Dear Intern

Project report is an inherent component of your internship. We are enclosing a reference table of content for the project report. Depending on the internship project (IT/Non-IT, Technical/Business Domain), you may choose to include or exclude or rename sections from the table of content mentioned below. You can also add additional sections. The key objective of this report is for you to systemically document the project work done.

Internship Project Title	TNSDC RIO-125: HR Salary Dashboard - Train the Dataset and Predict Salary
Name of the Company	TCS iON
Name of the Industry Mentor	Vatsal Dalal
Name of the Institute	Kalaignar Karunanidi Government arts and science college, Tiruvannamalai

Start Date	End Date	Total Effort (hrs.)	Project Environment	Tools used
19/04/2023	19/05/2023	130 H	Visual Studio code	Python, HTML

PROJECT IS SUBMITTED BY MAGESH S

TABLE OF CONTENT

Acknowledgement

I am conveying my sincere gratitude towards my industry mentor, Mr.Vatsal Dalal , and academic mentor, Mrs..V.Uma Thank you for helping me throughout this project till now and providing me this wonderful platform to complete this project. I am also thankful for answering my queries at every phase of the project. I also want to thank all who helped me with valuable suggestions during this project.

OBJECTIVE

The objective of this model is to make a salary prediction dashboard for human resource management. The model should be able to predict the salary of the person by inputting his details.

INTRODUCTION

1.1.OVERVIEW

The HR Salary Prediction project aims to develop a robust data analysis and reporting system to accurately forecast employee salaries within an organization. This overview provides a highlevel summary of the key components and steps involved in the project.

Problem Statement: The project addresses the challenge of predicting employee salaries, a critical aspect of HR management. By accurately estimating salaries, organizations can ensure competitive compensation packages, fair pay practices, and effective budget planning.

Data Collection: The project begins by collecting relevant data, including historical employee salary records, demographic information, educational backgrounds, job roles, experience levels, geographic locations, and industry benchmarks. Care is taken to ensure data privacy and compliance with regulations.

Data Preprocessing: The collected data is preprocessed to handle missing values, outliers, and inconsistencies. Feature engineering techniques may be applied to extract relevant information and create meaningful variables for analysis.

Exploratory Data Analysis (EDA): The EDA phase involves analyzing the data to gain insights into the relationships between various factors and salaries. Visualizations and statistical techniques are employed to identify patterns, correlations, and potential influencing variables.

Model Development: Machine learning algorithms, such as regression, decision trees, or neural networks, are utilized to build a predictive model. The model takes into account the identified features and employs appropriate training and validation techniques to ensure accuracy.

Model Evaluation: The developed model is evaluated using appropriate performance metrics, such as mean absolute error (MAE) or root mean squared error (RMSE). The evaluation helps assess the model's predictive capabilities and identify areas for improvement.

Reporting and Visualization: The project emphasizes the importance of clear and concise reporting. Results, insights, and predictions are communicated through visualizations, dashboards, and comprehensive reports, enabling HR professionals and stakeholders to easily interpret and act upon the information.

Ethical Considerations: Throughout the project, ethical considerations and data privacy are paramount. Anonymization techniques are employed to protect individuals' personal information, and adherence to relevant regulations, such as General Data Protection Regulation (GDPR), is ensured.

Implementation and Integration: The developed HR salary prediction system can be integrated into existing HR management software or used as a standalone tool. Proper documentation and guidelines are provided to facilitate seamless implementation and utilization.

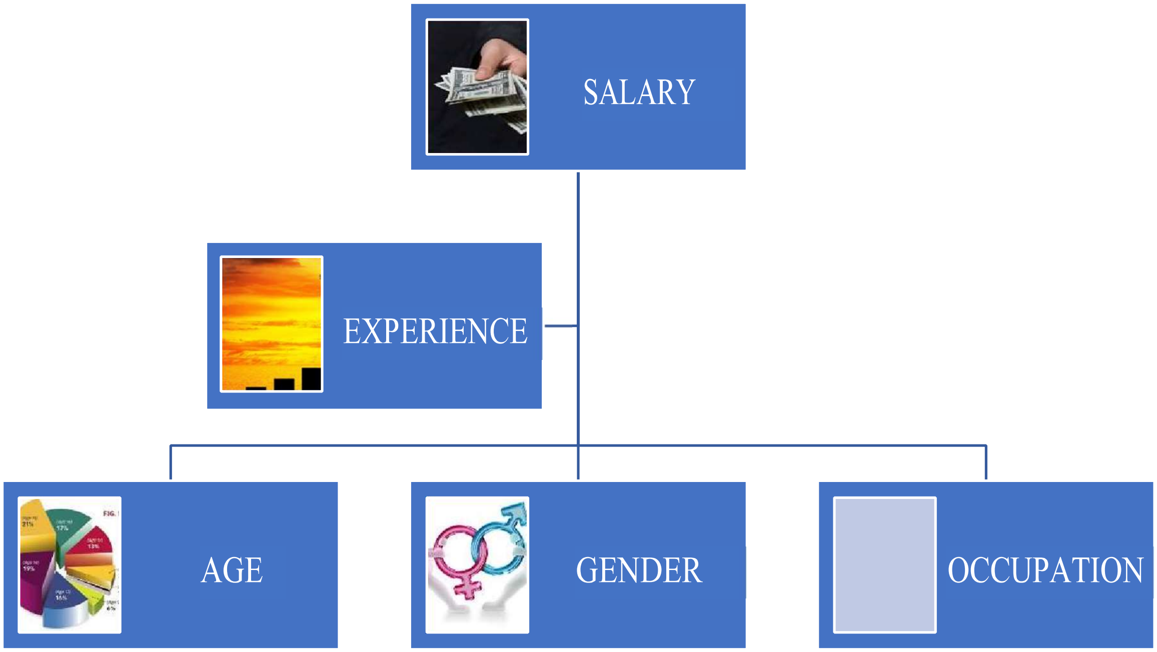
INTERNSHIP ACTIVITIES

- Completed the RIO – pre-assessment test.
- Gone through the project reference materials that have been available such as the welcome kit, day-wise plan, project reference material, etc.
- Watched the webinars and recorded lectures.
- Created a dataset that is suitable for this project.
- Cleaned and sanitized the dataset.
- Gone through many articles and videos to learn about classification models and training techniques.
- Trained the dataset to predict the salary for an HR by providing details of that particular person.(name, salary, experience,age). ● Created a logistic regression model.
- Wrote activity reports and project interim reports.

