Lagdu Singh Charitable Trust's (Regd.)
THAKUR COLLEGE OF
ENGINEERING & TECHNOLOGY
Autonomous College Affiliated to University of Mumbai
Approved by All India Council for Technical Education(AICTE) and Government of Maharashtra(GoM)
Website : www.tcetmumbai.in
Conferred Autonomous Status by University Grants Commission (UGC) for 10 years w.e.f. A.Y 2019-20
Amongst Top 200 Colleges in the Country, Ranked 193rd in NIRF India Ranking 2019 in Engineering College category
• ISO 9001:2015 Certified • Programmes Accredited by National Board of Accreditation (NBA), New Delhi
• Institute Accredited by National Assessment and Accreditation Council (NAAC), Bangalore
Estd. in 2001

A.Y 2021-22
Institutional Summer Internship

Track: Machine Learning Using Python

Project Report on

Salary Prediction using Machine Learning

Team Member Details:

Yash Jain (56) (SE Comp A)

Aman Jaiswal (57) (SE Comp A)

Anup Jaiswal (58) (SE Comp A)

Anand Jaiswar (62) (SE Comp A)

Mrs. Vaishali Nirgude

Assistant Professor
Department of Computer Engineering

TOPIC INDEX

LIST OF FIGURES

Chapter 1: - Problem Description:

Now days, Major reason an employee switches the company is the salary of the employee. Employees keep switching the company to get the expected salary. And it leads to loss of the company and to overcome this loss we came with an idea what if the employee gets the desired/expected salary from the Company or Organization. In this Competitive world everyone has a higher expectation and goals. But we cannot randomly provide everyone their expected salary there should be a system which should measure the ability of the Employee for the Expected salary. We cannot decide the exact salary, but we can predict it by using certain data sets. A prediction is an assumption about a future event.

In this project the main aim is predicting salary and making a suitable user-friendly graph. So that an Employee can get the desired salary based on his qualification and hard work. For developing this system, we are using a Linear regression algorithm of supervised learning in machine learning.

Linear regression algorithm in machine learning is a supervised learning technique to approximate the mapping function to get the best predictions. The main goal of regression is the construction of an efficient model to predict the dependent attribute from a bunch of attribute variables. A regression problem is when the output value is real or a continuous value like salary.

Chapter 2: - Literature Survey

Paper 1: "Salary Prediction Using Machine Learning"

Summary: Our salary prediction system is aimed toward providing better assistance to the school students regarding the salary that they will aspect after completing their course. Not only they're going to be ready to get a thought of their deserving salary but also, they will get to understand about the talents that they have to satisfy their professional goals. this may enhance the motivation of scholars who are enrolled in education institutes and supply better assistance also. Through this paper we've tried to provide a system for salary prediction during which data processing technique is employed. during this system the profile of student are going to be compared with graduated student. we've used data mining techniques for comparison as they perform best. we've also performed an experiment on student data set using 10-fold cross-validation.

The major focus of educational data mining is to assist the institutes to arrange their students far better and aid them to reinforce their performance. Various machine learning concepts are applied to review the data collected from learning. Figure1 shows the picturing of application of educational data mining system. As now seen, many students select their course on the idea of trend, or they select course on the idea of their peer suggestion. The impact of all such is that the performance of scholars goes to pot, and this tend to throw in the towel from institutes. The performance of such students can be enhanced if we offer them with samples of their college graduates as they need already taken very same path and that they may need faced same issues. Their career and aid could help the scholars to be motivated for his or her courses and be focused

This paper provides a system for salary prediction in which data mining technique is used. In this system the profile of student will be compared with graduated student. We have used data mining techniques for comparison as they perform best. We have also performed an experiment on student data set using 10- fold cross-validation. We concluded that decision tree(J48) provides the best result. But KNN will give better performance if featured attribute is less.

Paper 2: "Salary Prediction in the IT Job Market with Few High-Dimensional Samples"

Summary: This work focuses on the challenge of predicting the salary offered by companies through job posts on the web. Instead of focusing on international and

multidomain web portals, which are abundant in terms of number of posts, this work analyses job posts collected from Tecnoempleo, an e-Recruitment website specialized in IT jobs for young people in Spain. Domain and geographical restrictions of the website make salary prediction a challenging task. In fact, the number of posts including an explicit indication of the salary, collected in 5 months daily, is only ≈ 4,000. Moreover, each post is retrieved as a vector of ≈ 2,000 features. From a machine learning perspective, the task is difficult because of the limited number of samples, the relatively high dimensionality and the presence of noise.

