Building Predictive Models

- ➢ Creating Dummy variables

```
year_type_calendar year_type_fiscal year2014 year2015 year2016 year2017 year2018 year2019
                 0                1        0        0        0        1        0        0
                 1                0        0        0        0        0        0        0
                 1                0        0        0        0        0        1        0
                 0                1        0        1        0        0        0        0
                 0                1        0        1        0        0        0        0
                 1                0        0        0        0        0        0        0
year2028 organization_group_community_health organization_group_culture_recreation
       0                                   0                                     0
       0                                   0                                     0
       0                                   0                                     0
       0                                   0                                     0
       0                                   1                                     0
       0                                   1                                     0
organization_group_general_city_responsibilities
                                               0
                                               0
                                               0
                                               0
                                               0
                                               0
organization_group_human_welfare_neighborhood_development organization_group_public_protection
                                                        0                                    0
                                                        0                                    1
                                                        0                                    0
                                                        0                                    0
                                                        0                                    0
                                                        0                                    0
```

- Hold Out Evaluation

```
> dim(subdata)
[1] 150000     22
> subdata=subdata[sample(nrow(subdata)),]
> select.data = sample(1:nrow(subdata),0.7*nrow(subdata))
> train.data=subdata[select.data,]
> test.data=subdata[-select.data,]
> dim(train.data)
[1] 105000     22
> dim(test.data)
[1] 45000     22
>
```

- Weak Co relations and Transformation

```
#transformation for overtime and other_salaries
t=compdata$overtime*compdata$overtime
cor(compdata$total_compensation,t, method = "pearson")
t=log(compdata$overtime)
cor(compdata$total_compensation,t, method = "pearson")
t=1/(compdata$overtime)
cor(compdata$total_compensation,t, method = "pearson")
compdata=select(compdata,-c(overtime))

t=compdata$other_salaries*compdata$other_salaries
cor(compdata$total_compensation,t, method = "pearson")
t=log(compdata$other_salaries)
cor(compdata$total_compensation,t, method = "pearson")
t=1/(compdata$other_salaries)
cor(compdata$total_compensation,t, method = "pearson")
compdata=select(compdata,-c(other_salaries))
```

As we can observe that the variables overtime and other salaries have a have a weak co relation even after performing transformation.

Thus, we would remove these variables.

Search Algorithm – Backward Elimination, Feature Selection Criteria – AIC

Model 1:

The first model was being built here.

Residual analysis was being performed on the model -

Following steps were performed –

- ✓ Checking the normality test
- ✓ Checking the variance
- ✓ Jarque-Bera Test
- ✓ Calculation of the RMSE

Check the multicollinearity using the VIF.

```
> cor(train.data$total_salary,train.data$retirement, method="pearson")
[1] 0.9448298
> cor(train.data$total_salary,train.data$salaries, method="pearson")
[1] 0.9681689
> cor(train.data$total_salary,train.data$organization_group_culture_recreation, method="pearson")
[1] -0.1523838
> cor(train.data$total_salary,train.data$department_code_lib, method="pearson")
[1] -0.04342369
> cor(train.data$total_salary,train.data$department_code_rec, method="pearson")
[1] -0.146247
> cor(train.data$total_salary,train.data$health_and_dental, method="pearson")
[1] 0.5831619
> cor(train.data$total_salary,train.data$other_benefits, method="pearson")
[1] 0.711008
> cor(train.data$total_salary,train.data$total_benefits, method="pearson")
[1] 0.900791
> cor(train.data$total_salary,train.data$department_code_fam, method="pearson")
[1] -0.03540061
> cor(train.data$organization_group_culture_recreation,train.data$total_benefits, method="pearson")
[1] -0.1372857
> cor(train.data$organization_group_culture_recreation,train.data$retirement, method="pearson")
[1] -0.1457864
> cor(train.data$organization_group_culture_recreation,train.data$salaries, method="pearson")
[1] -0.1441378
> cor(train.data$organization_group_culture_recreation,train.data$department_code_lib, method="pearson")
[1] 0.4834578
> cor(train.data$organization_group_culture_recreation,train.data$department_code_rec, method="pearson")
[1] 0.7805858
> cor(train.data$organization_group_culture_recreation,train.data$health_and_dental, method="pearson")
[1] -0.1159762
> cor(train.data$organization_group_culture_recreation,train.data$other_benefits, method="pearson")
[1] -0.09886695
> cor(train.data$organization_group_culture_recreation,train.data$department_code_fam, method="pearson")
[1] 0.2408308
> cor(train.data$salaries,train.data$total_benefits, method="pearson")
[1] 0.9218288
```

From the VIF calculated and after checking the co-relations we can give a conclusion that some columns can be removed having higher multi collinearity.

```
> train.data=select(train.data,-c(retirement))
> train.data=select(train.data,-c(total_salary))
> train.data=select(train.data,-c(total_benefits))
> train.data=select(train.data,-c(salaries))
>
> test.data=select(test.data,-c(retirement))
> test.data=select(test.data,-c(total_salary))
> test.data=select(test.data,-c(total_benefits))
> test.data=select(test.data,-c(salaries))
```

Model after resolving the multi-collinearity-

```
> #build model again after removing multicoll
> m5=lm(train.data$total_compensation ~ .,data=train.data)
> summary(m5)

Call:
lm(formula = train.data$total_compensation ~ ., data = train.data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.42101 -0.01761 -0.00060  0.01786  0.48592 
```

```
Step:  AIC=-657467.7
train.data$total_compensation ~ year_type_calendar + year2015 + 
    year2016 + year2017 + year2018 + year2019 + organization_group_community_health + 
    organization_group_culture_recreation + organization_group_general_city_responsibi
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.04366 on 104835 degrees of freedom
Multiple R-squared:  0.8373,	Adjusted R-squared:  0.837 
F-statistic:  3289 on 164 and 104835 DF,  p-value: < 2.2e-16
```

After resolving the multi collinearity we can observe that the adjusted R2 changes to 0.837.

Residual analysis was being performed again on the new data after removing the variables overtime and other salaries.

Following steps were performed –

- ✓ Checking the normality test

- ✓ Checking the variance
- ✓ Jarque Bera Test
- ✓ Calculation of the RMSE

7 Predicting Total Compensation

Search Algorithm –Forward Elimination, Feature Selection Criteria – AIC

Model 1 –

The first model was being built here.

Residual analysis was being performed again on the data again using forward elimination with AIC.

Following steps were performed –

- ✓ Checking the normality test
- ✓ Checking the variance
- ✓ Jarque Bera Test
- ✓ Calculation of the RMSE

After this we checked the multi collinearity (certain columns with VIF more than 4 were removed after rechecking the correlations.)

Model 2:

The second model was built here.

Residual analysis was being performed again on the model again using forward elimination with AIC.

Following steps were performed:

- ✓ Checking the normality test

- ✓ Checking the variance
- ✓ Jarque Bera Test
- ✓ Calculation of the RMSE

Thus, after building the forward and backward models for total compensation, a comparison was being done between both of them.

The model with the lowest RMSE was chosen as a better model amongst both of them.

8 Predicting Salary

Search Algorithm –Backward Elimination, Feature Selection Criteria – AIC

Model 1:

The first model was being built here.

Residual analysis was being performed again on the data again using forward elimination with AIC.

Following steps were performed –

- ✓ Checking the normality test
- ✓ Checking the variance
- ✓ Jarque Bera Test
- ✓ Calculation of the RMSE

After this we checked the multi collinearity (certain columns with VIF more than 4 were removed after rechecking the correlations.)

VIF calculation:

2^{ND} MODEL:

```
> cor(train.data$total_salary,train.data$retirement, method="pearson")
[1] 0.9448298
> cor(train.data$total_salary,train.data$salaries, method="pearson")
[1] 0.9681689
> cor(train.data$total_salary,train.data$organization_group_culture_recreation, method="pearson")
[1] -0.1523838
> cor(train.data$total_salary,train.data$department_code_lib, method="pearson")
[1] -0.04342369
> cor(train.data$total_salary,train.data$department_code_rec, method="pearson")
[1] -0.146247
> cor(train.data$total_salary,train.data$health_and_dental, method="pearson")
[1] 0.5831619
> cor(train.data$total_salary,train.data$other_benefits, method="pearson")
[1] 0.711008
> cor(train.data$total_salary,train.data$total_benefits, method="pearson")
[1] 0.900791
> cor(train.data$total_salary,train.data$department_code_fam, method="pearson")
[1] -0.03540061
> cor(train.data$organization_group_culture_recreation,train.data$total_benefits, method="pearson")
[1] -0.1372857
> cor(train.data$organization_group_culture_recreation,train.data$retirement, method="pearson")
[1] -0.1457864
> cor(train.data$organization_group_culture_recreation,train.data$salaries, method="pearson")
[1] -0.1441378
> cor(train.data$organization_group_culture_recreation,train.data$department_code_lib, method="pearson")
[1] 0.4834578
> cor(train.data$organization_group_culture_recreation,train.data$department_code_rec, method="pearson")
[1] 0.7805858
> cor(train.data$organization_group_culture_recreation,train.data$health_and_dental, method="pearson")
[1] -0.1159762
> cor(train.data$organization_group_culture_recreation,train.data$other_benefits, method="pearson")
[1] -0.09886695
> cor(train.data$organization_group_culture_recreation,train.data$department_code_fam, method="pearson")
[1] 0.2408308
> cor(train.data$salaries,train.data$total_benefits, method="pearson")
[1] 0.9218288
```

From the VIF calculated and after checking the co-relations we can give a conclusion that some columns can be removed having higher multi co linearity.

```
> train_s.data=train.data
> test_s.data=test.data
> train_s.data=select(train_s.data,-c(retirement))
> train_s.data=select(train_s.data,-c(total_salary))
> train_s.data=select(train_s.data,-c(total_benefits))
> train_s.data=select(train_s.data,-c(union_fighters))
Error in map_lgl(.x, .p, ...) : object 'union_fighters' not found
> train_s.data=select(train_s.data,-c(union_firefighters))
> test_s.data=test.data
> test_s.data=select(test_s.data,-c(retirement))
> test_s.data=select(test_s.data,-c(total_salary))
> test_s.data=select(test_s.data,-c(total_benefits))
> test_s.data=select(test_s.data,-c(union_firefighters))
```