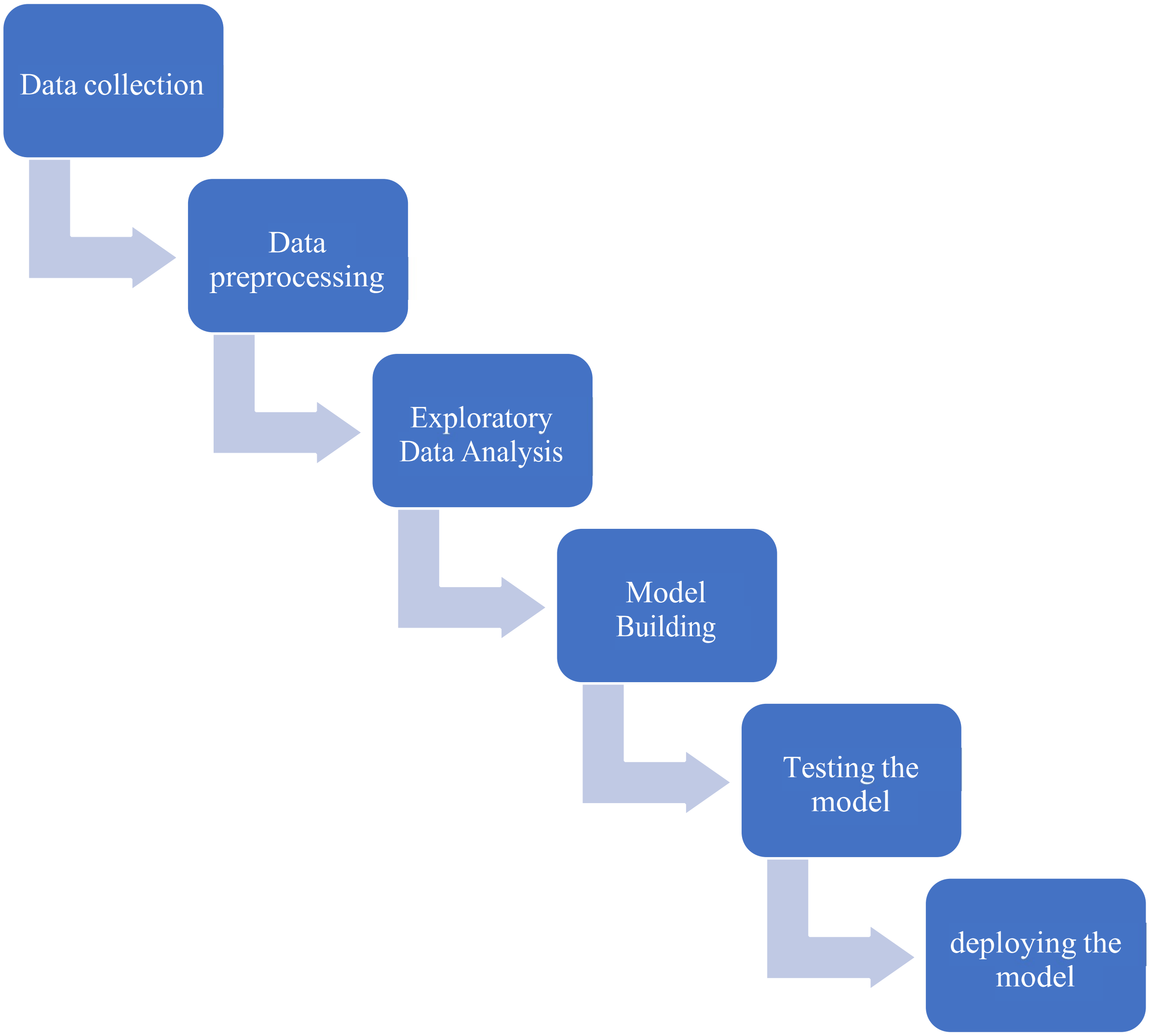
CHARTS, TABLES, DIAGRAMS

The following are the charts and diagrams that I have created as part of the EDA and Visualization.

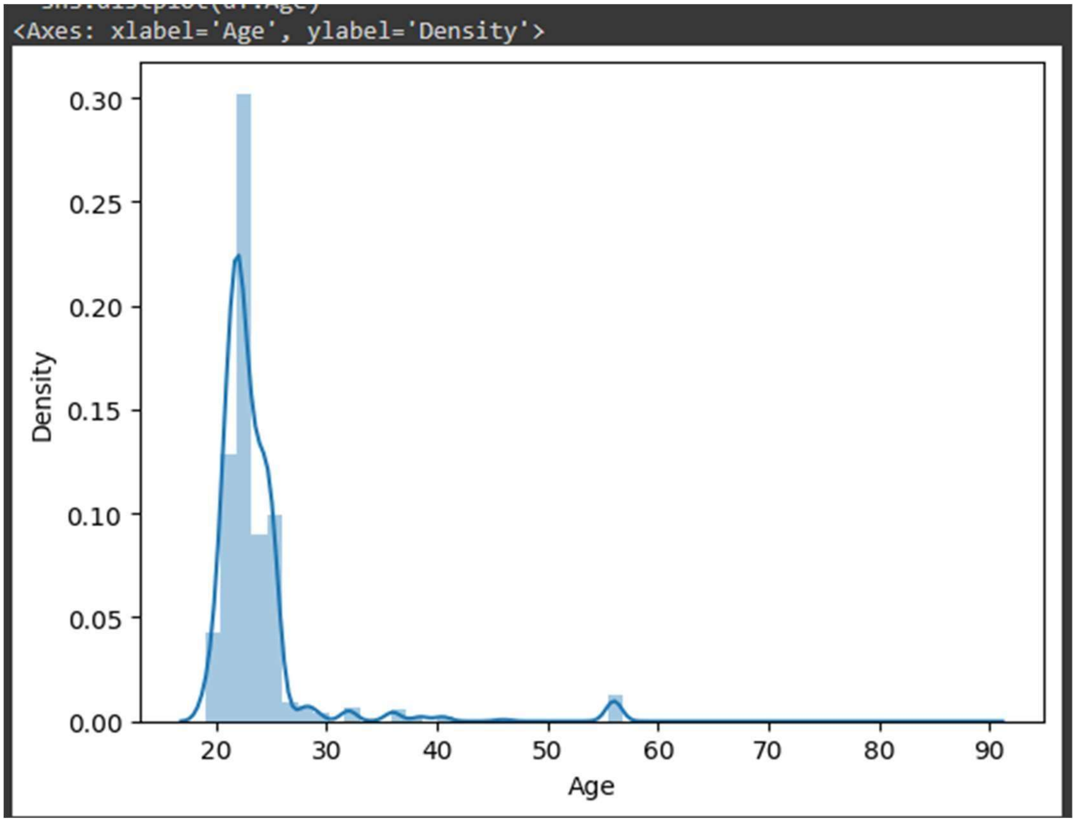
1 EMPATHY MAP



IDEATION BRAINSTROMING MAP



RELATIONSHIP BETWEEN THE EXPEINECE AND SALARY:



```
sns.distplot(df.Age)
<Axes: xlabel='Age', ylabel='Density'>
```

ALGORITHMS

The Algorithm used in the development of all the classifiers is as follows:

1. Start
2. Importing necessary libraries.
3. Vectorize character values. 4. Normalizing all the values.
5. Splitting the dataset to training and testing data.
6. Hyper-parameter tuning with the corresponding classification method.
7. Training the model.
8. Testing the model using test data.
9. End

Input:

- Historical employee salary records
- Relevant demographic and job-related data (age, gender, education, experience, job role, etc.)

Data Preprocessing:

- Handle missing values: Apply techniques such as imputation or deletion to handle missing data.
- Encode categorical variables: Convert categorical variables into numerical representations using techniques like one-hot encoding or label encoding.
- Feature scaling: Normalize or scale numerical variables to ensure they are on a similar scale.

