A PROJECT REPORT ON

PREDICTING THE SALARIES OF PROFESSORS

Submitted to Osmania University
in partial fulfilment of the requirements for the award of

MASTER OF SCIENCE IN STATISTICS

DEPARTMENT OF STATISTICS
UNIVERSITY COLLEGE OF SCIENCE

OSMANIA UNIVERSITY HYDERABAD – INDIA

By

M.PRATHIMA	**Roll No:1007-17-507-041**
S.SHRAVANI	**Roll No:1007-17-507-017**
M.LIKITHA	**Roll No:1007-17-507-034**
P.MADHURI	**Roll No:1007-17-507-016**
P.LAVANYA	**Roll No:1007-17-507-042**
B.MADHAN MOHAN	**Roll No:1007-17-507-033**
R.DIVYA	**Roll No:1007-17-507-025**

Under the Supervision of

Dr. M. VENUGOPALA RAO

2018

A PROJECT REPORT ON

PREDICTING THE SALARIES OF PROFESSORS

Submitted to Osmania University
in partial fulfilment of the requirements for the award of

MASTER OF SCIENCE IN STATISTICS

DEPARTMENT OF STATISTICS
UNIVERSITY COLLEGE OF SCIENCE

OSMANIA UNIVERSITY HYDERABAD – INDIA

By

M.PRATHIMA	**Roll No:1007-17-507-041**
S.SHRAVANI	**Roll No:1007-17-507-017**
M.LIKITHA	**Roll No:1007-17-507-034**
P.MADHURI	**Roll No:1007-17-507-016**
P.LAVANYA	**Roll No:1007-17-507-042**
B.MADHAN MOHAN	**Roll No:1007-17-507-033**
R.DIVYA	**Roll No:1007-17-507-025**

Under the Supervision of

Dr. M. VENUGOPALA RAO

2018

CERTIFICATE

This is to certify that

Ms. M.PRATHIMA	**Roll No:1007-17-507-041**
Ms. S.SHRAVANI	**Roll No:1007-17-507-017**
Ms. M.LIKITHA	**Roll No:1007-17-507-034**
Ms. P.MADHURI	**Roll No:1007-17-507-016**
Ms. P.LAVANYA	**Roll No:1007-17-507-042**
Mr. B.MADHAN MOHAN	**Roll No:1007-17-507-033**
Ms. R.DIVYA	**Roll No:1007-17-507-025**

have submitted the project titled **"PREDICTING THE SALARIES OF PROFESSOR'S"** in partial fulfilment for the degree of Master of Science in Statistics.

Head

Department of Statistics

Internal Examiner External Examiner

DECLARATION

The research presented in this project has been carried out in the **Department of Statistics, Osmania University, Hyderabad.** The work is original has not been submitted so far, in part or full, for any other degree of diploma of any university.

M.PRATHIMA

S.SHRAVANI

M.LIKITHA

P.MADHURI

P.LAVANYA

B.MADHAN MOHAN

R.DIVYA

Department of Statistics

Osmania University

Hyderabad – 500 007, T.S.

INDIA

ACKNOWLEDGEMENTS

I deem it a great pleasure to express my deep sense of gratitude and indebtedness to my research supervisor **Dr. M. VENUGOPALA RAO,** Statistics department, University College of Science, Osmania University for his valuable guidance, and enlightening discussions throughout the progress of my project work.

I also express my sincere and heartfelt thanks to **PROF.C.JAYALAKSHMI ,**Head of Department, Department of Statistics, Osmania University for providing the necessary support and facilities in the department for completion of this work successfully.

It is indeed with great pleasure I record my thanks to **Dr.G.JAYASREE ,** Chairperson, Board of Studies , Department of Statistics , Osmania University for having provided with all the facilities to carry out our work.

I thank **Dr.N.Ch.BHATRACHARYULU** , **Dr.K.VANI , Dr.S.A.JYOTHI RANI , Dr.G.SIRISHA , Mrs.J.L.PADMA SHREE** for their encouragement and constant help during the research.

I would like to express my deepest gratitude to **T.SANDHYA, BALA KARTHIK** for their advice, guidance and involvement at various stages of this work , I would also like to thank them for their understanding and constant encouragement throughout this project.

I thank all Non-Teaching members of the Department of Statistics, who helped me during my Thesis work.

I am thankful to the Osmania University for permitting me to carry out this work

CONTENTS

Page No.

Chapter - I

Introduction

INTRODUCTION

1.1 Scope of the problem:

This problem is related to the financial and academic decision area.

One of the leading university in united states desired to ascertain

- What are parameters to evaluate salaries.
- What are the characteristics taken to evaluate the academic 9 months salaries for professors (Assistant professors, Associate professors, professors).
- An algorithm, by which we predict new respondent's salary .

Source:

We have extracted the data from the below website https://vincentarelbundock.github.io/Rdatasets/doc/carData/Salaries.html

1.2 Description of data:

Prediction of Salary for Professor's:

The 2008-09 nine-month academic salary for Assistant Professors, Associate Professors and Professors in a college in the US. The data were collected as the part of the on-going effort of the college's administration to monitor salary between male and female faculty members.

▶ **Usage:**

Salaries

▶ **Format:**

A data frame with 200 observations on the following 6 variables.

- **Rank**

  a factor with 3 levels

  1.Prof

  2.AssocProf

  3.AsstProf

- **Discipline**

  a factor with 2 levels

  - A ("Theoretical departments")
  - B ("Applied department").

- **Yrs.since.phd**

  years since PhD.

- **Yrs. service**

  years of service.

- **Sex**

  a factor with 2 levels

  - female (1)
  - male (0)

- **Salary**

  nine-month salary, in dollars.

1.3 Review of chapters:

Chapter 2 gives the brief introduction about machine learning techniques like need of ML today, types of ML Algorithms and various models in each algorithm and what technique to use when and how to validate, Tune the ML algorithms and how to measure the performance of the ML model

Chapter 3 describes the various results obtained for the problem. This section contains all the outputs generated through the ML algorithms applied on the data as well as validation and performance matrices

Chapter 4 describes the summary and conclusions followed by Bibliography.

Chapter-2

Review of Machine Learning Process

2.0 Need of Machine Learning

In this age of modern technology, there is one resource that we have in abundance: a large amount of structured and unstructured data. In the second half of the twentieth century, machine learning evolved as a subfield of artificial intelligence that involved the development of self-learning algorithms to gain knowledge from that data in order to make predictions. Instead of requiring humans to manually derive rules and build models from analysing large amounts of data, machine learning offers a more efficient alternative for capturing the knowledge in data to gradually improve the performance of predictive models, and make data-driven decisions. Not only is machine learning becoming increasingly important in computer science research but it also plays an ever greater role in our everyday life.

2.1 Machine Learning Process

The CRISP-DM (Cross-Industry Standard Process for Data Mining) Process was designed specifically for the data mining. However, it is flexible and thorough enough that it can be applied to any analytical project whether it is predictive analytics, data science, or Machine learning. The Process has the following six phases

- Business Understanding
- Data Understanding
- Data preparation
- Modelling
- Evaluation
- Deployment

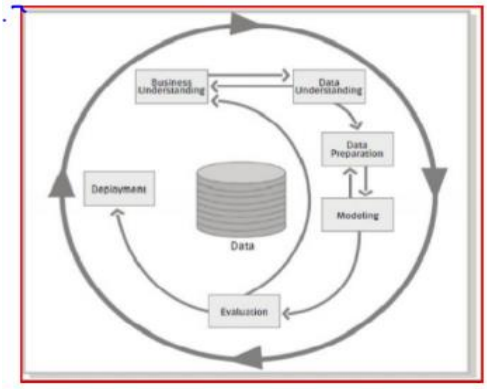


Fig 2.1 CRISP DM

And, each phase has different steps covering important tasks which are mentioned below:

2.1.1 Business Understanding

It is very important step of the process in achieving the success. The purpose of this step is to identify the requirements of the business so that you can translate them into analytical objectives. It has the following tasks:

- Identify the Business objective
- Assess the situation
- Determine the Analytical goals
- Produce a project plan

2.1.2 Data Understanding

After enduring the all-important pain of the first step, you can now get your hands on the data. The task in this process consist the following

- Collect the data
- Describe the data
- Explore the data

- Verify the data Quality

2.1.3 Data Preparation

This step is relatively self-explanatory and in this step the goal is to get the data ready to input in the algorithms. This includes merging, feature engineering, and transformations. If imputation for missing values / outliers is needed then, it happens in this step. The key five tasks under this step are as follows:

- Select the data
- Clean the data
- Construct the data
- Integrate the data
- Format the data

2.1.4 Modeling

Oddly, this process step includes the consideration that you already thought of and prepared for. In this, one will need atleast a modicum of an idea about how they will be modelling. Remember, that this is flexible, iterative process and some strict linear flow chart such as an aircrew checklist.

Below are the tasks in this step:

- Select a modelling technique
- Generate a test design
- Build a model
- Assess a Model

Both cross validation of the model (using train/test or K fold validation) and model assessment which involves comparing the models with the choosen criterion (RMSE, Accuracy, ROC) will be performed under this phase.

2.1.5 Evaluation

In the evaluation process, the main goal is to confirm that the work that has been done and the model selected at this point meets the business objective. Ask yourself and others, have we achieved the definition of success? And, here are the tasks in this step:

- Evaluate the results
- Review the process
- Determine the next steps

2.1.6 Deployment

If everything is done according to the plan up to this point, it might come down to flipping a switch and your model goes live. Here are the tasks in this step:

- Deploying the plan
- Monitoring and maintenance of the plan
- Producing the final report

2.2 Types of Machine Learning

Broadly, the Machine Learning Algorithms are classified into 3 types.

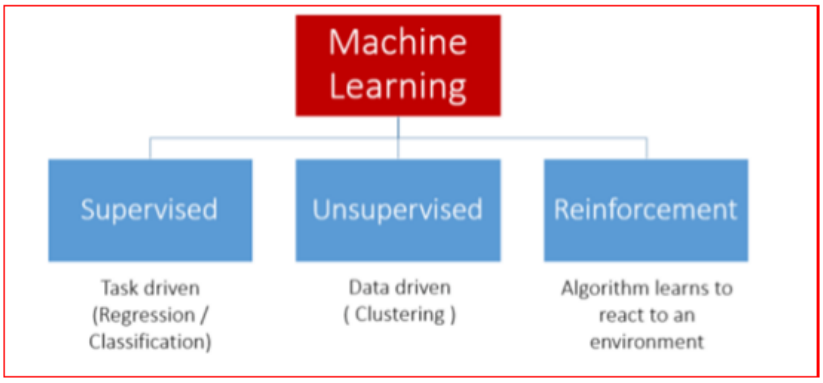


Fig 2.2 Types of machine learning.

2.2.1 Supervised Learning

This algorithm consists of a target / outcome / dependent variable which is to be predicted from a given set of predictors / independent variables. Using these set of variables, we generate a function that maps inputs to desired output. The training process continues until the model achieves a desired level of accuracy on the training data.