Model after resolving the multi-collinearity-

```
Console  Terminal ×  Jobs ×
C:/Users/raina/Downloads/527-Data Analytics/
job_clerk                                    1.431e-03  9.060e-04   1.580 0.114147
job_court_legal_and_legislative             -9.304e-03  4.594e-04 -20.250  < 2e-16 ***
job_director_and_chief_staff                 1.024e-02  1.078e-03   9.507  < 2e-16 ***
job_electrical_and_electronics_engineer      1.047e-03  6.116e-04   1.712 0.086948 .
job_engineer                                 2.775e-03  4.025e-04   6.892 5.51e-12 ***
job_food_and_purchaser                      -3.305e-03  6.350e-04  -5.204 1.95e-07 ***
job_hospital_and_emergency                  -1.828e-03  8.149e-04  -2.243 0.024912 *
job_industrial_and_materials                 7.737e-03  1.202e-03   6.436 1.23e-10 ***
job_manager                                 -2.244e-04  6.516e-04  -0.344 0.730602
job_mayoral_staff                           -1.001e-02  1.670e-03  -5.996 2.03e-09 ***
job_medical_health_and_diagnostic_expert     3.809e-03  3.928e-04   9.699  < 2e-16 ***
job_museum_and_art_supervisor               -3.533e-03  1.094e-03  -3.230 0.001238 **
job_others                                   8.463e-06  2.510e-04   0.034 0.973103
job_police_and_investigation                -1.135e-03  4.440e-04  -2.556 0.010599 *
job_power_and_fire_executive                -6.976e-03  8.501e-04  -8.207 2.30e-16 ***
job_public_relations_and_child_support      -1.732e-03  4.586e-04  -3.778 0.000158 ***
job_safety_and_security                     -8.729e-03  1.033e-03  -8.451  < 2e-16 ***
job_technician_and_tech_expert              -1.870e-03  5.527e-04  -3.382 0.000719 ***
job_transit_and_transport                   -7.344e-03  4.399e-04 -16.696  < 2e-16 ***
job_utility_and_janitorial_services          1.487e-03  4.661e-04   3.191 0.001420 **
job_water_services_and_welfare               3.661e-03  7.240e-04   5.056 4.29e-07 ***
other_benefits                               6.331e-02  1.060e-03  59.734  < 2e-16 ***
total_compensation                           7.529e-01  1.037e-03 725.900  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01651 on 104835 degrees of freedom
Multiple R-squared:  0.9659,	Adjusted R-squared:  0.9658
F-statistic: 1.809e+04 on 164 and 104835 DF,  p-value: < 2.2e-16

>
```

Model 2 –

The second model was built here

Residual analysis was being performed again on the data using forward elimination with AIC.

Following steps were performed:

- ✓ Checking the normality test
- ✓ Checking the variance
- ✓ Jarque Bera Test
- ✓ Calculation of the RMSE

9 Predicting Salary

Search Algorithm –Forward Elimination, Feature Selection Criteria – AIC

Model 1:

The first model was being built here.

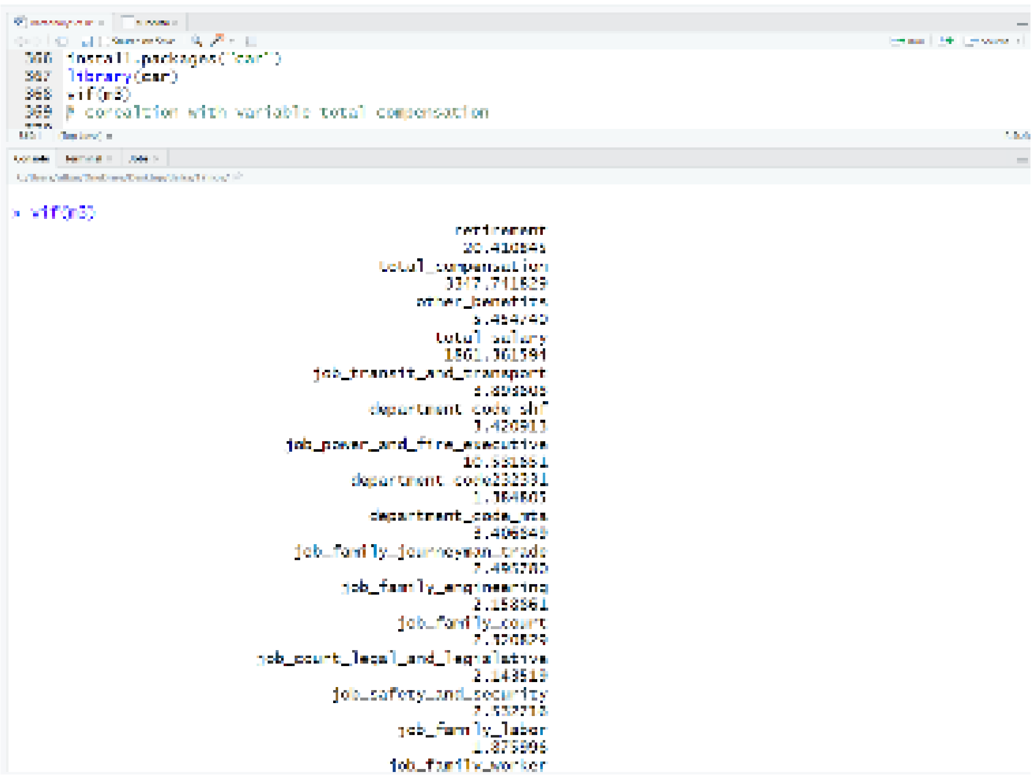
Residual analysis was being performed again on the data again using forward elimination with AIC.

Following steps were performed:

- ✓ Checking the normality test
- ✓ Checking the variance
- ✓ Jarque Bera Test
- ✓ Calculation of the RMSE

After this we checked the multi collinearity (certain columns with VIF more than 4 were removed after rechecking the correlations.)

VIF calculation



Model 2:

The second model was built here

Residual analysis was being performed again on the data using forward elimination with AIC.

Following steps were performed:

- ✓ Checking the normality test
- ✓ Checking the variance
- ✓ Jarque Bera Test
- ✓ Calculation of the RMSE

Thus, after building the forward and backward models for total compensation, a comparison was being done between both.

The model with the lowest RMSE was chosen as a better model amongst both.

5.2. Evaluations and Results

Given a same problem, you may have several solutions or build several models

Evaluate your solutions based on selected metrics and compare them

To evaluate which model is the best, we need to test all the models against the test data.

For total compensation we have built two models including the forward elimination and the backward elimination.

A conclusion would be given based on comparing their RMSE values.

5.2.1 Search Algorithm – Backward Elimination, Feature Selection Criteria – AIC

Model 1:

```
#total_compensation
m4=lm(train.data$total_compensation ~ .,data=train.data)
summary(m4)

m3=step(m4, direction = "backward", trace = T)

summary(m3)

# residual analysis

res=rstandard(m3)
```

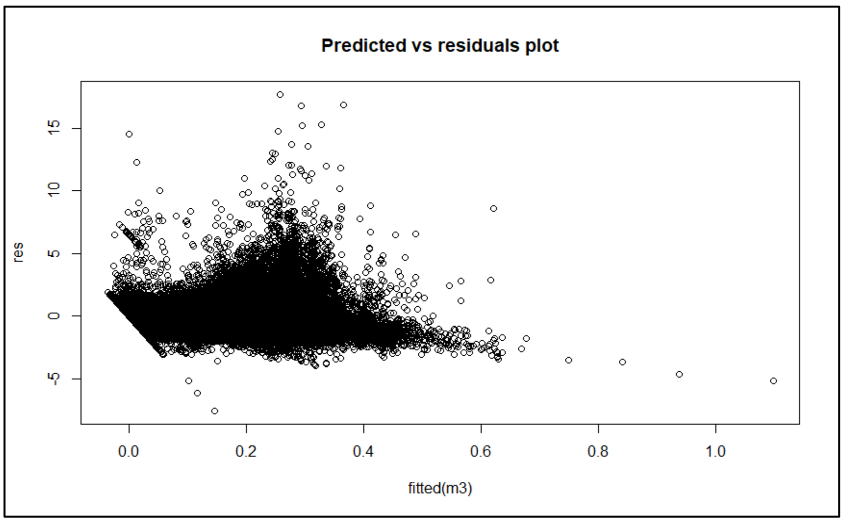
AIC, ADJUSTED R2 AND RMSE VALUES –

```
Step:  AIC=-1320830
train.data$total_compensation ~ year_type_calendar + year2015 +
    year2016 + year2017 + year2019 + organization_group_community_health +
    organization_group_culture_recreation + organization_group_human_welfare_neighborhood_
```

```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.001855 on 104952 degrees of freedom
Multiple R-squared:  0.9997,    Adjusted R-squared:  0.9997
F-statistic: 7.592e+06 on 47 and 104952 DF,  p-value: < 2.2e-16
```

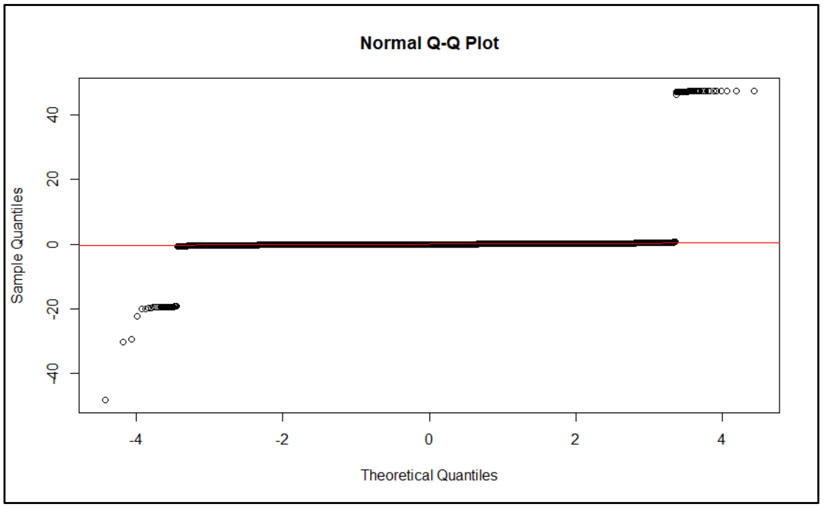
Residual Analysis:

Check the variance:



Here we can observe that the variance is constant as the spread is constant and points are randomly scattered. There is no pattern here.

Normality Test by QQ plot:



Here we can observe that the residual is normally distributed as most of the points are lying near the straight line.

Jarque-Bera Test:

```
> jarque.bera.test(res)

	Jarque Bera Test

data:  res
X-squared = 1.7737e+10, df = 2, p-value < 2.2e-16
```

From the above test, we can conclude that p-value is less than 0.05, so the model follows normality.

In order to check the multicollinearity using the VIF.

