MACHINE LEARNING PROJECT

PREDICT CUSTOMER MODE OF TRANSPORT

Shweta Gupta

PGPBABI

**TABLE OF CONTENTS**

* Background and Objectives
* Data Understanding
* EDA - Continuous Variables
* EDA – Categorical Variables
* Feature Engineering – Treating multicollinearity using PCA analysis
* Summary EDA
* Slicing the data as Train and Test
* SMOTE- Balancing the Data
* Model building
* Logistic Regression
* KNN
* Naïve Bayes
* Bagging
* Boosting
* Choosing the best fit model
* Insights from the analysis

**PROBLEM AT HAND**

* The project requires us to **understand what mode of transport employees prefers to commute to their office**.
* The data '[Cars.csv](https://olympus.greatlearning.in/courses/8747/files/923956/download?verifier=J7wNhWvVZ2Pkdx6uKGb95mmPemZSuM7nvTxtMiW5&wrap=1)' includes employee information about their mode of transport as well as their personal and professional details like age, salary, work exp.
* We need to predict **whether or not an employee will use Car as a mode of transport.**
* Also, **which variables are a significant predictor** behind this decision?

**OBJECTIVES**

**EDA**

* Perform an EDA on the data
* Illustrate the insights based on EDA
* Check for Multicollinearity - Plot the graph based on Multicollinearity & treat it.

**Data Preparation**

Prepare the data for analysis (SMOTE)

**Modeling**

Create multiple models and explore how each model perform using appropriate model performance metrics

* KNN
* Naive Bayes (is it applicable here? comment and if it is not applicable, how can you build an NB model in this case?)
* Logistic Regression
* Apply both bagging and boosting modeling procedures to create 2 models and compare its accuracy with the best model of the above step.

**Actionable Insights & Recommendations**

* Summarize your findings from the exercise in a concise yet actionable note

**DATA UNDERSTANDING**

The dataset contains **8 Independent variables** and **1 Dependent variable called Transport**.

There is a total of 444 observations across 9 variables as given below:

**Data Dictionary**

|  |  |
| --- | --- |
| Age | Age of the Employee in Years |
| Gender | Gender of the Employee |
| Engineer | For Engineer =1 , Non Engineer =0 |
| MBA | For MBA =1 , Non MBA =0 |
| Work Exp | Experience in years |
| Salary | Salary in Lakhs per Annum |
| Distance | Distance in Kms from Home to Office |
| License | If Employee has Driving Licence -1, If not, then 0 |
| **Transport** | **Mode of Transport** |

**CHECK DEPENDENT VARIABLE DISTRIBUTION**

**Dependent variable Transport– Proportion Table**

2Wheeler Car Public Transport

0.1869369 0.1373874 0.6756757

* The data is skewed towards public transport and hence it is imbalanced as those who travel by Car only constitute **13.7%** of total observations.

**EXPLORATORY DATA ANALYSIS**

**FIVE POINT SUMMARY**

Age Gender Engineer MBA Work.Exp

Min. :18.00 Female:128 0:109 0 :331 Min. : 0.0

1st Qu.:25.00 Male :316 1:335 1 :112 1st Qu.: 3.0

Median :27.00 NA's: 1 Median : 5.0

Mean :27.75 Mean : 6.3

3rd Qu.:30.00 3rd Qu.: 8.0

Max. :43.00 Max. :24.0

Salary Distance license Transport

Min. : 6.50 Min. : 3.20 0:340 2Wheeler : 83

1st Qu.: 9.80 1st Qu.: 8.80 1:104 Car : 61

Median :13.60 Median :11.00 Public Transport:300

Mean :16.24 Mean :11.32

3rd Qu.:15.72 3rd Qu.:13.43

Max. :57.00 Max. :23.40

* High variation in salary
* Most people use public transport
* Most people are engineers
* Males employees are more than females
* Only 1/4th of total respondents have license

**CHECK FOR MISSING VALUES**

**Missing Values**

Age Gender Engineer MBA Work.Exp Salary Distance

0 0 0 1 0 0 0

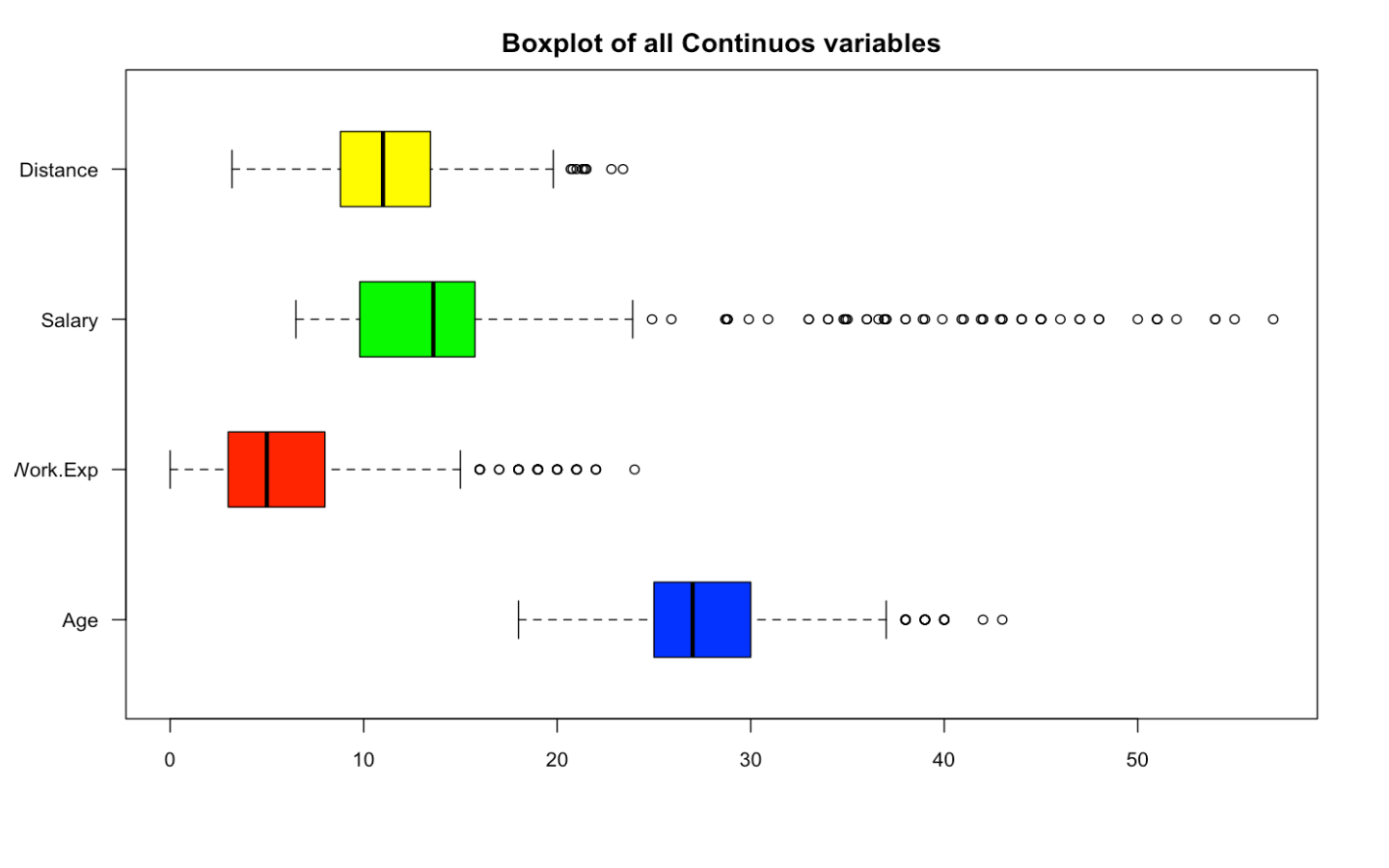
license Transport

0 0

* Only 1 missing value present in MBA variable

**IDENTIFY PRESENCE OF OUTLIERS IN CONTINUOUS VARIABLES**

**Boxplot**



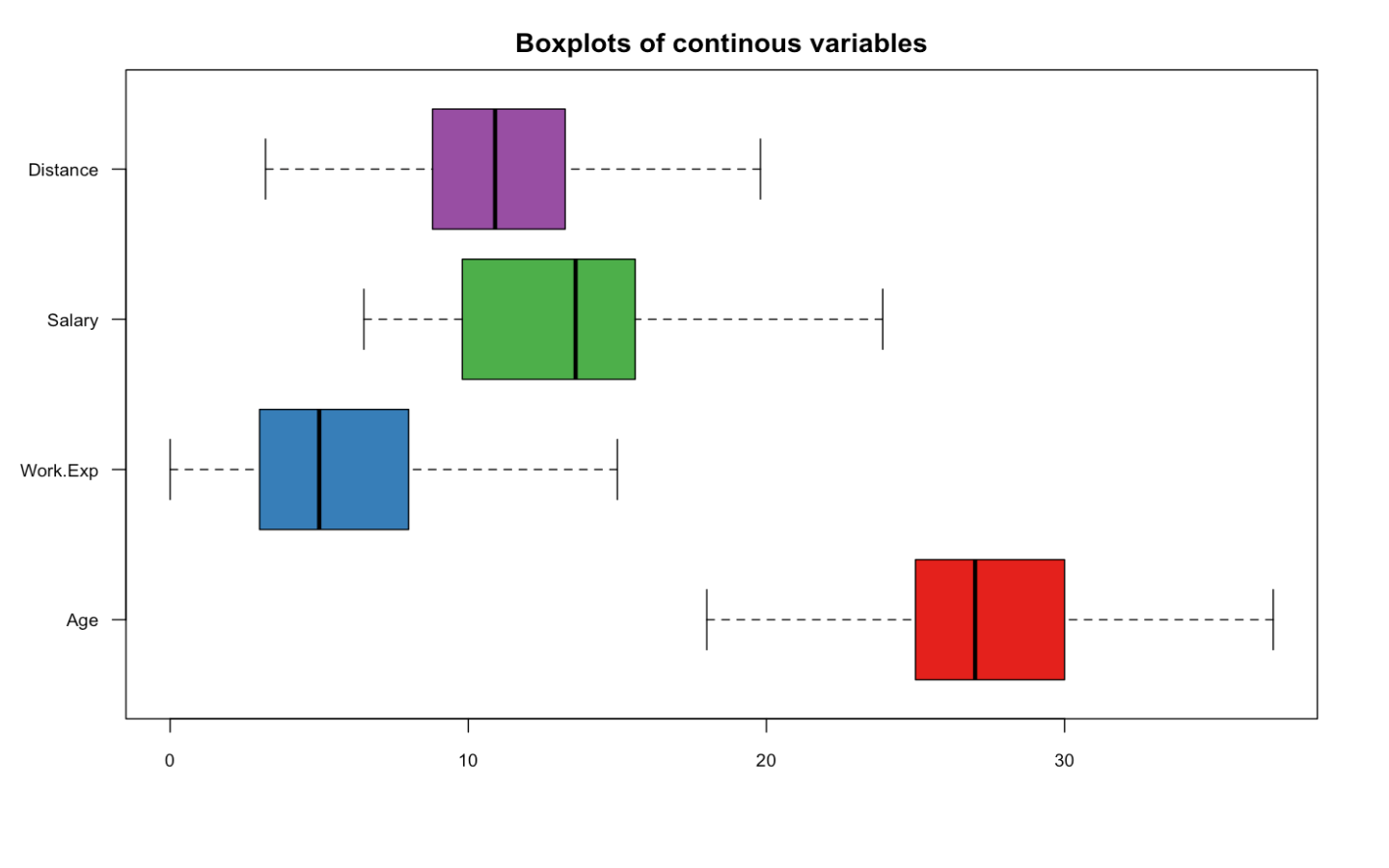
* All continuous variables in our data have outliers as indicated in the boxplot above.
* No of outliers by Variables: - Age – 25; Work Exp – 38; Salary – 59; Distance – 9