After analyzing key aspects from the job market, we assess the relevance of the features that can be used to predict salaries. Results indicate that some features, such as experience, job stability or certain job roles (i.e., Team Leader and IT Architect) contribute significantly to the final salary perceived by employees. Furthermore, we observe that posts can be arranged into 5 different skill-based profiles, namely: Back-end developer, Systems Administrator, .Net developer, Java developer and Front-end developer. Such profiles seem to be similarly paid, even though the demand for Back-end developers (including Java and .Net technologies) is higher than that for the rest of professionals. Finally, this work classifies job posts according to the offered salary range in a noisy and example scarce context. After collection, features are preprocessed, and the dimensionality is reduced by 10 times by using a customized procedure exploiting the domain knowledge. Embedded feature selection or other state-of-the-art filter methods are not beneficial in terms of classification accuracy. We compare 1206 International Journal of Computational Intelligence Systems, Vol. 11 (2018) 1192-1209 several models including logistic regression, nearest neighbors, MLPs, SVMs, random forests, adaptive boosting and voting classifiers based on all or part of them. Experiments show that ensembles based on decision trees behave generally better and that a voting committee based on them leads to an accuracy of ≈ 84%

Paper 3: “Predictive Analysis of HR Salary using Machine Learning Techniques”

Summary: Information irregularity amongst employers and employees has become a problem that needs immediate solving. The probable applicants are most often kept blind with regards to the interview procedure and only are aware of it at the end. In the meantime, the employers must be committed to rightly meeting up with the candidate's prospects for making new HR strategies that satisfy the demands of the applicant. Therefore, one must be vigilant enough to not offer too low a salary, which would result in the decline in not just the salary but also will build more irresponsible, lack-luster individuals with longer untaken positions. Whilst the vice-versa would also be a cause of concern leading to wastage of company’s vital resources. Therefore, it is imperative to supply an unbiased salary for an employee which he/she truly deserves and must be

right to the market demands. This paper is based on predicting the salary by training a Machine Learning model and performing comparative analysis on Logistic Regression and Support Vector Machine using their classification reports.

The main aim of this review paper focuses on finding the right future salary of an applicant based on some parameter of a particular domain. The algorithms used are Logistic Regression and Support Vector Machines to train and perform predictions using the ML model. Once both are imported the classification report is used as a comparison criterion to examine the overall efficiency of both algorithms. Out of the two, it is Support Vector Machine which is more accurate with an accuracy (F1- Score) of 89% accuracy for salary<=50K and 57% for salary>50K. All this is only possible if proper data cleaning is done by removing all the missing, incorrect and noisy data from the dataset to get an efficient result. Hence, this model has the capability to act as an aid for HR to predict salary precisely quite conveniently

Chapter 3: - Dataset Description

Our dataset consists of 30*2 rows by columns. the columns are salary of the employee and the years of experiences that are done by the employees. the type of data in our dataset is in float. There is no categorical value is present in dataset and no null values are present. 608.2 bytes of data is there (memory usage). total 30 employee's information are there. the mean of years of experience and salary is 5.313333 and 76003.000000 respectively.

The standard deviation of years of experience and salary is 2.837888 and 27414.429785 respectively. The minimum year of experience of employee in dataset is 1.100000 and maximum is 10.500000. The minimum salary of employee is 37731.00000 and maximum salary is 122391.000000

Salary is the dependent variable whereas years of experience is independent variable.

Chapter 4: - Exploratory Data Analysis

Exploratory Data Analysis refers to the critical process of performing initial investigations on data to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations

EDA explained using sample Data set:

To share my understanding of the concept and techniques I know, I'll take an example of salary prediction Quality data set which is available on UCI Machine Learning Repository and try to catch hold of as many insights from the data set using EDA.

To starts with, I imported necessary libraries (for this example pandas, NumPy, matplotlib and seaborn) and loaded the data set.

The describe () function in pandas is very handy in getting various summary statistics. This function returns the count, mean, standard deviation, minimum and maximum values and the quantiles of the data.

```
dataset.describe()
```

	YearsExperience	Salary
count	30.000000	30.000000
mean	5.313333	76003.000000
std	2.837888	27414.429785
min	1.100000	37731.000000
25%	3.200000	56720.750000
50%	4.700000	65237.000000
75%	7.700000	100544.750000
max	10.500000	122391.000000

Figure 1 Description of Dataset

A box plot (or box-and-whisker plot) shows the distribution of quantitative data in a way that facilitates comparisons between variables. The box shows the quartiles of the dataset while the whiskers extend to show the rest of the distribution.

