**PREDICTION OF SALARY HIKE BASED ON EXPERIENCE**

****

**Prepared by: Tarun Mondal**

**Table of Contents**

[Objective 3](#_Toc174732908)

[Methodology 3](#_Toc174732909)

[End-to-end process with solution architecture 3](#_Toc174732910)

[Importing libraries in Python 3](#_Toc174732911)

[Data exploration 4](#_Toc174732912)

[Data preprocessing 5](#_Toc174732913)

[Exploratory data analysis (EDA) 5](#_Toc174732914)

[Model development 8](#_Toc174732915)

[Time Taken 10](#_Toc174732916)

[Challenges faced 10](#_Toc174732917)

[Complexity level 10](#_Toc174732918)

# Objective

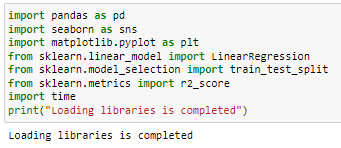
The objective of this project is to build a prediction model for Salary Hike based on experience using machine learning.

# Methodology

The main purpose of this project is to predict salary hikes using experience based on a fictional dataset. The dataset contains 30 observations and 2 variables (Salary and YearsExperience). A machine learning approach has been adopted for predicting Salary based on Years of Experience. The target variable in this project is ‘numerical (continuous)’, suggesting the selection of a regression model. Due to the presence of a single feature (independent variable) (‘YearsExperience’), a simple linear regression model has been developed for predicting ‘Salary’ using Python programming language in Jupyter Notebook IDE.

# End-to-end process with solution architecture

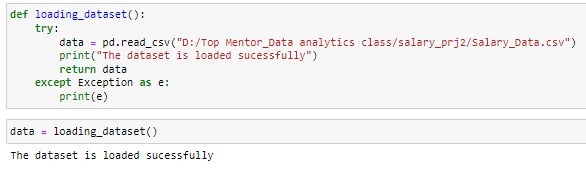
## Importing libraries in Python

******

***Figure 1: Importing libraries in Python***

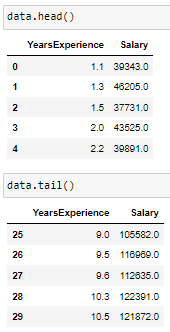
Pandas library has been imported into Python for data loading and manipulation, whereas Seaborn and Matplotlib libraries have been used for data visualisations. For developing a machine learning model (Linear regression), ‘LinearRegression’ module has been imported from scikit-learn framework, whereas the evaluation metric (R2-score) has been imported from ‘sklearn.metrics module (***Refer to Figure 1***).

## Data exploration

******

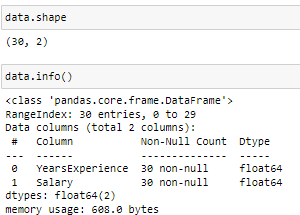
***Figure 2: Data loading***

The dataset has been loaded in Python (Jupyter Notebook Environment) using the ‘read’ function from pandas library (***Refer to Figure 2***). Try-catch block has been used for better handling of errors.

******

***Figure 3: Data loading***

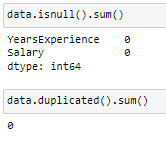
The head and tail of the ‘Salary’ dataset have been calculated using the ‘head ()’ and ‘tail ()’ functions from pandas library (***Refer to Figure 3***).

******

***Figure 4: Shape and info of the dataset***

The shape and info of the dataset have been calculated using the ‘shape’ method and ‘info ()’ function from pandas library, from which it can be stated that the dataset contains 30 valid observations and 2 variables (YearsExperience and salary) (***Refer to Figure 4***).

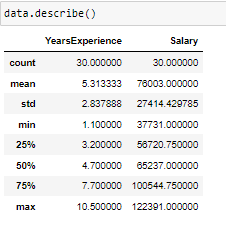
## Data preprocessing

******

***Figure 5: Missing and duplicate values in the dataset***

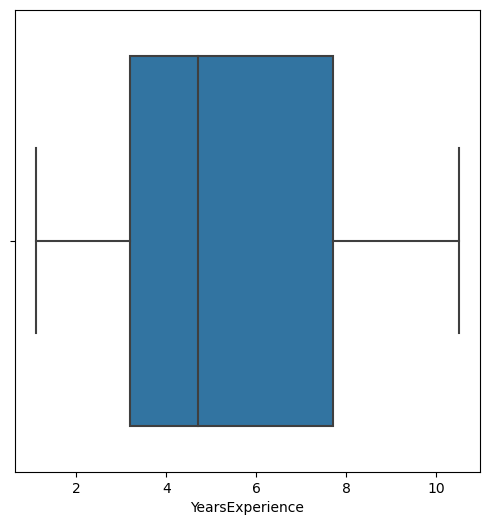
Missing values in the ‘Salary’ dataset have been checked using ‘isnull ().sum ()’ function from pandas library, from which it can be observed that the dataset contains 0 missing values for both variables. This indicates that data errors are absent in the dataset, thus, data cleaning steps like dropping null values or filling the null values are not adequately required in this project. Additionally, duplicate values have been checked and no duplicate values have been identified.