**NEXT STEPS – DATA PREPARATION AND FEATURE ENGINEERING**

* Treat Outliers using MICE imputation
* Treat Missing Values using MICE imputation
* Change the Class of variables as appropriate if needed
* Recode the Dependent variable Transport to Binary
  + Car to be coded as 1 and rest other modes of transport coded as 0
* Create additional variable **Education** by combining Engineer and MBA variables
  + Those who have done both Engineering and MBA is coded as Both or 3, Those who have done only MBA are coded as 2, Those who have done only Engineering are coded as 1
* Remove variables MBA and Engineer from the data
* Check for Multicollinearity among Independent variables
* Run PCA to treat multicollinearity and combine correlated variables in a factor
* Arrive at final dataset post the above steps

**UNIVARIATE ANALYSIS**

**BOXPLOT OF CONTINOUS VARIABLES POST TREATING OUTLIERS**



* We performed MICE imputation to treat outliers and missing values
* Post MICE imputation there are no outliers present in the data
* From the above boxplot we can infer that median age is around 27 years, median work exp is around 5 years, median salary around 14 lakhs per annum and median distance is around 11 KM
* While Age and work exp are right skewed towards older adults and more experience, Salary is left skewed towards lower salaries.
* Distance follows near normally distributed boxplot

**Missing Values**

Age Gender Engineer MBA Work.Exp Salary Distance

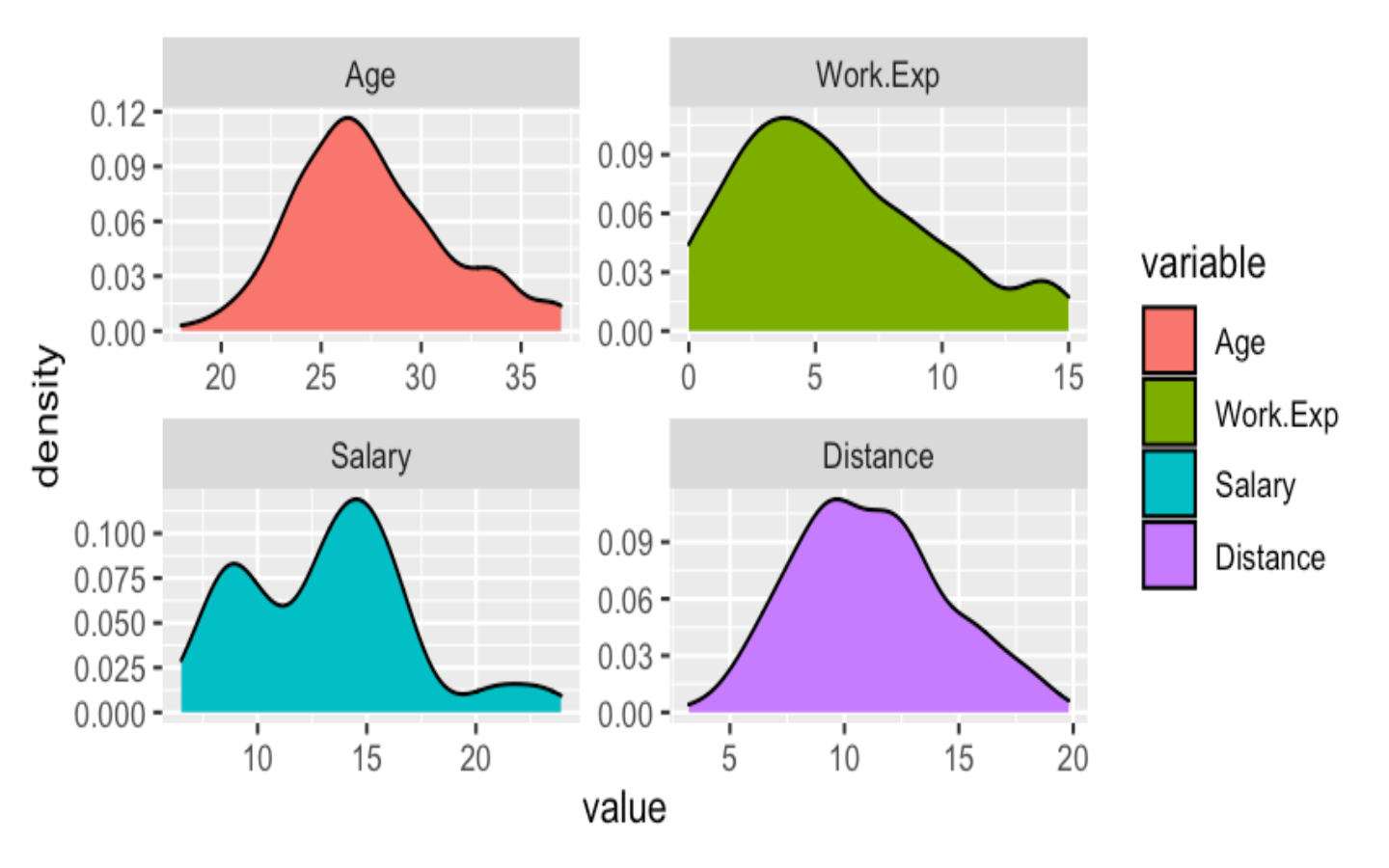
0 0 0 0 0 0 0

license Transport Transport2 education

0 0 0 0

* No missing values in the data as well post MICE treatment

**DENSITY PLOT OF CONTINUOUS VARIABLES**

****

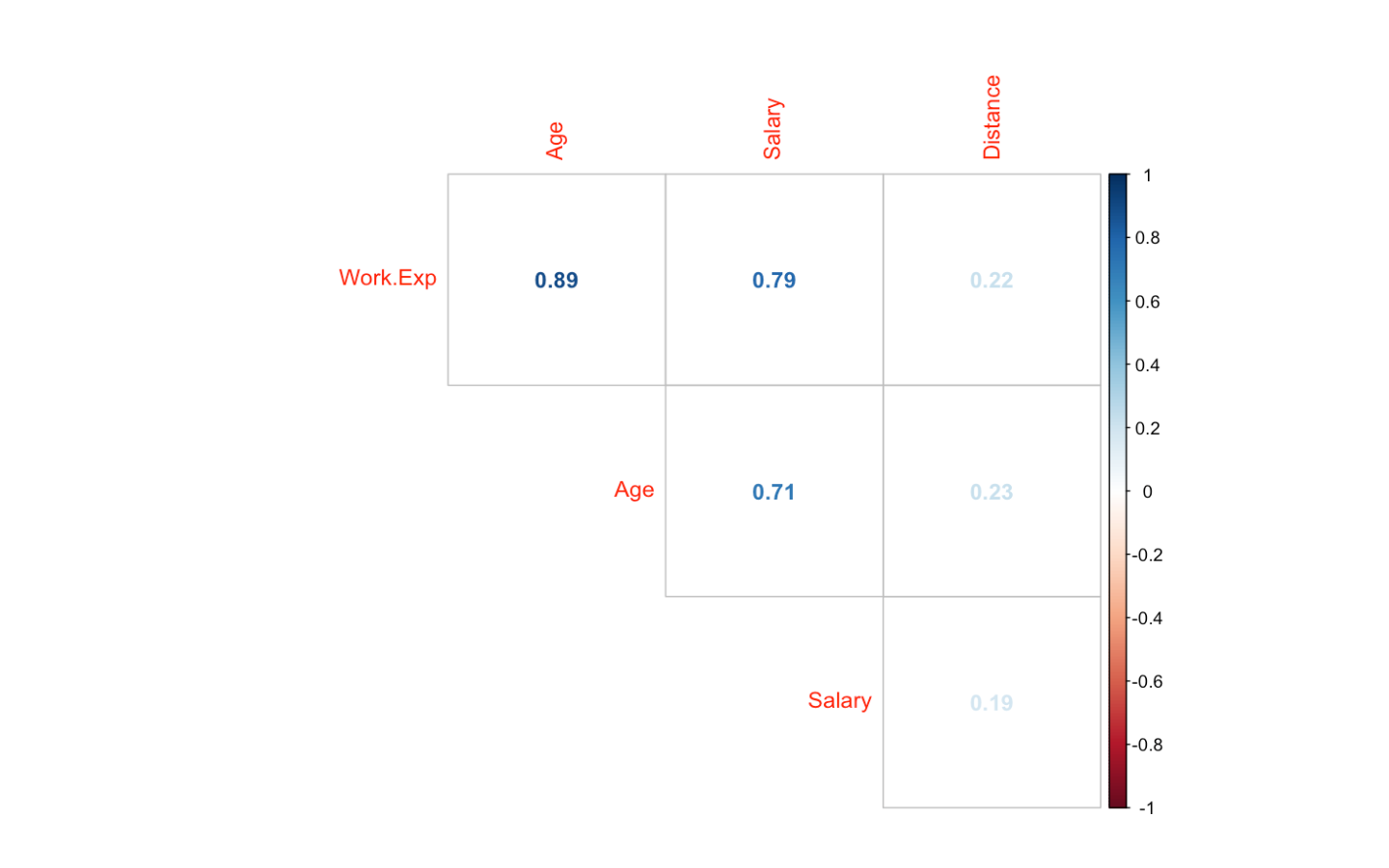
* Age, Work experience and Distance are normally distributed
* Work experience variable is right skewed indicating higher presence of people with more experience
* Salary shows 2 peaks, indicating extreme values in data