The box plot (box and whisker diagram) is a standardized way of displaying the distribution of data based on the five number summary:

- Minimum

- First quartile
- Median
- Third quartile
- Maximum.

```
sns.boxplot(x = dataset['Salary'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fed3461a810>
```

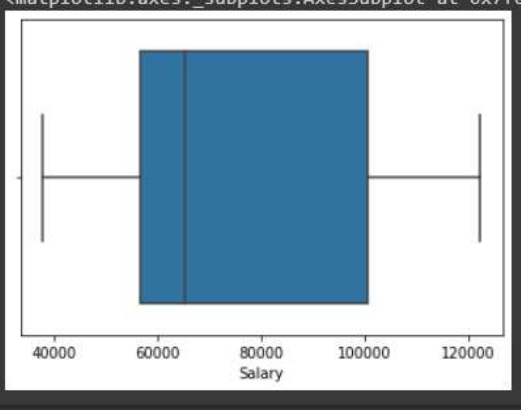


Figure 2 box plot

Now, let's also the columns and their data types. For this, we will use the info () method.

```
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30 entries, 0 to 29
Data columns (total 2 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   YearsExperience  30 non-null     float64
 1   Salary           30 non-null     float64
dtypes: float64(2)
memory usage: 608.0 bytes
```

Figure 3 Info of Dataset

Chapter 5: - Data Preprocessing

Data Preprocessing in Machine learning

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model. When creating a machine learning project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So, for this, we use data preprocessing task.

Why do we need Data Preprocessing?

A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Data preprocessing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.

It involves below steps:

- **Getting the dataset**

  To create a machine learning model, the first thing we required is a dataset as a machine learning model completely works on data. The collected data for a particular problem in a proper format is known as the dataset.

  Dataset may be of different formats for different purposes, such as, if we want to create a machine learning model for business purpose, then dataset will be different with the dataset required for a liver patient. So, each dataset is different from another dataset. To use the dataset in our code, we usually put it into a CSV file.

- **Importing libraries**

  In order to perform data preprocessing using Python, we need to import some predefined Python libraries. These libraries are used to perform some specific jobs. There are three specific libraries that we will use for data preprocessing, which are: NumPy, Matplotlib and Pandas.

- **Importing datasets**

Now we need to import the datasets which we have collected for our machine learning project. But before importing a dataset, we need to set the current directory as a working directory. Now to import the dataset, we will use read_csv () function of panda's library, which is used to read a csv file and performs various operations on it. Using this function, we can read a csv file locally as well as through an URL.

- **Finding Missing Data**

  There are mainly two ways to handle missing data, which are:

  By deleting the row: The first way is used to commonly deal with null values. In this way, we just delete the specific row or column which consists of null values. But this way is not so efficient and removing data may lead to loss of information which will not give the accurate output.

  By calculating the mean: In this way, we will calculate the mean of that column or row which contains any missing value and will put it on the place of missing value. This strategy is useful for the features which have numeric data such as age, salary, year, etc. Here, we will use this approach.

- **Encoding Categorical Data**

Since machine learning model completely works on mathematics and numbers, but if our dataset would have a categorical variable, then it may create trouble while building the model. So, it is necessary to encode these categorical variables into numbers.

- **Splitting dataset into training and test set**

In machine learning data preprocessing, we divide our dataset into a training set and test set. This is one of the crucial steps of data preprocessing as by doing this, we can enhance the performance of our machine learning model.

Suppose if we have given training to our machine learning model by a dataset and we test it by a completely different dataset. Then, it will create difficulties for our model to understand the correlations between the models.

If we train our model very well and its training accuracy is also very high, but we provide a new dataset to it, then it will decrease the performance. So, we always try to make a machine learning model which performs well with the training set and with the test dataset.

Chapter 6: - Choice of Model

Regression is a method of modelling a target value based on independent predictors. This method is mostly used for forecasting and finding out cause and effect relationship between variables. Regression techniques mostly differ based on the number of independent variables and the type of relationship between the independent and dependent variables.

Linear Regression

Simple linear regression is a type of regression analysis where the number of independent variables is one and there is a linear relationship between the independent(x) and dependent(y) variable. The red line in the above graph is referred to as the best fit straight line. Based on the given data points, we try to plot a line that models the points the best. The line can be modelled based on the linear equation shown below.

y = a_0 + a_1 * x ## Linear Equation

The motive of the linear regression algorithm is to find the best values for a_0 and a_1. Before moving on to the algorithm, let's have a look at two important concepts you must know to better understand linear regression**.**

Figure 4 Linear Regression Model for Training and Testing

Polynomial Regression

In polynomial regression, the relationship between the independent variable x and the dependent variable y is described as an nth degree polynomial in x. Polynomial

regression, abbreviated E (y |x), describes the fitting of a nonlinear relationship between the value of x and the conditional mean of y. It usually corresponded to the least-squares method. According to the Gauss Markov Theorem, the least square approach minimizes the variance of the coefficients. This is a type of Linear Regression in which the dependent and independent variables have a curvilinear relationship, and the polynomial equation is fitted to the data; we'll go over that in more detail later in the article. Machine learning is also referred to as a subset of Multiple Linear Regression. Because we convert the Multiple Linear Regression equation into a Polynomial Regression equation by including more polynomial elements.