The process of Supervised Learning model is illustrated in the below picture:

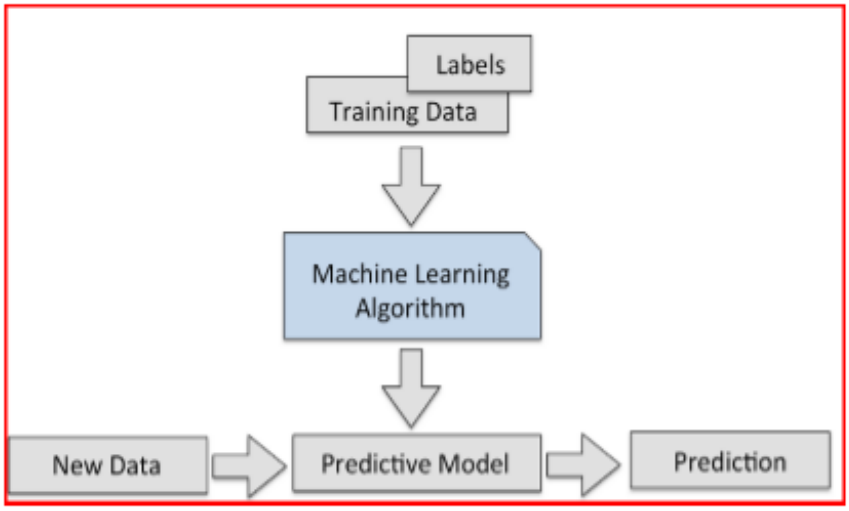


Fig 2.2.1 Supervised learning.

Examples of Supervised Learning: Regression, Decision Tree, Random Forest, KNN, Logistic Regression,…etc

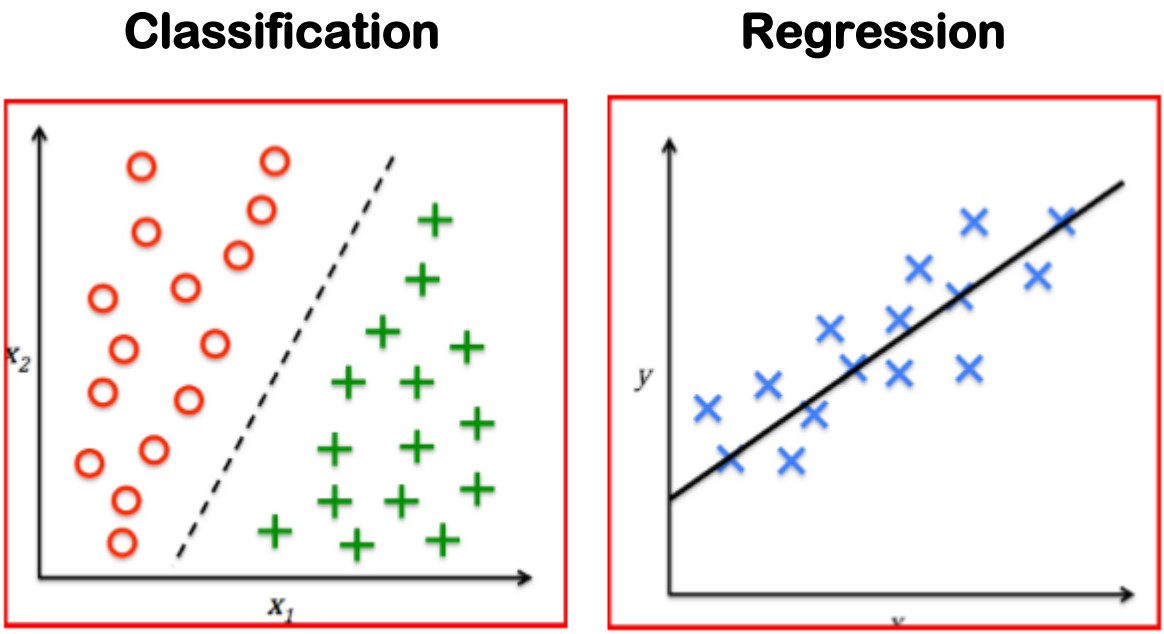


Fig 2.2.1(1) Classification.

Fig 2.2.1(2) Regression.

2.2.2 Unsupervised Learning

In this algorithm, we will not have any target or outcome variable to predict / estimate. It is used for clustering population into different groups, which is widely used for segmenting customers in different groups for specific intervention. (More of Exploratory Analysis)

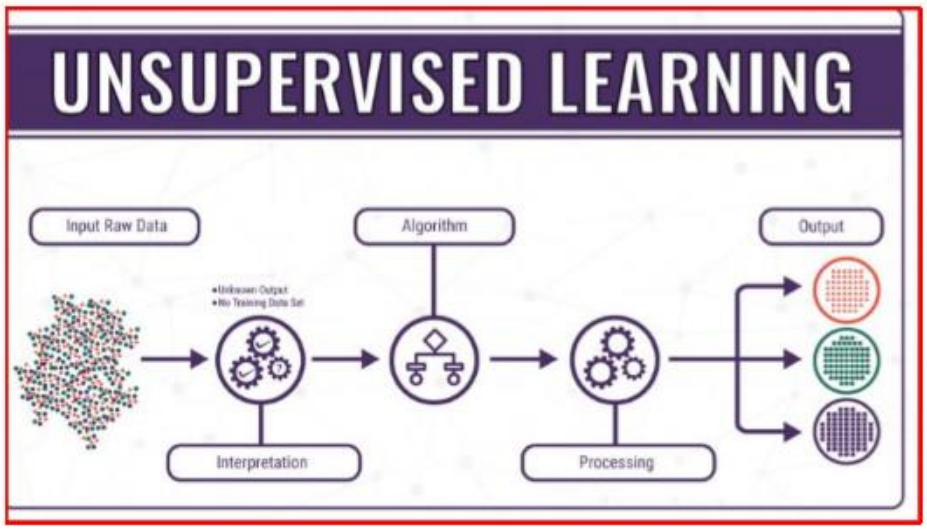
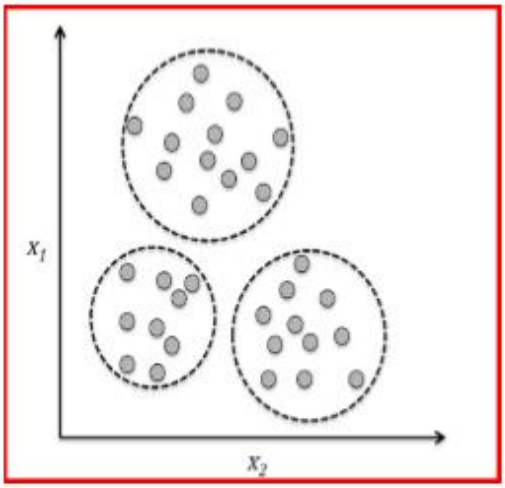


Fig 2.2.2 Unsupervised learning

Examples of Unsupervised Learning: Data reduction techniques, Cluster Analysis, Market Basket Analysis,…etc

Cluster Analysis **Data Reduction Techniques**



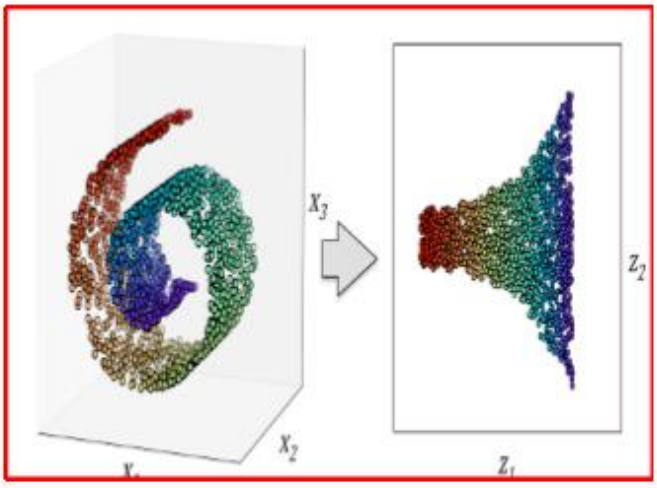


Fig 2.2.2(1) Cluster Analysis.

Fig 2.2.2(2) Data Reduction Techniques.

2.2.3 Reinforcement Learning

Using this algorithm, the machine is trained to make specific decisions.

It works this way: the machine is exposed to an environment where it trains itself continually using trial and error. This machine learns from past experience and tries to capture the best possible knowledge to make accurate business decisions.

The process of reinforcement learning is illustrated in the below picture:

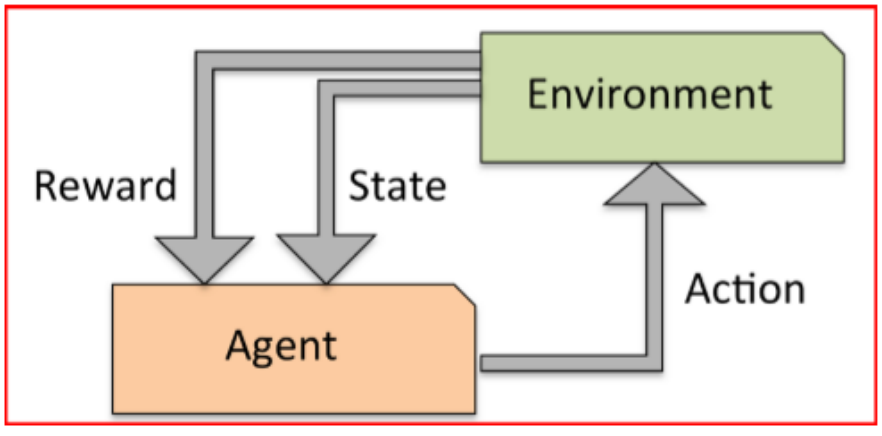


Fig 2.2.3 Reinforcement learning

Examples of Reinforcement Learning: Markov Decision Process, Self-driving cars,...etc

2.3 Choosing the algorithm

Choosing the right algorithm will depend on the type of the problem we are solving and also depends on the scale of the dependent variable. In case of continuous target variable, we will use regression algorithms and in case of categorical target, we will use classification algorithms and for the model which doesn't have target variable, we will use either cluster analysis / data reduction techniques.

Below picture describes the process of choosing the right algorithm:

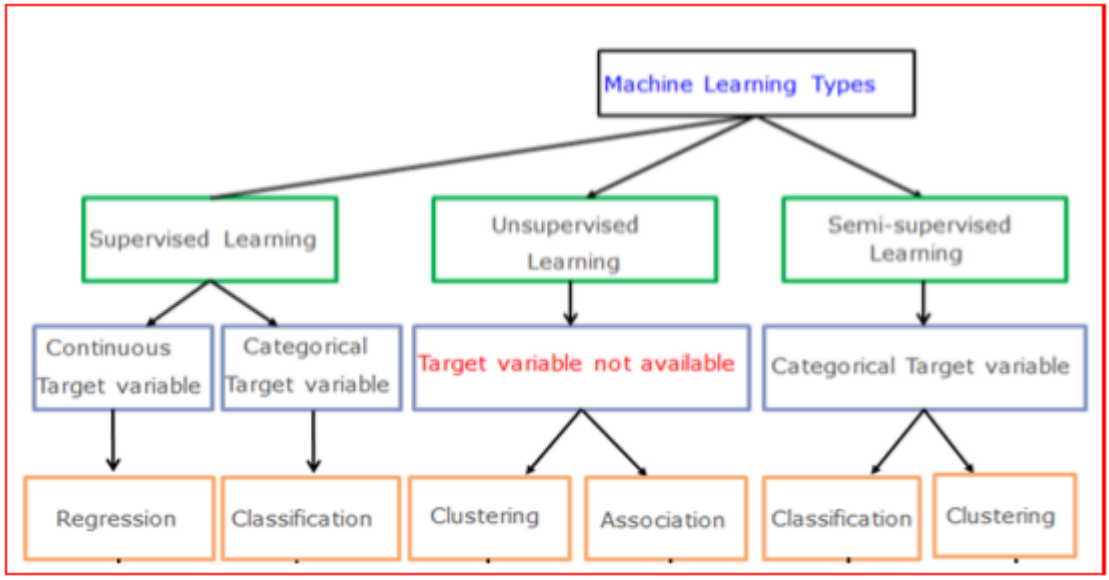


Fig 2.3 Algorithm.