**BIVARIATE ANALYSIS**



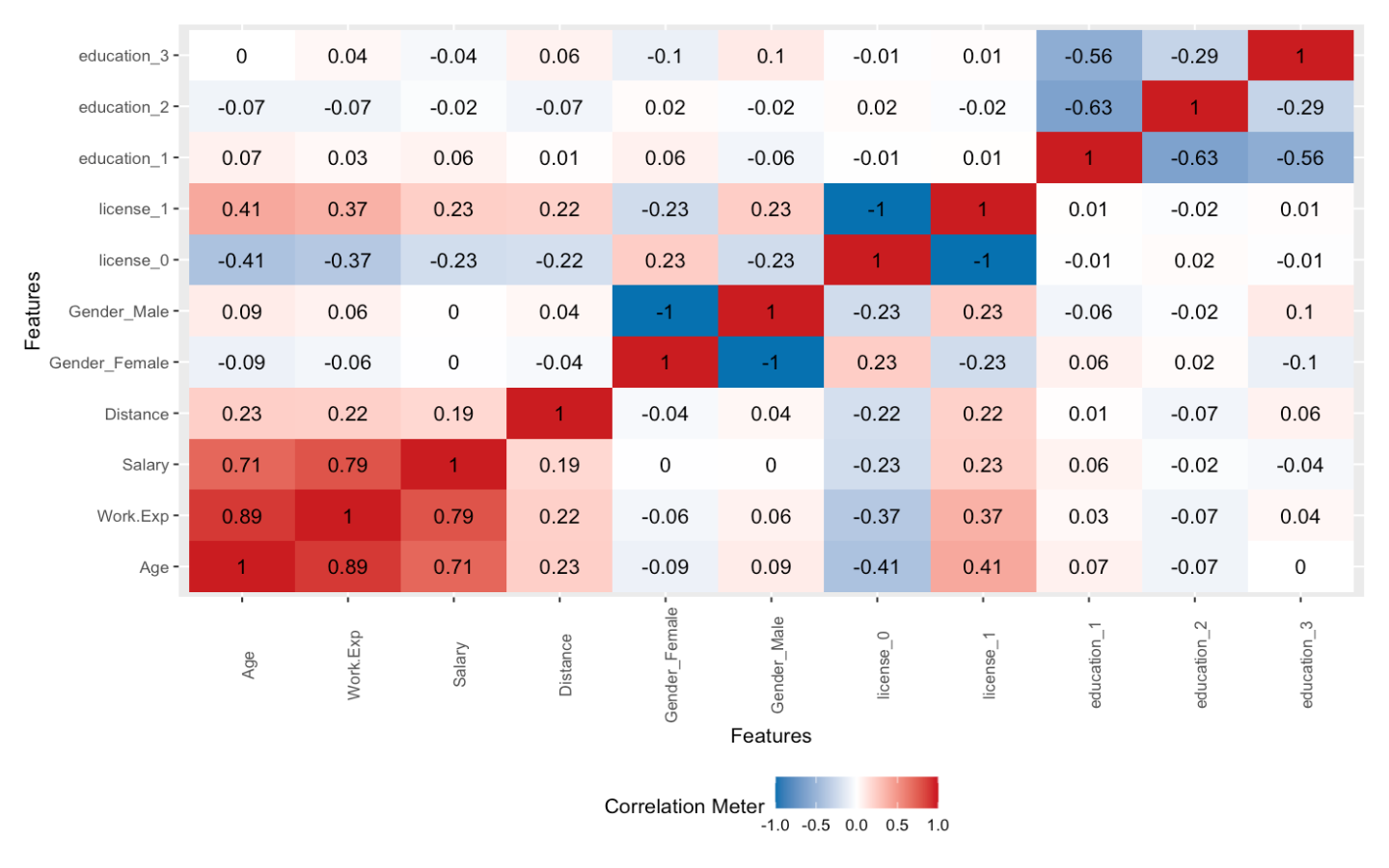
* Those using car as a mode of transport have a higher age, work experience, salary and distance from home compared to those who do not use car

**CHECK MULTICOLLINEARITY AMONG CONTINUOUS AND CATEGORICAL INDEPENDENT VARIABLES**

**Correlation among continuous variables**

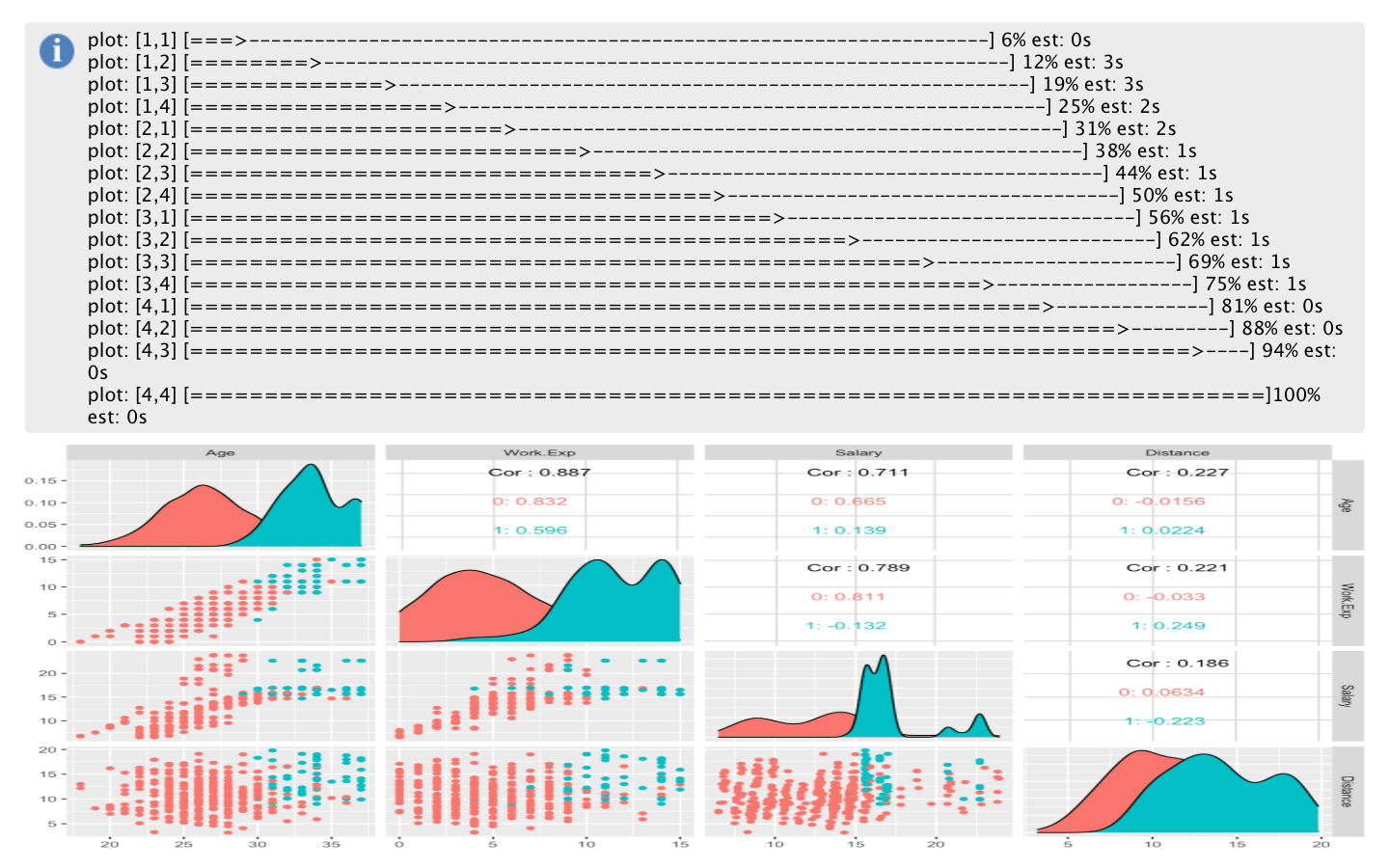
* Work Experience, Age and Salary are highly correlated variables