## Exploratory data analysis (EDA)

******

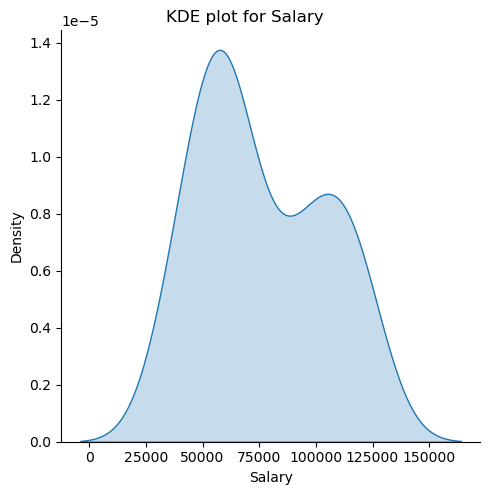
***Figure 6: Summary statistics***

The mean value of ‘YearsExperience’ is 5.31 with a standard deviation of 2.83. On the other hand, the mean salary is 76003.00 with a standard deviation of 27.414.42, indicating a substantial variability in salary of the employees.

******

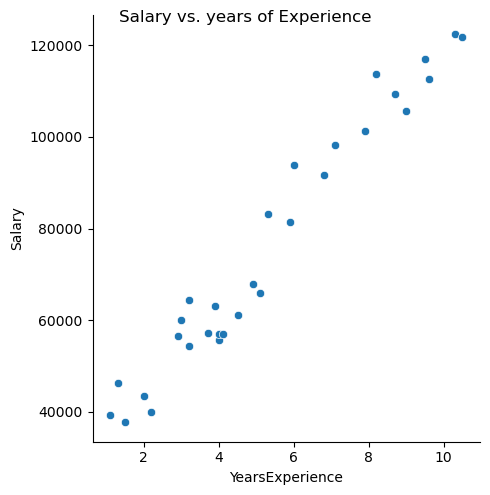
***Figure 7: Box plot for ‘YearsExperence’***

Box plot for the variable ‘YearsExperience’ indicates that no outliers are present in the dataset.

******

***Figure 8: Kernel density plot for Salary***

Kernel density plot for salary shows that the variable has followed a normal distribution and salary of the majority of the employees lies in the range of 25000 to 75000.

******

***Figure 9: Scatter plot between ‘YearsExperience’ and ‘Salary’***

Scatter plot between ‘YearsExperience’ and ‘Salary’ shows a linear upward relationship, indicating with enhancement of years of experience, salary of the employees has increased.

## Model development

******

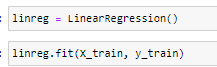
***Figure 10: Feature and target***

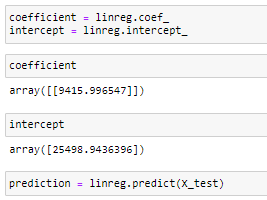
Feature and target variable have been selected and reshaped to (-1,1) to convert them into vertical arrays.

******

***Figure 11: train-test split***

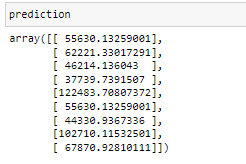
70-30% splitting has been performed by using the train\_test\_split function from ‘scikit-learn’ framework, through which 70% of data has been used to train the model and remaining 30% of data has been used to test the model performance. A random state of 1234 has been considered to maintain the same output across multiple runs.

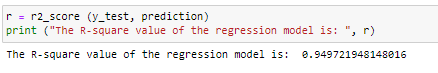


******

***Figure 12: Simple linear regression model***

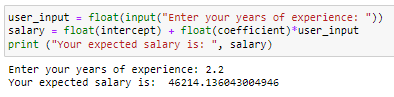
The coefficient and intercept obtained from the simple linear regression model are 9415.99 and 25498.943 (***Refer to Figure 12***). Thus, the equation of the regression line is: Salary = 25498.943 + (9415.99 \*YearsExperience).



******

***Figure 13: R-square***

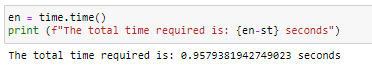
The R-square of the regression model is 0.9497, indicating that within the model, feature (YearsExperience) has explained almost 94.97% variability in the target variable (Salary).

******

***Figure 14: Prediction***

An employee having an experience of 2.2 years, is expected to earn a salary of 46214.136 (***Refer to Figure 14***). Using this model, users can estimate their salary based on years of experience they are having, allowing them to make informed decisions regarding joining a company in the offered salary or not.

# Time Taken

******

***Figure 15: Execution time***

The total time elapsed for the complete execution of the code is 0.9579 seconds, indicating high speed of the model in predicting salary.

# Challenges faced

No such challenges have been faced in this project.

# Complexity level

The dataset is very small and contains only 2 variables with no data errors (such as missing values, duplicate values and outliers), reflecting the project is very simple.