2.3.1 Types of Regression Algorithms

There are many Regression algorithms in machine learning, which will be used in different regression applications. Some of the main regression algorithms are as follows:

A) Simple Linear Regression:-

In simple linear regression, we predict scores on one variable from the data of second variable. The variable we are forecasting is called the criterion variable and referred to as Y. The variable we are basing our predictions on is called the predictor variable and denoted as X.

B) Multiple Linear Regression:-

Multiple linear regression is one of the algorithms of regression technique, and is the most common form of linear regression analysis. As a predictive analysis, the multiple linear regression is used to explain the relationship between one dependent variable with two or more independent variables. The independent variables can be either continuous or categorical.

C) Polynomial Regression:-

Polynomial regression is another form of regression in which the maximum power of the independent variable is more than one. In this regression technique, the best fit line is not a straight line instead it is in the form of a curve.

D) Support Vector Machines:-

Support Vector Machines can be applied to regression problems as well as Classification. It contains all the features that characterises maximum margin algorithm. Linear learning machine maps a non-linear function into high dimensional kernel-induced feature space. The system capacity will be controlled by parameters that do not depend on the dimensionality of feature space.

E) Decision Tree Regression:-

Decision tree builds regression models in the form of a tree structure. It breaks down the data into smaller subsets and while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes.

F) Random Forest Regression:-

Random Forest is also one of the algorithms used in regression technique. It is very a flexible, easy to use machine learning algorithm that produces, even without hyper - parameter tuning, a great result most of the time. It is also one of the most widely used algorithms because of its simplicity and the fact that it can used for both regression and classification tasks. The forest it builds is an ensemble of Decision Trees, most of the time trained with the “bagging” method.

Other than these we have regularized regression models like Ridge, LASSO and Elastic Net regression which are used to select the key parameters and these is also Bayesian regression which works with the Bayes theorem.

2.3.2 Types of Classification Algorithms

There are many Classification algorithms in machine Learning, which can be used for different classification applications. Some of the main classification algorithms are as follows:

a) Logistic Regression/Classification:-Logistic regression falls under the category of supervised learning; it measures the relationship between the dependent variable which is categorical with one or more than one independent variables by estimating probabilities using a logistic/sigmoid function. Logistic regression can generally be used when the dependent variable is Binary or Dichotomous. It means that the dependent variable can take only two possible values like “Yes or No”, “Living or dead”.

b) K -Nearest Neighbours:- k-NN algorithm is one of the most straightforward algorithms in classification, and it is one of the most used ML algorithms. An object is classified by a majority vote of its neighbours, with the object being assigned to the class most common among its k nearest neighbours. It can also use for regression — output

is the value of the object (predicts continuous values). This value is the average (or median) of the values of its k nearest neighbours.

c) Naive Bayes:- Naive Bayes is a type of Classification technique based on Bayes’ theorem, with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a Particular feature in a class is unrelated to the presence of any other function. Naive Bayes

model is accessible to build and particularly useful for extensive datasets.

d) Decision Tree Classification:- Decision tree builds classification models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision node has two or more branches. Leaf node represents a classification or decision. The first decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data.

e) Support Vector Machines:- A Support Vector Machine is a type of Classifier, in which a discriminative classifier is formally defined by a separating hyperplane. The algorithm outputs an optimal hyperplane which categorises new examples. In two dimensional space, this hyperplane is a line dividing a plane in two parts where in each class lay in either side.

f) Random Forest Classification:- Random Forest is a supervised learning algorithm. It creates a forest and makes it somehow random. The forest it builds is an ensemble of Decision Trees, most of the times the decision tree algorithm trained with the "bagging" method. The general idea of the bagging method is that a combination of learning models increases the overall result. And Random Forest is also very powerful to find the variable importance in classification/ Regression problems.

2.3.3 Types of Unsupervised Learning

Clustering is the type of unsupervised learning in which an unlabelled data is used to draw inferences. It is the process of grouping similar entities together. The goal of this unsupervised machine learning technique is to find similarities in the data points and group similar data points together and also to figure out which cluster should a new data point belong to.

Types of Clustering Algorithms:- There are many Clustering algorithms in machine learning, which can be used for different clustering applications. Some of the main clustering algorithms are as follows:

a) Hierarchical Clustering:- Hierarchical clustering is one of the algorithms of clustering technique, in which similar data is grouped in a cluster. It is an algorithm that builds the hierarchy of clusters. This algorithm starts with all the data points assigned to a bunch of their own. Then, two nearest groups are merged into the same cluster. In the end, this algorithm terminates when there is only a single cluster left.

It starts by assigning each data point to its bunch. Finds the closest pair using Euclidean distance and merges them into one cluster. This process is continued until all data points are clustered into a single cluster.

b) K -Means Clustering:- K-Means clustering is one of the algorithms of clustering technique, in which similar data is grouped into a cluster. K-means is an iterative algorithm that aims to find local maxima in each iteration. It starts with K as the input which is the desired number of clusters. Input k centroids in random locations in your space. Now, with the use of the Euclidean distance method, calculates the distance between

data points and centroids, and assign data point to the cluster which is close to its centroid. Re calculate the cluster centroids as a mean of data points attached to it. Repeat until no further changes occur.

Types of Dimensionality Reduction Algorithms:-

There are many dimensionality reduction algorithms in machine learning, which are applied for different dimensionality reduction applications. One of the main dimensionality reduction techniques is Principal Component Analysis (PCA) / Factor Analysis.

Principal Component Analysis (Factor Analysis):- Principal Component Analysis is one of the algorithms of Dimensionality reduction. In this technique, it transforms data into a new set of variables from input variables, which are the linear combination of real variables. These Specific new set of variables are known as principal components. As a result of the transformation, the first primary component will have the most significant possible variance, and each following component in has the highest possible variance under the constraint that it is orthogonal to the above components. Keeping only the best m < n components, reduces the data dimensionality while retaining most of the data information.

2.4 Choosing and comparing models through Pipelines

When you work on machine learning project, you often end up with multiple good models to choose from. Each model

will have different performance characteristics. Using resampling methods like k-fold cross validation; you can get an estimate of how accurate each model may be on unseen data. You need to be able to use these estimates to choose one or two best models from the suite of models that you have created.

2.4.1 Model Validation

When you are building a predictive model, you need to evaluate the capability or generalization power of the model on unseen data. This is typically done by estimating accuracy using data that was not used to train the model, often referred as cross validation.

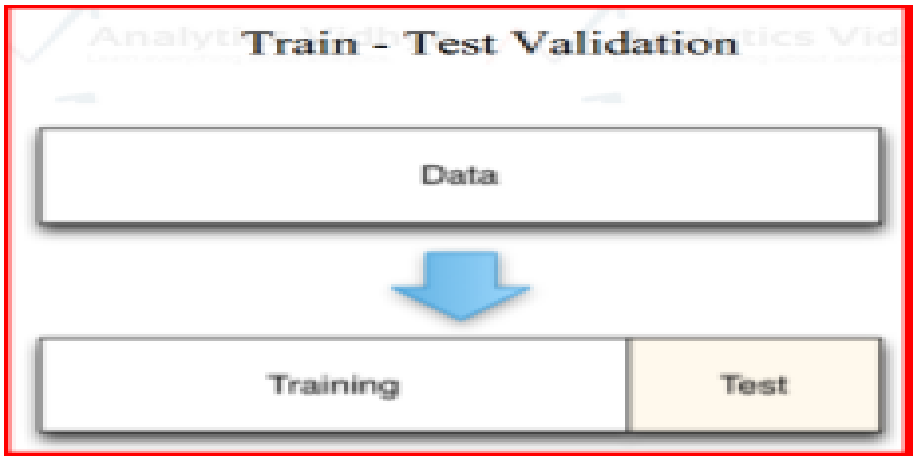


Fig 2.4.1 Model Validation.

A few common methods used for Cross Validation:

1) The Validation set Approach (Holdout Cross validation)

In this approach, we reserve large portion of dataset for training and rest remaining portion of the data for model validation.

Ideally people will use 70-30 or 80-20 percentages for training and validation purpose respectively.

A major disadvantage of this approach is that, since we are training a model on a randomly chosen portion of the dataset, there is a huge possibility that we might miss-out on some interesting information about the data which, will lead to a higher bias.

2) K-fold cross validation

As there is never enough data to train your model, removing a part of it for validation may lead to a problem of under fitting. By reducing the training data, we risk losing important patterns/ trends in data set, which in turn increases error induced by bias. So, what we require is a method that provides ample data for training the model and also leaves ample data for validation. K Fold cross validation does exactly that.

In K Fold cross validation, the data is divided into k subsets. Now the holdout method is repeated k times, such that each time, one of the k subsets is used as the test set/ validation set and the other k-1 subsets are put together to form a training set. The error estimation is averaged over all k trials to get total effectiveness of our model. As can be seen, every data point gets to be in a validation set exactly once, and gets to be in a training set k-1 times. This significantly reduces the bias as we are using most of the data for fitting, and also significantly reduces variance as most of the data is also being used in validation set. Interchanging the training and test sets also adds

to the effectiveness of this method. As a general rule and empirical evidence, K = 5 or 10 is preferred, but nothing's fixed and it can take any value.

Below are the steps for it:

- Randomly split your entire dataset into k"folds"
- For each k-fold in your dataset, build your model on k – 1 folds of the dataset. Then, test the model to check the effectiveness for kth fold.
- Record the error you see on each of the predictions. Repeat this until each of the k-folds has served as the test set.
- The average of your k recorded errors is called the cross-validation error and will serve as your performance metric for the model.

Below is the visualization of a k-fold validation when k=5.