Types of Polynomial Regression

A quadratic equation is a general term for a second-degree polynomial equation. This degree, on the other hand, can go up to nth values. Polynomial regression can so be categorized as follows:

1. Linear – if degree as 1

2. Quadratic – if degree as 2

3. Cubic – if degree as 3 and goes on, based on degree.

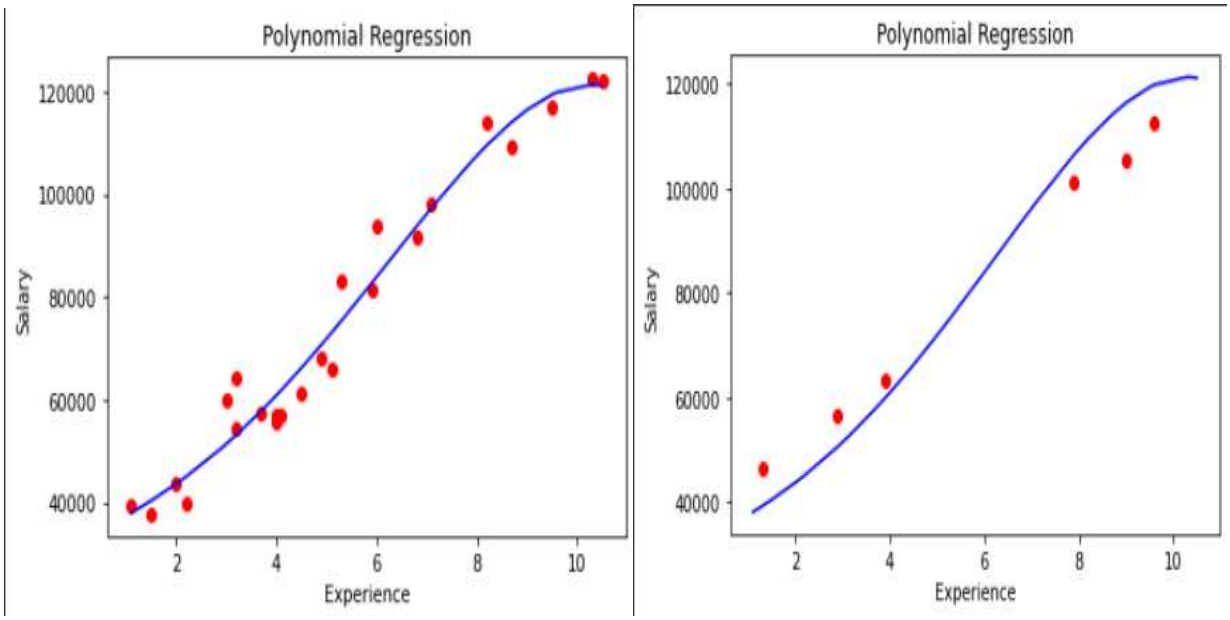


Figure 5 Polynomial Regression for Training and Testing

Decision Tree Algorithm

Decision Tree is one of the most used, practical approaches for supervised learning. It can be used to solve both Regression and Classification tasks with the latter being put more into practical application.

It is a tree-structured classifier with three types of nodes. The Root Node is the initial node which represents the entire sample and may get split further into further nodes. The Interior Nodes represent the features of a data set, and the branches represent the decision rules. Finally, the Leaf Nodes represent the outcome. This algorithm is very useful for solving decision-related problems.

