CAPSTONE PROJECT

PROJECT TITLE

Presented By:
Arshad Pasha
Seshadripuram Degree College
BCA Final Year

edunet foundation

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

- he project aims to predict whether an employee's annual salary exceeds $50,000 based on various demographic and employment-related features. In today's competitive job market, understanding salary patterns is crucial for both employers and employees for fair compensation decisions. Traditional manual salary assessment methods are time-consuming and may be biased. There is a need for an automated, data-driven approach to predict salary categories accurately. The challenge lies in identifying the most significant factors that influence high-income earnings and building a reliable prediction model. This project addresses the need for an intelligent system that can assist HR departments, job seekers, and policymakers in making informed decisions about compensation structures.

SYSTEM APPROACH

The "System Approach" section outlines the overall strategy and methodology for developing and implementing the Employee Salary Prediction system.

System Requirements:

- **Python 3.7 or higher**
- **Jupyter Notebook environment**
- **Minimum 4GB RAM for data processing**
- **Sufficient storage for dataset and model files**
- **Web browser for Streamlit application**

SYSTEM APPROACH

Libraries Required to Build the Model:

- **pandas - Data manipulation and analysis**
- **matplotlib - Data visualization and plotting**
- **scikit-learn - Machine learning algorithms and preprocessing**
- **streamlit - Web application development**
- **joblib - Model serialization and saving**
- **numpy - Numerical computations (implicit dependency)**

SYSTEM APPROACH

Technology Stack:

- **Programming Language: Python**
- **Development Environment: Jupyter Notebook**
- **Web Framework: Streamlit**
- **Machine Learning: Scikit-learn**
- **Data Processing: Pandas**
- **Visualization: Matplotlib**

ALGORITHM & DEPLOYMENT

- tep-by-Step Procedure to Complete the Project:
-
- 1. Data Loading and Exploration:
- - Load the adult.csv dataset using pandas
- - Examine data structure with head(), tail(), and shape()
- - Identify data types and missing values
-
- 2. Data Preprocessing:
- - Handle missing values by replacing '?' with 'Others'
- - Remove irrelevant categories ('Without-pay', 'Never-worked')
- - Detect and remove outliers using boxplot visualization
- - Filter age range (17-75 years) and education levels (5-16)

ALGORITHM & DEPLOYMENT

3. Feature Engineering:
   - Remove redundant features (education column)
   - Apply Label Encoding to categorical variables:
     * workclass, marital-status, occupation, relationship
     * race, gender, native-country

4. Model Development:
   - Split data into features (X) and target variable (y)
   - Implement train-test split (80-20 ratio)
   - Test multiple algorithms:
     * Logistic Regression
     * Random Forest Classifier
     * K-Nearest Neighbors (KNN)
     * Support Vector Machine (SVM)
     * Gradient Boosting Classifier

ALGORITHM & DEPLOYMENT

- 5. Model Evaluation:
- - Compare accuracy scores across all models
- - Generate classification reports
- - Visualize model performance using bar charts
- - Select best performing model
-
- 6. Model Deployment:
- - Save the best model using joblib
- - Create Streamlit web application (app.py)
- - Implement user interface for single predictions
- - Add batch prediction functionality
- - Enable model downloading capabilities

RESULT

```
data.shape
```

Python

```
(46720, 15)
```

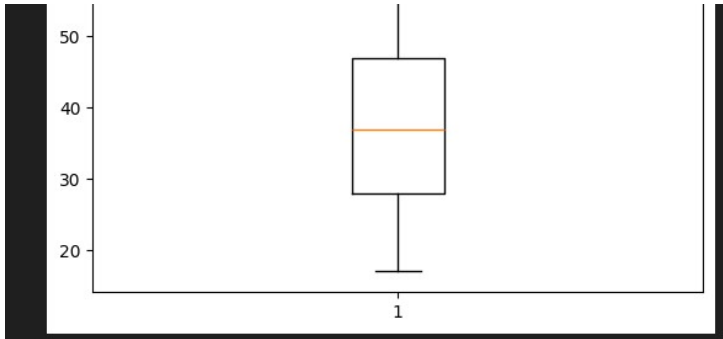
```
data=data.drop(columns=['education']) #redundant features removal
```

