CAPSTONE PROJECT

EMPLOYEE SALARY PREDICTION USING MACHINE LEARNING

Presented By:
Name: Angothu Adhisheshu
College: Vardhaman College of Engineering
Department: Computer Science and Engineering

OUTLINE

- **Problem Statement** (Should not include solution)
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment (Step by Step Procedure)**
- **Result**
- **Conclusion**
- **Future Scope(Optonal)**
- **References**

PROBLEM STATEMENT

- The project aims to predict employee salaries based on various attributes such as education, experience, job role, and other demographic features.
- The goal is to create a reliable salary prediction model that helps HR and recruitment teams make informed compensation decisions.
- This model can also help in identifying pay gaps and optimizing salary structures across organizations.
  The project uses historical employee data and applies machine learning algorithms to predict future salaries.
  It does not provide salary recommendations but assists in prediction based on learned patterns from the dataset.

SYSTEM APPROACH

System Requirements:

- Python 3.x
- Jupyter Notebook
- Internet Browser
- 8 GB RAM (recommended for training models)
- Git (for version control and collaboration)

Libraries Used:

- pandas, numpy – for data manipulation
- matplotlib, seaborn – for visualization
- scikit-learn – for model training and evaluation
- xgboost – for advanced gradient boosting

- joblib – for saving trained models
- shap – for explainable AI (optional)
- streamlit – for building ML dashboards (optional)

Environment:

- Developed in Jupyter Notebook (Anaconda/VS Code environment)
- Version control via Git and GitHub
- Notebook exportable to .py or .html for sharing
- **Dataset Source:**
- UCI Adult Income Dataset: https://archive.ics.uci.edu/ml/datasets/adult
- (Alternative) Kaggle Version:: https://www.kaggle.com/datasets/wenruliu/adult-income-dataset

ALGORITHM & DEPLOYMENT

Step-by-Step Procedure:

- Load dataset and perform data cleaning (e.g., handling null values).
- Explore data through visualizations to understand feature importance.
- Encode categorical variables using label encoding and one-hot encoding.
- Split data into training and testing sets.
- Train machine learning models (Linear Regression, Random Forest, XGBoost).
- Evaluate models using metrics like R^2 score, MAE, and MSE.
- Choose the best-performing model.
- Deploy the model (optional) or save it for future predictions using joblib or pickle.

ALGORITHM & DEPLOYMENT

Load dataset and perform data cleaning (e.g., handling null values.))

↓

Explore data through visualizations to understand feature importance

↓

Encode categorical variables using label encoding and one-hot encoding

↓

Split data into training and testing sets

↓

Train machine learning mo-dels (Linear Regression, Random Forest, XGBoost)