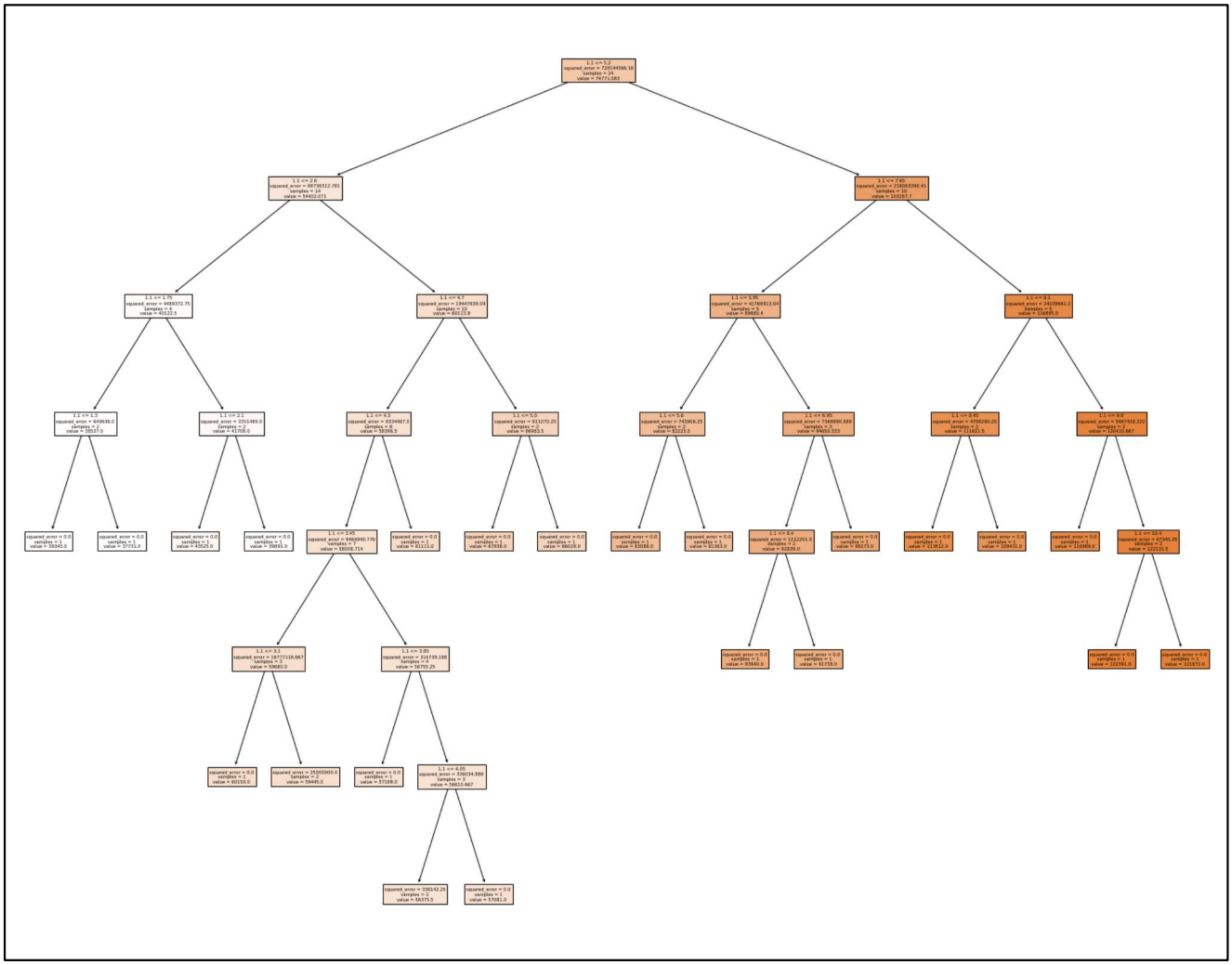


Figure 6 Decision Tree Structure

With a particular data point, it is run completely through the entirely tree by answering True/False questions till it reaches the leaf node. The final prediction is the average of the value of the dependent variable in that leaf node. Through multiple iterations, the Tree can predict a proper value for the data point

Chapter 7: - Model training and testing

A common problem in machine learning algorithms is their tendency to "memorize" the data they have been trained on. In data science terms this is called over-fitting the data and can make your model look great on the training data, but it has no ability to generalize on new data that is given. To gain perspective into how the model is doing we use a train/test split. Simply put we just select a certain percentage of our data and withhold it from the "eyes" of the machine learning algorithm. In this way, at the end of training there is still a chunk of data that we can test the model on and see how it performs in comparison to the training data. Often when you compare the score of the test and training data the training data will have a lower score but in good models the test score should be close by. Let's test this quickly with the logistic regression classifier from Scikit-Learn.

Let's first import the train_test_split model from the Scikit library. Then we assign the "features" or columns that we want to use for prediction to "x" and our "labels" or column that we are predicting to "y". Next, we use both x and y as our data for the train_test_split function also specifying the test size that we want. Additionally, we can assign a random_state which is optional and does not affect the quality of our data.

Next, we plug our cleaned and split data into our logistic regression algorithm, which we must import. We first train or "fit" our model on the training split. Then to get a proper score of the model we score it based on the test data. You can also score the model on the training data which can give us an idea of the score points lost between the training and test runs. This can provide hints into how much the model is over-fitting.

Now that we have our data split and we know a basic way of testing our model and on the test set, we need to figure out where to start in finding the best model for our application. There are many ways to go about this, however a rather simple and good first pass attempt is to just train a few different models on the same data and see what score they each achieve "out of the box". As we will see later, we can take the top two best performing models and tune them each individually to get even better results. In order to score each of the model we will be using an indicator called Area under the Curve which will be discussed later. The code below shows each of the models I used in this example. If you are new to ML, it is not important just yet to understand how the model works under the hood but more importantly to understand how they initially score.