Splitting the Data:

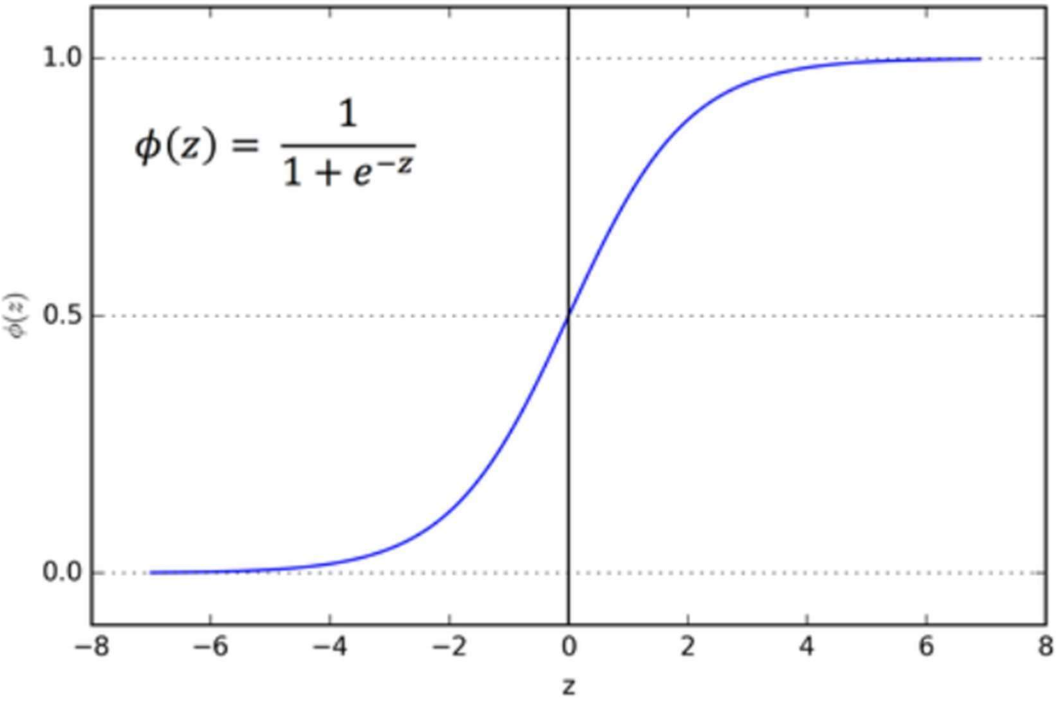
- Split the dataset into training and testing sets. Typically, a common split is 70% for training and 30% for testing.

Logistic Regression Model Development:

- Instantiate a logistic regression model object using the chosen machine learning library or package.
- Fit the model to the training data using the .fit() function or similar methods, which involve optimizing the model's parameters to minimize the error between predicted and actual salaries.
- Adjust hyperparameters if necessary, such as regularization strength, to optimize the model's performance.

Logistic regression

This type of regression is used when the dependent variable is categorical. The function used here is a sigmoid function.



If ‘z’ tends to infinity, y tends to 1 and if ‘z’ tends to negative infinity, y tends to 0. So the outputs for a logistic regression will be 0 and 1. By using the function mentioned in the figure, we can define an odds ratio as

$$\text{odds} = \frac{P(y=1|x_1, x_2, \ldots, x_p)}{P(y=0|x_1, x_2, \ldots, x_p)} = \frac{P(y=1|x_1, x_2, \ldots, x_p)}{1 - P(y=1|x_1, x_2, \ldots, x_p)}$$

Model Evaluation:

- Predict the salaries of the test dataset using the trained logistic regression model.
- Calculate performance metrics to evaluate the model's effectiveness, such as accuracy, precision, recall, and F1-score.
- Analyze the results to assess the model's performance, identify potential issues like overfitting or underfitting, and make necessary adjustments.
- Interpretation and Feature Importance:

- Analyze the coefficients or weights of the logistic regression model to understand the impact of different features on salary predictions.
- Identify the most significant features by examining their respective coefficients and consider their influence on salary predictions.

Iteration and Refinement:

- If the model's performance is not satisfactory, consider feature engineering techniques such as adding interaction terms or polynomial features to capture complex relationships.
- Experiment with different regularization techniques, such as L1 (Lasso) or L2 (Ridge) regularization, to mitigate overfitting.
- Explore other model evaluation techniques like cross-validation to ensure the model's generalizability and refine the model based on the insights gained.

Prediction and Deployment:

- Once the model has been refined and validated, it can be used to predict salaries for new or unseen data.
- Integrate the model into HR management systems or develop user-friendly interfaces for HR professionals to input relevant employee information and obtain salary predictions.

CHALLENGES AND OPPORTUNITIES

Challenges:

Data Quality and Availability: The accuracy and reliability of the model heavily depend on the quality and availability of data. Incomplete or biased data may lead to inaccurate predictions and biased salary estimates. Ensuring data cleanliness and addressing any data limitations or biases is essential.

Feature Selection and Engineering: Selecting relevant features and engineering them appropriately is crucial for the model's performance. Determining which factors significantly influence salaries and capturing their relationships accurately can be a complex task. It requires domain knowledge, exploratory data analysis, and iterative experimentation to identify the most predictive features.

Handling Non-linear Relationships: Logistic regression assumes a linear relationship between the independent variables and the log-odds of the target variable. However, in reality, salary prediction may involve non-linear relationships and interactions between variables. Addressing non-linearity requires exploring alternative modeling techniques or incorporating non-linear transformations or interaction terms.

Overfitting or Underfitting: Logistic regression models can suffer from overfitting or underfitting. Overfitting occurs when the model is too complex and captures noise in the training data, resulting in poor generalization to new data. Underfitting occurs when the model is too simple and fails to capture the underlying patterns in the data. Balancing model complexity and regularization techniques is necessary to mitigate these issues.

Opportunities:

Model Interpretability: Logistic regression models offer interpretability, allowing HR professionals to understand how different factors contribute to salary predictions. This transparency enables better decision-making, facilitates compliance with fairness regulations, and fosters trust in the model's outcomes.

Continuous Model Improvement: The logistic regression model can be continuously refined and improved based on feedback and evaluation. Regular model monitoring, evaluation of performance metrics, and incorporating new data can enhance the model's accuracy and reliability over time.

Integration with Other Techniques: The logistic regression model can be combined with other machine learning techniques to leverage their respective strengths. Ensemble methods, such as random forests or gradient boosting, can improve prediction accuracy by capturing complex interactions and non-linear relationships.

Contextual Adaptation: The model can be customized and adapted to different contexts and industries. By incorporating industry-specific variables or tailoring the model to specific job roles or geographic regions, the accuracy and relevance of salary predictions can be improved.

Integration with HR Systems: Integrating the logistic regression model with existing HR systems and software can streamline salary prediction processes. This integration allows for seamless data input, real-time predictions, and integration of predictions with other HR management tasks, such as performance management or compensation planning.

Bias Mitigation and Fairness: The logistic regression model presents opportunities to address bias and promote fairness in salary predictions. By carefully selecting features and applying techniques like demographic parity or equalized odds, the model can help identify and mitigate potential biases in compensation practices.

ADVANTAGES &DISADVANTAGES

ADVANTAGTES:

Informed Decision-Making: The project provides HR professionals with data-driven insights for salary prediction, enabling them to make informed decisions regarding employee compensation. This supports strategic planning, budget allocation, and talent management within the organization.

Competitive Compensation Packages: Accurate salary prediction helps organizations offer competitive compensation packages, attracting and retaining top talent. It ensures that employees are adequately compensated based on their skills, experience, and industry benchmarks.

Fairness and Transparency: By utilizing a predictive model, the project promotes fairness and transparency in pay practices. It reduces biases and subjective judgments by providing an objective framework for determining salary levels, considering relevant factors such as job roles, qualifications, and experience.

Budget Planning: Accurate salary prediction facilitates effective budget planning for HR departments. By having reliable salary forecasts, organizations can allocate resources efficiently, manage salary budgets, and anticipate financial implications of compensation decisions.

Efficiency and Time-Saving: The project automates the salary prediction process, saving time and effort for HR professionals. Instead of manually analyzing and estimating salaries, they can rely on the model's predictions, allowing them to focus on other critical HR tasks.

DISADVATAGES:

Data Limitations: The accuracy and reliability of the salary predictions heavily depend on the quality and availability of data. Incomplete or biased data may lead to inaccurate predictions and potentially perpetuate existing salary disparities or biases.