After checking the multicollinearity and removing the variables having correlation greater than +- 0.9 (around +- 1), we will built a new model.

```
> cor(train.data$total_salary,train.data$retirement, method="pearson")
[1] 0.9448298
> cor(train.data$total_salary,train.data$salaries, method="pearson")
[1] 0.9681689
> cor(train.data$total_salary,train.data$organization_group_culture_recreation, method="pearson")
[1] -0.1523838
> cor(train.data$total_salary,train.data$department_code_lib, method="pearson")
[1] -0.04342369
> cor(train.data$total_salary,train.data$department_code_rec, method="pearson")
[1] -0.146247
> cor(train.data$total_salary,train.data$health_and_dental, method="pearson")
[1] 0.5831619
> cor(train.data$total_salary,train.data$other_benefits, method="pearson")
[1] 0.711008
> cor(train.data$total_salary,train.data$total_benefits, method="pearson")
[1] 0.900791
> cor(train.data$total_salary,train.data$department_code_fam, method="pearson")
[1] -0.03540061
> cor(train.data$organization_group_culture_recreation,train.data$total_benefits, method="pearson")
[1] -0.1372857
> cor(train.data$organization_group_culture_recreation,train.data$retirement, method="pearson")
[1] -0.1457864
> cor(train.data$organization_group_culture_recreation,train.data$salaries, method="pearson")
[1] -0.1441378
> cor(train.data$organization_group_culture_recreation,train.data$department_code_lib, method="pearson")
[1] 0.4834578
> cor(train.data$organization_group_culture_recreation,train.data$department_code_rec, method="pearson")
[1] 0.7805858
> cor(train.data$organization_group_culture_recreation,train.data$health_and_dental, method="pearson")
[1] -0.1159762
> cor(train.data$organization_group_culture_recreation,train.data$other_benefits, method="pearson")
[1] -0.09886695
> cor(train.data$organization_group_culture_recreation,train.data$department_code_fam, method="pearson")
[1] 0.2408308
> cor(train.data$salaries,train.data$total_benefits, method="pearson")
[1] 0.9218288
```

From the VIF calculated and after checking the correlations we can give a conclusion that some columns can be removed having higher multi collinearity.

```
> train.data=select(train.data,-c(retirement))
> train.data=select(train.data,-c(total_salary))
> train.data=select(train.data,-c(total_benefits))
> train.data=select(train.data,-c(salaries))
>
> test.data=select(test.data,-c(retirement))
> test.data=select(test.data,-c(total_salary))
> test.data=select(test.data,-c(total_benefits))
> test.data=select(test.data,-c(salaries))
```

Model after resolving the multi-collinearity-

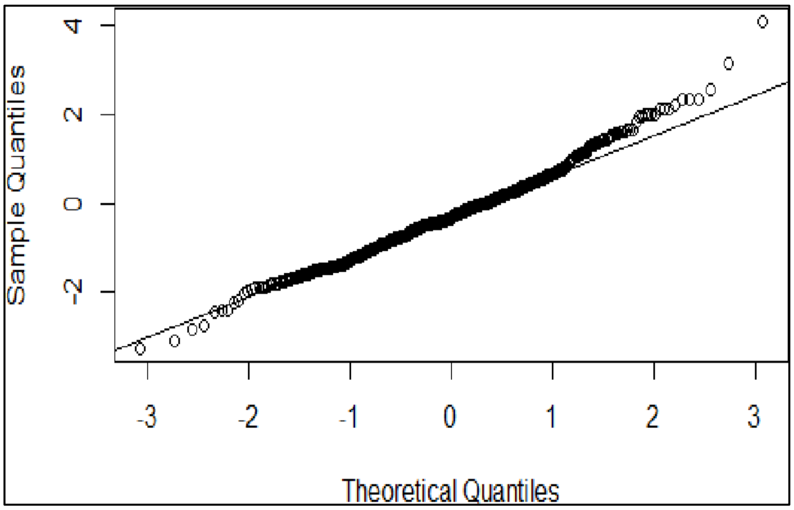
```
> #build model again after removing multicoll
> m5=lm(train.data$total_compensation ~ .,data=train.data)
> summary(m5)

Call:
lm(formula = train.data$total_compensation ~ ., data = train.data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.42101 -0.01761 -0.00060  0.01786  0.48592
```

```
Step:  AIC=-657467.7
train.data$total_compensation ~ year_type_calendar + year2015 + 
    year2016 + year2017 + year2018 + year2019 + organization_group_community_health + 
    organization_group_culture_recreation + organization_group_general_city_responsibi
```

Model 2:

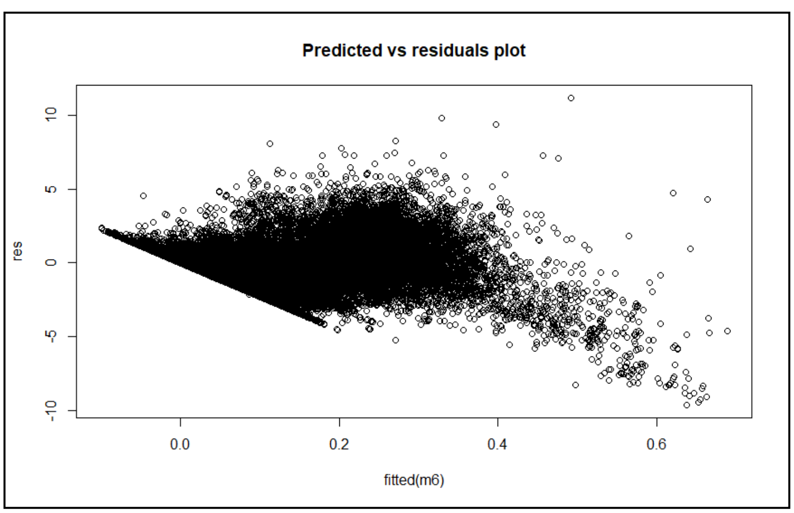
```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.04366 on 104835 degrees of freedom
Multiple R-squared:  0.8373,    Adjusted R-squared:  0.837
F-statistic:  3289 on 164 and 104835 DF,  p-value: < 2.2e-16
```

Normality Test



Here we can observe that the residual is normally distributed as most of the points are lying near the straight line.

Residual Plot



Here we can observe that the variance is constant as the spread is constant and points are randomly scattered. There is no pattern here.

- Jarque-Bera Test

```
> jarque.bera.test(res)

	Jarque Bera Test

data:  res
X-squared = 235949, df = 2, p-value < 2.2e-16

>
```

From the above test, we can conclude that p-value is less than 0.05, so the model follows normality.

- RMSE

```
[163] "job_utility_and_janitorial_services"
[164] "job_water_services_and_welfare"
[165] "health_and_dental"
[166] "other_benefits"
[167] "total_compensation"
> y1=predict.glm(m6,test.data)
> y=test.data[,167]
> rmse_2 = sqrt((y-y1)%*%(y-y1)/nrow(test.data))
> rmse_2
           [,1]
[1,] 0.04321609
>
```

5.2.1 Search Algorithm – Forward Selection -Feature Selection Criteria – AIC

Creating Model

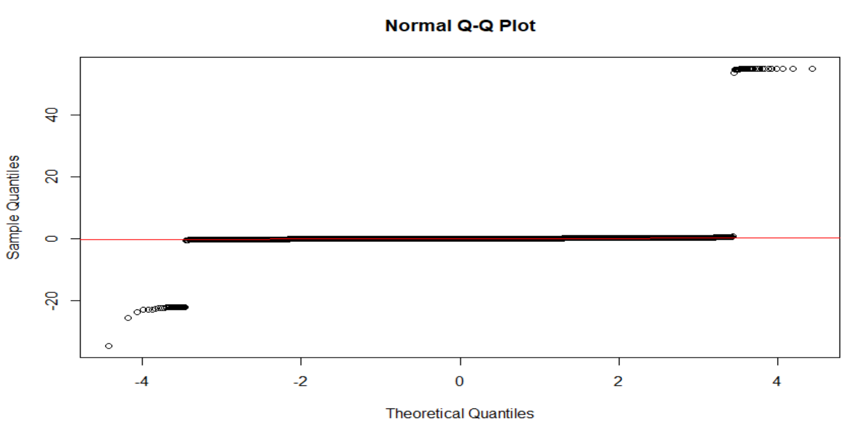
Model 1:

Calculating AIC, ADJUSTED R2 AND RMSE VALUES

```
Residuals:
      Min        1Q    Median        3Q       Max
-0.055893 -0.000086 -0.000019  0.000050  0.088444

Coefficients:
                                                Estimate Std. Error  t value Pr(>|t|)
(Intercept)                                    8.857e-05  1.654e-05    5.354 8.61e-08 ***
total_benefits                                 2.239e-01  2.043e-04 1095.608  < 2e-16 ***
total_salary                                   7.790e-01  2.188e-04 3560.000  < 2e-16 ***
health_and_dental                             -7.291e-04  4.888e-05  -14.915  < 2e-16 ***
organization_group_public_protection           7.989e-05  2.047e-05    3.903 9.52e-05 ***
salaries                                      -1.346e-03  3.056e-04   -4.405 1.06e-05 ***
year2019                                      -1.447e-04  1.804e-05   -8.019 1.08e-15 ***
retirement                                    -1.787e-03  2.461e-04   -7.260 3.90e-13 ***
department_code_cpc                            3.238e-04  6.881e-05    4.706 2.54e-06 ***
other_benefits                                -4.125e-04  1.107e-04   -3.725 0.000196 ***
year_type_calendar                            -5.885e-05  1.120e-05   -5.254 1.49e-07 ***
year2015                                       7.274e-05  1.907e-05    3.813 0.000137 ***
job_engineer                                   1.172e-04  2.924e-05    4.007 6.16e-05 ***
union_sheriffs_managers_and_supervisors_association 4.414e-04  1.021e-04    4.325 1.53e-05 ***
year2017                                       4.737e-05  1.489e-05    3.182 0.001465 **
job_family_nursing                             7.800e-05  2.171e-05    3.592 0.000328 ***
job_court_legal_and_legislative               -8.893e-05  2.978e-05   -2.987 0.002822 **
department_code_dpw                            6.940e-05  2.857e-05    2.429 0.015149 *
job_family_budget_admn_stats_analysis          8.795e-05  2.933e-05    2.999 0.002710 **
job_family_police_services                     2.087e-04  3.471e-05    6.013 1.82e-09 ***
job_power_and_fire_executive                   1.205e-04  3.089e-05    3.900 9.62e-05 ***
year2016                                       3.551e-05  1.839e-05    1.931 0.053522 .
job_police_and_investigation                  -1.240e-04  3.191e-05   -3.885 0.000103 ***
job_apprentice_and_media                       2.383e-04  9.094e-05    2.621 0.008769 **
department_code_cat                            1.267e-04  6.905e-05    1.835 0.066574 .
job_family_clerical_secretarial_steno         -5.294e-05  2.528e-05   -2.094 0.036268 *
organization_group_general_city_responsibilities -2.797e-06  1.911e-05   -0.146 0.883598
union_municipals                               1.175e-04  4.173e-05    2.815 0.004881 **
job_family_management_and_development_agency  -9.459e-05  4.442e-05   -2.129 0.033245 *
department_code_dph                           -9.787e-05  2.744e-05   -3.567 0.000361 ***
organization_group_community_health            9.752e-05  3.283e-05    2.971 0.002970 **
year2014                                      -3.225e-05  1.923e-05   -1.677 0.093508 .
organization_group_culture_recreation         -3.271e-05  2.062e-05   -1.586 0.112668
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.001613 on 104967 degrees of freedom
Multiple R-squared:  0.9998,    Adjusted R-squared:  0.9998
F-statistic: 1.472e+07 on 32 and 104967 DF,  p-value: < 2.2e-16
```