Python

```
data
```

Python

	age	workclass	fnlwgt	educational-num	marital-status	occupation	relationship	race	gender	capital-gain	capital-loss	hours-per-week	native-country
0	25	Private	226802	7	Never-married	Machine-op-inspct	Own-child	Black	Male	0	0	40	United-States
1	38	Private	89814	9	Married-civ-spouse	Farming-fishing	Husband	White	Male	0	0	50	United-States
2	28	Local-gov	336951	12	Married-civ-spouse	Protective-serv	Husband	White	Male	0	0	40	United-States
3	44	Private	160323	10	Married-civ-spouse	Machine-op-inspct	Husband	Black	Male	7688	0	40	United-States
4	18	Others	103497	10	Never-married	Others	Own-child	White	Female	0	0	30	United-States
...	...	...	...	...	...	...	...	...	...	...	...	...	...
48837	27	Private	257302	12	Married-civ-spouse	Tech-support	Wife	White	Female	0	0	38	United-States
48838	40	Private	154374	9	Married-civ-spouse	Machine-op-inspct	Husband	White	Male	0	0	40	United-States
48839	58	Private	151910	9	Widowed	Adm-clerical	Unmarried	White	Female	0	0	40	United-States



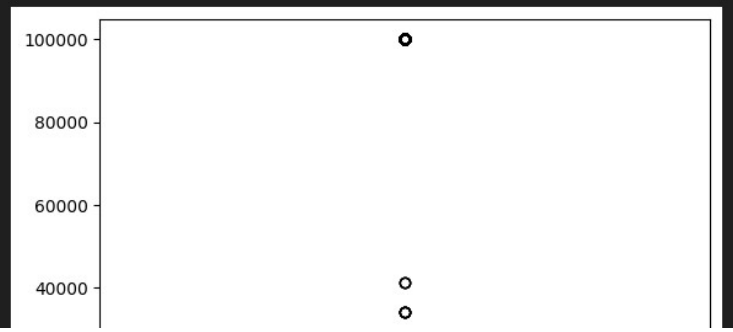

```
data.shape
```

Python

```
(48438, 15)
```

```
plt.boxplot(data['capital-gain'])
plt.show()
```

Python



edunet foundation

RESULT

- **Attach your Github link**
- https://github.com/pashaarshad/AICTE-B2_AI--2025-26-/blob/main/Employee%20Salary%20Prediction/employee%20salary%20prediction.ipynb

CONCLUSION

- The Employee Salary Prediction project successfully demonstrates the application of machine learning techniques for classification problems. The implemented solution effectively processes demographic and employment data to predict salary categories with reasonable accuracy. The automated model comparison approach ensures optimal algorithm selection, while the Streamlit deployment provides an accessible interface for end-users.

FUTURE SCOPE(OPTIONAL)

Potential enhancements and expansions for the system:

1. Advanced Feature Engineering:

- **Include additional features like industry type, company size**
- **Implement feature selection techniques**
- **Add polynomial features for better model performance**

2. Model Improvements:

- **Implement ensemble methods combining multiple algorithms**
- **Add hyperparameter tuning using Grid Search or Random Search**
- **Incorporate deep learning models for complex pattern recognition**

REFERENCES

- 1. UCI Machine Learning Repository - Adult Data Set
- https://archive.ics.uci.edu/ml/datasets/adult
- 2. Scikit-learn Documentation
- https://scikit-learn.org/stable/
- 3. Streamlit Documentation
- https://docs.streamlit.io/

THANK YOU

This capstone project demonstrates the complete machine learning pipeline from data preprocessing to model deployment, showcasing practical skills in data science and software development for real-world applications.