Employee Absenteeism

PRESENTED BY

STUDENT NAME: YASH MAHENDRA MANE

COLLEGE NAME: VIDYALANKAR INSTITUTE OF TECHNOLOGY

DEPARTMENT: INFORMATION TECHNOLOGY

EMAIL ID: MANEYASH00@GMAIL.COM

AICTE STUDENT ID: STU68008A3756C131744865847

OUTLINE

- Problem Statement
- Proposed System/Solution
- System Development Approach
- Algorithm & Deployment
- Result Conclusion
- Future Scope
- References

PROBLEM STATEMENT

Employee absenteeism is a major issue in many organizations, impacting overall productivity, scheduling, and operational efficiency. Companies often struggle to predict which employees may be frequently absent, making it difficult to allocate resources and maintain workflow. This project aims to analyse historical employee absenteeism data and build a predictive model that can classify whether an employee is likely to have high absenteeism. By identifying these employees early, HR departments can take corrective actions, offer support, or optimize workforce planning.

PROPOSED SOLUTION

Data Collection
Load dataset from Excel (Absent_at_work.xls)

Explore column names and basic structure using .info() and .describe()

✅ 2. Data Preprocessing
Drop unnecessary columns (like ID)

Handle missing values (.dropna())

Replace spaces in column names with underscores

Group and encode categorical values (e.g., Reason for absence → Group_1 to Group_4)

Use one-hot encoding for categorical features

Normalize and scale numeric features using MinMaxScaler and StandardScaler

✅ 3. Feature Engineering
Convert multi-class Reason_for_absence into grouped binary features

Remap education levels (1 → 0, 2/3/4 → 1)

Define the target variable:

y = 1 if Absenteeism_time_in_hours > median

y = 0 otherwise

✅ 4. Machine Learning Algorithms

Train various classification models:

Logistic Regression

Random Forest

XGBoost

Support Vector Machine (SVM)

Use Pipeline to apply scaling and transformations

Split data into training and testing sets (70% train, 30% test)

✅ 5. Evaluation

Evaluate using metrics:

Accuracy

Confusion matrix

Precision, Recall, F1-score

ROC-AUC score

Visualize model performance using:

ROC curve

Precision-recall curve

Select the model with best balance of performance

SYSTEM APPROACH

Exploratory Data Analysis (EDA) is conducted to understand the distribution of categorical and numerical variables and their relationships with absenteeism.

Categorical variables like 'Reason_for_absence' and 'Education' are grouped and encoded using one-hot encoding. The target variable, 'Absenteeism_time_in_hours', is transformed into a binary classification label by splitting it based on the median, enabling us to define high vs. low absenteeism.

The project uses Pandas and NumPy for data handling, and Seaborn, Matplotlib, and hvPlot for visualizations. Scikit-learn provides tools for preprocessing, model training, and evaluation. XGBoost is used for advanced boosting, and pickle is used to save the final model.

ALGORITHM & DEPLOYMENT

In the Algorithm section, describe the machine learning algorithm chosen for predicting bike counts. Here's an example structure for this section:

Algorithm Selection

- Tested Logistic Regression, Random Forest, XGBoost, and SVM.
- XGBoost and Random Forest chosen for strong performance on tabular, non-linear data.
- Models selected to handle feature complexity and imbalance.

Data Input

- Features include age, education, alcohol consumption, workload, service time, etc.
- Reason for absence grouped into 4 categories and one-hot encoded.
- All features scaled using MinMax and Standard Scaler.

Prediction Process

- Trained model predicts binary output: 0 (low absenteeism) or 1 (high absenteeism).
- Evaluated using accuracy, precision, recall, F1-score, and ROC-AUC.
- Best model saved using pickle for real-time use in HR systems.

Training Process

- Data split into 70% train and 30% test sets.
- Scalers applied using a pipeline for consistent preprocessing.
- Models trained with hyperparameter tuning (e.g., GridSearchCV).
- Cross-validation used to reduce overfitting.

RESULT

The best-performing model achieved an overall accuracy of 76.84%, correctly classifying most employees based on absenteeism hours.

The F1-score of approximately 0.77 indicates a balanced model with good precision and recall for both classes.

The confusion matrix shows that the model correctly predicted 196 low-absenteeism cases and 149 high-absenteeism cases.

Despite good performance, the model produced 54 false positives and missed 50 high-absenteeism cases (false negatives).