Residual Analysis-

Normality Test-



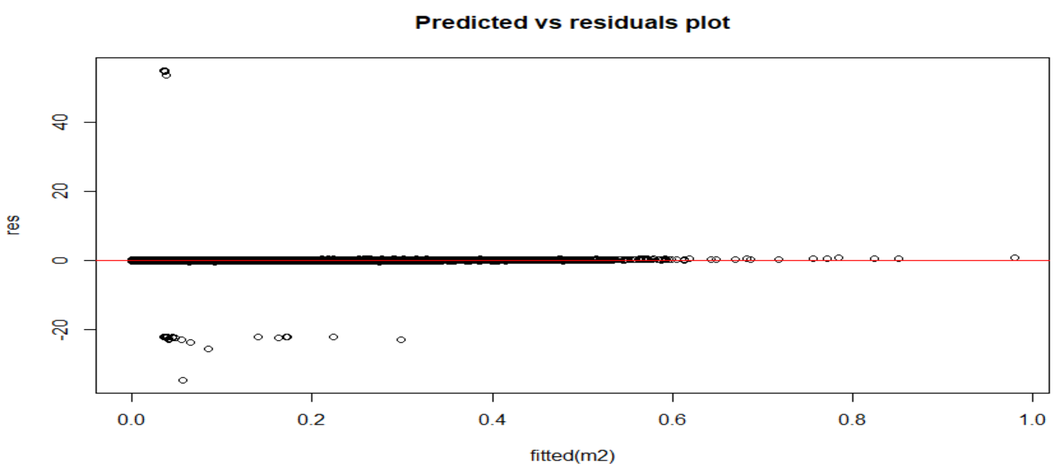
Here we can observe that the residual is normally distributed as most of the points are lying near the straight line.

Jarque Bera Test-

```
> jarque.bera.test(res)

        Jarque Bera Test

data:  res
X-squared = 3.0136e+10, df = 2, p-value < 2.2e-16

>
```

From the above test, we can conclude that p-value is less than 0.05, so the model follows normality.

Check the variance:



Here we can observe that the variance is constant as the spread is constant and points are randomly scattered. There is no pattern here.

In order to calculate the value of RMSE –

```
> y1=predict.glm(m2,test.data_c)
> y=test.data_c[,171]
> rmse = sqrt((y-y1)%*%(y-y1)/nrow(test.data_c))
> rmse
            [,1]
[1,] 0.002159817
>
```

In order to check the multicollinearity using the VIF.

After checking the multicollinearity and removing the variables having correlation greater than +- 0.9 (around +- 1), we will built a new model.

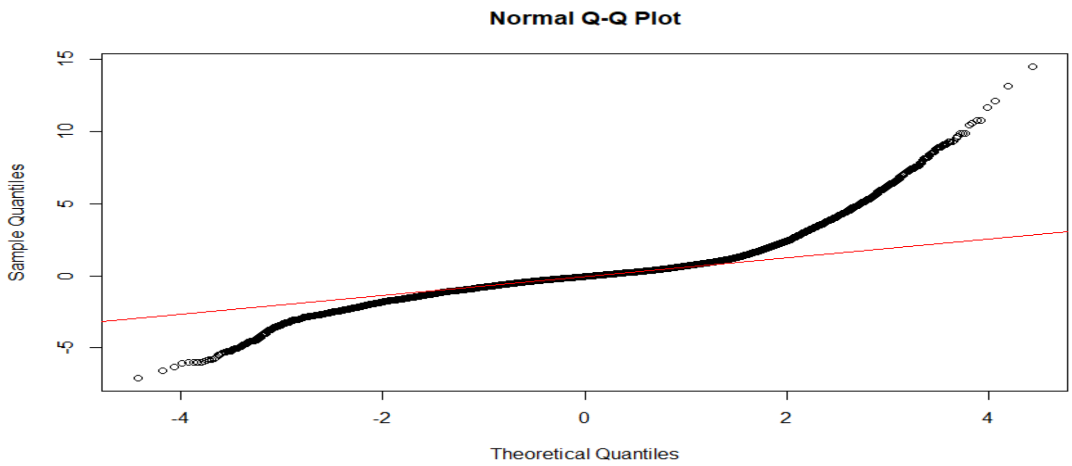
From the VIF calculated and after checking the co-relations we can give a conclusion that some columns can be removed having higher multi collinearity.

Model after resolving the multi-collinearity:

Model 2:

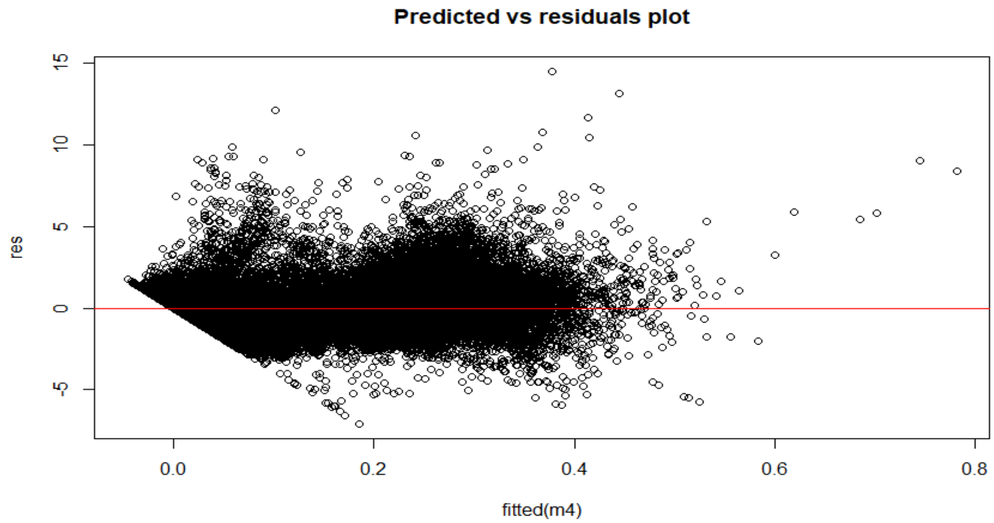
```
job_food_and_purchaser                        -9.366e-04  9.934e-04   -0.943 0.345745
job_hospital_and_emergency                     1.089e-02  1.290e-03    8.441  < 2e-16 ***
job_industrial_and_materials                   3.824e-03  1.897e-03    2.016 0.043842 *
job_manager                                   -4.655e-03  1.030e-03   -4.518 6.24e-06 ***
job_mayoral_staff                             -1.503e-02  2.632e-03   -5.712 1.12e-08 ***
job_medical_health_and_diagnostic_expert       7.088e-03  6.225e-04   11.387  < 2e-16 ***
job_museum_and_art_supervisor                  5.746e-03  1.692e-03    3.396 0.000684 ***
job_others                                     2.291e-04  3.953e-04    0.580 0.562236
job_police_and_investigation                  -1.589e-02  6.992e-04  -22.732  < 2e-16 ***
job_power_and_fire_executive                   2.038e-02  1.340e-03   15.203  < 2e-16 ***
job_public_relations_and_child_support        -2.516e-03  7.270e-04   -3.460 0.000540 ***
job_safety_and_security                        1.654e-02  1.627e-03   10.164  < 2e-16 ***
job_technician_and_tech_expert                -1.501e-03  8.659e-04   -1.734 0.082996 .
job_transit_and_transport                      1.576e-03  6.980e-04    2.258 0.023965 *
job_utility_and_janitorial_services            7.361e-04  7.386e-04    0.997 0.318920
job_water_services_and_welfare                 1.239e-03  1.161e-03    1.068 0.285634
health_and_dental                             -6.194e-02  7.562e-04  -81.912  < 2e-16 ***
other_benefits                                 5.573e-02  1.808e-03   30.824  < 2e-16 ***
total_benefits                                 7.799e-01  1.801e-03  433.023  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.02605 on 104835 degrees of freedom
Multiple R-squared:  0.942,	Adjusted R-squared:  0.9419
F-statistic: 1.038e+04 on 164 and 104835 DF,  p-value: < 2.2e-16
```

Normality Test



Here we can observe that the residual is normally distributed as most of the points are lying near the straight line.

Residual Plot



Here we can observe that the variance is constant as the spread is constant and points are randomly scattered. There is no pattern here.

- JarqueBera Test

```
> jarque.bera.test(res)

        Jarque Bera Test

data:  res
X-squared = 463277, df = 2, p-value < 2.2e-16
```

From the above test, we can conclude that p-value is less than 0.05, so the model follows normality.

- RMSE

```
[164] "job_water_services_and_welfare"
[165] "health_and_dental"
[166] "other_benefits"
[167] "total_benefits"
[168] "total_compensation"
> y1=predict.glm(m4,test.data)
> y=test.data[,168]
> rmse_1 = sqrt((y-y1)%*%(y-y1)/nrow(test.data))
> rmse_1
           [,1]
[1,] 0.02598934
```

Here, we can observe that the variance is constant as the spread is constant and points are not scattered.

A final conclusion can be given based on the comparison of the two models built on total compensation including the backward and the forward .

Comparing the backward and forward models for total compensation -

Measures	Backward Elimination	Forward Selection
ADJ R2	0.837	0.9419
RMSE	0.0431	0.0259

As we can observe that the RMSE for the forward selection is less as compared to the backward one. Thus, we would prefer the forward model for total compensation instead of the backward one.