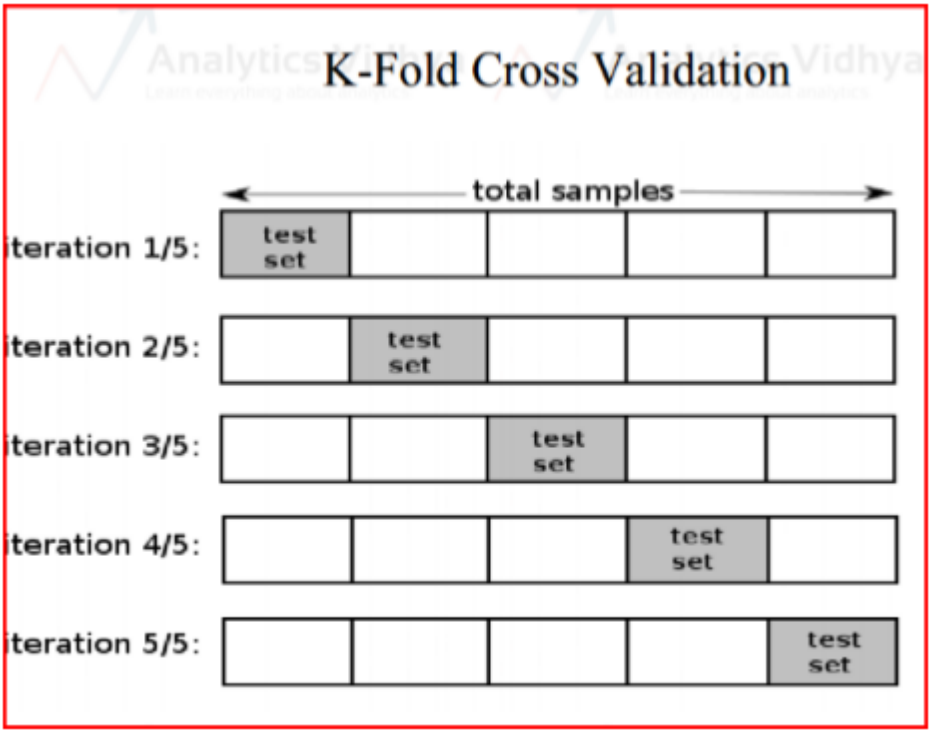


Fig 2.4.1(1) k-Fold cross validation

How to choose K:

- Smaller dataset: 10-fold cross validation is better
- Moderate dataset: 5 or 6 fold cross validation works mostly
- Big dataset: Train – Val split for validation

Other than this, we have Leave one out cross validation (LOOCV), in which each record will be left over from the training and then, the same will be used for testing purpose. This process will be repeated across all the respondents.

2.5 Model Diagnosis with over fitting and under fitting

2.5.1 Bias and Variance

A fundamental problem with supervised learning is the bias variance trade-off. Ideally, a model should have two key characteristics

- Sensitive enough to accurately capture the key patterns in the training dataset.
- Generalized enough to work well on any unseen dataset.

Unfortunately, while trying to achieve the above-mentioned first point, there is an ample chance of over-fitting to noisy or unrepresentative training data points leading to a failure of generalizing the model. On the other hand, trying to generalize a model may result in failing to capture important regularities.

If model accuracy is low on a training dataset as well as test dataset, the model is said to be under-fitting or that the model

has high bias. The Bias refers to the simplifying assumptions made by the algorithm to make the problem easier to solve. To solve an under-fitting issue or to reduce bias, try including more meaningful features and try to increase the model complexity by trying higherorder interactions

The Variance refers to sensitivity of a model changes to the training data. A model is giving high accuracy on a training dataset, however on a test dataset the accuracy drops drastically then, the model is said to be over-fitting or a model that has high variance.

To solve the over-fitting issue Try to reduce the number of features, that is, keep only the meaningful features or try regularization methods that will keep all the features. Ideal model will be the trade-off between Underfitting and over fitting like mentioned in the below picture.

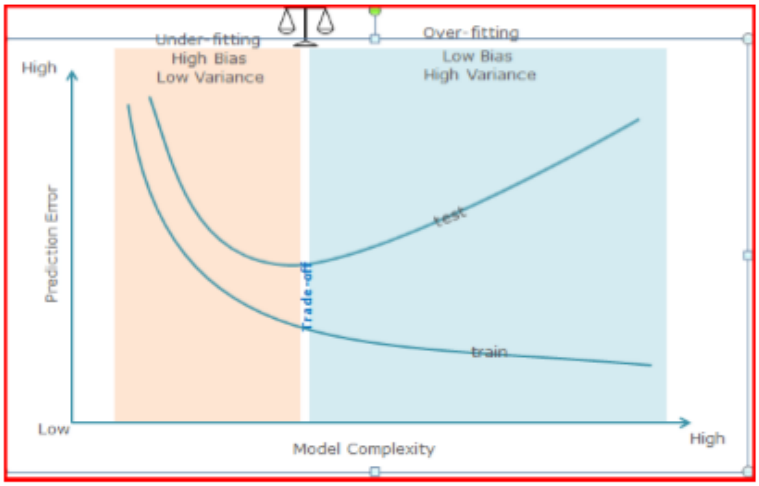


Fig 2.5.1 Bias and Variance

And, the Hyperparameters will be tuned in the below mentioned ways to reach the optimal solution:

1) Grid Search

2) Random Search

3) Manual Tuning

2.5.2 Model Performance Matrix

Model evaluation is an integral part of the model development. Based on model evaluation and subsequent comparisons, we can take a call whether to continue our efforts in model enhancement or cease them and select the final model that should be used / deployed.

1. Evaluating Classification Models

Confusion Matrix

Confusion matrix is one of the most popular ways to evaluate a classification model. A confusion matrix can be created for a binary classification as well as a multi-class classification model.

A confusion matrix is created by comparing the predicted class label of a data point with its actual class label. This comparison is repeated for the whole dataset and the results of this comparison are compiled in a matrix or tabular format

Table 2.5.2 Confusion Matrix.

		Predicted classed		
		Positive (C_0)	Negative (C_1)	
Actual class	Positive (C_0)	a = number of correctly Classified c0 cases	c = number of c_0 cases Incorrectly classified as c_1	Precision = a/(a + c)
	Negative (C_0)	b = number of c_1 cases Incorrectly classified as c_0	d = number of correctly classified c_1 cases	
		Sensitivity (Recall) = a/(a+b)	Specificity = d/c+d	Accuracy = (a+b)(a+b+c+d)

Specificity : The ratio of actual negative cases that are identified correctly.

shows an example confusion matrix.

. Example of classifications Accuracy measurement

		Predicted classed		
		Positive (C_0)	Negative (C_1)	
Actual class	Positive (C_0)	80	30	Precision = 70/110=0.63
	Negative (C_1)	40	90	
		Recall=80/120=0.67	Specificity = 90/240=0.75	Accuracy = 80+90/240=0.71

And, below are the various measures that will be used to assess the performance of the model based on the requirement of the problem and as well as data.

Table 2.5.2(1) Various measures used to assess performance.

Metric	Description	Formula
Accuracy	What% of predictions were Correct?	(TP + TN)/(TP + TN + EP + FN)
Misclassification rate	What % of prediction is wrong?	(FP + FN)/(TP + TN + FP + FN)
True positive rate OR Sensitivity or recall (completeness)	What % of positive cases did Model catch?	TP/(FN + TP)
False positive Rate	What % 'NO' were predicted as 'Yes'?	FP/FP+TN)
Specificity	What % 'NO' were predicted as 'NO'?	TN/(TN + FP)
Precision(exactness)	What % of positive predictions Were correct?	TP(TP + FP)
FI score	Weighted average of precision And recall	2*((precision*recall)/ (precision + recall))

2. Regression Model Evaluation

A regression line predicts the y values for a given x value. Note that the values are around the average. The prediction error (called as root-mean-square error or RSME) is given by the following formula:

$$RMSE = \sqrt{\frac{\sum_{k=0}^{n} \left(\hat{Y}_k - y_k \right)^2}{n}}$$

And, the regression will also assessed by R square (Co-efficient of determination).

3. Evaluating Unsupervised Models

The Unsupervised algorithms will be assessed by the profile of the factors/ clusters which were derived through the models.

2.6 Overall Process of Machine Learning

To put overall process together, below is the picture that describes the road map for building ML Systems

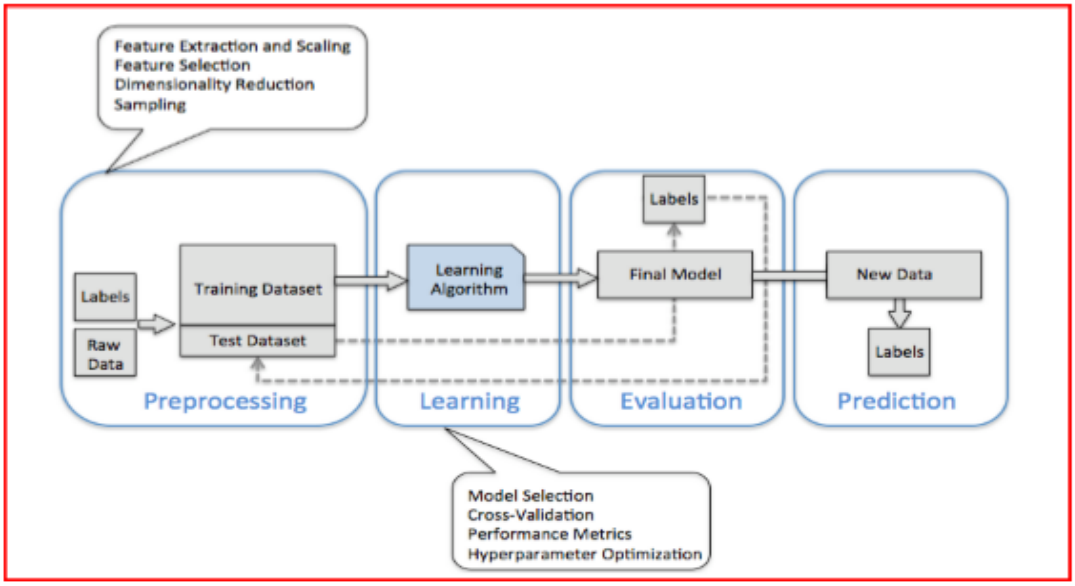


Fig 2.6 Process of machine learning.

CHAPTER- 3

Machine Learning - At Work

3.1 An Approach to the Problem:

In order to carry out the analysis, we have extracted 200 records from the university and the information of the same is mentioned in Chapter 1.

In this Chapter, we are going to discuss about the results of different Machine Learning methods used in order to obtain the solution for the problem mentioned in Chapter 1.

As mentioned in Chapter 2, the first step of a ML Algorithm is Data cleaning and preparing data for the modelling . As a first step, we have to check whether the data was read properly, and all the scale types are as per the data.

```
'data.frame':   200 obs. of  6 variables:
 $ rank          : int  1 1 1 1 1 1 1 1 1 1 ...
 $ discipline    : Factor w/ 2 levels "A","B": 2 2 2 2 2 2 2 2 2 2 ...
 $ yrs.since.phd: int  19 20 45 40 30 45 21 18 20 12 ...
 $ yrs.service   : int  18 16 39 41 23 45 20 18 18 3 ...
 $ sex           : int  0 0 0 0 0 0 0 1 0 0 ...
 $ salary        : int  139750 173200 115000 141500 175000 147765 119250 129000 104800 117150 ...
```

Output 3.1(1) description of the data

Understanding data using Descriptive Statistics:

We will look at the summary of the data.