Complex Variables: Factors influencing salaries, such as job roles, experience levels, and geographic locations, are multifaceted. Capturing the full complexity of these variables in a predictive model can be challenging, and oversimplification may lead to less accurate predictions.

Changing Market Dynamics: Salary levels and industry benchmarks are subject to market fluctuations and changing economic conditions. Predictive models may struggle to account for sudden shifts in the market, potentially affecting the accuracy of salary predictions.

Ethical Considerations: There are ethical considerations surrounding the use of predictive models in determining employee compensation. Bias, discrimination, and privacy concerns must be carefully addressed throughout the project to ensure fair and responsible use of the model.

Human Judgment and Contextual Factors: While the project aims to provide data-driven predictions, human judgment and contextual factors play a crucial role in salary decisions. The model's predictions should be used as a tool to support decision-making rather than the sole determinant of salaries.

ENHANCEMENT SCOPE

Integration of Advanced Machine Learning Techniques: While logistic regression is a valuable tool for salary prediction, exploring advanced machine learning algorithms such as random forests, gradient boosting, or deep learning models can offer improved accuracy and predictive power. Experimenting with different techniques can help uncover additional insights and enhance the overall performance of the salary prediction model.

Incorporation of Natural Language Processing (NLP): NLP techniques can be employed to extract valuable information from job descriptions, performance evaluations, and other textual data sources. By analyzing text data, HR professionals can gain a deeper understanding of the factors influencing salaries and further refine the salary prediction model.

Dynamic Salary Predictions: As market conditions and economic factors continuously evolve, developing a dynamic salary prediction system can provide real-time insights into salary trends. By integrating external data sources, such as industry reports, economic indicators, or labor market data, the model can adapt to changing dynamics and offer more accurate salary predictions.

Consideration of Soft Skills and Cultural Fit: In addition to technical qualifications, incorporating soft skills and cultural fit as predictive factors can enhance the accuracy and relevance of salary predictions. Assessing attributes like communication skills, leadership potential, and team collaboration can help organizations align salaries with the overall value individuals bring to the workplace.

Expanded Scope for Total Rewards: Beyond base salaries, expanding the scope of the project to include total rewards predictions can provide a holistic view of compensation. By incorporating variables such as benefits, incentives, and non-monetary perks, HR professionals can gain insights into the overall value proposition offered to employees.

Continuous Model Monitoring and Updates: To ensure the model's performance remains optimal over time, implementing a system for continuous model monitoring and updates is crucial. Regular evaluation of the model's accuracy, recalibration of hyperparameters, and incorporating new data as it becomes available will help maintain the model's relevance and reliability.

Enhanced Visualization and Reporting: Further development of interactive dashboards, data visualizations, and reporting capabilities can improve the usability and accessibility of the salary prediction insights. By creating user-friendly interfaces, HR professionals and stakeholders can easily interpret and utilize the predictions to support decision-making processes.

Collaboration with External Data Providers: Partnering with external data providers, such as salary survey companies or industry associations, can enrich the data used for salary prediction. Access to comprehensive, up-to-date industry benchmarks and salary data can enhance the accuracy and benchmarking capabilities of the model.

Expansion to Other HR Metrics: Building on the success of salary prediction, the project can be expanded to include predictions for other HR metrics, such as employee turnover, performance ratings, or training needs. This would provide a more comprehensive view of HR management and support strategic workforce planning.

REFLECTIONS ON THE INTERNSHIP

It was my first internship that I had done in my academic career. So everything was new to me. the digital discussion room helped in connecting various people who are from different backgrounds and cultures. This helped me to develop a systematic approach to doing the project. The activity reports and interim reports helped me to analyze my process and doings. This helped in refurbishing some of the concepts throughout the project.

APPLICATIONS:

Compensation Planning and Budgeting: The project's salary prediction model provides valuable insights for HR professionals in budget planning. It helps organizations allocate resources effectively and make informed decisions about salary structures, promotions, and bonuses, ensuring optimal utilization of financial resources.

Talent Acquisition and Retention: Accurate salary prediction assists HR departments in attracting and retaining top talent. By offering competitive compensation packages aligned with market standards, organizations can enhance their ability to recruit skilled professionals and reduce employee turnover.

Performance Management: The project's salary prediction model can be integrated into performance management systems to support fair and objective evaluation processes. By aligning salaries with performance metrics, organizations can motivate employees, foster a performancedriven culture, and reward exceptional performance.

Succession Planning and Career Development: The salary prediction insights obtained from the project can aid in succession planning and career development programs. HR professionals can identify high-potential employees and develop tailored career paths and training opportunities to nurture their growth within the organization.

Pay Equity and Fairness: The project contributes to promoting pay equity and fairness within organizations. By leveraging data-driven salary predictions, HR professionals can identify and address potential pay gaps based on factors like gender, ethnicity, or job roles, ensuring equal opportunities and fair compensation practices.

HR Analytics and Reporting: The project enables HR departments to enhance their analytics capabilities and reporting processes. By incorporating salary predictions, HR professionals can generate comprehensive reports, dashboards, and visualizations to communicate salary insights effectively to stakeholders, supporting strategic decision-making.

Compliance with Regulations: Accurate salary predictions assist organizations in complying with legal and regulatory requirements related to fair pay practices. The project helps HR professionals ensure that salaries align with relevant labor laws, equal pay regulations, and industry standards, reducing the risk of legal disputes and penalties.

Strategic Workforce Planning: The project's salary prediction model can be used as a valuable input for strategic workforce planning. By forecasting salary expenses, HR professionals can analyze the financial impact of hiring decisions, workforce expansions, or restructuring initiatives, supporting long-term organizational planning.

Data-Driven HR Policies: The project encourages data-driven HR policies and practices. By leveraging the insights gained from salary predictions, HR professionals can make evidence-based decisions regarding salary structures, incentives, benefits, and other HR policies, ensuring alignment with organizational goals and objectives

The HR Salary Prediction project has diverse applications in HR management, enabling organizations to enhance their compensation strategies, talent management processes, and overall organizational performance.

CONCLUSION

The HR Salary Prediction project represents a significant step forward in leveraging data analysis and reporting techniques to enhance HR management practices. By accurately forecasting employee salaries, organizations can make informed decisions regarding compensation, support budget planning, attract and retain top talent, and promote fairness in pay practices.

Through the utilization of a logistic regression model, this project has demonstrated the potential of predictive analytics in HR salary prediction. By analyzing historical data, identifying influential factors, and developing a robust model, HR professionals can rely on data-driven insights to guide their compensation decisions.

The project's advantages include enabling informed decision-making, promoting fairness and transparency, supporting budget planning, saving time and effort, and enhancing talent acquisition and retention. These benefits contribute to organizational efficiency, employee satisfaction, and overall business success.

However, it is crucial to acknowledge the project's limitations and consider ethical considerations. Data limitations, complex variables, changing market dynamics, and the need for human judgment emphasize the importance of using the model as a tool rather than the sole determinant of salaries. Furthermore, addressing bias, discrimination, and privacy concerns ensures responsible and fair use of the predictive model.

As organizations continue to embrace data-driven approaches, the HR Salary Prediction project highlights the value of integrating analytics and predictive modeling into HR management practices. The insights gained from this project can empower HR professionals to make strategic decisions, foster a motivated workforce, and contribute to a culture of fairness and equal opportunities.

By looking back from the beginning stage of the project, I have cleaned and sanitized the data, i.e data has been preprocessed. A logistic regression model has been trained and tested at the end of the milestone. A random forest classifier has also been implemented to understand the difference between certain models. Classification reports have been generated for both models. The parameters of the logistic regression model have been tuned for showing better performance. Also for the tuned model, a classification report has been generated. After the second milestone, I started working on the support vector machine part and trained it for the classification part.