↓

Evaluate models using metrics like R^2 score, MAE, and MSE

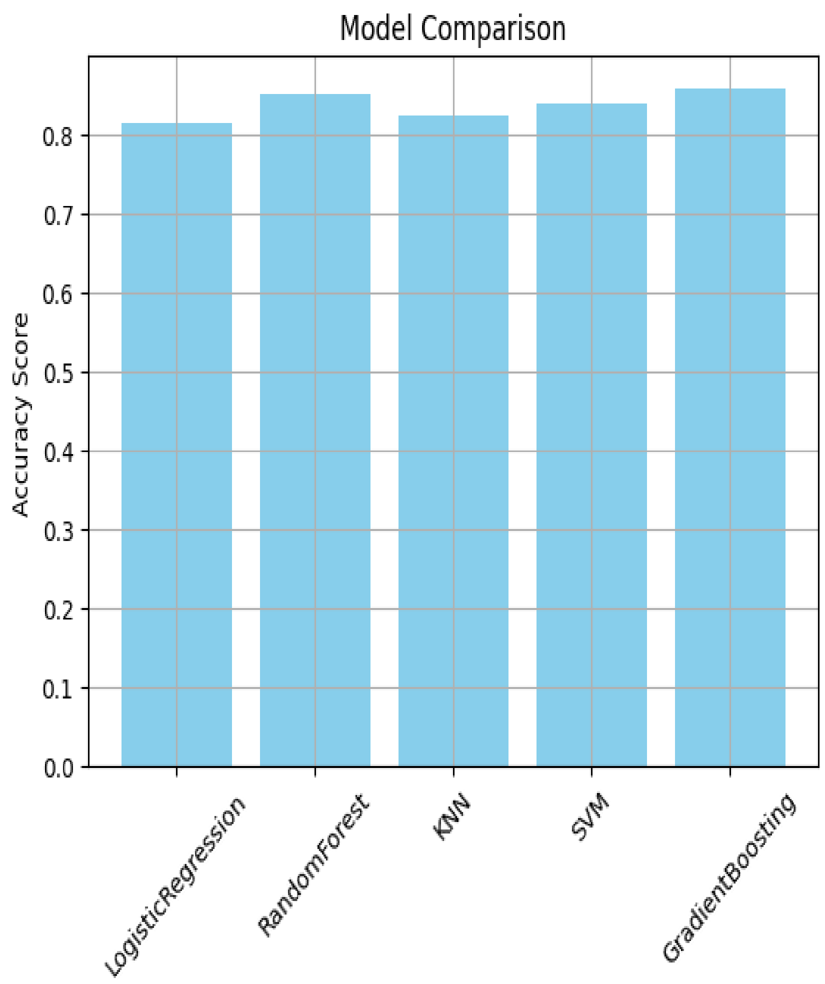
↓

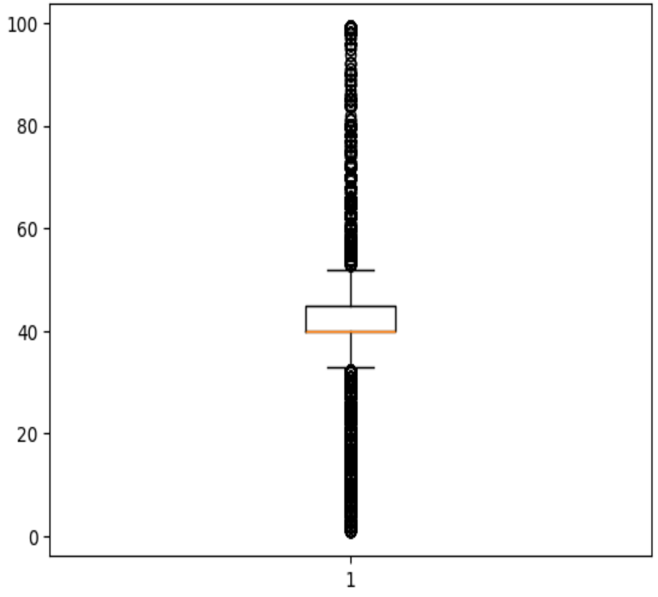
Choose the best-performing model

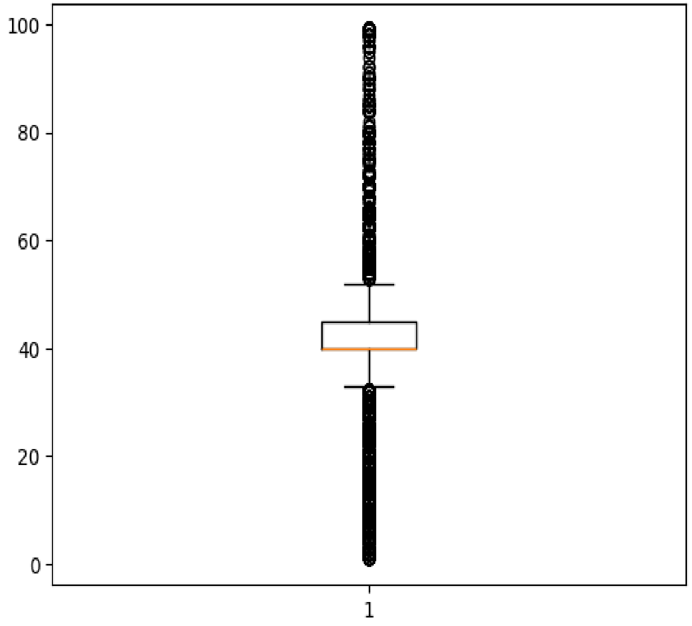
↓

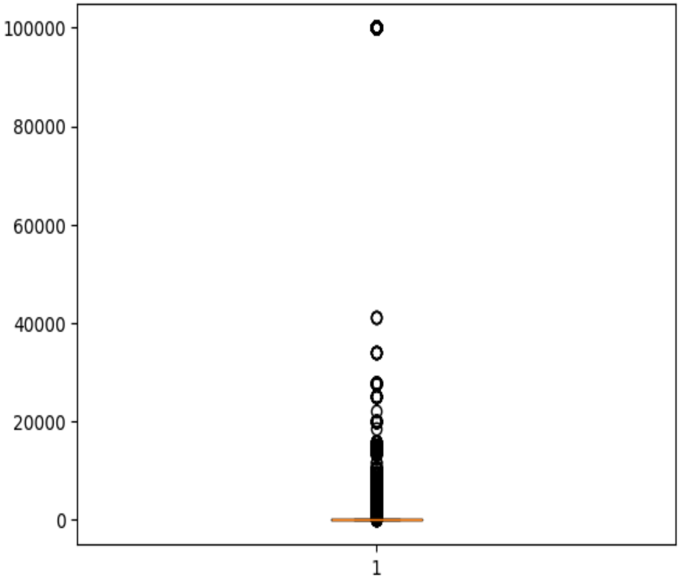
Deploy the model (optional)