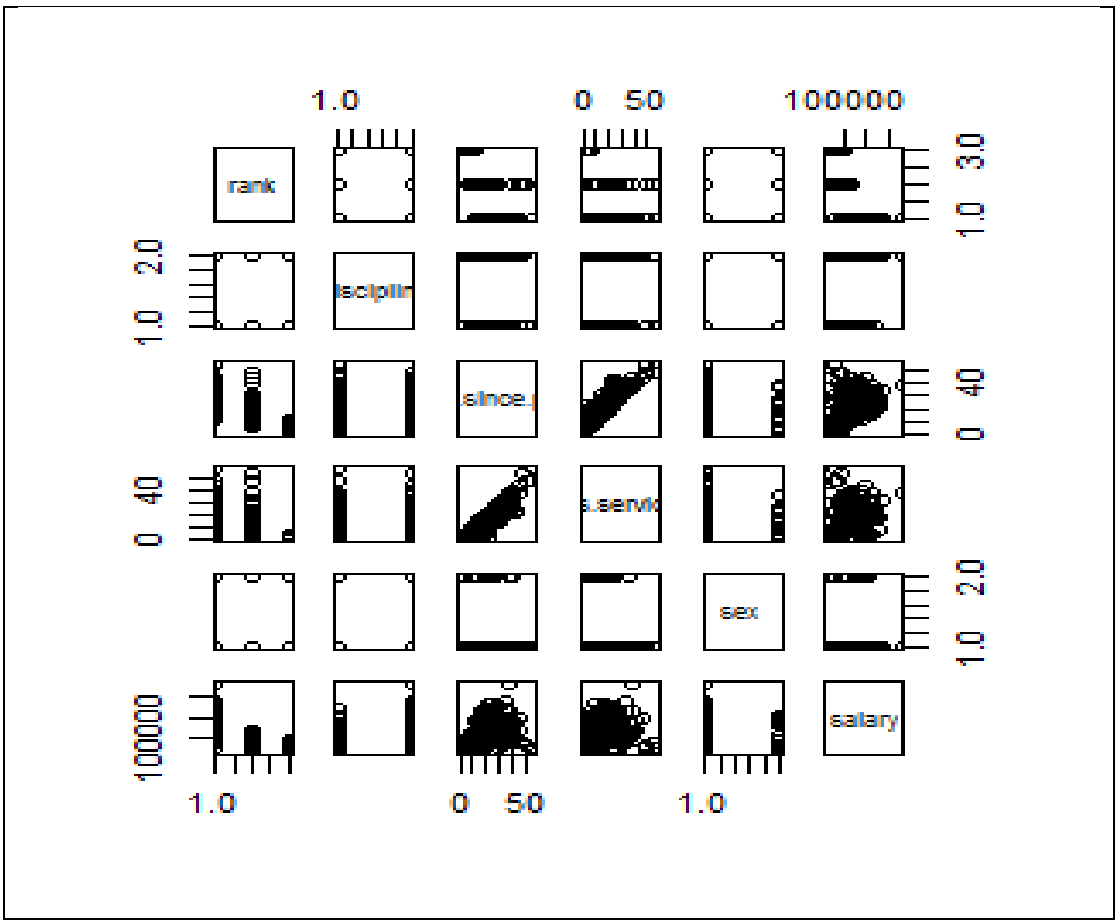
```
 rank    discipline yrs.since.phd    yrs.service    sex         salary
 1:112   A: 66      Min.    : 1.00   Min.    : 0.00   0:176   Min.    : 62884
 2: 43   B:134      1st Qu.: 9.00   1st Qu.: 5.00   1: 24   1st Qu.: 83888
 3: 45              Median :19.00   Median :13.00           Median :104015
                    Mean   :19.42   Mean   :15.17           Mean   :110004
                    3rd Qu.:27.25   3rd Qu.:23.00           3rd Qu.:129923
                    Max.   :56.00   Max.   :57.00           Max.   :231545
>
```

Output 3.1(2) summary of the data

For continous variables we determined central tendency and dispersion. For categorical variables we determined frequency

Understanding data visually:

Also, look at the data visually to understand the relationships between and within the variables



Output 3.1(3) data visualization

Checking for missing Values:

Then, check if there are any missing values in the data

```
rank    discipline yrs.since.phd  yrs.service        sex       salary
   0             0             0             0          0            0
```

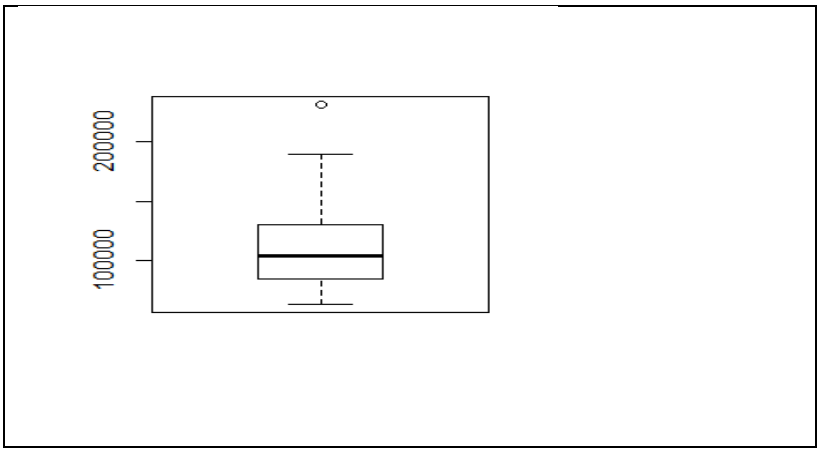
Output 3.1(4) missing values

There are no missing values in the given data.

Checking for Outliers:

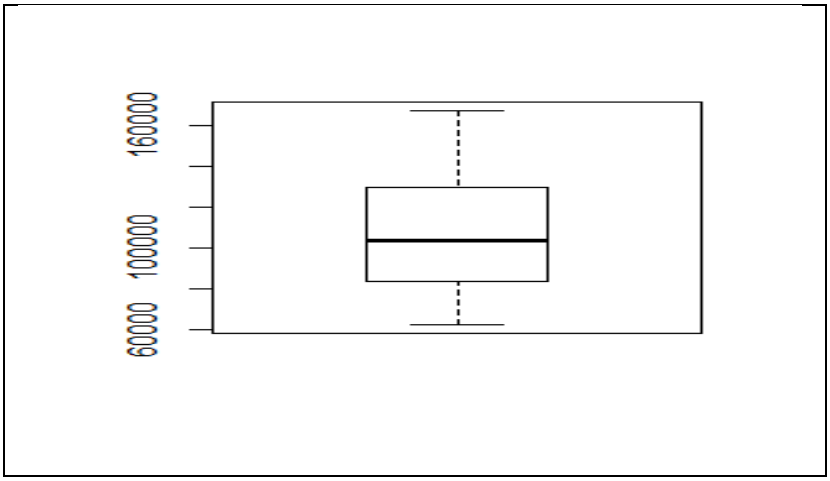
We used Box-plots to check for Outliers in each of the continuous variables.

BOXPLOT FOR SALARY:



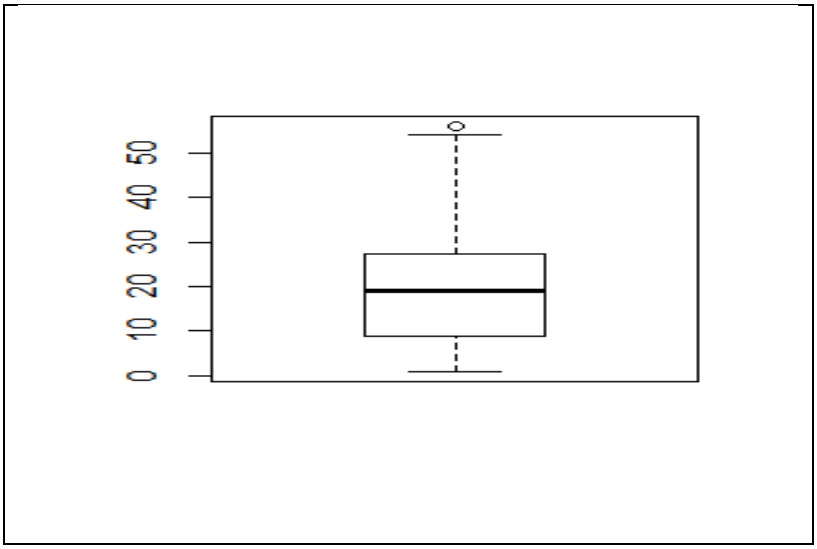
Output 3.1(5) boxplot with outliers of the salary

Values more than 95th percentile will be imputed using the 95th percentile value and the values less than 5th percentile will be imputed using 5th percentile value.



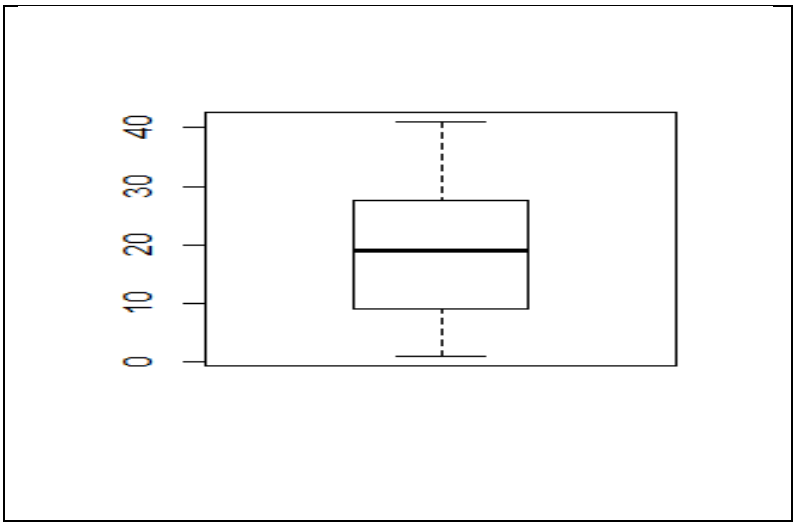
Output 3.1(6) boxplot after removing outliers for salary

BOXPLOT FOR YRS.SINCE.PHD :



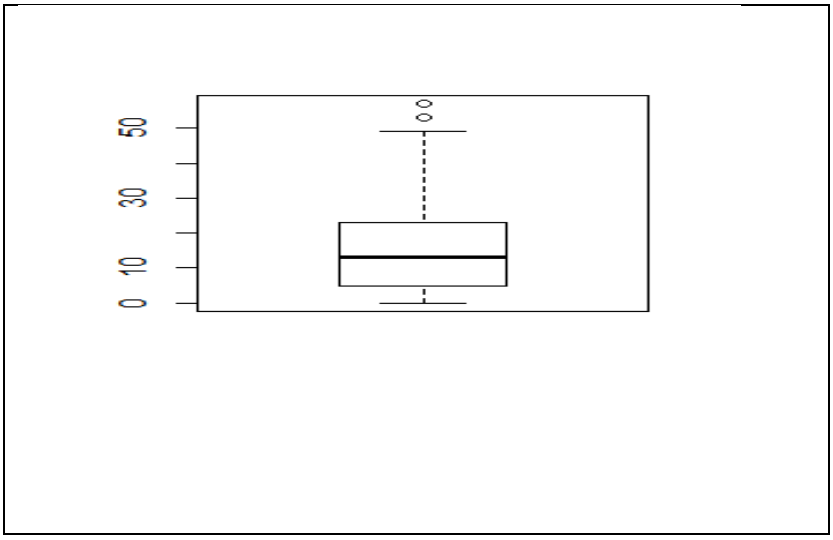
Output 3.1(7) boxplot with outliers of the yrs.since.phd

Values more than 95th percentile will be imputed using the 95th percentile value and the values less than 5th percentile will be imputed using 5th percentile value.