1. Linear Regression: - R2 Score: 0.9514303308376894
2. Polynomial Regression: - R2 Score: 0.9655409645531071 (train set), 0.9292176784490374(test set)
3. Decision Tree Regressor: - R2 Score: 0.9283821242117826

Chapter 8: - Result analysis and Discussion

So, with the help of various analysis of many models, we can get our predicted results.

1. Linear Regression: In statistics, regression is a technique that can be used to analyze the relationship between predictor variables and a response variable.
   Mean absolute error: 4753.740341419587
   Mean squared error: 33491976.48457026
   Root mean squared error: 5787.225283723648
   R2 Score: 0.9514303308376894

This means our accuracy of Linear Regression is **95.143%.**

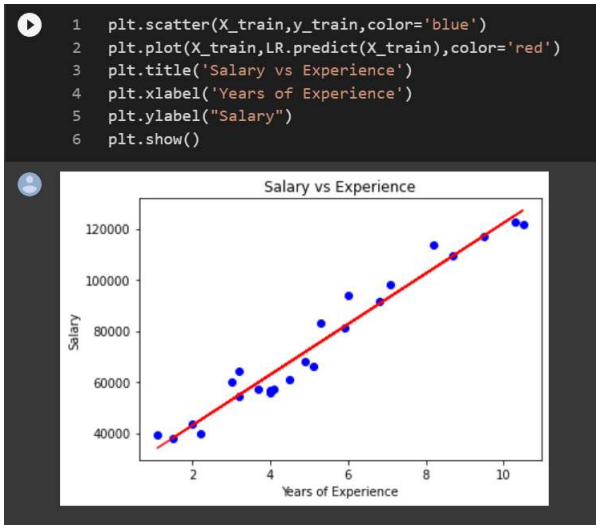


Figure 8Linear Regression Model for Training Set

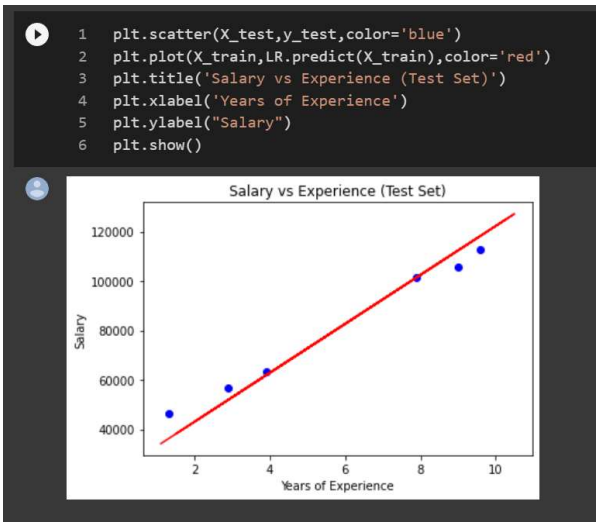


Figure 7 Linear Regression Model for Testing Set

2. Decision Tree Regressor: Decision tree builds regression 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 result is a tree with decision nodes and leaf nodes.
   Mean absolute error: 6317.583333333333
   Mean squared error: 49385228.541666664
   Root mean squared error: 7027.462453949268
   R2 Score: 0.9283821242117826

This means our accuracy of Decision Tree Regressor is 92.838%.