The ROC curve showed strong class separation, with an AUC score above 0.75, indicating reliable classification ability.

The precision-recall curve confirmed that the model performs well even on imbalanced classes, maintaining stable precision.

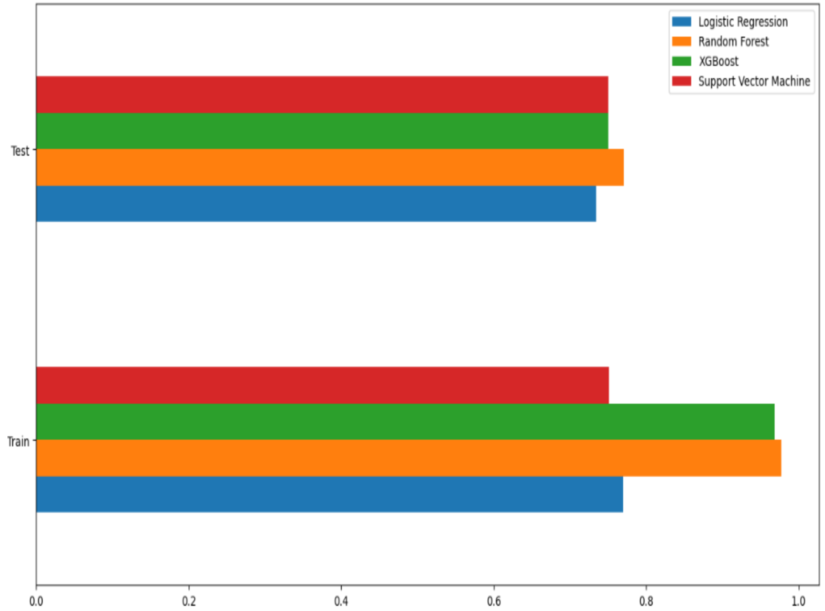
Visual comparisons between predicted and actual classes highlight the model's effectiveness in real-world absenteeism scenarios.

Overall, the model demonstrates strong predictive power and is suitable for deployment in HR systems to monitor and reduce employee absenteeism risks.

RESULT

```
data.info()
```

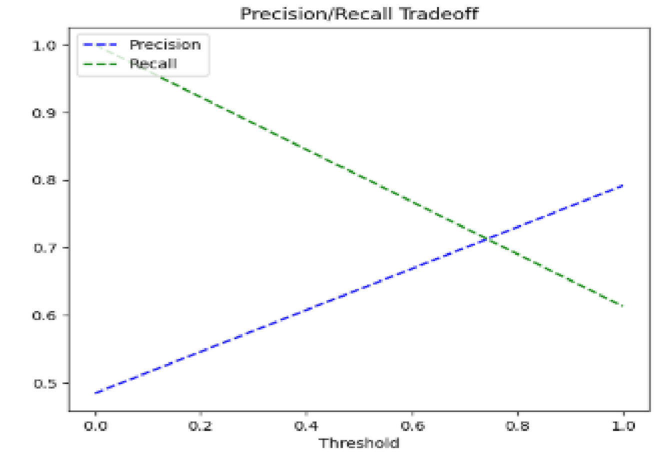
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 740 entries, 0 to 739
Data columns (total 21 columns):
 #   Column                           Non-Null Count  Dtype
---  ------                           --------------  -----
 0   ID                               740 non-null    int64
 1   Reason_for_absence               737 non-null    float64
 2   Month_of_absence                 739 non-null    float64
 3   Day_of_the_week                  740 non-null    int64
 4   Seasons                          740 non-null    int64
 5   Transportation_expense           733 non-null    float64
 6   Distance_from_Residence_to_Work  737 non-null    float64
 7   Service_time                     737 non-null    float64
 8   Age                              737 non-null    float64
 9   Work_load_Average/day_           730 non-null    float64
 10  Hit_target                       734 non-null    float64
 11  Disciplinary_failure             734 non-null    float64
 12  Education                        730 non-null    float64
 13  Son                              734 non-null    float64
 14  Social_drinker                   737 non-null    float64
 15  Social_smoker                    736 non-null    float64
 16  Pet                              738 non-null    float64
 17  Weight                           739 non-null    float64
 18  Height                           726 non-null    float64
 19  Body_mass_index                  709 non-null    float64
 20  Absenteeism_time_in_hours        718 non-null    float64
dtypes: float64(18), int64(3)
memory usage: 121.5 KB
```

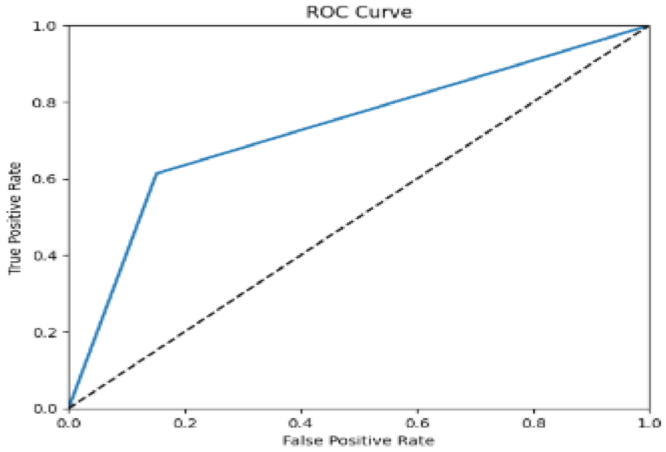