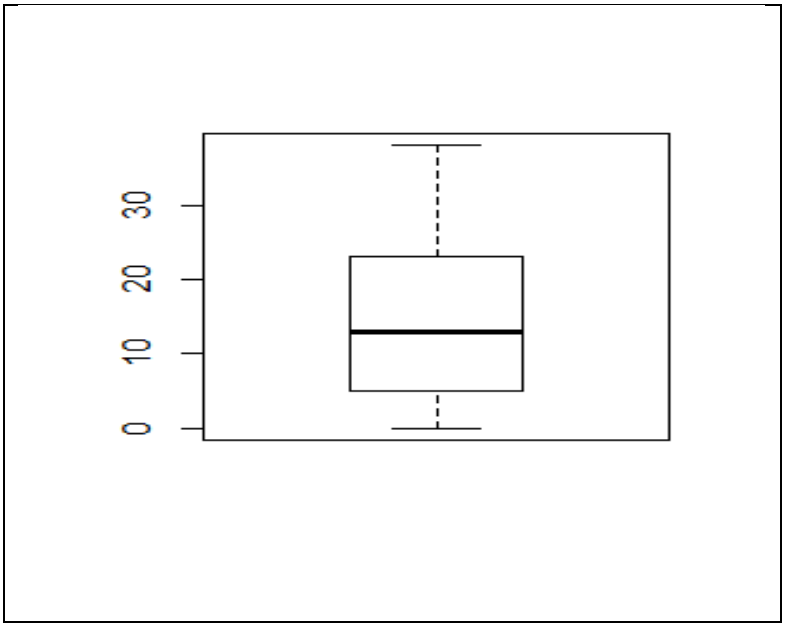
Output 3.1(8) boxplot after removing outliers of the yrs.since.phd

BOXPLOT FOR YRS.SERVICE:



Output 3.1(9) boxplot with outliers of yrs.service

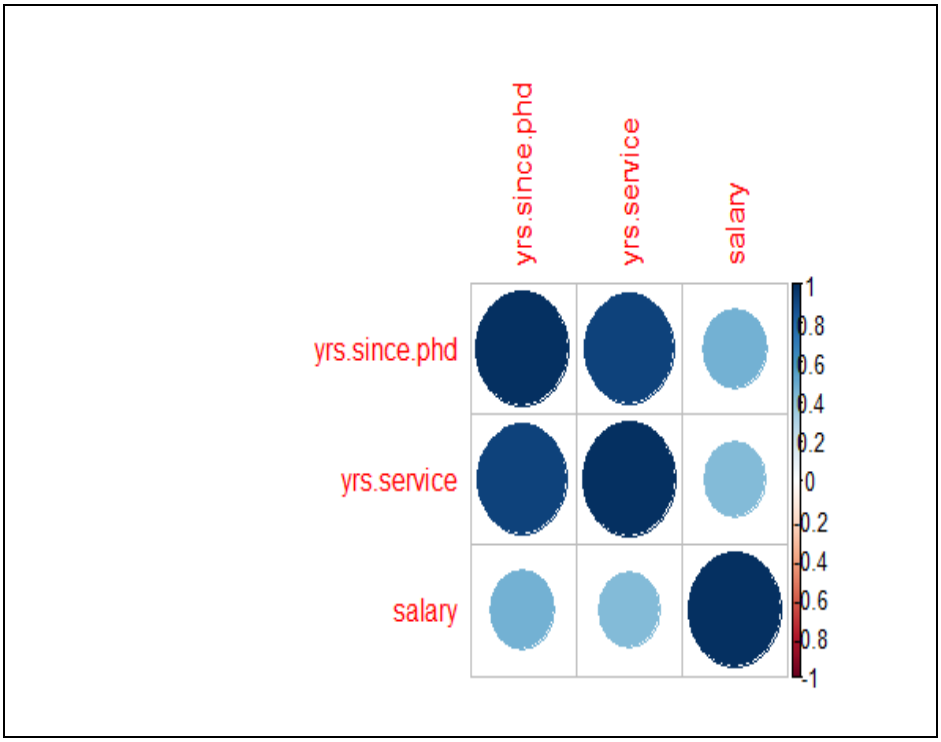
Values more than 95th percentile will be imputed using the 95th percentile value and the values less than 5th percentile will be imputed using 5th percentile value.



Output 3.1(10) boxplot after removing outliers of yrs.service

Understanding relationships between variables:

For the continuous variables, we will look at the Correlation plots between variables to understand the relationships between variables.



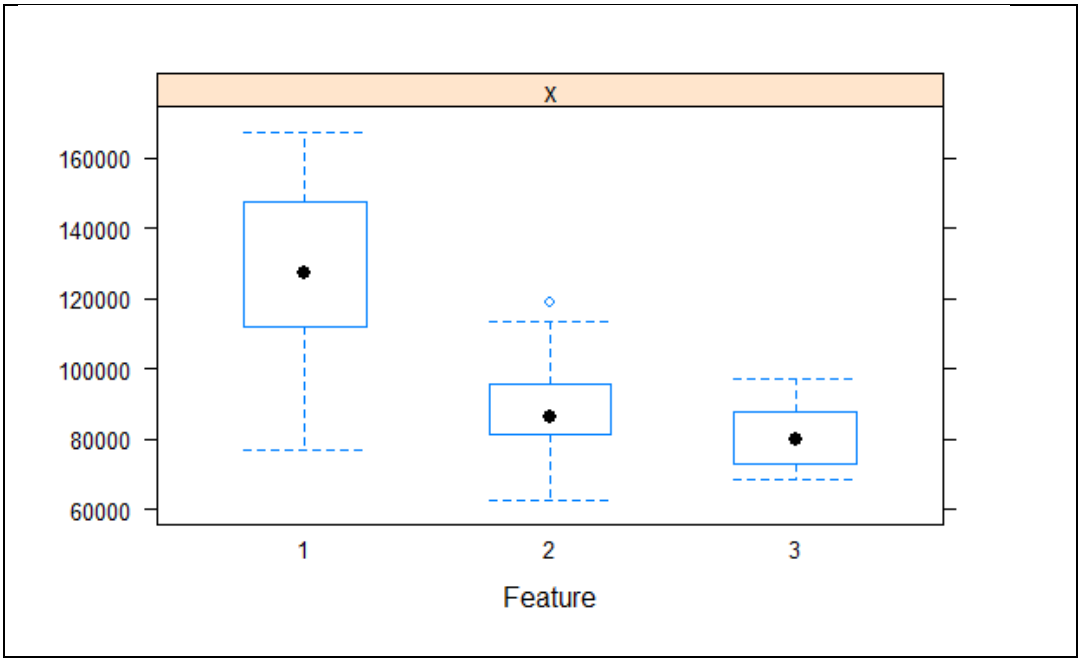
Output 3.1(11) correlation plot

Here, the circle size refers to the strength of the relation and colour refers to the direction of the relationship.

From the plot, we can see that yrs.since.phd and yrs.service are highly positively correlated.

For the continuous vs categorical variable, we will look at Feature plots to understand the relationships.

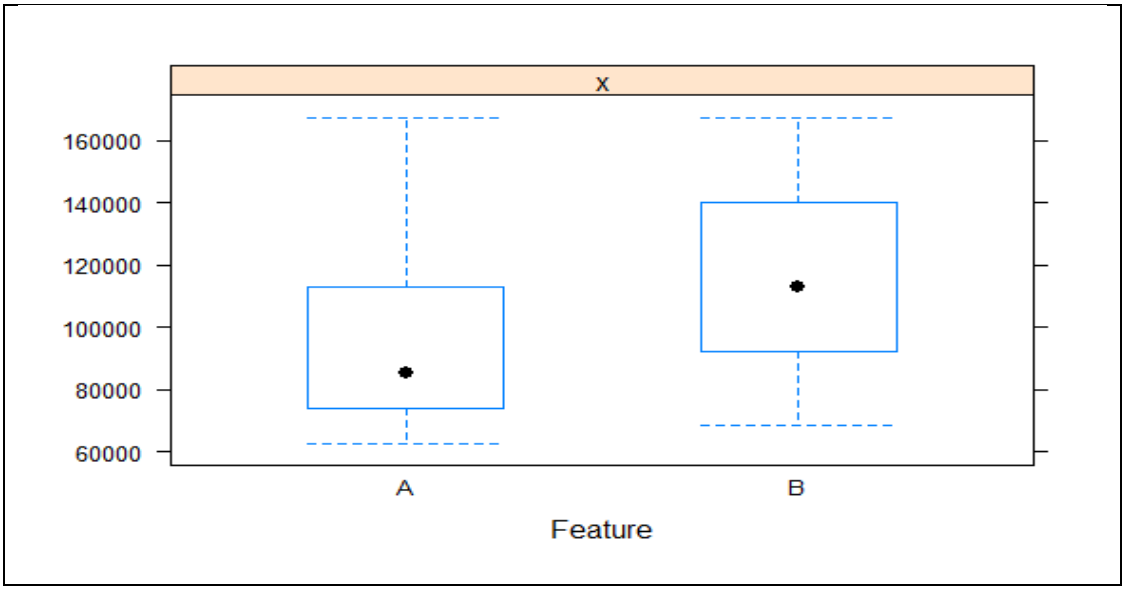
Rank vs Salary:



Output 3.1(12) feature plot of the salary vs rank

From the above feature plot, we observe that rank 1 i.e professors have more salary when compared to ranks 2 and 3.

