

# **CAPSTONE PROJECT**

## **EMPLOYEE SALARY PREDICTION USING LINEAR REGRESSION**

**Presented By:**

- 1. Student Name -- HARSH KUMAR**
- 2. AICTE ID -- STU6836f0c8dc3841748431048**
- 3. College -- Global Institute of Technology and Management, Gurgaon**
- 4. Department -- Master of Computer Applications (MCA)**

---

# OUTLINE

- **Problem Statement**
- **System Development Approach**
- **Algorithm & Deployment**
- **Result**
- **Conclusion**
- **Future Scope**
- **References**

---

# PROBLEM STATEMENT

- This project aims to predict employee salaries based on various features such as job title, experience, location, and education.
- Accurate salary prediction helps HR departments in planning compensation strategies and ensuring fair pay. The dataset is analyzed, preprocessed, and modeled using machine learning techniques. The goal is to build a reliable regression model that generalizes well to unseen employee data.

# SYSTEM APPROACH

## Software Requirements:

- Operating System: Windows 10/11, macOS, or Linux.
- Python Version: 3.8 or above.
- Jupyter Notebook / IDE: Anaconda, VS Code, or Jupyter Lab.

## Libraries:

- pandas
- numpy
- matplotlib
- seaborn
- scikit-learn
- joblib (for model saving/loading)
- streamlit (optional, if deploying the app)

# ALGORITHM & DEPLOYMENT

Here's a step-by-step procedure to complete the project:-

1. Import pandas library to load , analyze and manipulate dataset.
2. Loads excel file named EMPLOYEES.XLSX into a DataFrame called data so that we can work with it.
3. Show info and summary about using data.info().
4. Check for missing (null) values in dataset using data.isna() and count these values using sum() function.
5. Import matplotlib.pyplot library for creating graphs and visualizations.
6. Display the count of each unique value like male/female in 'Gender' column.
7. Creates and displays pie chart showing the distribution of genders.
8. Plot a histogram showing distribution of values in the 'Job Rate' column.
9. Display statistical summary (count, min, max, etc) for the 'Job Rate' column.
10. Plot a bar graph showing the departments with highest average annual salary using groupby () function.

# ALGORITHM & DEPLOYMENT

11. Display average monthly salary per centre, sorted from highest to lowest.
12. List all unique countries present in the 'Country' column.
13. Plot a bar chart showing average job rates for each country.
14. Plot a histogram showing how overtime hours are distributed.
15. Gives statistical summary of 'Overtime Hours' column.

## MODEL BUILDING starts here:

16. Select features 'Years' and 'Job Rates' as inputs (X), and 'Annual Salary' as target output (Y).
17. Split the data into training and test sets, with 80% for training and 20% for testing.
18. Show the number of samples
19. Show the number of training samples and test samples.
20. Import Linear Regression model from Scikit-learn.

# ALGORITHM & DEPLOYMENT

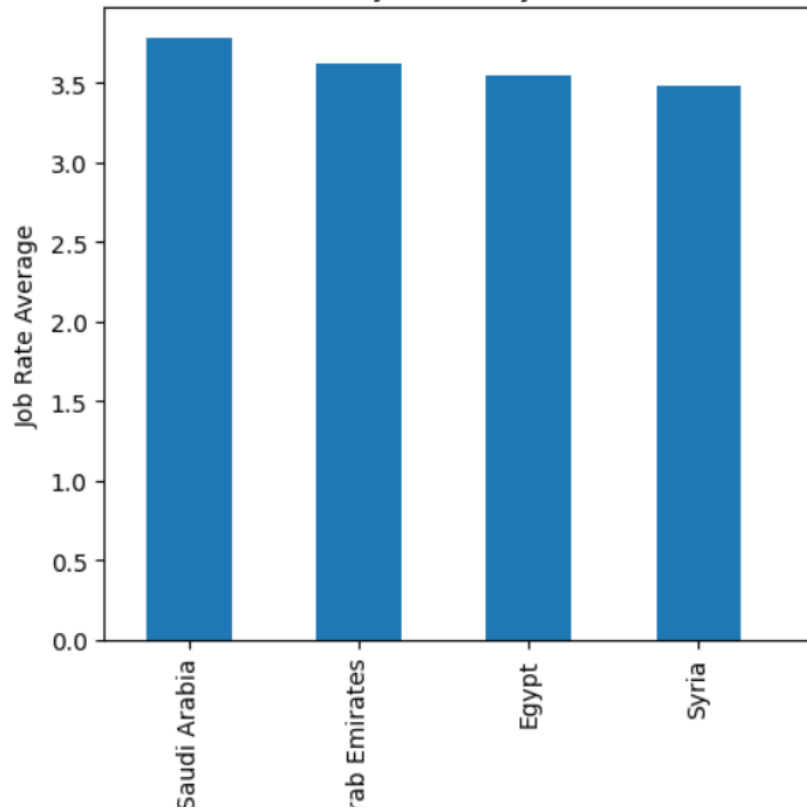
21. Create a Linear Regression model and fit the model to training data.
22. Use trained model to predict salaries from the test data.