10 Salary using backward elimination

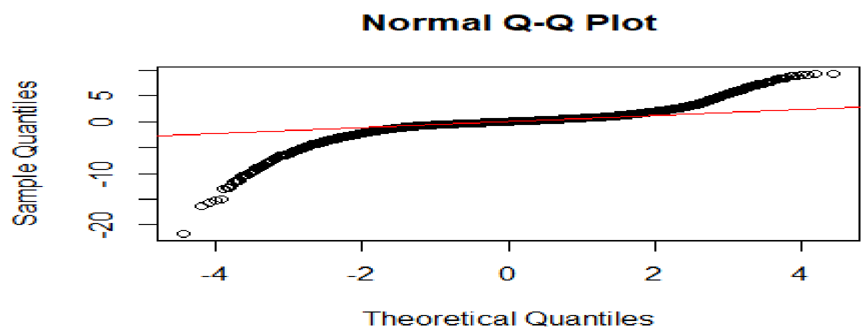
Model 1 –

Step 2 – Calculating AIC , ADJUSTED R2 AND RMSE VALUES –

```
Source
Console   Terminal ×   Jobs ×
C:/Users/raina/Downloads/527-Data Analytics/
job_analyst                                              1.23e-05 ***
job_animal_control_and_environmentalist                  0.000137 ***
job_asphalt_and_asr_officer                              0.054949 .
job_automotive_and_car                                   1.40e-11 ***
job_chemists_and_pharmacists                              < 2e-16 ***
job_court_legal_and_legislative                           < 2e-16 ***
job_director_and_chief_staff                              < 2e-16 ***
job_electrical_and_electronics_engineer                  0.050753 .
job_engineer                                             0.140970
job_food_and_purchaser                                   1.58e-14 ***
job_hospital_and_emergency                               1.18e-07 ***
job_industrial_and_materials                             1.06e-08 ***
job_mayoral_staff                                        1.18e-15 ***
job_medical_health_and_diagnostic_expert                 5.94e-11 ***
job_museum_and_art_supervisor                            0.000172 ***
job_others                                               7.42e-05 ***
job_police_and_investigation                             3.76e-07 ***
job_power_and_fire_executive                              < 2e-16 ***
job_public_relations_and_child_support                   0.005312 **
job_safety_and_security                                   < 2e-16 ***
job_technician_and_tech_expert                           5.15e-05 ***
job_transit_and_transport                                 < 2e-16 ***
job_utility_and_janitorial_services                      0.018769 *
job_water_services_and_welfare                           0.001527 **
total_salary                                              < 2e-16 ***
retirement                                                < 2e-16 ***
other_benefits                                            < 2e-16 ***
total_benefits                                            < 2e-16 ***
total_compensation                                       1.08e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01507 on 104883 degrees of freedom
Multiple R-squared:  0.9715,	Adjusted R-squared:  0.9715 
F-statistic: 3.083e+04 on 116 and 104883 DF,  p-value: < 2.2e-16

> |
```

Residual Analysis-

Normality Test-



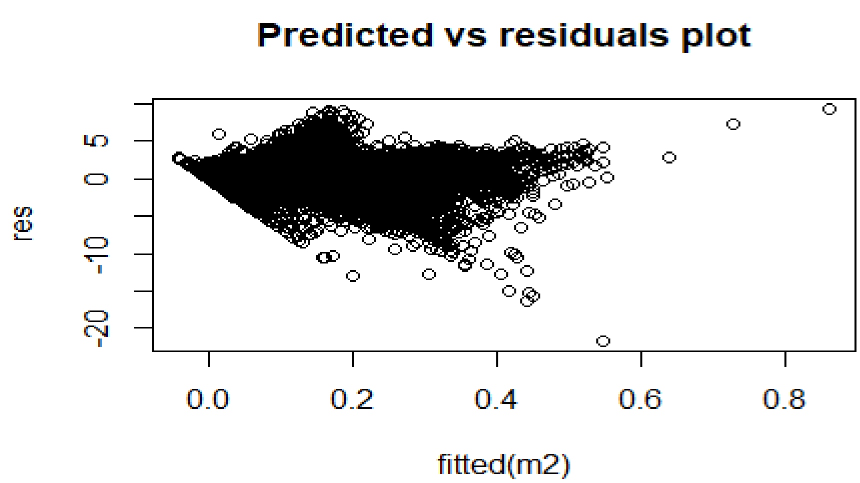
Here we can observe that the residual is normally distributed as most of the points are lying near the straight line.

Jarque Bera Test-

```
> jarque.bera.test(res)

        Jarque Bera Test

data:  res
X-squared = 1278454, df = 2, p-value < 2.2e-16
```

From the above test, we can conclude that p-value is less than 0.05, so the model follows normality.

In order to check the variance, we draw residual plots –



Here we can observe that the variance is constant as the spread is constant and points are randomly scattered. There is no pattern here.

Calculate the value of RMSE –

```
[1] 105000    170
> dim(test.data)
[1] 45000   170
> y1=predict.glm(m2,test.data)
> y=test.data[,165]
> rmse_1 = sqrt((y-y1)%*%(y-y1)/nrow(test.data))
> rmse_1
           [,1]
[1,] 0.01514243
> |
```

Check the multicollinearity using the VIF.

After checking the multicollinearity and removing the variables having co linearity greater than 0.09 a new model was being built.

From the VIF calculated and after checking the co-relations we can give a conclusion that some columns can be removed having higher multi co l

```
> train_s.data=train.data
> test_s.data=test.data
> train_s.data=select(train_s.data,-c(retirement))
> train_s.data=select(train_s.data,-c(total_salary))
> train_s.data=select(train_s.data,-c(total_benefits))
> train_s.data=select(train_s.data,-c(union_fighters))
Error in map_lgl(.x, .p, ...) : object 'union_fighters' not found
> train_s.data=select(train_s.data,-c(union_firefighters))
> test_s.data=test.data
> test_s.data=select(test_s.data,-c(retirement))
> test_s.data=select(test_s.data,-c(total_salary))
> test_s.data=select(test_s.data,-c(total_benefits))
> test_s.data=select(test_s.data,-c(union_firefighters))
```

Model after resolving the multi-collinearity-

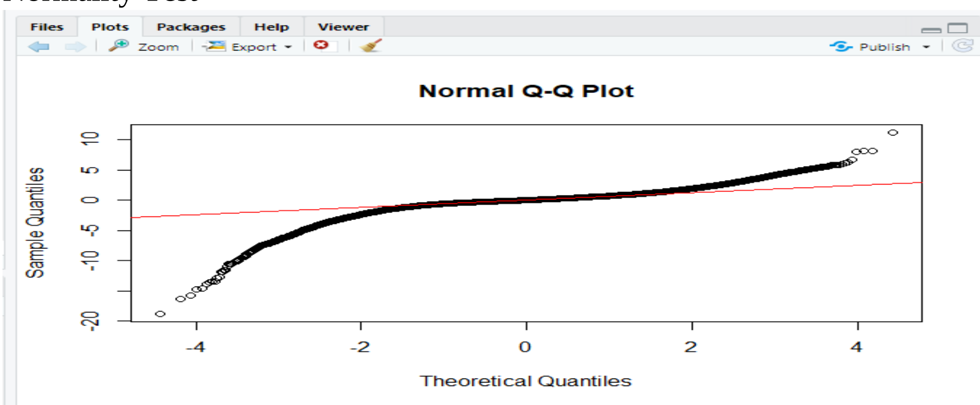
Model 2 -

Console Terminal Jobs

C:/Users/raina/Downloads/527-Data Analytics/

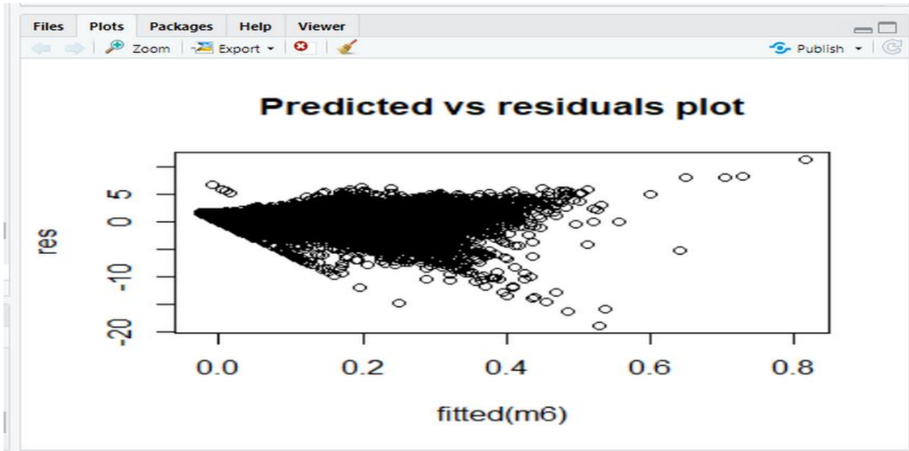
```
job_clerk                                    1.431e-03  9.060e-04   1.580 0.114147    
job_court_legal_and_legislative             -9.304e-03  4.594e-04 -20.250  < 2e-16 ***
job_director_and_chief_staff                 1.024e-02  1.078e-03   9.507  < 2e-16 ***
job_electrical_and_electronics_engineer      1.047e-03  6.116e-04   1.712 0.086948 .  
job_engineer                                 2.775e-03  4.025e-04   6.892 5.51e-12 ***
job_food_and_purchaser                      -3.305e-03  6.350e-04  -5.204 1.95e-07 ***
job_hospital_and_emergency                  -1.828e-03  8.149e-04  -2.243 0.024912 *  
job_industrial_and_materials                 7.737e-03  1.202e-03   6.436 1.23e-10 ***
job_manager                                 -2.244e-04  6.516e-04  -0.344 0.730602    
job_mayoral_staff                           -1.001e-02  1.670e-03  -5.996 2.03e-09 ***
job_medical_health_and_diagnostic_expert     3.809e-03  3.928e-04   9.699  < 2e-16 ***
job_museum_and_art_supervisor               -3.533e-03  1.094e-03  -3.230 0.001238 ** 
job_others                                   8.463e-06  2.510e-04   0.034 0.973103    
job_police_and_investigation                -1.135e-03  4.440e-04  -2.556 0.010599 *  
job_power_and_fire_executive                -6.976e-03  8.501e-04  -8.207 2.30e-16 ***
job_public_relations_and_child_support      -1.732e-03  4.586e-04  -3.778 0.000158 ***
job_safety_and_security                     -8.729e-03  1.033e-03  -8.451  < 2e-16 ***
job_technician_and_tech_expert              -1.870e-03  5.527e-04  -3.382 0.000719 ***
job_transit_and_transport                   -7.344e-03  4.399e-04 -16.696  < 2e-16 ***
job_utility_and_janitorial_services          1.487e-03  4.661e-04   3.191 0.001420 ** 
job_water_services_and_welfare               3.661e-03  7.240e-04   5.056 4.29e-07 ***
other_benefits                               6.331e-02  1.060e-03  59.734  < 2e-16 ***
total_compensation                           7.529e-01  1.037e-03 725.900  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01651 on 104835 degrees of freedom
Multiple R-squared:  0.9659,	Adjusted R-squared:  0.9658 
F-statistic: 1.809e+04 on 164 and 104835 DF,  p-value: < 2.2e-16

> 
```

Normality Test



Here we can observe that the residual is normally distributed as most of the points are lying near the straight line.

Residual Plot



Here we can observe that the variance is constant as the spread is constant and points are randomly scattered. There is no pattern here.

- JarqueBera Test

```
> jarque.bera.test(res)

        Jarque Bera Test

data:  res
X-squared = 1135360, df = 2, p-value < 2.2e-16

>
```

From the above test, we can conclude that p-value is less than 0.05, so the model follows normality.