**Correlation among Categorical and Continuous Independent variables**

****

* No major multicollinearity among factor and continuous variables

**EDA BIVARIATE ANALYSIS - CONTD**

**PAIR PLOTS OF CONTINUOUS VARIABLES**

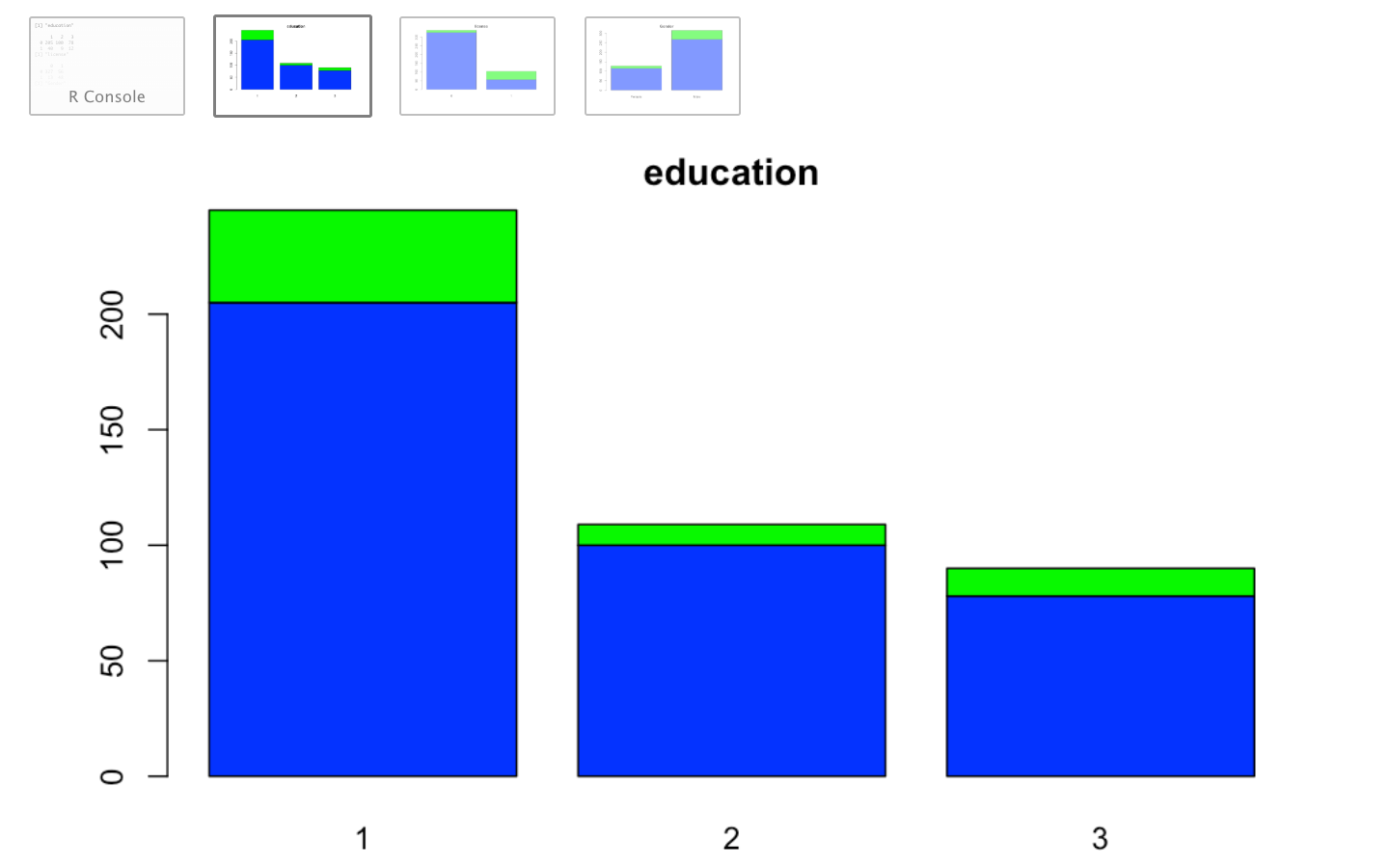
*Color by - Transport*

*Green – Car, Red – All other modes*

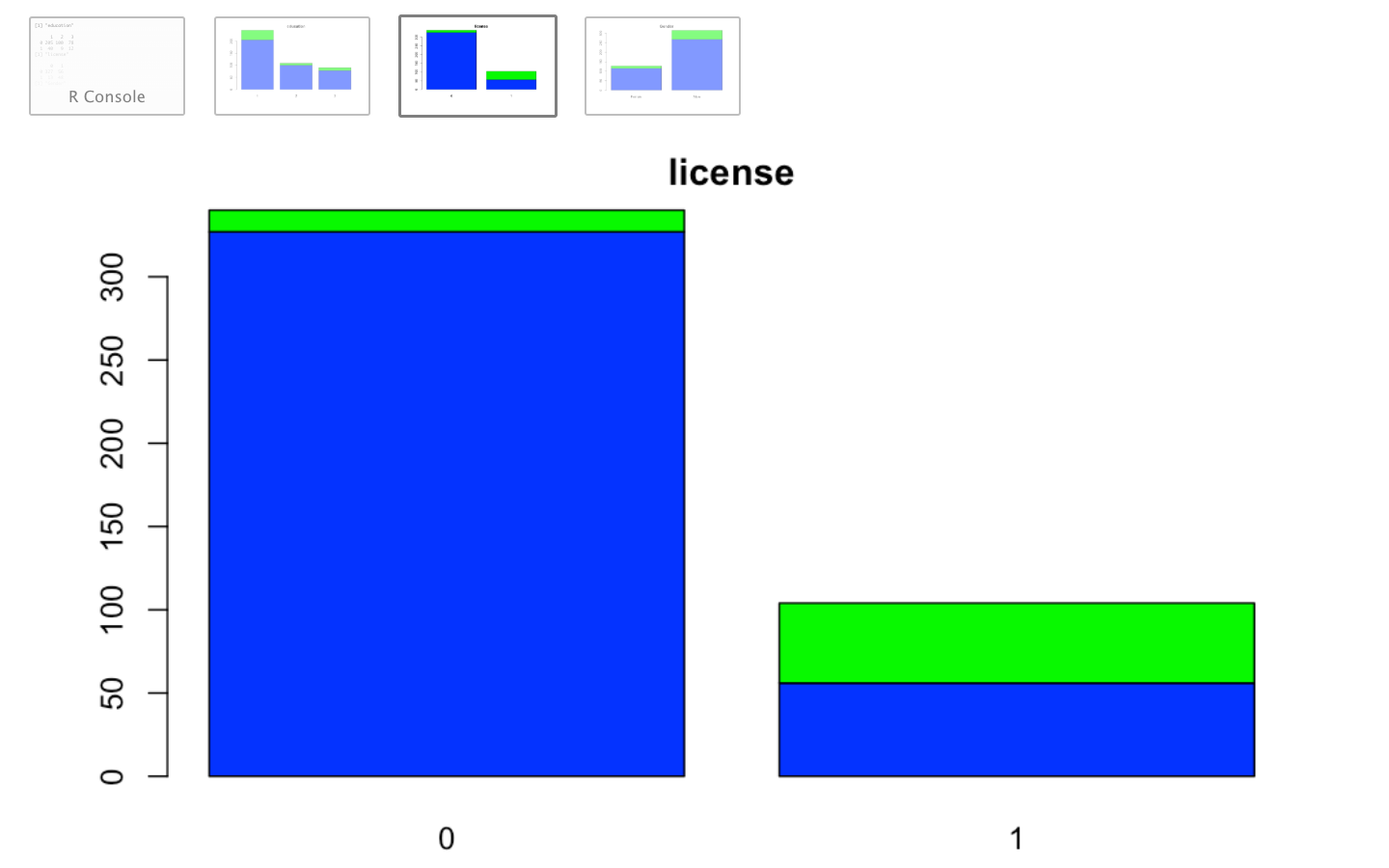
* The scatter plots clearly indicate a linear relationship between Age, Work Exp and Salary.

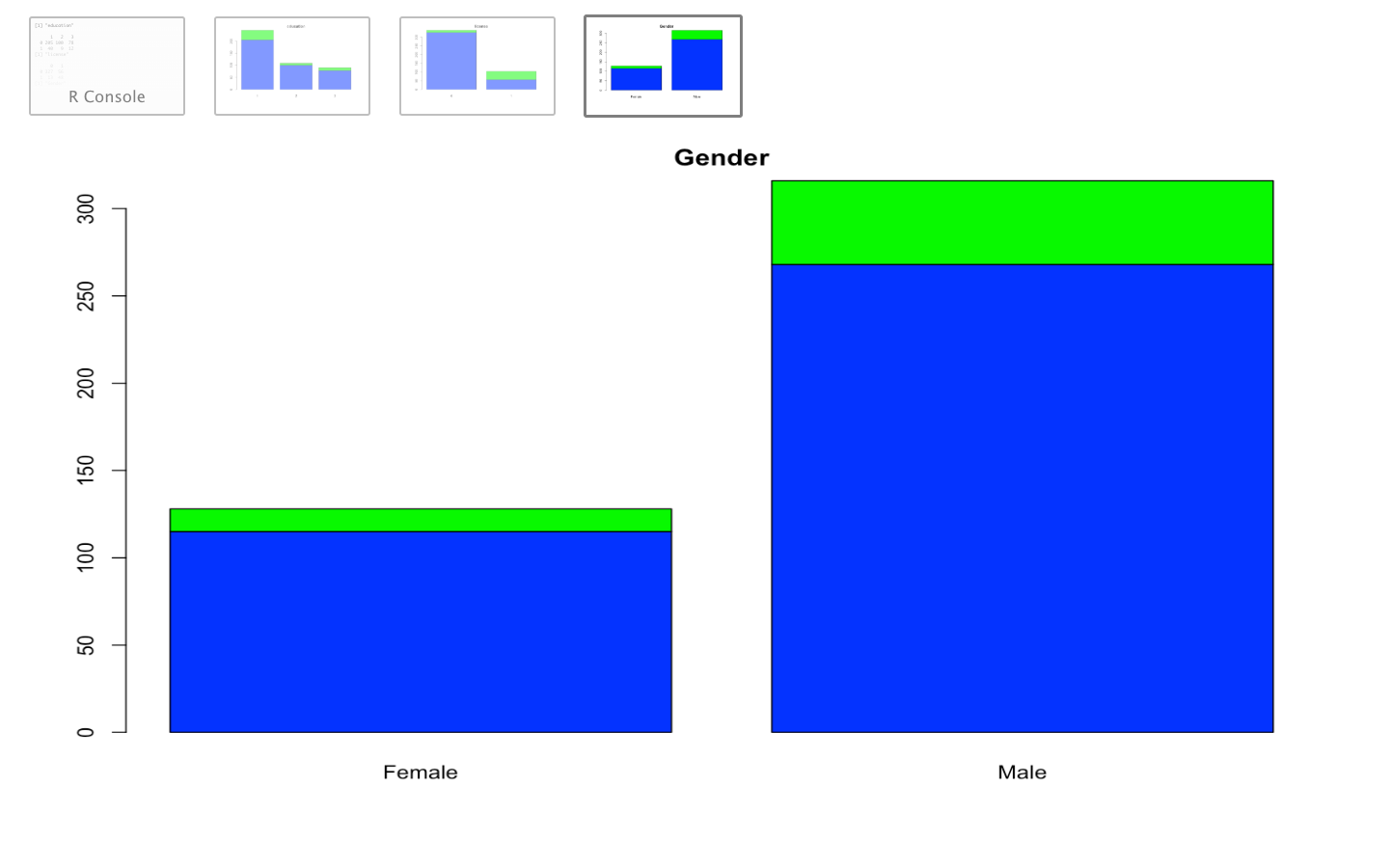
**BIVARIATE ANALYSIS – CATEGORICAL VARIABLES TO TRANSPORT**

**FREQUENCY TABLE/BAR PLOT W.R.T TRANSPORT**









***Legend***

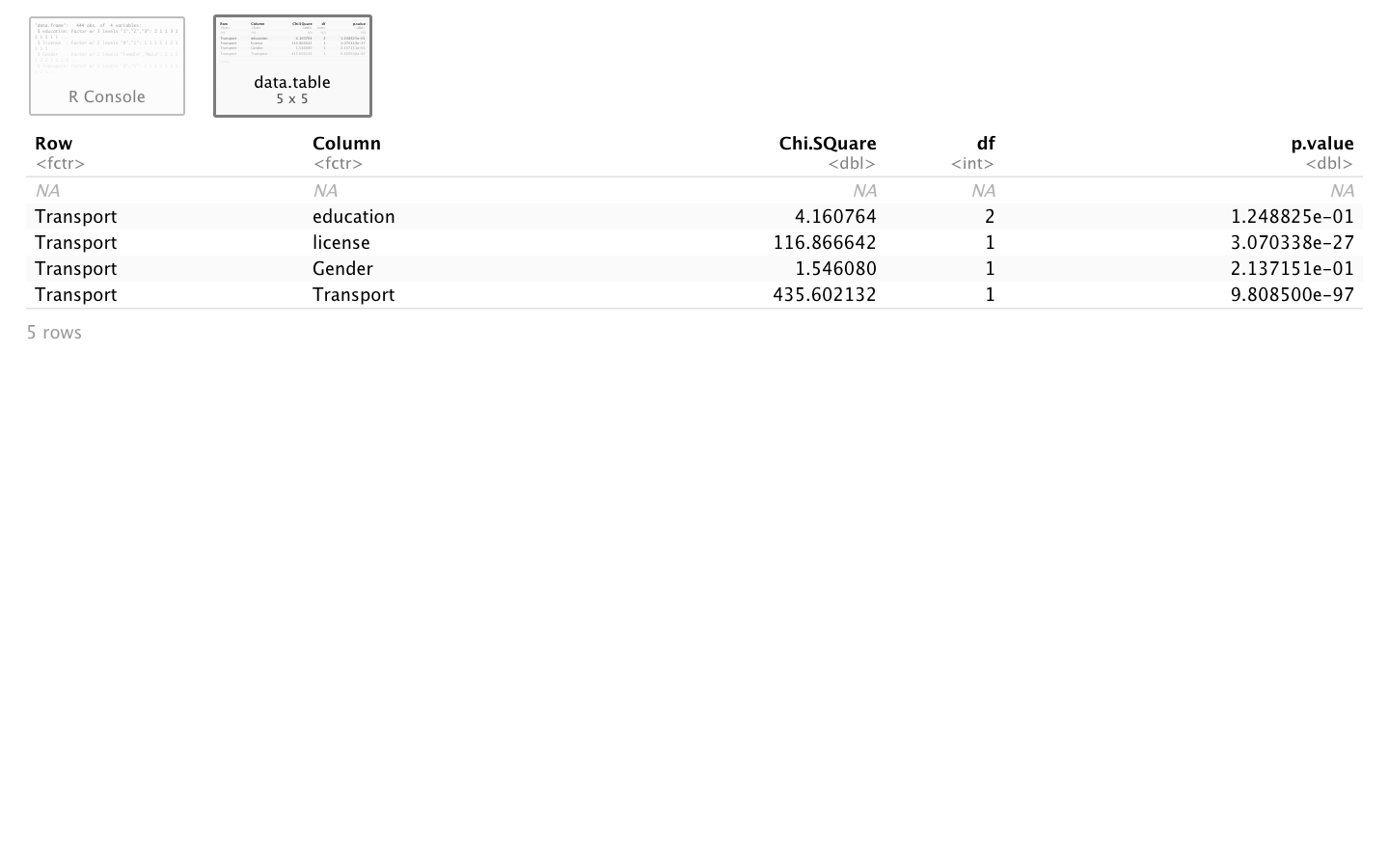
*Transport: 1- (Green)Car, 0 - (Blue)all other modes;*