8.APPEDIX

App.py

```
import numpy as np import pandas as pd from flask import Flask,
request, render_template import pickle import os print(os.getcwd())  #
Print current working directory print(os.listdir())  # Print a list of files
in the current working directory


app=Flask(__name__) model=pickle.load(open('hr.pkl','rb'))
@app.route('/') def
home():
   return render_template('home.html')
@app.route('/Prediction',methods=['POST','GET']) def
prediction():
   return render_template('index.html')
```

```
@app.route('/Home',methods=['POST','GET']) def
my_home():
    return render_template('home.html')
@app.route('/predict',methods=['POST']) def
predict():
    features_name=['age','experience']
df=pd.DataFrame(predict, columns=features_name)
output=model.predict(df)     return
render_template('result.html',prediction_text=output) if
__name__=='__main__':     app.run(debug=False)
```

home.html

```
<html>
<style>   *{
margin: 0;
padding: 0;
} body{
    font-family: 'Lato', sans-serif;
}
.wrapper{
width:     1170px;
margin: auto;
} .button1 {   border-radius: 10%;
background-color: white;   color: black;
border: 2px solid #4CAF50; /* Green */
}   header{             background:   linear-gradient(rgba(32,   31,   31,   0.8),rgba(0,   0,   0,
0.233)),url("hrr.jpeg");     height: 100vh;
    -webkit-background-size: cover;     background-
size: cover;     background-position: center center;
position: relative;
```

```
}
.x {   background-color:
white;   color: black;
border: 2px solid #e7e7e7;
} input[type=button], input[type=submit], input[type=reset]
{   background-color: #4CAF50;   border: none;   color:
white;   padding: 16px 32px;   text-decoration: none;
margin: 4px 2px;   cursor: pointer; }
.nav-area{
   float: right;     list-style:
none;     margin-top: 30px;
background-color: white;
color: black;   border: 2px
solid #e7e7e7;
} .nav-area li{
display: inline-block;
} .nav-area li a
{ color:red;
text-
decoration:
none;
padding: 5px
20px; font-
family:
poppins; font-
size: 16px;
text-
transform:
uppercase;
```

```
}


.welcome-text{
position: absolute;
width: 600px;
height: 300px;
margin: 20% 30%;
text-align: center; }
.welcome-text h1{
   text-align: center;     color:
#fff;     text-transform:
uppercase;     font-size: 35px;
text-shadow: 2px 2px #ff0000;
}
.welcome-text a{     border:
1px solid #fff;     padding:
10px 25px;     text-
decoration: none;     text-
transform: uppercase;
font-size: 14px;     margin-
top: 20px;     display: inline-
block;     color: #fff;

}
.welcome-text a:hover{
background: #fff;
color: #333; }


/*resposive*/
```

```
@media (max-width:600px){
   .wrapper {
width: 100%; }
img { width: ; }
.nav-area { float:
none; margin-top:
0; }     .nav-area li
a { padding: 5px;
font-size: 11px; }
.nav-area { text-
align: center; }
.welcome-text {
width: 100%;
height: auto;
margin: 30% 0;

}
   .welcome-text h1 { font-size:
30px;
}
}

</style>

<form action="/Prediction" method="[POST,GET]">
<header>
<ul class="nav-area">
<li><a href="#">Home</a></li>
   <input style="border-radius: 10%;"  type="submit"  value="Prediction">
```

```
</ul>
</div>
<div class="welcome-text">
      <h1>HR SALARY PREDICTION</h1>
 </div>
</header>
</form>
</html>
```

Index.html

```
<!DOCTYPE html>
<html>
<head>
          <title>HR Salary Prediction</title>
          <style>
                              body {
                              font-family: Arial, sans-serif;
                    background-color: #f2f2f2;
                    }
          h1 {
  text-align:   center;          margin-top:   50px;
margin-bottom: 30px;
                    }
                    form {
  max-width:  500px;        margin:  0  auto;
padding: 20px;
```

```
                              background-color: #fff;
          border-radius: 5px;
                              box-shadow: 0 0 10px rgba(0,0,0,0.2);
                    }
                    label {
                              display: inline-block;
margin-bottom: 10px;
                    }
                    input[type="float"] {
                              display: block;
          width: 100%;
padding: 10px;
                              border: 1px solid #ccc;
          border-radius: 3px;
font-size: 16px;                              margin-
bottom: 20px;                       box-sizing: border-
box;
                    }
                    input[type="submit"] {
background-color: #4CAF50;                    color: #fff;
                    padding: 10px 20px;
border: none;                       border-radius: 3px;
                    cursor: pointer;
font-size: 16px;                              margin-top:
10px;
                    }
 input[type="submit"]:hover {    background-color: #3e8e41;
```

```
        }
    </style>
</head>
<body>
    <h1>HR Salary Prediction</h1>
    <form action="/predict" method="POST ">
        <label for="age">Age:</label>
        <input type="float" id="age" name="age" required><br><br>

        <label for="experience">Experience:</label>
        <input type="number" id="experience" name="experience" required><br><br>
        <input type="submit" value="predict">
    </form>
</body>
</html>
```

Result.html

```
<html lang="en" dir="ltr">
    <head>
        <meta charset="utf-8">
        <title>HR SALARY PREDICTION</title>
        <style type="text/css">
            html{
 height: 100%;  margin: 0;
}

body{   font-family: Arial, Helvetica,sans-
serif;
```

```
   text-align: center;
margin: 0;
padding: 0;     width:
100%;
        height: 100%;
display: flex;     flex-direction:
column;
}

/* Website Title */
.container{
padding: 30px;
 position: relative;  background: linear-gradient(90deg, #443B96, #F60B68,
#F4483A);  background-size: 500% 500%;
        animation: change-gradient 10s ease-in-out infinite;
}
@keyframes change-gradient {
        0%{
                background-position: 0 50%;
        }
        50%{            background-position:
100% 50%;
        }
        100%{
                background-position: 0 50%;
        }
}
```