- RMSE

```
> y1=predict.glm(m6,test_s.data)
> y=test_s.data[,164]
> rmse_1 = sqrt((y-y1)%*%(y-y1)/nrow(test_s.data))
> rmse_1
           [,1]
[1,] 0.01673998
>
```

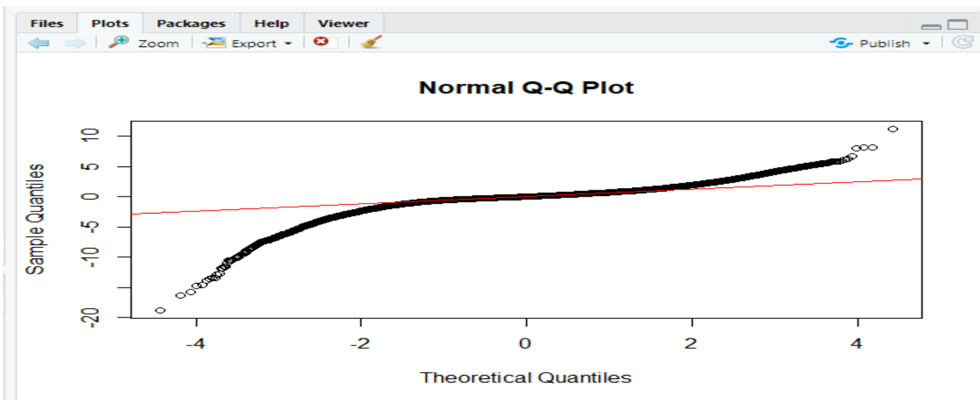
A final conclusion can be given based on the comparison of the two models built on total compensation including the backward and the forward model.

Model 2

Model after resolving the multi-collinearity-

```
Console   Terminal ×   Jobs ×
C:/Users/raina/Downloads/527-Data Analytics/
job_clerk                                        1.431e-03  9.060e-04   1.580 0.114147
job_court_legal_and_legislative                 -9.304e-03  4.594e-04 -20.250  < 2e-16 ***
job_director_and_chief_staff                     1.024e-02  1.078e-03   9.507  < 2e-16 ***
job_electrical_and_electronics_engineer          1.047e-03  6.116e-04   1.712 0.086948 .
job_engineer                                     2.775e-03  4.025e-04   6.892 5.51e-12 ***
job_food_and_purchaser                          -3.305e-03  6.350e-04  -5.204 1.95e-07 ***
job_hospital_and_emergency                      -1.828e-03  8.149e-04  -2.243 0.024912 *
job_industrial_and_materials                     7.737e-03  1.202e-03   6.436 1.23e-10 ***
job_manager                                     -2.244e-04  6.516e-04  -0.344 0.730602
job_mayoral_staff                               -1.001e-02  1.670e-03  -5.996 2.03e-09 ***
job_medical_health_and_diagnostic_expert         3.809e-03  3.928e-04   9.699  < 2e-16 ***
job_museum_and_art_supervisor                   -3.533e-03  1.094e-03  -3.230 0.001238 **
job_others                                       8.463e-06  2.510e-04   0.034 0.973103
job_police_and_investigation                    -1.135e-03  4.440e-04  -2.556 0.010599 *
job_power_and_fire_executive                    -6.976e-03  8.501e-04  -8.207 2.30e-16 ***
job_public_relations_and_child_support          -1.732e-03  4.586e-04  -3.778 0.000158 ***
job_safety_and_security                         -8.729e-03  1.033e-03  -8.451  < 2e-16 ***
job_technician_and_tech_expert                  -1.870e-03  5.527e-04  -3.382 0.000719 ***
job_transit_and_transport                       -7.344e-03  4.399e-04 -16.696  < 2e-16 ***
job_utility_and_janitorial_services              1.487e-03  4.661e-04   3.191 0.001420 **
job_water_services_and_welfare                   3.661e-03  7.240e-04   5.056 4.29e-07 ***
other_benefits                                   6.331e-02  1.060e-03  59.734  < 2e-16 ***
total_compensation                               7.529e-01  1.037e-03 725.900  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01651 on 104835 degrees of freedom
Multiple R-squared:  0.9659,	Adjusted R-squared:  0.9658
F-statistic: 1.809e+04 on 164 and 104835 DF,  p-value: < 2.2e-16

>
```

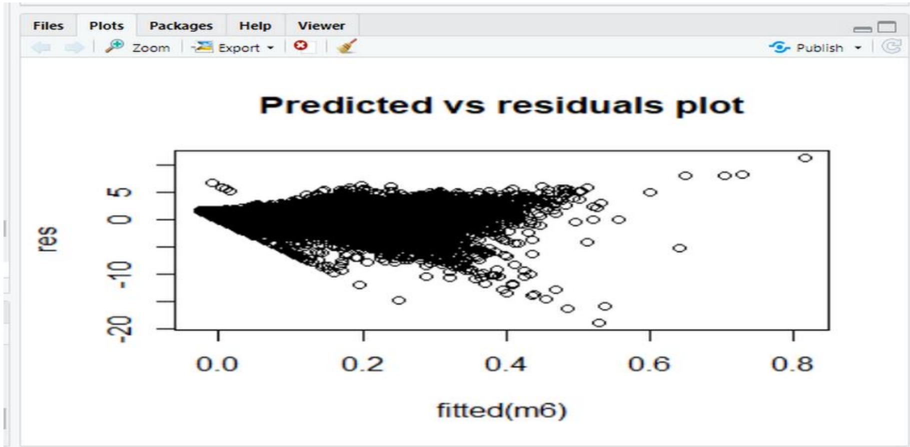
Normality Test



Here we can observe that the residual is normally distributed as most of the points are lying near the straight line.

Residual Plot

Here we can observe that the variance is constant as the spread is constant and points are randomly scattered. There is no pattern here.



- Jarque-Bera Test

```
> jarque.bera.test(res)

        Jarque Bera Test

data:  res
X-squared = 1135360, df = 2, p-value < 2.2e-16

>
```

From the above test, we can conclude that p-value is less than 0.05, so the model follows normality.

- RMSE

```
> y1=predict.glm(m6,test_s.data)
> y=test_s.data[,164]
> rmse_1 = sqrt((y-y1)%*%(y-y1)/nrow(test_s.data))
> rmse_1
           [,1]
[1,] 0.01673998
>
```

A final conclusion can be given based on the comparison of the two models built on total compensation including the backward and the forward.

11 Salary using forward elimination

Model 1 –

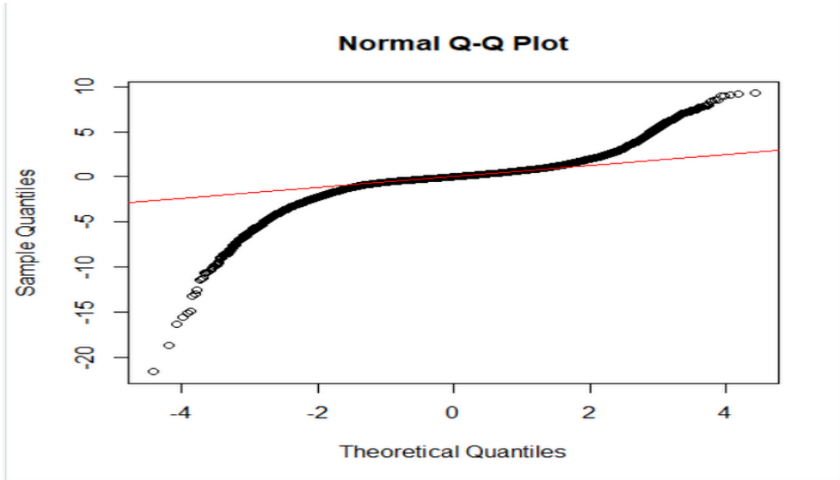
Step 2 – Calculating AIC , ADJUSTED R2 AND RMSE VALUES –

```r
#salaries
m1=lm(train.data$salaries ~ .,data=train.data)
summary(m1)
install.packages("leaps")
library(leaps)

#names(subdata)
m2=step(m1, direction = "backward", trace = T)

#forwad
base=lm(salaries~retirement, data=train.data)
m3=step(base, scope=list(upper=m1, lower=~1),direction="forward",trace=F)
summary(m3)
```

```
department_code_cfc                                      0.0051381  0.0024182   2.125 0.033606 *
department_code229347                                    0.0028036  0.0016197   1.731 0.083471 .
job_asphalt_and_asr_officer                             -0.0023650  0.0010481  -2.257 0.024035 *
year2019                                                -0.0007872  0.0002073  -3.798 0.000146 ***
year2018                                                -0.0007052  0.0001966  -3.587 0.000335 ***
organization_group_human_welfare_neighborhood_development -0.0009807  0.0002957  -3.316 0.000912 ***
department_code_reg                                     -0.0017830  0.0007122  -2.503 0.012301 *
department_code229313                                    0.0027815  0.0017118   1.625 0.104189
department_code_ttx                                     -0.0014743  0.0007171  -2.056 0.039804 *
department_code_hrd                                     -0.0013156  0.0006451  -2.039 0.041410 *
job_others                                              -0.0004663  0.0001965  -2.374 0.017617 *
job_public_relations_and_child_support                  -0.0008677  0.0003945  -2.200 0.027827 *
union_employees                                          0.0003220  0.0002066   1.559 0.119056
department_code229982                                   -0.0010403  0.0005676  -1.833 0.066812 .
department_code232300                                   -0.0013428  0.0007053  -1.904 0.056919 .
department_code_rec                                     -0.0008956  0.0005163  -1.735 0.082820 .
department_code_una                                     -0.0032590  0.0019256  -1.693 0.090553 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01502 on 104885 degrees of freedom
Multiple R-squared:  0.9716,    Adjusted R-squared:  0.9715
F-statistic: 3.142e+04 on 114 and 104885 DF,  p-value: < 2.2e-16
```

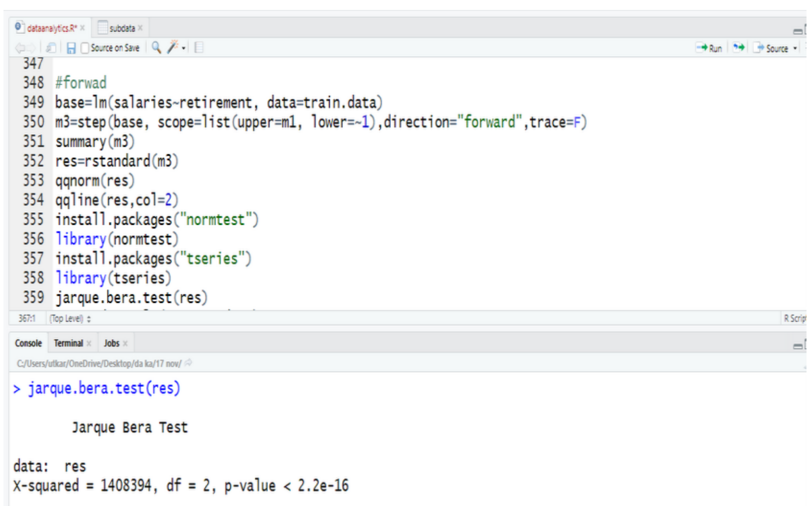
Residual Analysis-

Normality Test-



Here we can observe that the residual is normally distributed as most of the points are lying near the straight line.