*License: 0 – No license, 1 – has license;*

*Education :1 – Engineer, 2 – MBA, 3 – Both*

* Engineers, Males and those who possess a license are more likely to use car as a mode of transport.

**CHI SQUARE TEST TO DETERMINE IMPORTANT CATEGORICAL VARIABLES WRT TRANSPORT**



* All categorical variables are significant in explaining transport as P value is very small for all of them

**TREATING MULTICOLLINEARITY IN THE DATA**

**TEST FOR POSSIBILITY OF DIMENSIONALITY REDUCTION**

Cortest.bartlet test

[1] 249.7168

$p.value

[1] 4.714646e-51

$df

[1] 6

* Cortest bartlet test checks whether the correlation matrix is an identity matrix or not. If it is not an identity matrix then there is a possibility of dimensionality reduction.
* Since P value is much lower than 0.95 confidence level, hence the test indicates there is a scope dimensionality reduction in our data

**TEST FOR SAMPLE ADEQUACY**

Kaiser-Meyer-Olkin factor adequacy

Call: KMO(r = cormat)

Overall MSA = 0.72

MSA for each item =

Age Work.Exp Salary Distance

0.70 0.64 0.83 0.96

* Since overall MSA is more than 0.5, hence we can safely assume that our sample is sufficient for conducting PCA

**PERFORMING PCA**

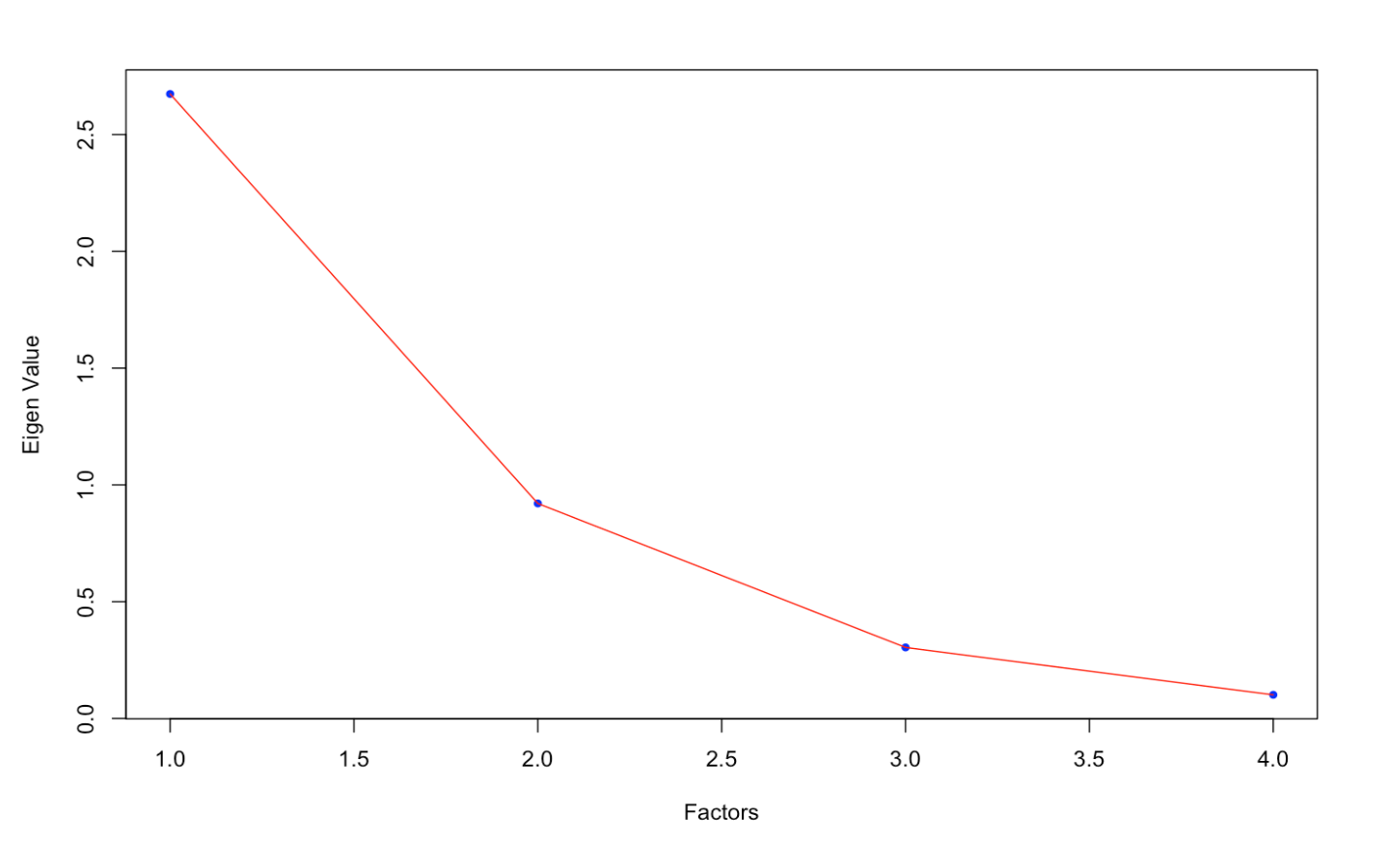
**KEY STEPS**

* We perform PCA on our numerical variables in the dataset to identify and replace correlated variables with factors
* We create a dataset with only numeric variables
* Identify number of factors
* Identify the variables to be replaced with the factor, name the factor
* Create a dataset after removing the correlated variables and including the new factor to represent the correlated variables

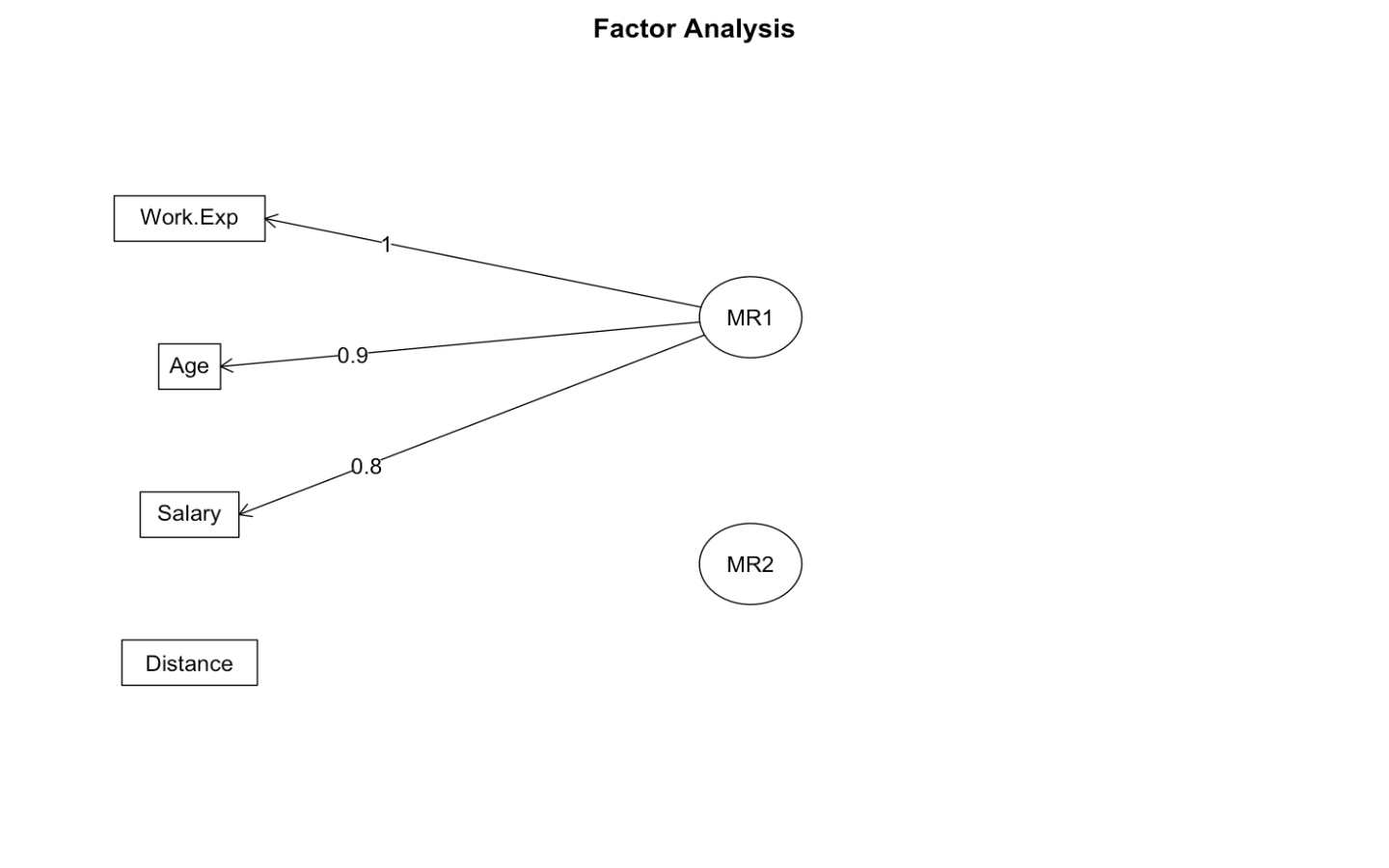
**Eigen values**

> ev

[1] **2.6738772 0.9206283** 0.3041800 0.1013145

**Scree plot**

* Since eigen value below 1 is insignificant, we can look at forming two factors where our eigen value is 0.92 and starts declining significantly after that.