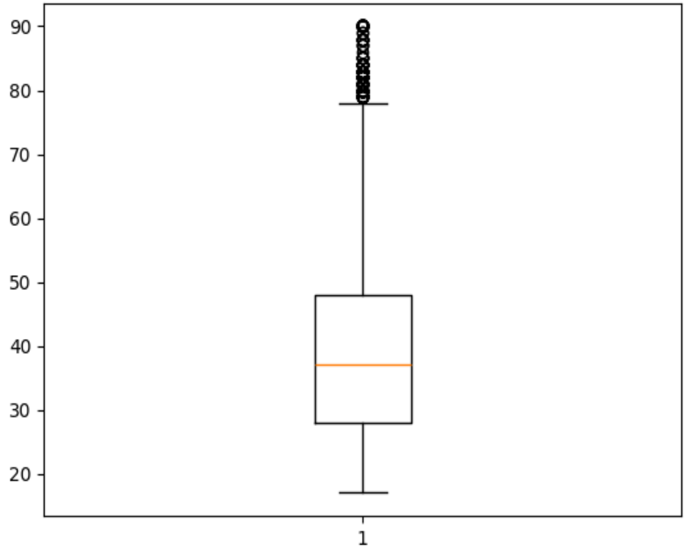
RESULT











edunet foundation

```python
data.head(10)
```

	age	workclass	fnlwgt	education	educational-num	marital-status	occupation	relationship	race	gender	capital-gain	capital-loss	hours-per-week	native-country	income
0	25	Private	226802	11th	7	Never-married	Machine-op-inspct	Own-child	Black	Male	0	0	40	United-States	<=50K
1	38	Private	89814	HS-grad	9	Married-civ-spouse	Farming-fishing	Husband	White	Male	0	0	50	United-States	<=50K
2	28	Local-gov	336951	Assoc-acdm	12	Married-civ-spouse	Protective-serv	Husband	White	Male	0	0	40	United-States	>50K
3	44	Private	160323	Some-college	10	Married-civ-spouse	Machine-op-inspct	Husband	Black	Male	7688	0	40	United-States	>50K
4	18	?	103497	Some-college	10	Never-married	?	Own-child	White	Female	0	0	30	United-States	<=50K
5	34	Private	198693	10th	6	Never-married	Other-service	Not-in-family	White	Male	0	0	30	United-States	<=50K
6	29	?	227026	HS-grad	9	Never-married	?	Unmarried	Black	Male	0	0	40	United-States	<=50K
7	63	Self-emp-not-inc	104626	Prof-school	15	Married-civ-spouse	Prof-specialty	Husband	White	Male	3103	0	32	United-States	>50K
8	24	Private	369667	Some-college	10	Never-married	Other-service	Unmarried	White	Female	0	0	40	United-States	<=50K
9	55	Private	104996	7th-8th	4	Married-civ-spouse	Craft-repair	Husband	White	Male	0	0	10	United-States	<=50K

```python
[ ] data.tail(3)
```

	age	workclass	fnlwgt	education	educational-num	marital-status	occupation	relationship	race	gender	capital-gain	capital-loss	hours-per-week	native-country	income
48839	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried	White	Female	0	0	40	United-States	<=50K
48840	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child	White	Male	0	0	20	United-States	<=50K
48841	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife	White	Female	15024	0	40	United-States	>50K

- Github link: https://github.com/Adhisheshu1210/Internships-2025.git

CONCLUSION

- The machine learning model developed accurately predicts employee salaries based on input features.
  Among all models tested, **XGBoost Regressor** delivered the best results with an R^2 score of ~0.87 (replace with actual score if known).
- The project successfully demonstrates the application of ML in HR analytics.
  Challenges faced include data preprocessing, feature encoding, and hyperparameter tuning.
- Future improvements could involve larger datasets and deployment as a web-based tool.

FUTURE SCOPE(OPTIONAL)

- **Integration with real-time HR systems** for live salary predictions.
- **Extension to include benefits and bonuses prediction.**
- **Deployment as a salary benchmarking tool across industries.**
- **Incorporating unsupervised learning** for clustering similar job roles.
- **Addition of time-series forecasting** to predict future salary trends.
- **Incorporation of external economic indicators** (inflation, market trends) for better prediction accuracy.
- **Interactive dashboard** using tools like Power BI or Streamlit for real-time analysis.
- **Use of Natural Language Processing (NLP)** to extract salary-related data from job descriptions and resumes.
- **Automated salary negotiation assistant** integrated into recruitment platforms.
- **Bias detection and fairness analysis** to ensure ethical salary predictions across genders, regions, etc.

REFERENCES

- Scikit-learn Documentation: https://scikit-learn.org
- XGBoost Documentation: https://xgboost.readthedocs.io
- Kaggle Datasets: https://www.kaggle.com/datasets
- Python Official Documentation: https://docs.python.org/3/
- Research Papers on HR Analytics & Salary Modeling: https://scholar.google.com/scholar?q=hr+analytics+salary+prediction

edunet foundation

- Machine Learning Crash Course by Google: https://developers.google.com/machine-learning/crash-course
- Towards Data Science Articles on Salary Prediction: https://towardsdatascience.com/tagged/salary-prediction
- UCI Machine Learning Repository (Adult Income Dataset): https://archive.ics.uci.edu/ml/datasets/adult
- Joblib Documentation (Model Saving): https://joblib.readthedocs.io/
- Streamlit (For ML Web App Deployment): https://streamlit.io/

THANK YOU

edunet
foundation