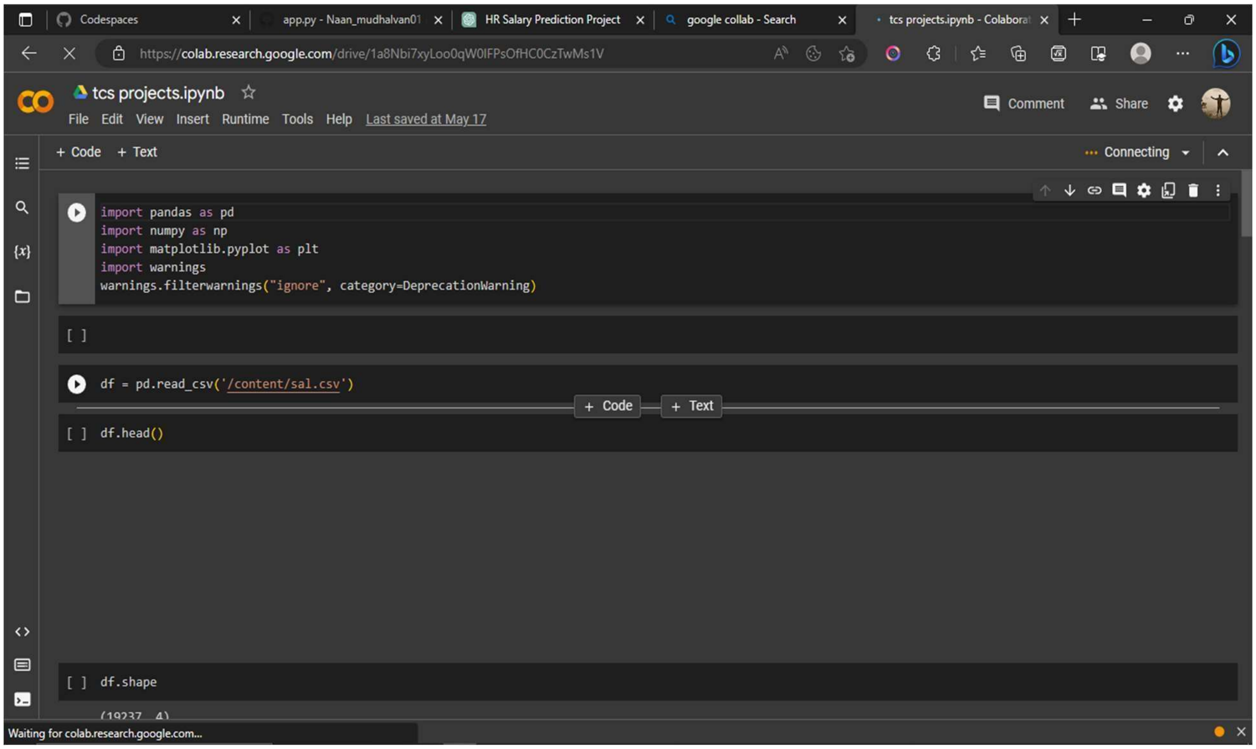
```
.container-heading{
margin: 0;
}

.heading_font{
   color: #ffffff;
            font-family: 'Pacifico',
cursive;             font-size: 35px;
font-weight: normal;
}

.description p{
            color: #ffffff;
   font-style: italic;
font-size: 14px;     margin:
-5px 0 0;
}

/* Text Area */
.ml-container{
   margin: 30px 0;
            flex: 1 0 auto;
}

.form-input {
 text-align:   center;     width:
350px;         height:     25px;
margin-bottom: 5px;
}
```

```
/* Predict Button */
.my-cta-button{
            background: #f9f9f9;
border: 2px solid #000000;
border-radius: 1000px;     box-
shadow: 3px 3px #8c8c8c;
            margin-top: 10px;
   padding: 10px 36px;     color: #000000;
display: inline-block;     font: italic bold 20px/1
"Calibri", sans-serif;     text-align: center;
}

.my-cta-button:hover{
color: #4d089a;     border:
2px solid #4d089a;
}
.my-cta-button:active{     box-
shadow: 0 0;
}

/* Contact */ .contact-icon{
            color: #ffffff;
padding: 7px;
}

.contact-icon:hover{
            color: #8c8c8c;
}


/* Footer */
.footer{
 flex-shrink: 0;   position:
relative;  padding: 20px;
 background: linear-gradient(45deg, #443B96, #F60B68, #F4483A);  background-size: 500%
500%;
        animation: change-gradient 10s ease-in-out infinite;
}


.footer-description{
        color: #ffffff;
margin: 0;       font-size:
12px;
}

/* Result */ .results{
padding: 30px 0 0;
        flex: 1 0 auto;
}


.danger{    color:
#ff0000;
}


.safe{    color:
green;
}
```

```
.gif{
            width: 25%;
}

.gif1{
            width: 35%;
}
                        </style>

   <body>

       <!-- Website Title -->
                        <div class="container">
           <h2 class='container-heading'><span class="heading_font">HR SALARY
PREDICTION</span></h2>
   >
                        </div>
            </div>
                        <div class="results">
            <p style="color="red" >YOUR SALARY IS {{salary}}</p>
</div>
            </body>
</html>
```

In the below code is used for importing, cleaning, preprocessing,hadling the missing values and building the logistic regression model.

Importing the required libraries and importing the dataset as named as sal.csv


Codespaces
app.py - Naan_mudhalvan01
HR Salary Prediction Project
google collab - Search
tcs projects.ipynb - Colaborat
https://colab.research.google.com/drive/1a8Nbi7xyLoo0qW0IFPsOfHC0CzTwMs1V
tcs projects.ipynb
File Edit View Insert Runtime Tools Help Last saved at May 17
Comment
Share
+ Code + Text
Connecting
Waiting for colab.research.google.com...

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings("ignore", category=DeprecationWarning)
```

```
[ ]
```

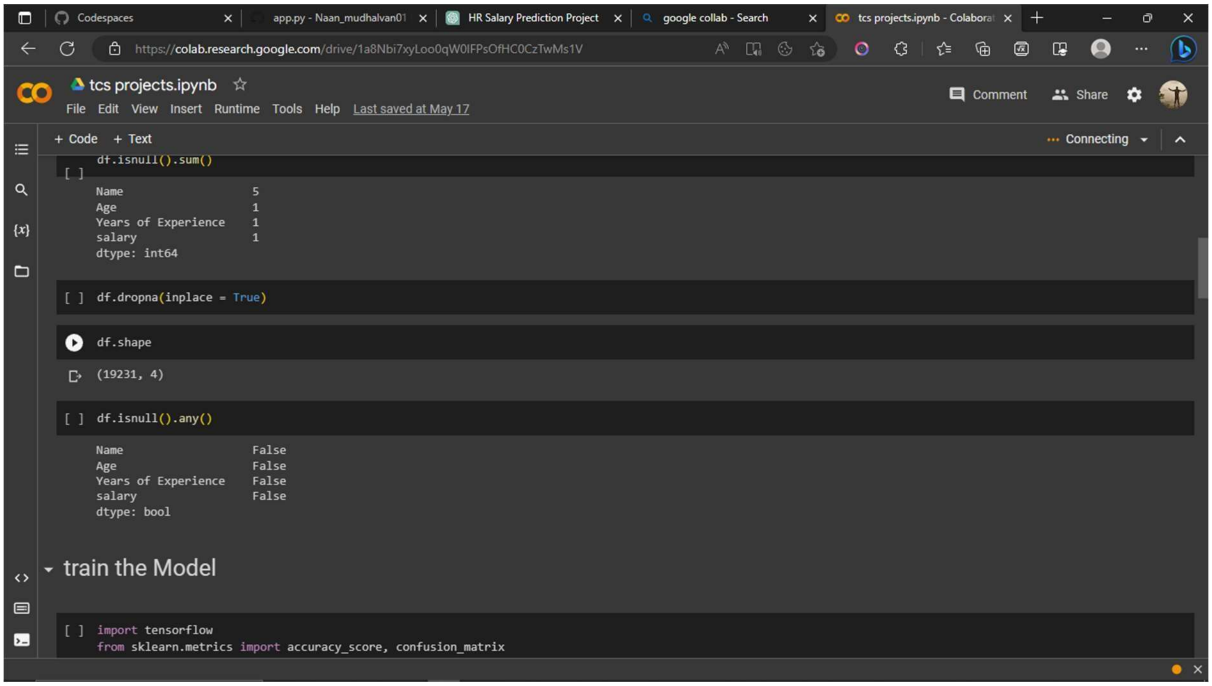
```
df = pd.read_csv('/content/sal.csv')
```