FA diagram

* The above diagram shows that **Work experience, Age and Salary** can be combined into a single factor
* Hence we combine these 3 variables into One variable in our dataset and will name it as SEC

Let’s review the new dataset that we formed as below:

'data.frame': 444 obs. of 6 variables:

$ Gender : Factor w/ 2 levels "Female","Male": 2 1 2 1 2 2 2 1 2 2 ...

$ Distance : num 3.2 3.3 4.1 4.5 4.6 4.8 5.1 5.1 5.1 5.2 ...

$ license : Factor w/ 2 levels "0","1": 1 1 1 1 1 2 1 1 1 1 ...

$ Transport2: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...

$ education : Factor w/ 3 levels "1","2","3": 2 1 1 3 1 1 1 1 1 1 ...

$ SEC : num -0.433 -0.56 0.303 -0.192 -0.449 ...

* Therefore in our new dataset post dimensionality reduction, our variables have come down to 6 from initial 9 variables
* WE have removed Age, Salary and work experience with the new factor named as SEC based on our PCA results

**SUMMARISING EDA**

* Salary density plot shows 2 peaks, indicating extreme values in dataset
* Most employees use public transport and are engineers
* Males employees are more than female employees
* Only 1/4th of total respondents have license
* Age, Work experience and Distance are normally distributed
* Work experience variable is right skewed indicating higher presence of people with more experience
* Median age is around 27 years, median work exp is around 5 years, median salary around 14 lakhs per annum and median distance is around 11 KM
* While Age and work exp are right skewed towards older adults and more experience, Salary is left skewed towards lower salaries.
* The scatter plots of continuous variables indicate a linear relationship between Age, Work Exp and Salary.These are highly correlated variables
* All categorical variables are significant in explaining transport as P value is very small for all of them
* Those using car as a mode of transport have a higher age, work experience, salary and distance from home compared to those who do not use car
* Engineers, Males and those who possess a license are more likely to use car as a mode of transport.
* To treat multicollinearity we perform PCA and create a new dataset by WE replacing Age, Salary and work experience with the new factor named as SEC based on our PCA results

**DATA PARTITIONING**

**KEY STEPS**

* Slice the data into Train and Test set in 70:30 ratio
* Check for the proportions with dependent variable in each data set to ensure that the data division has taken place properly

Dimensions of sliced datasets

**Original** Data dimensions - 444 6

**Test** Data dimensions - 132 6

**Train** Data dimensions - 312 6

Test data – Transport Distribution

0 1

0.8621795 0.1378205

Train data – Transport Distribution

0 1

0.8636364 0.1363636

* Same transport proportions achieved for our Train and Test Dataset ensuring that data slicing is achieved successfully

**SMOTE – DATA BALANCING**

* Our dependent variable Transport is SKEWED towards Class 0 compared to class 1
* This may lead to low accuracy of our model while predicting class 1
* Therefore to improve our accuracy of models with respect to sensitivity, We will use SMOTE ALGORITHM to balance the data by increasing the instances of class 1 before modelling the data
* This will help us improve the sensitivity metric or the measure by which we can predict the class 1 correctly
* We will perform SMOTE operation only on our Train dataset and keep our Test dataset original
* The parameters **perc.over** and **perc.under** control the amount of over-sampling of the minority class and under-sampling of the majority classes, respectively
* **perc.over** - for each case in the original data set belonging to the minority class, **perc.over/100** new examples of that class will be created
* The parameter **perc.under** controls the proportion of cases of the majority class that will be randomly selected for the final "balanced" data set. This proportion is calculated with respect to the newly generated minority class cases
* For K we will go with the default value of 5.
* **SMOTE** command used **= train.new <- SMOTE(Transport2 ~., perc.over = 300, train, k = 15,perc.under = 189**

**ORIGINAL TRAIN DATA : TRANSPORT FREQUENCY TABLE –**

***PROPORTIONS TO TOTAL 86:14***

0 1

269 43

**SMOTE BALANCED TRAIN DATA : TRANSPORT FREQUENCY TABLE - PROPORTIONS TO TOTAL 59:42**

0 1

243 172

**MODEL BUILDING – LOGISTIC REGRESSION**

**KEY STEPS**

* **BASE MODEL -1 :** Build Logistic regression using all independent variables as per **Original dataset (without the new variables and without data balancing (SMOTE))**
* **MODEL- 2 :** Build Logistic regression using independent variables as per new dataset (data treated for multicollinearity using PCA and SMOTE for data balancing)
* Perform stepwise logistic regression on our new dataset using both method to arrive at best fit model
* FINAL MODEL - Note the important independent variables in the above step and create a final model using those variables

**BASE MODEL – ORIGINAL DATA**

Call:

glm(formula = Transport2 ~ ., family = binomial, data = train2)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.84996 -0.06285 -0.01217 -0.00151 2.78130

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -45.3817 10.6442 -4.264 2.01e-05 \*\*\*

Age 1.0220 0.2897 3.528 0.000419 \*\*\*

GenderMale -0.9162 0.8493 -1.079 0.280690

Engineer1 0.6564 1.0683 0.614 0.538887

MBA1 -3.0982 1.0551 -2.936 0.003320 \*\*

Work.Exp -0.2244 0.2031 -1.105 0.269098

Salary 0.3386 0.1540 2.199 0.027878 \*

Distance 0.6407 0.1697 3.777 0.000159 \*\*\*

license1 3.3114 0.8743 3.787 0.000152 \*\*\*

---

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

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 250.216 on 311 degrees of freedom

Residual deviance: 55.812 on 303 degrees of freedom

AIC: 73.812

Number of Fisher Scoring iterations: 9

**VIF**

Age Gender Engineer MBA Work.Exp Salary Distance license

4.236096 1.254320 1.225036 1.739918 3.800168 1.822782 2.095208 1.517915

* Many insignificant variables present in our base model – Gender, Engineer and work.exp
* VIF value of 4 and above indicates the presence of multicollinearity in our base model – in our case Age and work.exp have high VIF

**MODEL 2 – USING PCA AND SMOTE DERIVED DATA**

Call:

glm(formula = Transport2 ~ ., family = binomial, data = train.new)

Deviance Residuals:

Min 1Q Median 3Q Max

-2.7628 -0.2084 -0.0482 0.1931 3.1945

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -6.76460 1.16365 -5.813 6.13e-09 \*\*\*

GenderMale -0.60318 0.48489 -1.244 0.214

Distance 0.32689 0.08146 4.013 6.00e-05 \*\*\*

license1 2.22005 0.44909 4.943 7.68e-07 \*\*\*

education2 -0.05442 0.52170 -0.104 0.917

education3 -0.40269 0.54745 -0.736 0.462

SEC 2.79907 0.33271 8.413 < 2e-16 \*\*\*

---

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

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 563.11 on 414 degrees of freedom

Residual deviance: 161.30 on 408 degrees of freedom

AIC: 175.3

Number of Fisher Scoring iterations: 7

**VIF**

GVIF Df GVIF^(1/(2\*Df))

Gender 1.106520 1 1.051913

Distance 1.060899 1 1.029999

license 1.126759 1 1.061489

education 1.080558 2 1.019558

SEC 1.066589 1 1.032758

fitting null model for pseudo-r2

llh llhNull G2 McFadden r2ML r2CU

-80.6489801 -281.5526052 401.8072502 **0.7135563** 0.6202379 0.8352954

* Even in our second model we notice presence of insignificant variables – Gender and Education
* However VIF values for all variables are below 4 indicating that multicollinearity is not present in the model

**STEPWISE LOGISTIC REGRESSION TO ARRIVE AT BEST FIT MODEL**

Stepwise Selection Method

-------------------------

Candidate Terms:

1 . Gender

2 . Distance

3 . license

4 . education

5 . SEC

Step 0: AIC = 565.1052

Transport2 ~ 1

**Enter New Variables**

-------------------------------------------------

Variable DF AIC BIC Deviance

-------------------------------------------------

SEC 1 211.184 219.241 207.184

license 1 420.298 428.354 416.298

Distance 1 458.135 466.191 454.135

education 1 557.896 569.981 551.896

Gender 1 564.229 572.286 560.229

-------------------------------------------------

✔ **SEC**

Step 1 : AIC = 211.1843

Transport2 ~ SEC

✔ **license**

Step 2 : AIC = 188.1701

Transport2 ~ SEC + license

✔ **Distance**

**Step 3 : AIC = 171.6481**

**Transport2 ~ SEC + license + Distance**

Final Model Output

------------------

Maximum Likelihood Estimates

-----------------------------------------------------------------

Parameter DF Estimate Std. Error z value Pr(>|z|)

-----------------------------------------------------------------

(Intercept) 1 -7.0523 1.0355 -6.8103 0.0000

SEC 1 2.9267 0.3646 8.0262 0.0000

Distance 1 0.3347 0.0783 4.2729 0.0000

license1 1 1.4450 0.4248 3.4013 7e-04

-----------------------------------------------------------------

**FINAL MODEL**

* Basis the above step, we include only SEC, Distance and License as predictor variables to arrive at our final model in multicollinearity and smote treated data.

**call:**

**glm(formula = Transport2 ~ license + SEC + Distance, family = binomial,**

**data = train.new)**

Deviance Residuals:

Min 1Q Median 3Q Max

-2.87561 -0.21615 -0.04735 0.19099 3.12900

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -7.08782 1.10052 -6.440 1.19e-10 \*\*\*

license1 2.06627 0.42826 4.825 1.40e-06 \*\*\*

SEC 2.74692 0.32526 8.445 < 2e-16 \*\*\*

Distance 0.31786 0.08011 3.968 7.25e-05 \*\*\*

---

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

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 563.11 on 414 degrees of freedom

Residual deviance: 163.65 on 411 degrees of freedom

AIC: 171.65

Number of Fisher Scoring iterations: 7

**VIF**

license SEC Distance

1.042134 1.026079 1.023055

fitting null model for pseudo-r2

llh llhNull G2 McFadden r2ML r2CU

-81.8240575 -281.5526052 399.4570954 0.7093827 0.6180812 0.8323909

**INTERPRETATION OF FINAL MODEL**

* All variables are significant with a p value much lower than 0.05 in final model
* Although \*AIC value is more than our base model, but our final model does not suffer from multicollinearity as indicated by VIF values of both models. (VIF) gauges – how much the variance of regression coefficient is inflated due to multicolinearity. hence we will go ahead with this model
* Also our final model \*mcfadden R square\* is 0.70 indicating it explains 70% of variation in our dependent variable, which suggests it’s a robust model

1. \*AIC The Akaike information criterion (AIC) is a measure of the relative quality of statistical models for a given set of data. Given a collection of models for the data, AIC estimates the quality of each model, relative to each of the other models. Given a set of candidate models for the data, the preferred model is the one with the minimum AIC value. AIC rewards goodness of fit (as assessed by the likelihood function), but it also includes a penalty that is an increasing function of the number of estimated parameters.
2. \* McFadden pseudo R-square: McFadden pseudo-R Squared: Logistic regression models are fitted using the method of maximum likelihood - i.e. the parameter estimates are those values which maximize the likelihood of the data which have been observed. McFadden's R squared measure is defined as (1-LogLc/LogLnull) where Lc denotes the (maximized) likelihood value from the current fitted model, and Lnull denotes the corresponding value but for the null model - the model with only an intercept and no covariates.