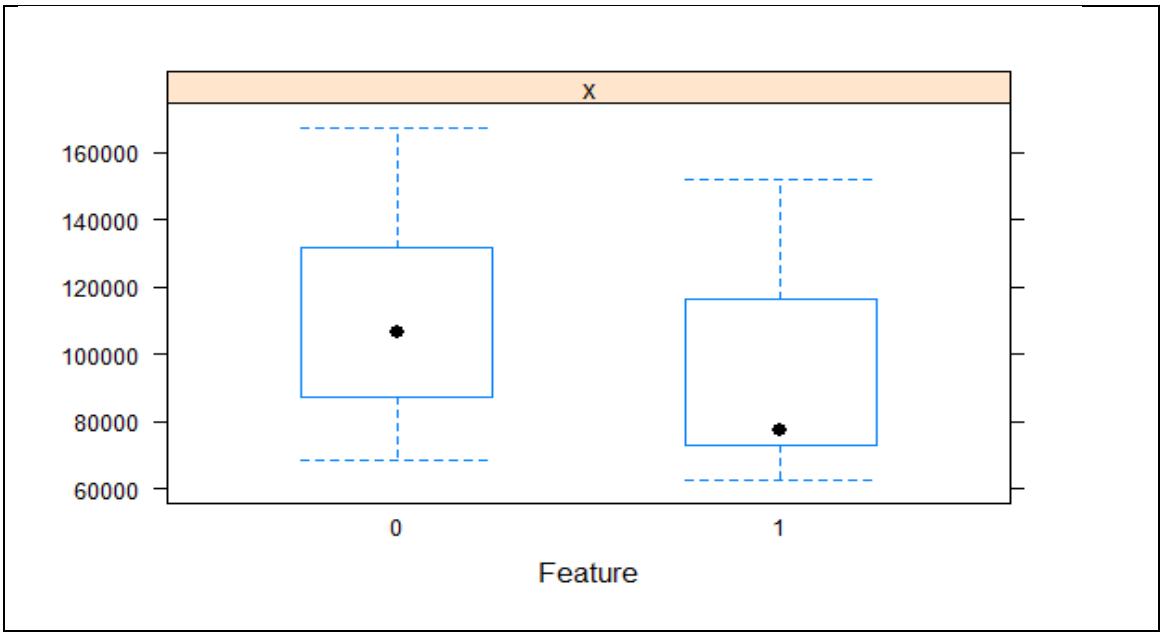
Discipline vs salary:



Output 3.1(13) feature plot for discipline vs salary

From the above feature plot, we observe that discpline “B “i.e applied department has more salary when compared with “A”.

Sex vs salary:



Output 3.1(14) feature plot of the salary vs sex

From the above feature plot, we observe that male has slightly higher salary than female.

Checking for the significance difference between variables:

To test the significant difference between categorical vs continuous variables, we will look at the ANOVA value.

Rank vs Salary:

```
             Df    Sum Sq   Mean Sq F value Pr(>F)
data$rank     2 1.017e+11 5.085e+10   147.3 <2e-16 ***
Residuals   197 6.802e+10 3.453e+08
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Output 3.1(15)ANOVA table of rank and salary

From the above table, as the p value is <0.05, we can conclude that there is more significant relationship between Rank and Salary.

Salary vs Sex:

```
             Df    Sum Sq   Mean Sq F value  Pr(>F)
data$sex      1 6.186e+09 6.186e+09    7.49 0.00677 **
Residuals   198 1.635e+11 8.259e+08
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Output 3.1(16)ANOVA table of sex and salary

From the above table, as the p value is < 0.05, we can conclude that there is significant relationship between sex and Salary.

Salary vs Discipline :

```
                Df    Sum Sq   Mean Sq F value   Pr(>F)
data$discipline  1 1.761e+10 1.761e+10   22.93 3.29e-06 ***
Residuals      198 1.521e+11 7.682e+08
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Output 3.1(17)ANOVA table of discipline and salary

From the above table, as the p value is < 0.05, we can conclude that there is significant relationship between discipline and Salary.

Cross validation

In order to validate the model , we will use Train Test and Hold out method.We will train and test the data using training sample and validate the same using hold out sample.

Split Data into Train & Test

We will split the data into train and test with 70% and 30% respectively.

The dimension of Train data :

“140 Observations “ “6 Variables”

The dimension of Test data:

“60 Observations” “6 Variables”

Running Pipeline using k-fold validation

We have used 10-fold validation in fitting the model and run a pipeline of algorithms to choose the algorithm that best fits the data.

As this is a Regression problem, we have run

1.Linear Model

2.Generalised Linear Model

3.Decision Tree

4.K-Nearest Neighbourhood

5.Support Vector Machine

6.Random Forest

1.LM:

The below is the output obtained from linear model.

```
Linear Regression

140 samples
  5 predictor

Pre-processing: re-scaling to [0, 1] (6)
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 127, 125, 126, 125, 126, 127, ...
Resampling results:

  RMSE      Rsquared   MAE
  16733.96  0.6684574  12696.28

Tuning parameter 'intercept' was held constant at a value of TRUE
```

Output 3.1(18) linear model

From the above, we can conclude that the model is able to explain 67% of the total variance in dependent variable

2.GLM:

Below is the output obtained from the general linear model.

```
Generalized Linear Model

140 samples
  5 predictor

Pre-processing: re-scaling to [0, 1] (6)
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 127, 125, 126, 125, 126, 127, ...
Resampling results:

  RMSE      Rsquared   MAE
  16733.96  0.6684574  12696.28

  ..
```

Output 3.1(19) general linear model

From the above, we can conclude that the model is able to explain 67% of the total variance in dependent variable

3.DECISION TREE

Below is the output obtained from the decision tree.

```
CART

140 samples
  5 predictor

Pre-processing: re-scaling to [0, 1] (6)
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 127, 125, 126, 125, 126, 127, ...
Resampling results across tuning parameters:

  cp    RMSE      Rsquared   MAE
  0.00  18444.68  0.6033279  13996.69
  0.05  19333.37  0.5629006  15464.92
  0.10  19147.06  0.5688265  15346.55

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was cp = 0.
```

Output 3.1(20) decision tree

From the above, we can conclude that the model is able to explain 60% of the total variance in dependent variable

4.KNN

Below is the output obtained from knn

```
k-Nearest Neighbors

140 samples
  5 predictor

Pre-processing: re-scaling to [0, 1] (6)
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 127, 125, 126, 125, 126, 127, ...
Resampling results across tuning parameters:

  k  RMSE      Rsquared   MAE
  5  18142.53  0.6110983  13543.59
  7  17724.82  0.6262221  13457.82
  9  17524.94  0.6335519  13300.86

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was k = 9.
```

Output 3.1(21) k-nearest neighboured

From the above, we can conclude that the model is able to explain 63% of the total variance in dependent variable

5.SVM

Below is the output obtained from support vector machine.

```
Support Vector Machines with Radial Basis Function Kernel

140 samples
  5 predictor

Pre-processing: re-scaling to [0, 1] (6)
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 127, 125, 126, 125, 126, 127, ...
Resampling results across tuning parameters:

  C     RMSE      Rsquared   MAE
  0.25  19933.12  0.5862363  15063.62
  0.50  18830.31  0.5994338  14247.28
  1.00  18241.41  0.6084280  13886.95

Tuning parameter 'sigma' was held constant at a value of 1.455068
RMSE was used to select the optimal model using the smallest value.
The final values used for the model were sigma = 1.455068 and C = 1.
```

Output 3.1(22) support vector machine

From the above, we can conclude that the model is able to explain 60% of the total variance in dependent variable

6.RANDOM FOREST

Below is the output obtained from random forest.

```
Random Forest

140 samples
  5 predictor

Pre-processing: re-scaling to [0, 1] (6)
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 127, 125, 126, 125, 126, 127, ...
Resampling results across tuning parameters:

  mtry  RMSE      Rsquared   MAE
  2     17530.81  0.6335160  13271.69
  4     18662.88  0.5823281  13783.55
  6     19162.21  0.5637713  14158.01

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was mtry = 2.
```

Output 3.1(23) random forest

From the above, we can conclude that the model is able to explain 63% of the total variance in dependent variable

Comparing algorithms

```
Rsquared
          Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
LM   0.4841037 0.6056505 0.6360820 0.6684574 0.7447965 0.8810851    0
GLM  0.4841037 0.6056505 0.6360820 0.6684574 0.7447965 0.8810851    0
CART 0.2987931 0.4700412 0.6043040 0.6033279 0.7082083 0.9057440    0
KNN  0.3544835 0.5475280 0.6382718 0.6335519 0.7008096 0.8547146    0
SVM  0.3052849 0.4929704 0.6215508 0.6084280 0.7305990 0.9444140    0
RF   0.3899295 0.5573032 0.6277416 0.6335160 0.7367717 0.9216459    0
```

Output 3.1(24) comparing alogrithums

From the above, by comparing 6 alogorithms,we conclude that Linear Model and Generalised Linear Model has 66% accuracy.

FINDING THE KEY VARIABLES USING RANDOM FOREST:

In order to find the key variables, we run the Random Forest using grid search and obtained key parameters.

```
                %IncMSE IncNodePurity
rank              43.43   40479257734
discipline        15.39    7669063280
yrs.since.phd     13.46   24743600828
yrs.service       10.63   18163318764
sex               -0.16    1196241655
```

Output 3.1(25) finding key variables using random forest

From the above table we can conclude that the key variables are rank, discipline, yrs.since.phd, yrs.service.

Since yrs.since.phd and yrs.service are highly correlated we consider any one i.e yrs.since.phd.

Using the obtained Key variables , we applied Generalised Linear Model and is the below output.

Genralised linear model using key variables:

```
Generalized Linear Model

140 samples
  3 predictor

Pre-processing: re-scaling to [0, 1] (4)
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 126, 125, 128, 126, 127, 127, ...
Resampling results:

  RMSE      Rsquared   MAE
  17142.62  0.6692964  13033.98
```

Output 3.1(26) fitting linear modell using key variables

The model has 66% accuracy with key variables. The model is explaining about 66% of the total variance in dependent variable.

Test Accuracy:

To validate the model , we have used hold out sample and got the below accuracy.

R-SQUARE OF TRAIN DATA IS:

0.649637247, i.e 64% accuracy

R-SQUARE OF TEST DATA IS:

0.705448543, i.e 70% accuracy

Since the accuracy of train and test data are more or less similar accurate . hence this model is generalised . so we can use this model to predict the future data.

CHAPTER-4

SUMMARY

SUMMARY

In order to solve the above problem, we have applied pipeline of techniques and shortlisted the Generalised linear regression based on the R-square value .and also, we applied the random forest technique to find the key variables for predicting the salaries and the key variables are rank, discipline, yrs.since.phd.

Hence, we have applied the linear regression model for these key variables on the train data and validate using test data. Below are the R-square values for train and test :

R-SQUARE OF TRAIN DATA IS:

0.649637247, i.e 64% accuracy

R-SQUARE OF TEST DATA IS:

0.705448543, i.e 70% accuracy

Since the accuracy of train and test data are more or less similar, this model is generalised . So ,we can use this model to predict the future data.

APPENDIX
R-CODE
DATA SET

R-CODE

```r
getwd()
setwd("D:/prathima")
#readind the data
data=read.csv("Final data.csv")
#Viewing the data
View(data)
#checking the dimension of data
dim(data)
#checking for the variables names in the data
names(data)
#viewing the head , tail of the data
head(data)
tail(data)
#to know the structure of the data
str(data)
#changing the datatypes into factors
data$rank=as.factor(data$rank)
data$sex=as.factor(data$sex)

str(data)
#to get descrptive statistics
summary(data)

#checking for missing values
table(is.na(data))

colSums(is.na(data))

# Checking for outliers in both Dependent and Independent variable

summary(data)
boxplot(data$salary)
boxplot(data$yrs.since.phd)
boxplot(data$yrs.service)
data$salary[data$salary>quantile(data$salary, 0.95)] <- quantile(data$salary, 0.95)
boxplot(data$salary)
data$yrs.since.phd[data$yrs.since.phd>quantile(data$yrs.since.phd, 0.95)] <- quantile(data$yrs.since.phd, 0.95
boxplot(data$yrs.since.phd)
data$yrs.service[data$yrs.service>quantile(data$yrs.service, 0.95)] <- quantile(data$yrs.service, 0.95)
boxplot(data$yrs.service)
```