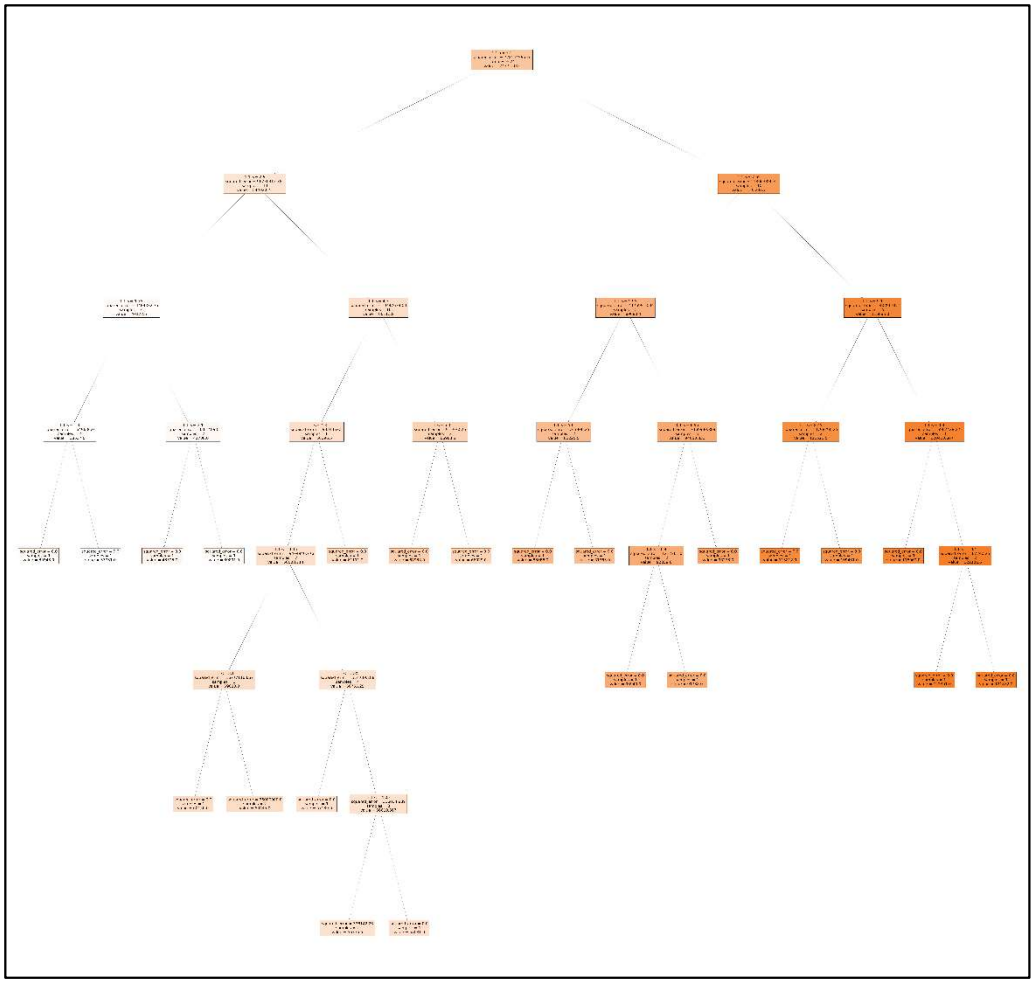


Figure 9Decision Tree Regressor

3. Polynomial Regression: Polynomial Regression is a regression algorithm that models the relationship between a dependent(y) and independent variable(x) as nth degree polynomial.
   Mean absolute error: 6607.2763311501085
   Mean squared error: 48809059.02376855
   Root mean squared error: 6986.348046280586
   R2 Score: 0.9292176784490374
   For Degree = 4

This means our accuracy of Polynomial Regression is 92.921% at Degree = 4.

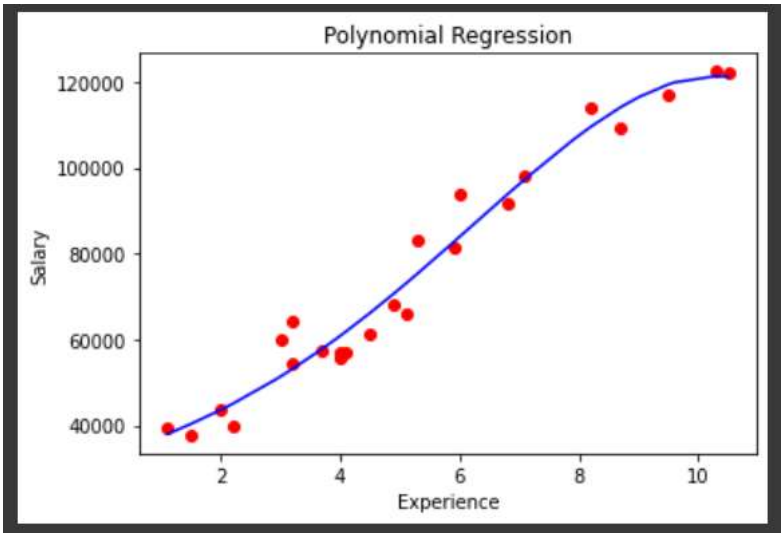


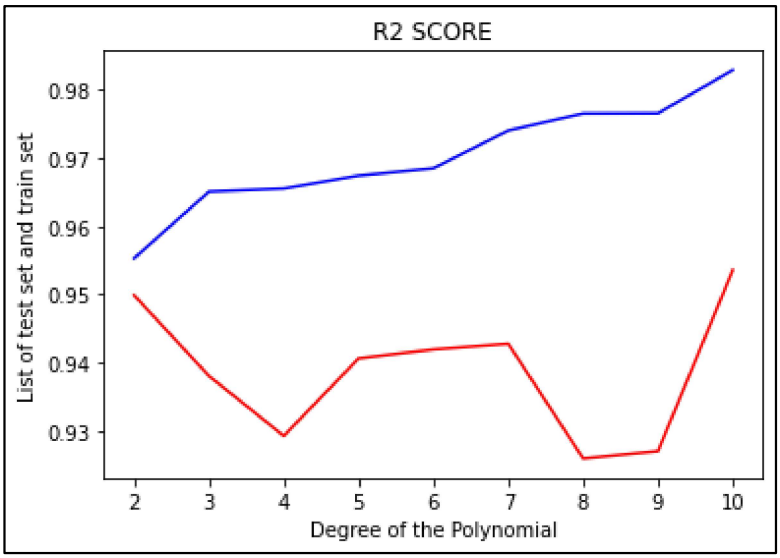
Figure 10Polynomial Regression Model for Training Set