Jarque Bera Test-

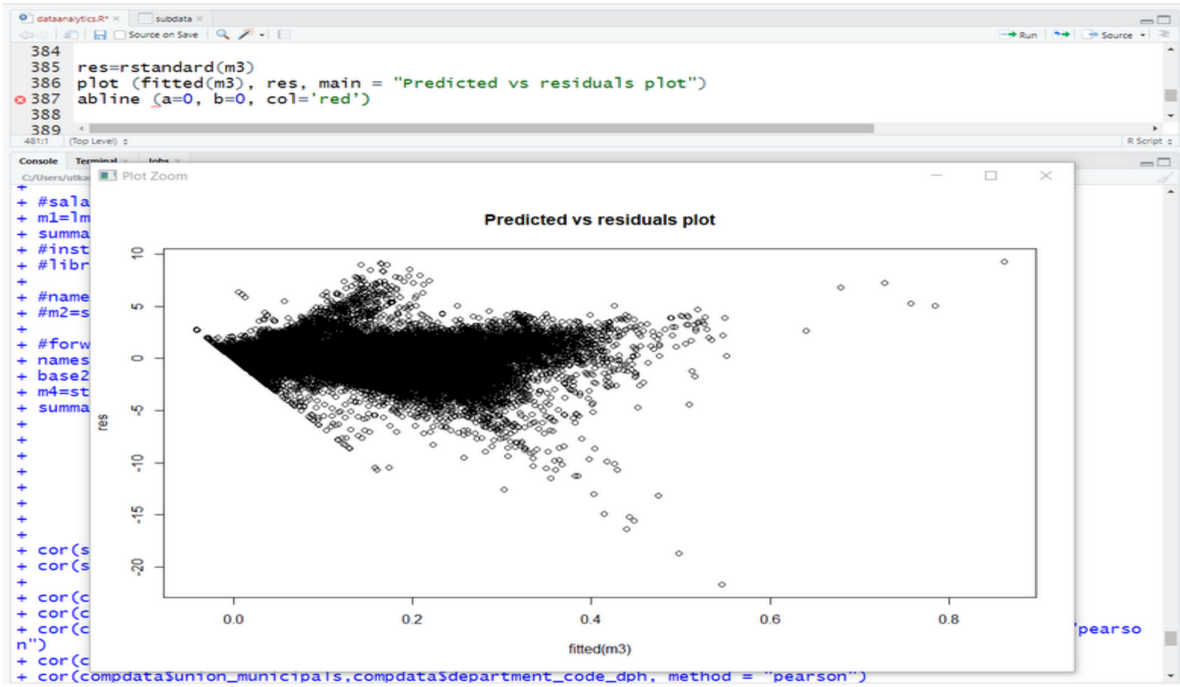


```
#forwad
base=lm(salaries~retirement, data=train.data)
m3=step(base, scope=list(upper=m1, lower=~1),direction="forward",trace=F)
summary(m3)
res=rstandard(m3)
qqnorm(res)
qqline(res,col=2)
install.packages("normtest")
library(normtest)
install.packages("tseries")
library(tseries)
jarque.bera.test(res)
```

```
> jarque.bera.test(res)

        Jarque Bera Test

data:  res
X-squared = 1408394, df = 2, p-value < 2.2e-16
```

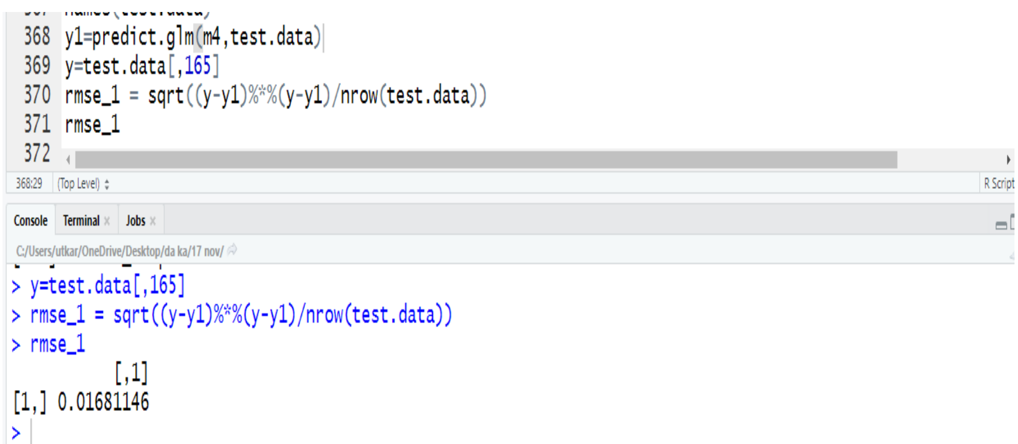
From the above test, we can conclude that p-value is less than 0.05, so the model follows normality.

Check the variance:



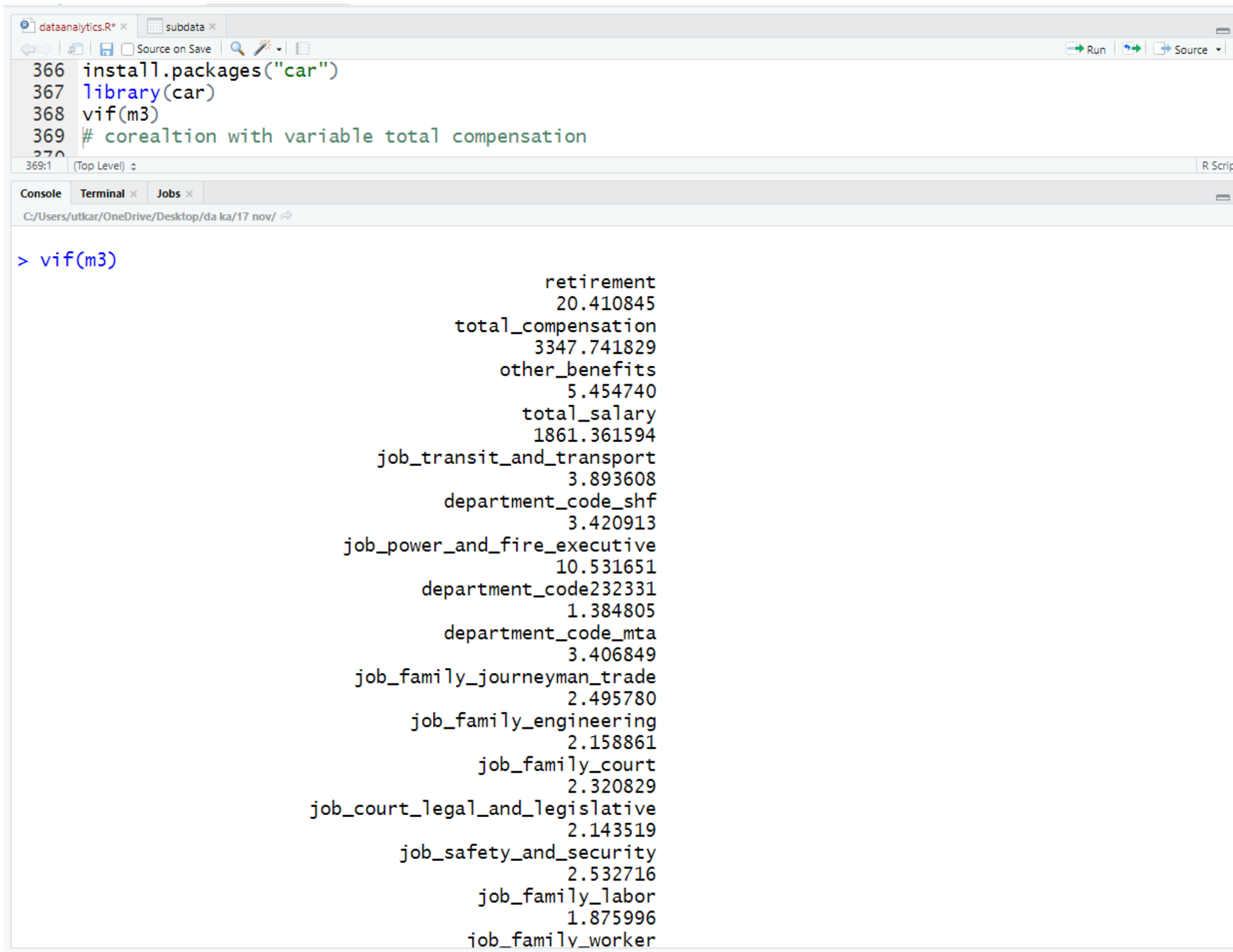
Here we can observe that the variance is constant as the spread is constant and points are randomly scattered. There is no pattern here.

In order to calculate the value of RMSE –



In order to check the multicollinearity using the VIF.

After checking the multicollinearity and removing the variables having collinearity greater than +-0.9 a new model was being built.



```
install.packages("car")
library(car)
vif(m3)
# corealtion with variable total compensation
```

```
> vif(m3)
                      retirement
                       20.410845
              total_compensation
                     3347.741829
                  other_benefits
                        5.454740
                    total_salary
                     1861.361594
      job_transit_and_transport
                        3.893608
             department_code_shf
                        3.420913
    job_power_and_fire_executive
                       10.531651
          department_code232331
                        1.384805
             department_code_mta
                        3.406849
    job_family_journeyman_trade
                        2.495780
         job_family_engineering
                        2.158861
               job_family_court
                        2.320829
  job_court_legal_and_legislative
                        2.143519
       job_safety_and_security
                        2.532716
               job_family_labor
                        1.875996
              job_family_worker
```

From the VIF calculated and after checking the co-relations we can give a conclusion that some columns can be removed having higher multi collinearity.