**ODDS RATIO**

(Intercept) license1 SEC Distance

8.352124e-04 7.895301e+00 1.559448e+01 1.374189e+00

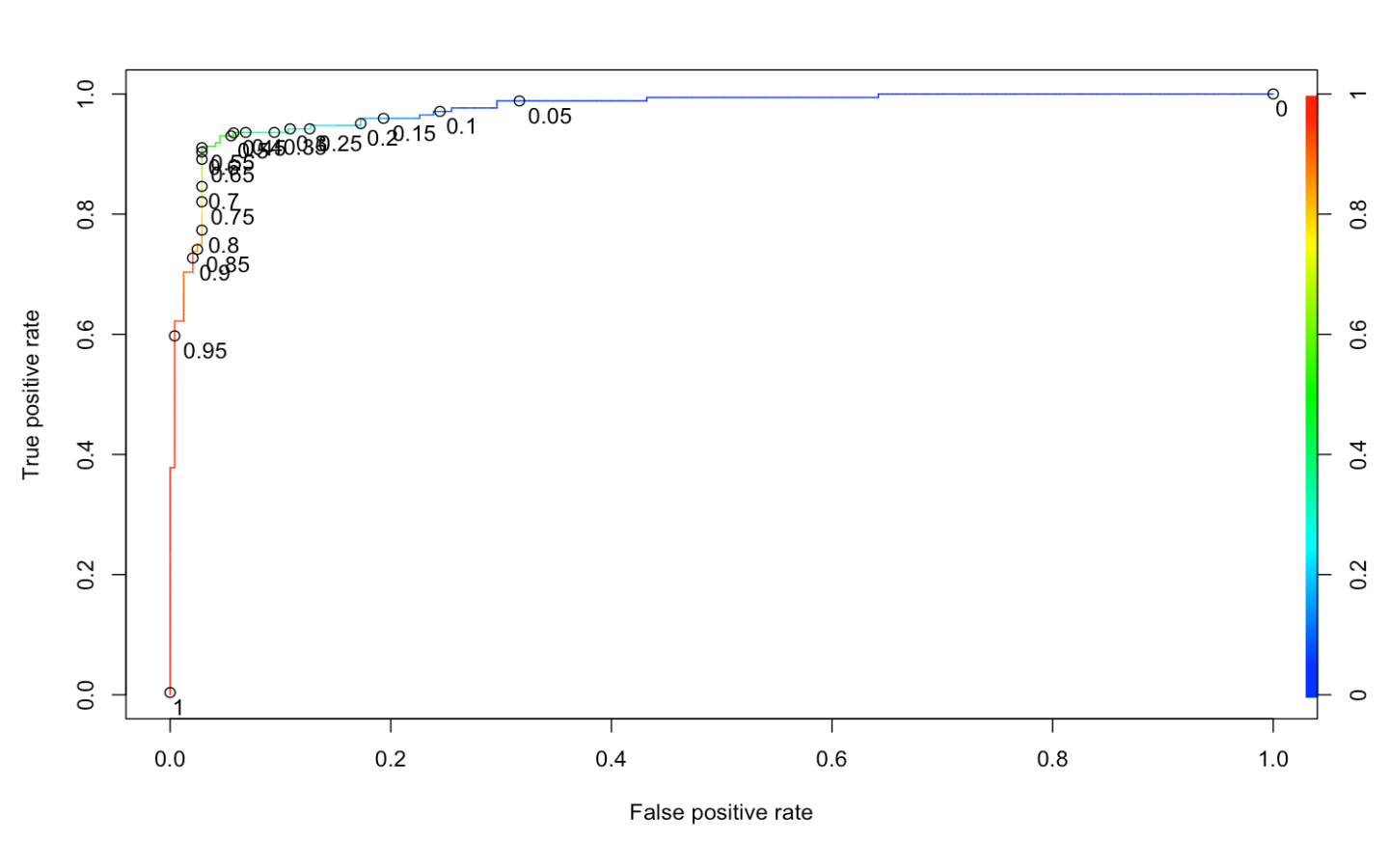
**PROBABILITY**

(Intercept) license1 SEC Distance

0.0008345154 0.8875810989 0.9397390110 0.5788035658

* The odds ratio and probability are helpful in explaining the impact of each independent variable on the dependent variable . For example if distance is increased by 1 KM then probability of mode of transport to be car will be increased by 57.8%

**ROC**



**AUC -** AUC of a classifier is equal to the probability that the classifier will rank a randomly chosen positive data point higher than a randomly chosen negative data point. Higher the probability better is the classifier.

0.9739449

**KS -** This performance measure is defined as maximum difference between TPR and FPR. Higher KS stat value indicates better model.

0.8849651

**GINI -** Gini coefficient is a ratio of two areas:

•  the area between the ROC curve and the random model line

•  top left triangle above the random model line – which is just 0.5

It can also be simplified as: (2 \* AUC – 1)

0.9478898

**CONFUSION MATRIX AND PERFORMANCE EVALUATION BETWEEN TRAIN AND TEST DATA FOR OUR FINAL MODEL**

Test data

Confusion Matrix and Statistics

Reference

Prediction 0 1

0 106 1

1 8 17

Accuracy : 0.9318

95% CI : (0.8745, 0.9684)

No Information Rate : 0.8636

P-Value [Acc > NIR] : 0.0106

Kappa : 0.7513

Mcnemar's Test P-Value : 0.0455

Sensitivity : 0.9444

Specificity : 0.9298

Pos Pred Value : 0.6800

Neg Pred Value : 0.9907

Prevalence : 0.1364

Detection Rate : 0.1288

Detection Prevalence : 0.1894

Balanced Accuracy : 0.9371

Train data

Confusion Matrix and Statistics

Reference

Prediction 0 1

0 230 12

1 13 160

Accuracy : 0.9398

95% CI : (0.9124, 0.9606)

No Information Rate : 0.5855

P-Value [Acc > NIR] : <2e-16

Kappa : 0.876

Mcnemar's Test P-Value : 1

Sensitivity : 0.9302

Specificity : 0.9465

Pos Pred Value : 0.9249

Neg Pred Value : 0.9504

Prevalence : 0.4145

Detection Rate : 0.3855

Detection Prevalence : 0.4169

Balanced Accuracy : 0.9384

**DEFINING THE ABOVE EVALUATION MEASURES**

* **Sensitivity** - Total no correct predictions of 1 out of total predictions of 1
* **Specificity** - Total no correct predictions of 0 out of total predictions of o
* **Accuracy** - Rati0 of correct predictions to total observations

**INTERPRETING THE ABOVE RESULTS**

* Accuracy of 93.9% for Train and 93.1% for Test Model indicates a good fit model. Even sensitivity and specificity are a good value at around 93%
* Sensitivity, Specificity, Accuracy, are all with in 10% difference for Test and Train data indicating a valid model

**MODEL BUILDING – KNN**

KNN is supervised classifier, which uses neighbor data points’ information to predict outcome variable. Neighbors are identified using distance measures such as Euclidean distance.

**KEY STEPS**

* Normalise all continuous variables, so that they are measured on similar scale to avoid scale bias in the model
* Run knn model
* Identify the k value (no of nearest neighbour cases/observations to target variable) which maximises accuracy or any other evaluation measure(for eg sensitivity, specificity )of interest, we chose accuracy in our case
* In our case k =5 gives maximum accuracy
* Check evaluation measures on Train and Test

**NORMALIZING CONTINUOUS VARIABLES**

Distance metric is highly influenced by the scale of the variable. Hence, it is important to standardize variables before utilizing them in model building. We will use min-max standardization method to bring all variables in same scale as below

scale = preProcess(train.new, method = "range")

train.norm.data = predict(scale, train.new)

test.norm.data = predict(scale, test)

**KNN Model**

* We will use train function from caret package to arrive at best fit KNN model

knn\_fit = train(Transport2 ~., data = train.norm.data, method = "knn",

trControl = trainControl(method = "cv", number = 3),

tuneLength = 10)

415 samples

5 predictor

2 classes: '0', '1'

No pre-processing

Resampling: Cross-Validated (3 fold)

Summary of sample sizes: 277, 276, 277

Resampling results across tuning parameters:

k Accuracy Kappa

5 0.8964481 0.7863694

7 0.8941021 0.7813981

9 0.8796441 0.7509169

11 0.8603204 0.7087936

13 0.8554895 0.6994633

15 0.8434817 0.6737669

17 0.8386508 0.6639199

19 0.8362354 0.6592157

21 0.8362354 0.6596208

23 0.8265388 0.6399846

Accuracy was used to select the optimal model using the largest value.

The final value used for the model was k = 5.

**CONFUSION MATRIX AND PERFORMANCE EVALUATION BETWEEN TRAIN AND TEST DATA FOR OUR KNN MODEL**

Model Evaluation – Test set

Confusion Matrix and Statistics

Reference

Prediction 0 1

0 103 2

1 11 16

Accuracy : 0.9015

95% CI : (0.8375, 0.9465)

No Information Rate : 0.8636

P-Value [Acc > NIR] : 0.1244

Kappa : 0.6546

Mcnemar's Test P-Value : 0.0265

Sensitivity : 0.8889

Specificity : 0.9035

Pos Pred Value : 0.5926

Neg Pred Value : 0.9810

Prevalence : 0.1364

Detection Rate : 0.1212

Detection Prevalence : 0.2045

Balanced Accuracy : 0.8962

Model Evaluation – Train set

Confusion Matrix and Statistics

Reference

Prediction 0 1

0 226 20

1 17 152

Accuracy : 0.9108

95% CI : (0.8792, 0.9364)

No Information Rate : 0.5855

P-Value [Acc > NIR] : <2e-16

Kappa : 0.8158

Mcnemar's Test P-Value : 0.7423

Sensitivity : 0.8837

Specificity : 0.9300

Pos Pred Value : 0.8994

Neg Pred Value : 0.9187

Prevalence : 0.4145

Detection Rate : 0.3663

Detection Prevalence : 0.4072

Balanced Accuracy : 0.9069

* Accuracy of 91% for Train and 90% for Test Model indicates a good fit model. However sensitivity and specificity are lower then logistic regression model
* Sensitivity, Specificity, Accuracy are all with in 10% difference for Test and Train data indicating a valid model

**Model Building – Naïve Bayes**

The e1071 package holds the naiveBayes function. It allows continuous and categorical features to be used in the naive bayes model. It is count-based classifier i.e. only thing it does is – count how often each variable’s distinct values occur for each class.