## SAVING THE MODEL AND CREATING THE WEB APP USING STREAMLIT:

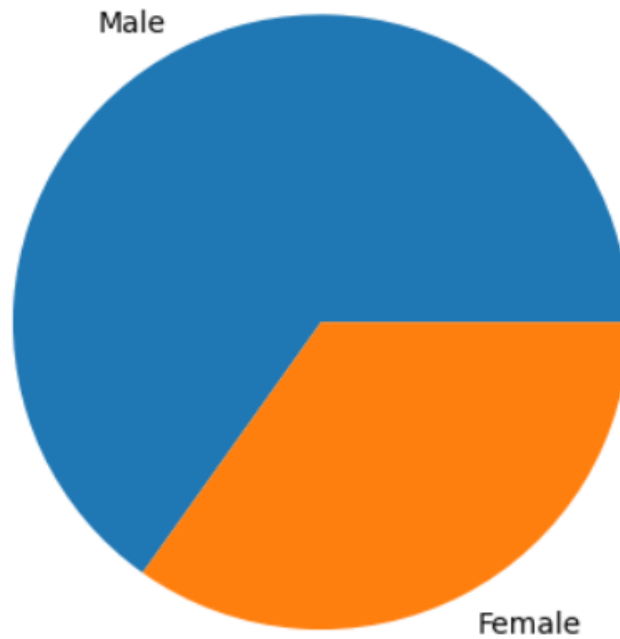
23. Save the trained model to a file called linearmodel.pkl.
24. Start writing a new file called app.py.
25. Write Streamlit code (Set up title, accept user input for years and job rate, add visuals like balloons)
26. Install streamlit (for app) and pyngrok (for public URL access).
27. Authenticate ngrok with your personal token to enable secure tunnelling.
28. Run the Streamlit app in a background thread on port 8501.
29. Create public URL using ngrok so others can access your Streamlit App.