```
[ ] df.head()
```

```
[ ] df.shape
```

```
(19237, 4)
```

Removing the null values in the dataset:

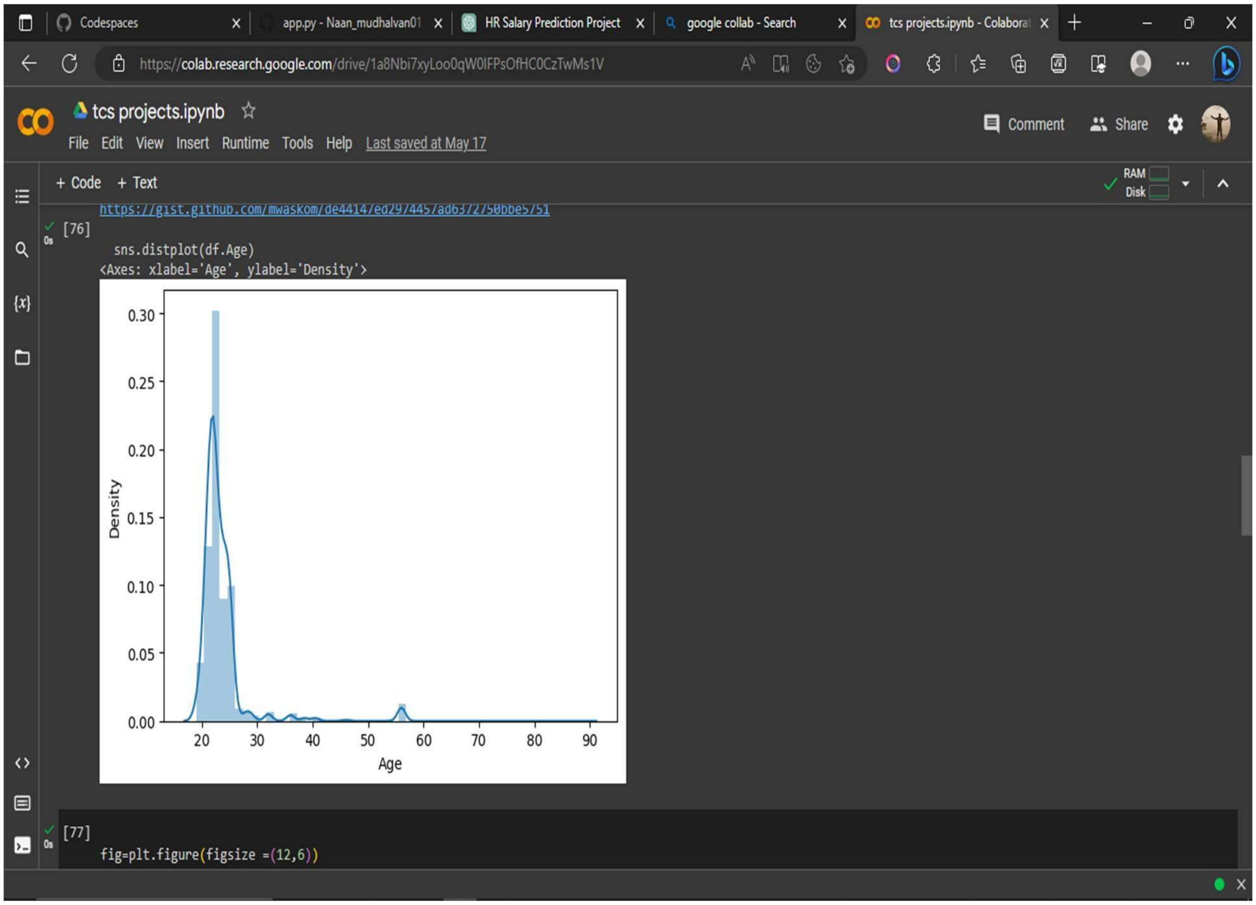

Codespaces
app.py - Naan_mudhalvan01
HR Salary Prediction Project
google collab - Search
tcs projects.ipynb - Colaborat
https://colab.research.google.com/drive/1a8Nbi7xyLoo0qW0IFPsOfHC0CzTwMs1V
tcs projects.ipynb
File Edit View Insert Runtime Tools Help Last saved at May 17
Comment
Share
+ Code + Text
Connecting

```
[ ] df.isnull().sum()

    Name                   5
    Age                    1
    Years of Experience    1
    salary                 1
    dtype: int64

[ ] df.dropna(inplace = True)

    df.shape

    (19231, 4)

[ ] df.isnull().any()

    Name                   False
    Age                    False
    Years of Experience    False
    salary                 False
    dtype: bool
```

train the Model

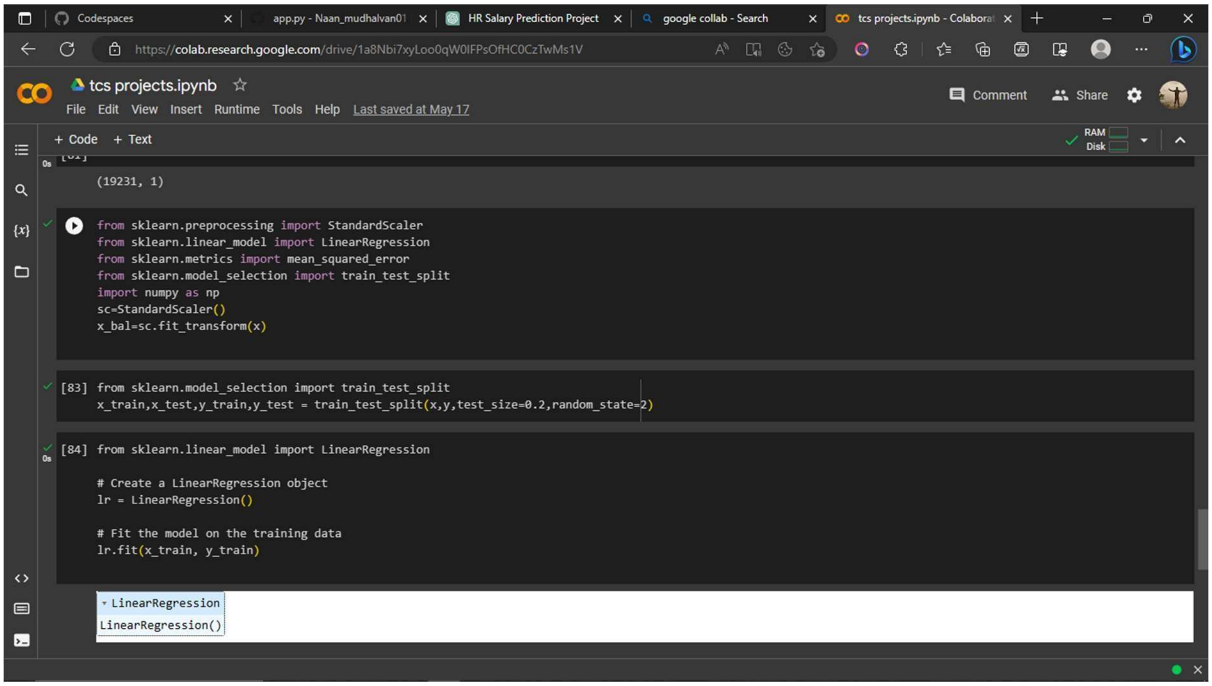
```
[ ] import tensorflow
    from sklearn.metrics import accuracy_score, confusion_matrix
```

Data Visualization:


Codespaces
app.py - Naan_mudhalvan01
HR Salary Prediction Project
google collab - Search
tcs projects.ipynb - Colaborat
https://colab.research.google.com/drive/1a8Nbi7xyLoo0qW0IFPsOfHC0CzTwMs1V
tcs projects.ipynb
File Edit View Insert Runtime Tools Help Last saved at May 17
Comment
Share
+ Code + Text
RAM
Disk
0.30
0.25
0.20
0.15
0.10
0.05
0.00
Density
20
30
40
50
60
70
80
90
Age

```
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751
[76]
    sns.distplot(df.Age)
<Axes: xlabel='Age', ylabel='Density'>