**WHETHER NAÏVE BAYES IS APPLICABLE HERE**

1. We have 5 independent variables out of which 3 are categorical and rest numeric, while Naïve Bayes works best with only categorical independent variables, it also accommodates the continuous /numerical independent variables , hence it is applicable in our case

**NAÏVE BAYES MODEL**

**Prior probabilities and conditional probabilities**

naiveBayes.default(x = train.norm.data[-c(4)], y = train.norm.data$Transport2)

Naive Bayes Classifier for Discrete Predictors

Call:

naiveBayes.default(x = train.norm.data[-c(4)], y = train.norm.data$Transport2)

A-priori probabilities:

train.norm.data$Transport2

0 1

0.5855422 0.4144578

Conditional probabilities:

Gender

train.norm.data$Transport2 Female Male

0 0.2757202 0.7242798

1 0.2267442 **0.7732558**

Distance

train.norm.data$Transport2 [,1] [,2]

0 0.4407283 0.1835037

1 **0.6801588** 0.1739603

license

train.norm.data$Transport2 0 1

0 **0.8271605** 0.1728395

1 0.2441860 **0.7558140**

education

train.norm.data$Transport2 1 2 3

0 0.4979424 0.2633745 0.2386831

1 **0.5465116** 0.1802326 0.2732558

SEC

train.norm.data$Transport2 [,1] [,2]

0 0.3725908 0.1871958

1 **0.7768932** 0.1153179

**CONFUSION MATRIX AND PERFORMANCE EVALUATION BETWEEN TRAIN AND TEST DATA FOR OUR NAÏVE BAYES MODEL**

Train Evaluation Measures

Confusion Matrix and Statistics

Reference

Prediction 0 1

0 233 17

1 10 155

Accuracy : 0.9349

95% CI : (0.9067, 0.9567)

No Information Rate : 0.5855

P-Value [Acc > NIR] : <2e-16

Kappa : 0.8652

Mcnemar's Test P-Value : 0.2482

Sensitivity : 0.9012

Specificity : 0.9588

Pos Pred Value : 0.9394

Neg Pred Value : 0.9320

Prevalence : 0.4145

Detection Rate : 0.3735

Detection Prevalence : 0.3976

Balanced Accuracy : 0.9300

'Positive' Class : 1

Test Evaluation measures

Confusion Matrix and Statistics

Reference

Prediction 0 1

0 106 2

1 8 16

Accuracy : 0.9242

95% CI : (0.8651, 0.9631)

No Information Rate : 0.8636

P-Value [Acc > NIR] : 0.02244

Kappa : 0.7179

Mcnemar's Test P-Value : 0.11385

Sensitivity : 0.8889

Specificity : 0.9298

Pos Pred Value : 0.6667

Neg Pred Value : 0.9815

Prevalence : 0.1364

Detection Rate : 0.1212

Detection Prevalence : 0.1818

Balanced Accuracy : 0.9094

'Positive' Class : 1

* Accuracy of 93% for Train and 92% for Test Model indicates a good fit model. However sensitivity IS lower thAn logistic regression model
* Sensitivity, Specificity, Accuracy are all with in 10% difference for Test and Train data indicating a valid model

**MODEL COMPARISON – LOGISTIC REGRESSION, KNN, NAÏVE BAYES**

* We will **compare Accuracy, Sensitivity and Specificity performance on Test data for our 3 models** to arrive at best fit model
* Since our objective is to identify employees who will use “car” as a mode of transport, we need to give more emphasis on improving sensitivity while balancing on accuracy and specificity measures.
* Hence, we will not just evaluate models based on accuracy on test data, we will also use sensitivity as metric to compare model performances

|  |  |  |  |
| --- | --- | --- | --- |
| **Metric** | **Logistic Regression** | **Naïve Bayes** | **KNN** |
| Accuracy | 0.9318 | 0.9242 | 0.9015 |
| Sensitivity | 0.9444 | 0.8889 | 0.8889 |
| Specificity | 0.9298 | 0.9298 | 0.9035 |

* From the above table, we can infer that Logistic Regression is the best fit model, which has performed the best on all 3 measures of importance in our test data

**MODEL BUILDING – BAGGING**

Bagging (aka Bootstrap Aggregating): is a way to decrease the variance of your prediction by generating additional data for training from your original dataset using combinations with repetitions to produce multisets of the same cardinality/size as your original data. Bagging is in parallel.

**MODEL**

bagging.car <- bagging(Transport2 ~.,

data=train.new,

control=rpart.control(maxdepth=5, minsplit=4))

**minsplit**

The minimum number of observations that must exist in a node, in order for a split to be attempted.

**maxdepth**

The maximum depth of any node of the final tree, with the root node counted as depth 0

**CONFUSION MATRIX AND PERFORMANCE EVALUATION BETWEEN TRAIN AND TEST DATA FOR OUR BAGGING MODEL**

Test

Confusion Matrix and Statistics

Reference

Prediction 0 1

0 110 1

1 4 17

Accuracy : 0.9621

95% CI : (0.9138, 0.9876)

No Information Rate : 0.8636

P-Value [Acc > NIR] : 0.000156

Kappa : 0.8497

Mcnemar's Test P-Value : 0.371093

Sensitivity : 0.9444

Specificity : 0.9649

Pos Pred Value : 0.8095

Neg Pred Value : 0.9910

Prevalence : 0.1364

Detection Rate : 0.1288

Detection Prevalence : 0.1591

Balanced Accuracy : 0.9547

Train

Confusion Matrix and Statistics

Reference

Prediction 0 1

0 234 9

1 9 163

Accuracy : 0.9566

95% CI : (0.9323, 0.9741)

No Information Rate : 0.5855

P-Value [Acc > NIR] : <2e-16

Kappa : 0.9106

Mcnemar's Test P-Value : 1

Sensitivity : 0.9477

Specificity : 0.9630

Pos Pred Value : 0.9477

Neg Pred Value : 0.9630

Prevalence : 0.4145

Detection Rate : 0.3928

Detection Prevalence : 0.4145

Balanced Accuracy : 0.9553

'Positive' Class : 1

* High Accuracy on both Train and Test dataset
* Even sensitivity and specificity are higher compared to our logistic regression model
* Best fit model till now

**MODEL BUILDING – BOOSTING**

Boosting is the idea of training the weak learners sequentially

There are various boosting methods as below

* AdaBoost (Adaptive Boosting) – building on weak learners combining decision stumps and weighting incorrect observations.
* Gradient Boosting – builds on each model, trying to fit the next model to the residuals of the previous model
* XGBoost (Extreme Gradient Boosting) – a specialized implementation of gradient bosting decision trees designed for performance. Three main types are: gradient boosting, stochastic gradient boosting and regularized gradient boosting

**KEY STEPS**

* We will build a XG boost model,
* Since XGBoost works with matrices that contain all numeric variables, we will convert our dataset into numeric matrices
* we also need to split the training data and the label of our dependent variable as matrices
* Arguments used:
  + ETA -  step size of each boosting step or shrinkage
  + max\_depth- Larger the depth, more complex the model; higher chances of overfitting. Larger data sets require deep trees to learn the rules from data.
  + min\_child\_weight- it blocks the potential feature interactions to prevent overfitting
  + Nrounds - Controls the maximum number of iterations. For classification, it is similar to the number of trees to grow.
  + early\_stopping\_rounds - Stop if no improvement for N consecutive trees

**XGB MODEL**

xgb.fit <- xgboost(

data = gd\_features\_train,

label = gd\_label\_train,

eta = 0.001,

max\_depth = 3,

min\_child\_weight = 3,

nrounds = 10,

nfold = 5,

objective = "binary:logistic", # for regression models

verbose = 0, # silent,

early\_stopping\_rounds = 10 # stop if no improvement for 10 consecutive trees

**)**

**CONFUSION MATRIX AND PERFORMANCE EVALUATION BETWEEN TRAIN AND TEST DATA FOR OUR BAGGING MODEL**

TEST

Confusion Matrix and Statistics

Reference

Prediction 0 1

0 102 2

1 12 16

Accuracy : 0.8939

95% CI : (0.8285, 0.9408)

No Information Rate : 0.8636

P-Value [Acc > NIR] : 0.18897

Kappa : 0.6351

Mcnemar's Test P-Value : 0.01616

Sensitivity : 0.8889

Specificity : 0.8947

Pos Pred Value : 0.5714

Neg Pred Value : 0.9808

Prevalence : 0.1364

Detection Rate : 0.1212

Detection Prevalence : 0.2121

Balanced Accuracy : 0.8918

TRAIN

Confusion Matrix and Statistics

Reference

Prediction 0 1

0 229 14

1 14 158

Accuracy : 0.9325

95% CI : (0.904, 0.9547)

No Information Rate : 0.5855

P-Value [Acc > NIR] : <2e-16

Kappa : 0.861

Mcnemar's Test P-Value : 1

Sensitivity : 0.9186

Specificity : 0.9424

Pos Pred Value : 0.9186

Neg Pred Value : 0.9424

Prevalence : 0.4145

Detection Rate : 0.3807

Detection Prevalence : 0.4145

Balanced Accuracy : 0.9305

* High Accuracy, Sensitivity and Specificity for Train model, however they decline for test set
* All performance measures are lower when compared to bagging model

**CHOOSING THE BEST FIT MODEL**

* We will **compare Accuracy, Sensitivity and Specificity performance on Test data for our 5 models** to arrive at best fit model
* Since our objective is to identify employees who will use “car” as a mode of transport, we need to give more emphasis on improving sensitivity while balancing on accuracy and specificity measures.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Metric | Logistic Regression | Naïve Bayes | KNN | Bagging | Boosting |
| Accuracy | 0.9318 | 0.9242 | 0.9015 | 0.9621 | 0.8939 |
| Sensitivity | 0.9444 | 0.8889 | 0.8889 | 0.9444 | 0.8889 |
| Specificity | 0.9298 | 0.9298 | 0.9035 | 0.9649 | 0.8947 |

* Bagging has improved accuracy and specificity while maintaining sensitivity from our previous best model – Logistic regression
* Hence we can go ahead with Bagging as final model

**INSIGHTS FROM ANALYSIS**

* While we can use Bagging model to predict whether an employee will use car or not as a mode of transport, we can also look at results from logistic regression model to understand which are the significant variables in determining the same
* For the data provided for our assignment, whether an employee will use “CAR” as a mode of transport is significantly affected by following variables:
  + - Distance : If distance is increased by 1 KM then probability of mode of transport to be car will be increased by 57.8%.
    - License : If the employee does not have a license then the probability of mode of transport to be car will be reduced by 88.8%.
    - SEC – combination factor of Age, work experience and Salary , any change in this variable has the highest impact on probability of using “CAR” as mode of transport
  + Other important variables are engineers who are more likely to use car as well as gender Male who are more likely to use car as mode of transport

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_