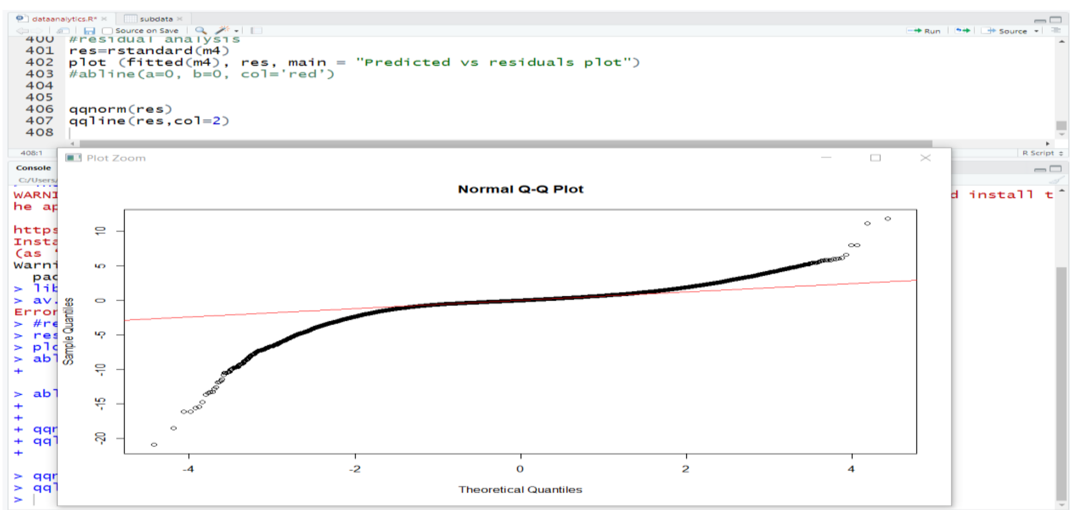
Model after resolving the multi-collinearity-

Model 2 -

```
400
401 #forwad
402 names(subdata)
403 base2=lm(salaries~total_compensation, data=train.data)
404 m4=step(base2, scope=list(upper=m1, lower=~1),direction="forward",trace=F)
405 summary(m4)
406
407
```

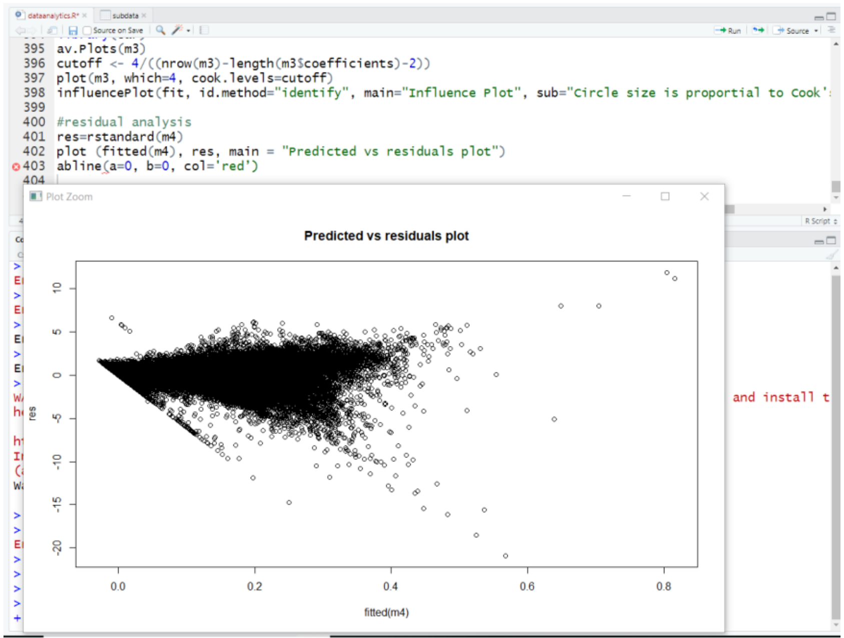
```
union_commissioner_no_benefits                          -0.0100063  0.0029848  -3.352 0.000801 ***
job_electrical_and_electronics_engineer                  0.0009322  0.0005678   1.642 0.100603
department_code_hhp                                     -0.0012413  0.0008422  -1.474 0.140493
department_code232073                                    0.0041016  0.0024734   1.658 0.097260 .
department_code_cfc                                      0.0049015  0.0027930   1.755 0.079272 .
department_code_pdr                                      0.0015182  0.0009744   1.558 0.119240
department_code_hrd                                     -0.0017006  0.0007337  -2.318 0.020454 *
department_code_eth                                     -0.0044428  0.0025839  -1.719 0.085542 .
department_code229982                                   -0.0011031  0.0006142  -1.796 0.072515 .
job_hospital_and_emergency                              -0.0019537  0.0007937  -2.461 0.013839 *
job_family_personnel                                     0.0007018  0.0003293   2.131 0.033082 *
job_accountant_fiscal_and_tax                            0.0021969  0.0007820   2.809 0.004965 **
job_family_payroll_billing_accounting                   -0.0015258  0.0007227  -2.111 0.034764 *
job_auditor_and_audio                                    0.0027436  0.0015299   1.793 0.072916 .
department_code_ttx                                     -0.0016638  0.0008636  -1.927 0.054019 .
organization_group_human_welfare_neighborhood_development -0.0007586  0.0004133  -1.835 0.066467 .
union_employees                                         -0.0047767  0.0010716  -4.458 8.30e-06 ***
union_board_members                                     -0.0053730  0.0014502  -3.705 0.000212 ***
union_auto_machinists                                   -0.0047684  0.0013472  -3.540 0.000401 ***
union_sheriffs_managers_and_supervisors_association     -0.0053381  0.0015836  -3.371 0.000750 ***
department_code232303                                    0.0033800  0.0022748   1.486 0.137333
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.01662 on 104881 degrees of freedom
Multiple R-squared:  0.9652,	Adjusted R-squared:  0.9651
F-statistic: 2.462e+04 on 118 and 104881 DF,  p-value: < 2.2e-16

>
```

Normality Test



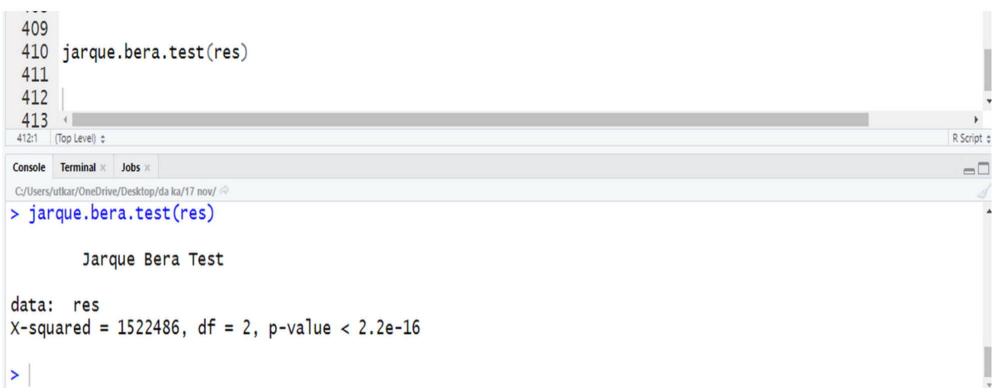
Here we can observe that the residual is normally distributed as most of the points are lying near the straight line.

Residual Plot



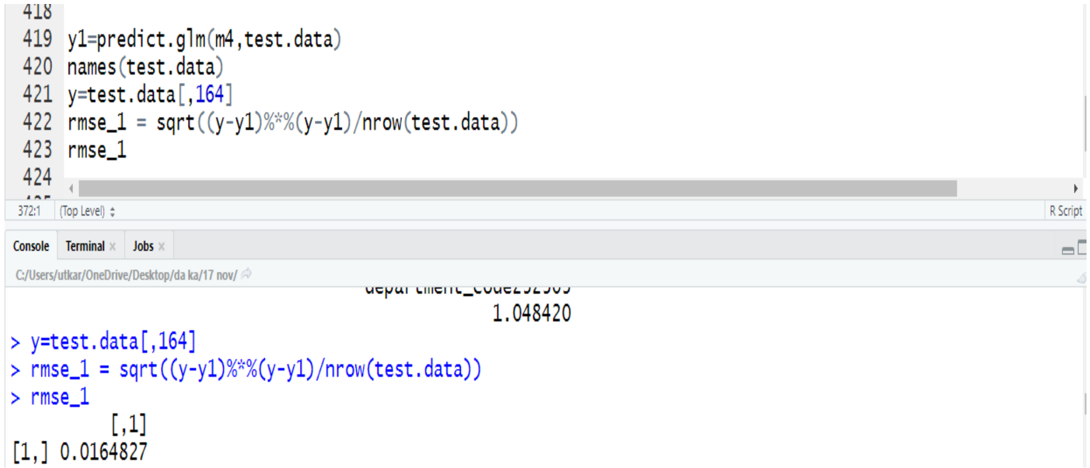
Here we can observe that the variance is constant as the spread is constant and points are randomly scattered. There is no pattern here.

- Jarque-Bera Test



From the above test, we can conclude that p-value is less than 0.05, so the model follows normality.

- RMSE



A final conclusion can be given based on the comparison of the two models built on total compensation including the backward and the forward.

Comparing the backward and forward models for total compensation -

Measures	Backward Elimination	Forward Selection
ADJ R2	0.9658	0.9651
RMSE	0.0167	0.0164

As we can observe that the RMSE for the forward selection is less as compared to the backward one.

Thus, we would prefer the forward model for salary prediction instead of the backward one.

5.3. Findings

For Total Compensation

We observed that the RMSE for the backward elimination model is less as compared to the forward selection model.

Thus, we would prefer the backward model for Total compensation prediction instead of the forward one.

For Salary

We observed that the RMSE for the forward selection is less as compared to the backward one.

Thus, we would prefer the forward model for salary prediction instead of the backward one.

ANOVA testing for job

At 95% confidence level, p-value is less than 0.05, we can reject null- hypothesis. Hence, the average salaries are not equal for all jobs.

BOX PLOT for organization

Using box plot, we can compare salaries between different organizations as variance is smaller and we can conclude that **Public Protection** has the highest average salary

6. Conclusions and Future Work

6.1. Conclusions

- We wanted to predict the total compensation of the employee based on various factors that will help the employers to decide what compensation should be given to employee in advance in order to keep tabs on their financial section.

- We wanted to predict the salaries of the employee based on benefits, compensation and job profile that will help the employees to aim for better job profiles based on high benefits.

- We wanted to tell if all the employees are same or different for various organizations or job profiles.
- We used Multiple linear regression to predict the compensation and benefits given to the employee based on salary, organization and job profile.

- We used Multiple linear regression to predict the salaries given to the employee based on organization and job profile and other factors.

- We will use ANOVA to compare average salaries of different employees based on job profiles and organization.

- In total compensation as we can observe that the RMSE for the forward selection is less as compared to the backward one.

- Thus, we would prefer the forward model for total compensation instead of the backward one.
- In salary we can observe that the RMSE for the backward selection is less as compared to the forward one.
- Thus, we would prefer the backward model for total compensation instead of the forward one.

6.2. Limitations

- Due to large dataset, we had to sample our data and then build the model as it was showing system limitations.

- There was issue in calculating Influence measures as it was not showing proper results, so we had to omit that part in our case.

6.3. Potential Improvements or Future Work

- Grouping of the job profiles in a better way in order to provide best association.
- Individual parameter test for each job profile in ANOVA testing in order to build better prediction model.
- Treatment of influential points; due to large dataset, influence measures wasn't giving proper results for influence points, so we can do it better on proper systems with enhanced specifications.
- Employees can use the predictive model to imply better strategies in terms of better job search which can provide better compensation and salary.
- Similarly, Employers can decide what compensation and salary should be given to the job seeker based on job and other factors in order to optimize their financial status.