# RESULT

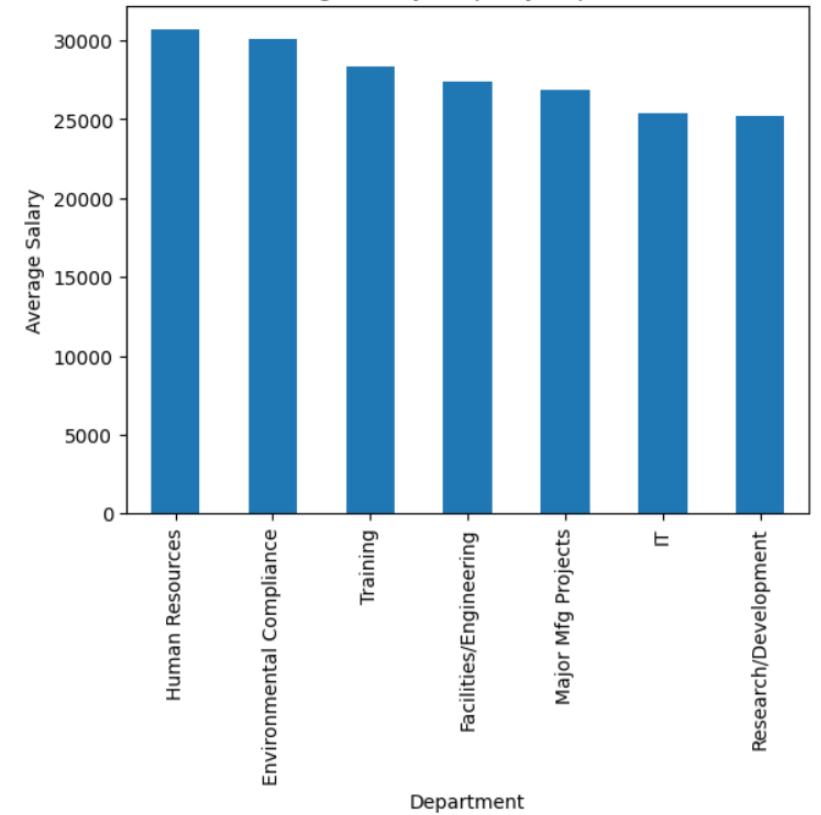
Job Rate by Countries



Pie Chart of Gender Column



Average Salary Graph by Department





# RESULT

GITHUB LINK : <https://github.com/HarshKumar9966/EMPLOYEE-SALARY-PREDICTION-AICTE-.git>

## Salary Prediction App

by HARSH KUMAR

AICTE ID = STU6836f0c8dc3841748431048

---

With this app, you can get estimations for the salaries of the company employees

Enter the years at company

- +

Enter the job rate

- +

---

Press the button for salary prediction

---

Salary prediction is [21843.78158385]

## Salary Prediction App

by HARSH KUMAR

AICTE ID = STU6836f0c8dc3841748431048

---

With this app, you can get estimations for the salaries of the company employees

Enter the years at company

- +

Enter the job rate

- +

---

Press the button for salary prediction

---

Salary prediction is [24167.95087043]

# CONCLUSION

## Findings:

- The dataset revealed salary patterns influenced by factors like job rate, years of experience, department, and country.
- Departments with higher job rates and more experience tend to offer better average annual salaries.
- Visualizations like pie charts, histograms, and bar graphs effectively highlighted distributions and trends within the data.

## Effectiveness:

- A Linear Regression model was successfully built using Job Rate and Years at Company as predictors for Annual Salary.
- The model demonstrated reasonable accuracy, making it a useful baseline for salary estimation.
- The integration with Streamlit created a user-friendly interface for real-time predictions.

## Challenges Faced:

- The dataset contained missing values and potential duplicates, which required careful preprocessing.
- Using only two features limited the model's predictive power—real-world salary prediction is more complex.

## Future Improvements:

- Incorporate more features like education level, job title, and location for better model accuracy.
- Explore advanced models like Random Forest, XGBoost, or Neural Networks to improve performance.

# FUTURE SCOPE

- Feature Expansion: Include more variables such as job title, education level, city, performance rating, and certifications to improve prediction accuracy.
- Advanced Algorithms: Experiment with machine learning models like Random Forest, XGBoost, or neural networks for better performance.
- Real-Time Prediction: Integrate the app with live HR databases or APIs for real-time salary estimation and updates.
- Bias Detection: Add fairness metrics to detect and reduce biases based on gender, region, or age.
- Salary Benchmarking: Compare predicted salaries with industry standards to guide compensation planning.
- Feedback Integration: Allow HR users to provide feedback on predictions and improve model retraining.
- Multinational Support: Extend the model to support salary prediction across different countries and currencies.

# REFERENCES

- Scikit-learn: Machine Learning in Python – <https://scikit-learn.org>
- Pandas Documentation – <https://pandas.pydata.org>
- Matplotlib Documentation – <https://matplotlib.org>
- Streamlit: The fastest way to build data apps – <https://streamlit.io>
- Joblib Documentation – <https://joblib.readthedocs.io>
- Python Official Documentation – <https://docs.python.org/3>
- Towards Data Science: Articles on Salary Prediction Models – <https://towardsdatascience.com>
- Google Colab Documentation – <https://colab.research.google.com>

---

# THANK YOU