The accuracy of Polynomial regression at various degrees is as follows.

```
[ ]   1  print("DEGREES\tR2 Score of test set\tR2 Score of train set\tSubtracted")
      2  for row in zip(numbers,list_of_test_set, list_of_train_set, subtracted) :
      3      print (str(row[0]) + "\t" + str(row[1]) + "\t" + str(row[2]) + "\t" + str(row[3]))
      4
```

```
DEGREES R2 Score of test set    R2 Score of train set   Subtracted
2       0.9498636647199424      0.955304761756877       0.005441097036934517
3       0.9380165041601258      0.9650971065646251      0.027080602404499277
4       0.9292176784490374      0.9655409645531071      0.03632328610406976
5       0.9405989041804101      0.9674386404500159      0.026839736269605785
6       0.9419112567180495      0.9685140015755337      0.02660274485748415
7       0.9427441108005268      0.9740706404312749      0.03132652963074811
8       0.9258948003308175      0.9765684285808384      0.05067362825002086
9       0.9269788799458184      0.9765951283202414      0.04961624837442302
10      0.9535742990612632      0.9829456831143499      0.029371384053086702
```

Figure 11R2 Score of all degrees in Polynomial Regression



Hence, we can easily say that our accuracy is maximum with Linear Regression Model i.e., 95.143 %.
Now we must also have a meaningful insight of our model. As accuracy is 95%, we must also implement our model in real life scenarios.
For this we defined a function named sal_pred that takes input the no. of Years or Experience in a company.

```
[38]  1  def sal_pred(a):
      2    new_sal_pred = LR.predict([[a]])
      3    print('The Predicted Salary of a person with {} years experience :- '.format(a),new_sal_pred)

[39]  1  no_yearsexperience = float(input("Enter your No. of Years "))
      2  sal_pred(no_yearsexperience)

Enter your No. of Years 7
The Predicted Salary of a person with 7.0 years experience :-  [92573.12657721]
```

Figure 12 Implementation of Our Linear Regression Model

Chapter 9: - Conclusion and Future Scope

1. In this project, we saw how we can build a machine learning model i.e., Regression model and predict the salary of the employees based on years of experience.
2. Here, we build a regression model and check the model RMSE which is equal to 5787.225283723648.
3. We also checked for R2 score of our model which is equal to 0.9514303308376894 or 95.143%. Which is an incredibly good R2 score.

Our solution helps to understand the future and capabilities of employees, it allows for this thanks to a well-chosen Linear Regression algorithm, and that's why we obtained about 95.143% final accuracy in each dataset. As a result, we can find out above all whether we should work more to improve our results through education to have opportunities in the future for decent working conditions. However, it should be remembered that many factors influence the future of a given person. Despite this, behavior during your study may answer the question of what personality a person has and thus predict the fortune.

In the future, we aim to further develop the project in such way that it can be determined at a young age whether a person will succeed. Some of the methods gave very bad results and some of the methods very good, even if all of them are very similar in assumptions. Therefore, in the future research we will concentrate on defining relations between input data and training procedure to achieve the best result of classification. We aim to include more models in future such as using Deep Learning Techniques.

Chapter 10: - References

I. Sananda Dutta, Airiddha Halder, Kousik Dasgupta," Design of a novel Prediction Engine for predicting suitable salary for a job" 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN).

II. Pornthep Khongchai, Pokpong Songmuang, "Improving Students' Motivation to Study using Salary Prediction System" 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)

III. Phuwadol Viroonluecha, Thongchai Kaewkiriya," Salary Predictor System for Thailand Labour Workforce using Deep Learning" The 18th International Symposium on Communications and Information Technologies (ISCIT 2018)

IV. https://www.datascience2000.in/2021/05/employee-salary-prediction-in-machine.html

V. http://ijasret.com/VolumeArticles/FullTextPDF/842_47._SALARY_PREDICTION_USING_MACHINE_LEARNING.pdf

VI. https://www.analyticsvidhya.com/blog/2021/05/all-you-need-to-know-about-your-first-machine-learning-model-linear-regression/

VII. https://towardsdatascience.com/machine-learning-basics-polynomial-regression-3f9dd30223d1

VIII. https://towardsdatascience.com/machine-learning-basics-decision-tree-regression-1d73ea003fda