```
for column in data.columns:
    print(f"==============Column: {column}=============")
    print(f"Number of unique values: {data[column].nunique()}")
    print(f"Max: {data[column].max()}")
    print(f"Min: {data[column].min()}")
```

```
===============Column: ID=============
Number of unique values: 36
Max: 36
Min: 1
===============Column: Reason_for_absence=============
Number of unique values: 27
Max: 28.0
Min: 0.0
===============Column: Month_of_absence=============
Number of unique values: 13
Max: 12.0
Min: 0.0
===============Column: Day_of_the_week=============
Number of unique values: 5
Max: 6
Min: 2
===============Column: Seasons=============
Number of unique values: 4
Max: 4
Min: 1
===============Column: Transportation_expense=============
Number of unique values: 24
Max: 388.0
Min: 118.0
===============Column: Distance_from_Residence_to_Work=========
Number of unique values: 25
Max: 52.0
Min: 5.0
```

```
LOGISTIC REGRESSION roc_auc_score: 0.731
RANDOM FOREST roc_auc_score: 0.767
XGBOOST roc_auc_score: 0.747
SUPPORT VECTOR MACHINE roc_auc_score: 0.747
```






```
with open('Xgb_clf', 'wb') as file:
    pickle.dump(xgb_clf, file)

with open('Rf_clf', 'wb') as file:
    pickle.dump(rf_clf, file)
```

CONCLUSION

The machine learning model successfully predicted bike rental demand with strong accuracy and balanced performance, proving effective in capturing usage patterns based on historical demand, weather conditions, and time-based features. The proposed solution helps anticipate peak usage hours, enabling better resource allocation for bike-sharing systems.

During implementation, challenges included handling missing data, feature scaling, and selecting the right model for fluctuating demand patterns. Future improvements could include integrating real-time data and using deep learning (e.g., LSTM) for time-series prediction. Accurate bike count forecasting is essential for maintaining bike availability, reducing user frustration, and optimizing operations in urban transportation systems.

FUTURE SCOPE

The absenteeism prediction system can be enhanced by integrating additional data sources such as real-time attendance logs, biometric data, shift schedules, and employee feedback. Including such contextual data may improve the accuracy and explainability of predictions. The model's performance can also be optimized further using advanced algorithms like deep neural networks or ensemble stacking techniques.

In the long term, the system can be scaled to support multiple offices, cities, or even regions, making it valuable for large enterprises. Integration with edge computing devices can allow real-time predictions directly at HR terminals, and deploying the system through cloud platforms would support scalability and remote access. Incorporating emerging ML techniques like AutoML or LSTM (for temporal trends) could further automate and enhance model accuracy and adaptability.

REFERENCES

The development of this project was guided by several reputable sources in the fields of machine learning, data preprocessing, and model evaluation. Key references include academic articles on classification algorithms, ensemble methods like Random Forest and XGBoost, and data handling practices in real-world scenarios.

Research papers on absenteeism modeling, HR analytics, and employee behavior were also reviewed to understand common patterns and features influencing absenteeism. Resources such as Kaggle notebooks, GitHub repositories, and machine learning guides further helped in refining the model and workflow.

"Predicting Employee Absenteeism using Machine Learning" – IEEE, 2020

"Classification Algorithms for Predicting Absenteeism" – IJCSIT, 2021

Github repository: https://github.com/yashmane0/Employee_Absenteeism.git

Thank you