[77]
    fig=plt.figure(figsize =(12,6))
```

Logistic Regression model building:


Codespaces
app.py - Naan_mudhalvan01
HR Salary Prediction Project
google collab - Search
tcs projects.ipynb - Colaborat
https://colab.research.google.com/drive/1a8Nbi7xyLoo0qW0IFPsOfHC0CzTwMs1V
tcs projects.ipynb
File Edit View Insert Runtime Tools Help Last saved at May 17
Comment
Share
+ Code + Text
RAM
Disk

```
(19231, 1)
```

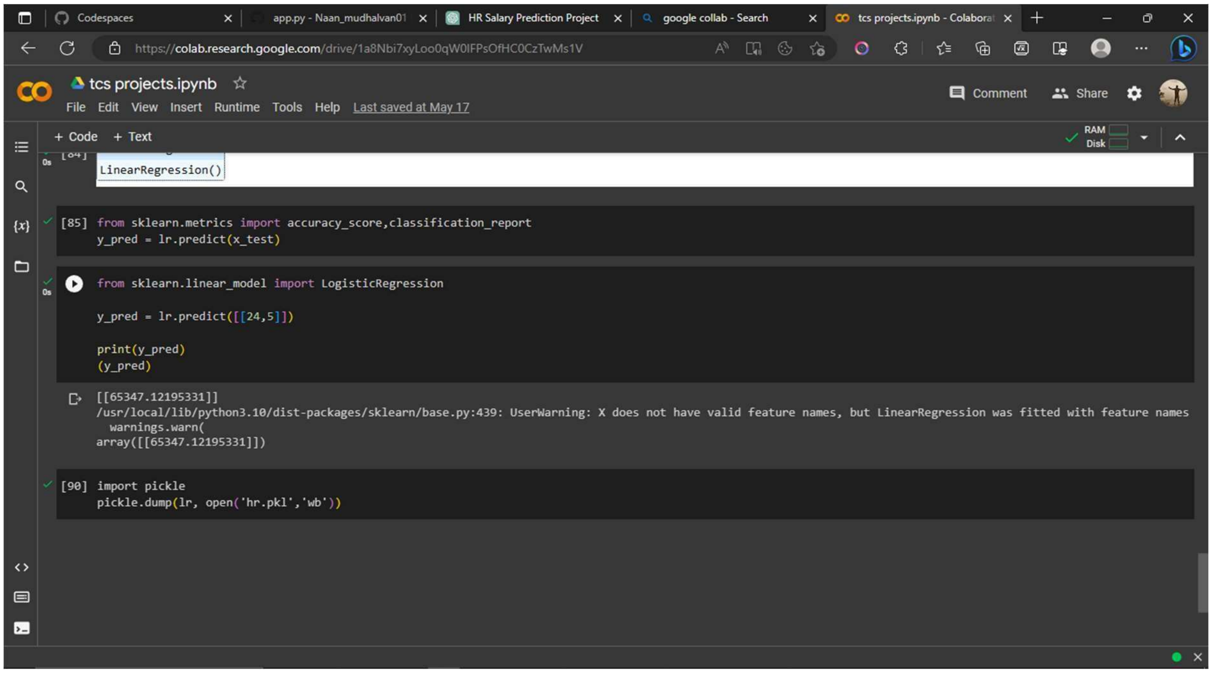
```
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
import numpy as np
sc=StandardScaler()
x_bal=sc.fit_transform(x)
```

```
[83] from sklearn.model_selection import train_test_split
     x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state=2)
```

```
[84] from sklearn.linear_model import LinearRegression

     # Create a LinearRegression object
     lr = LinearRegression()

     # Fit the model on the training data
     lr.fit(x_train, y_train)
```

```
▾ LinearRegression
LinearRegression()
```

Dump the logistic regression model into a pickle file,


Codespaces
app.py - Naan_mudhalvan01
HR Salary Prediction Project
google collab - Search
tcs projects.ipynb - Colaborat
https://colab.research.google.com/drive/1a8Nbi7xyLoo0qW0IFPsOfHC0CzTwMs1V
tcs projects.ipynb
File Edit View Insert Runtime Tools Help Last saved at May 17
Comment
Share
+ Code + Text
RAM
Disk

```
LinearRegression()
```

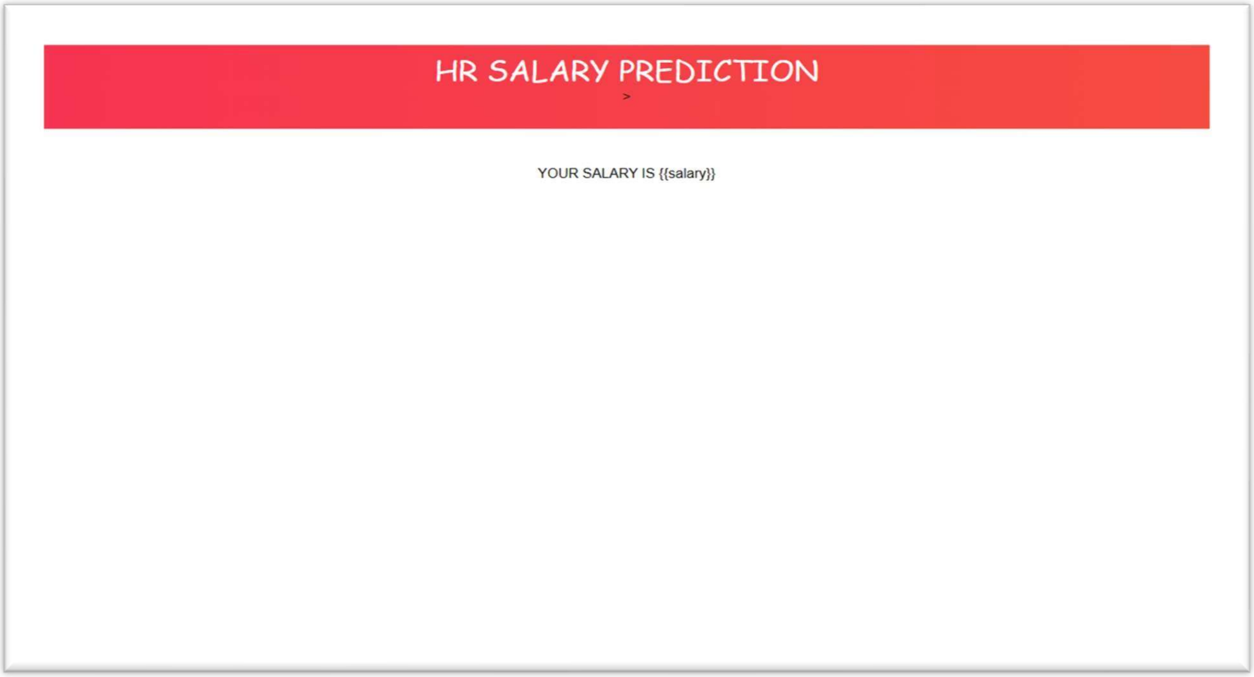
```
[85] from sklearn.metrics import accuracy_score,classification_report
     y_pred = lr.predict(x_test)
```

```
from sklearn.linear_model import LogisticRegression

y_pred = lr.predict([[24,5]])

print(y_pred)
(y_pred)
```

```
[[65347.12195331]]
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
  warnings.warn(
array([[65347.12195331]])
```

```
[90] import pickle
     pickle.dump(lr, open('hr.pkl','wb'))
```

OUTPUT PAGE:



YOUR SALARY IS {{salary}}

LINK TO CODE AND EXECUTABLE FILE

Link to the colab file: https://colab.research.google.com/drive/1G_qtRiWFV_7GSPUGFn4U6j-tcWatE1am?usp=drive_link

Link to the GitHub repository: https://github.com/magesh06069/TCS-iON-RIO25-Project-

Link for demo video: https://youtu.be/-x8QsB4X5N4