```r
data$yrs.service[data$yrs.service>quantile(data$yrs.service, 0.95)] <- quantile(data$yrs.service, 0.95)
boxplot(data$yrs.service)
####################
pre1=data[c(3,4,6)]
str(pre1)
library(corrplot)
pre1.cor  = cor(pre1)
print(pre1.cor)
corrplot(pre1.cor,  method="circle")
#######################################33
# Continuous vs categories
library(caret)
x <- data[,6]
y <- data[,1]
featurePlot(x=x, y=y, plot="box")

library(caret)
x <- data[,6]
y <- data[,2]
featurePlot(x=x, y=y, plot="box")

library(caret)
x <- data[,6]
y <- data[,5]
featurePlot(x=x, y=y, plot="box")

#anova
x1=aov(data$salary ~ data$rank)
summary(x1)
x2=aov(data$salary  ~ data$sex)
summary(x2)
x3=aov(data$salary ~ data$discipline)
summary(x3)
#######################################3
#statistical testss/ Filter Method
#######################################3
# Preparing data for Traing and tessting
############################3
train_rows<- sample(1:nrow(data), size=0.7*nrow(data))
train_rows
training <- data[train_rows, ]
test <- data[-train_rows, ]
dim(data)
dim(training)
dim(test)
names(training)
names(test)
############################3
# Evaluate Algorithms: Baseline
############################3
# Run algorithms using 10-fold cross validation
control <- trainControl(method="repeatedcv", number=10, repeats=3)
metric <- "RMSE"
# lm
set.seed(120)
fit.lm <- train(salary~., data=training, method="lm", metric=metric, preProc=c("range"), trControl=control)
print(fit.lm)
# GLM
set.seed(120)
fit.glm <- train(salary~., data=training, method="glm", metric=metric, preProc=c("range"), trControl=control)
print(fit.glm)
# CART
set.seed(120)
grid <- expand.grid(.cp=c(0, 0.05, 0.1))
fit.cart <- train(salary~., data=training, method="rpart", metric=metric, tuneGrid=grid, preProc=c("range"), tr
print(fit.cart)
# kNN
set.seed(120)
fit.knn <- train(salary~., data=training, method="knn", metric=metric, preProc=c("range"), trControl=control)
print(fit.knn)
# SVM
set.seed(120)
fit.svm <- train(salary~., data=training, method="svmRadial", metric=metric, preProc=c("range"), trControl=cont
print(fit.svm)
# RF
set.seed(120)
fit.rf <- train(salary~., data=training, method="rf", metric=metric, preProc=c("range"), trControl=control)
print(fit.rf)
results <- resamples(list(LM=fit.lm, GLM=fit.glm,CART=fit.cart, KNN=fit.knn, SVM=fit.svm,RF=fit.rf))
summary(results)
```

```
results <- resamples(list(LM=fit.lm, GLM=fit.glm,CART=fit.cart, KNN=fit.knn, SVM=fit.svm,RF=fit.rf))
summary(results)
dotplot(results)
####################################
library(randomForest)
set.seed(120)
fit.rffin <- randomForest(salary~., data=training, mtry=2,tree=500,
                          importance=TRUE, na.action=na.omit)
print(fit.rffin)
## Show "importance" of variables: higher value mean more important:
round(importance(fit.rffin), 2)

######################
# GLMFianl model with key variables
######################
set.seed(120)
fit.glm <- train(salary~ rank + discipline + yrs.since.phd  , data=training, method="glm", metric=metric, preP
print(fit.glm)
pred=predict(fit.glm,data=training)
xx=cbind(training$salary,pred)
write.csv(xx, 'trainpred2.csv')
######################

names(test)
test1=test[,c(1:4)]
str(test1)
predtest=predict(fit.glm,newdata=test)
zz=cbind(predtest,test$salary)
write.csv(zz, 'testpred2.csv')
######################

```

DATA SET

rank	discipline	yrs.since.phd	Yrs.service	sex	Salary
1	B	19	18	0	139750
1	B	20	16	0	173200
1	B	45	39	0	115000
1	B	40	41	0	141500
1	B	30	23	0	175000
1	B	45	45	0	147765
1	B	21	20	0	119250
1	B	18	18	1	129000
1	B	20	18	0	104800
1	B	12	3	0	117150
1	B	19	20	0	101000
1	A	38	34	0	103450
1	A	37	23	0	124750
1	A	39	36	1	137000
1	A	31	26	0	89565
1	A	36	31	0	102580
1	A	34	30	0	93904
1	A	24	19	0	113068
1	A	21	8	0	106294
1	A	35	23	0	134885
1	B	12	8	0	118223
1	B	20	4	0	132261
1	B	13	9	0	117256
1	B	22	21	0	155750
1	B	41	31	0	125196
1	B	23	2	0	146500
1	B	40	27	0	101299
1	B	38	38	0	231545
1	B	19	19	0	94384
1	B	25	15	0	114778
1	B	40	28	0	98193
1	B	23	19	1	151768
1	B	25	25	1	140096
3	B	4	3	0	79750
3	B	7	2	0	79800

3	B	1	1	0	77700
3	B	2	0	0	78000
3	B	5	3	0	82379
3	B	11	0	0	77000
3	B	7	2	0	79916
3	B	4	2	0	80225
3	B	4	2	1	80225
3	B	5	0	1	77000
3	B	7	4	0	86373
3	B	1	1	0	70768
3	B	11	3	1	74692
3	B	8	3	0	75044
3	B	3	2	0	75243
3	B	4	3	0	68404
3	B	8	3	0	73266
3	B	3	1	0	86100
3	B	6	2	0	84240
3	B	6	2	0	88825
3	B	2	2	0	88400
3	B	10	5	1	97032
3	B	4	0	0	84000
3	A	3	1	0	72500
3	A	4	1	0	72500
3	A	5	3	1	73500
3	A	2	0	1	72500
3	A	4	2	0	73000
3	A	3	1	1	72500
3	B	3	3	0	89942
3	B	4	4	0	92000
3	B	4	3	0	95079
3	B	4	4	0	92000
3	B	4	0	0	92000
3	B	1	0	0	88000
3	B	2	2	0	89516
2	B	6	6	0	97000
2	A	13	8	1	74830
2	B	23	23	0	93418
2	B	14	5	0	83900
2	B	9	8	0	90215

2	B	9	8	0	90304
2	B	17	12	0	95611
2	A	18	10	0	83850
2	A	11	8	0	82099
2	A	10	8	0	82600
2	A	15	8	0	81500
2	A	19	16	0	82100
2	A	25	22	1	62884
2	A	11	9	0	83001
2	A	10	8	1	77500
2	A	10	7	0	73877
2	A	15	10	0	81500
2	B	6	6	0	95408
2	B	22	7	0	98510
2	B	10	7	0	95436
2	B	19	19	0	86250
2	B	48	53	0	90000
2	A	9	7	0	70000
2	A	26	24	1	73300
2	A	12	8	0	83000
2	A	30	23	0	74000
2	A	41	33	0	88600
2	A	8	6	0	88650
2	A	49	49	0	81800
2	A	45	39	0	70700
2	B	12	9	1	71065
2	B	9	9	0	95642
2	B	10	10	0	99247
2	A	20	17	0	81285
2	A	13	8	0	78182
2	A	8	5	0	86895
1	B	25	25	0	172272
1	B	38	38	0	166024
1	B	21	20	0	123683
1	B	13	7	0	129676
1	B	30	14	0	102235
1	B	41	26	0	106689
1	B	42	25	0	133217
1	B	28	23	0	126933

1	B	16	5	0	153303
1	B	20	14	1	127512
1	A	31	28	0	113543
1	A	40	31	0	131205
1	A	20	16	0	112429
1	A	37	37	0	104279
1	A	12	0	1	105000
1	A	21	9	0	120806
1	A	30	29	0	148500
1	A	39	36	0	117515
1	A	14	14	0	115313
1	A	32	32	0	124309
1	A	24	22	0	97262
1	A	24	22	0	96614
1	A	54	49	0	78162
1	A	28	26	0	155500
1	A	32	30	0	113278
1	A	56	57	0	76840
1	A	35	25	0	168635
1	A	20	18	0	136000
1	A	16	14	0	108262
1	A	17	14	0	105668
1	A	21	18	0	152664
1	A	19	11	0	106608
1	B	27	27	0	112696
1	B	28	28	0	119015
1	B	27	27	0	156938
1	B	36	26	1	144651
1	B	14	12	0	128148
1	B	21	9	0	111168
1	B	21	21	0	118971
1	B	15	16	0	137167
1	B	26	19	0	176500
1	B	21	8	0	105890
1	B	16	16	0	167284
1	B	18	19	0	130664
1	B	25	18	0	181257
1	B	19	19	0	151575
1	B	37	24	0	93164

1	B	20	20	0	134185
1	B	28	25	0	111751
1	B	27	14	0	147349
1	B	11	11	0	142467
1	B	18	5	0	141136
1	B	26	22	0	150000
1	B	23	23	0	101000
1	B	33	30	0	134000
1	B	18	10	0	107500
1	B	25	19	0	153750
1	B	22	9	0	180000
1	B	43	22	0	133700
1	B	19	18	0	122100
1	B	34	33	0	189409
1	B	38	22	0	114500
1	B	40	40	0	119700
1	B	28	17	0	160400
1	B	17	17	0	152500
1	B	19	5	0	165000
1	B	35	33	0	162200
1	B	18	18	0	120000
1	B	20	20	0	163200
1	B	39	39	0	111350
1	B	15	7	0	128400
1	B	26	19	0	126200
1	B	16	11	0	145350
1	B	15	11	0	146000
1	B	13	11	0	119500
1	B	21	21	0	170000
1	B	23	10	0	145200
1	B	34	20	0	129600
1	A	20	20	0	122400
3	B	10	5	1	97032
3	B	4	0	0	84000
3	A	3	1	0	72500
3	A	4	1	0	72500
3	A	5	3	1	73500
3	A	2	0	1	72500
3	A	4	2	0	73000

3	A	3	1	1	72500
3	B	3	3	0	89942
2	B	10	7	0	95436
2	B	13	9	0	100944
2	B	8	8	0	100000
2	B	13	10	1	103750
2	B	28	28	0	106300
2	B	9	7	0	113600
2	B	11	1	0	118700

BIBLIOGRAPHY

1. Multivariate data analysis (Fifth Edition) --- Joseph F.Hair, Rolph E.Anderson, Ronald l Tatham and William C.Black

2. Data Mining- Theories, Algorithms, and Examples – NoNG YE

3. A Practical Guide to Data Mining for Business and Industry -- Andrea Ahlemeyer-Stubbe, Shirley Coleman

4. Data Mining and Predictive Analytics – Daniel T. Larose, Chantal D.Lorse

5. machine_learning_mastery_with_r. – Jason Brownlee

6. master_machine_learning_algorithms -- Jason Brownlee

7. statistical_methods_for_machine_learning - Jason Brownlee

8. Machine Learning Using R -- Karthik Ramasubramanian ,Abhishek Singh

9. Data Science for Business - Forster Provost & Tom Fawcett

10. Deep learning with Deep learning R by François Chollet