PROJECTS REPORT

Project Summaries

1) Forest Cover Type Prediction

GitHub Repository: Forest-Cover-Prediction

Objective: Predict the type of forest cover (1–7) for a 30×30m plot of land based on terrain and environmental features.

Approach:

- Conducted EDA and visualized correlations between elevation, soil types, and forest cover.
- Implemented Logistic Regression, Random Forest, XGBoost, and LightGBM models.
- Adjusted label encoding for multi-class compatibility (0–6 index for boosting models).
- Evaluated models on accuracy, precision, and confusion matrix visualization.

Results:

LightGBM achieved the best performance with an accuracy of 88.5%, followed closely by XGBoost (88.3%).

Key Learnings:

- Handling multi-class datasets effectively.
- Importance of proper label encoding for boosted models.
- Balancing model performance and computational efficiency.

2) Vehicle Price Prediction

GitHub Repository: Vehicle-Price-Prediction

Objective: Predict the price of used vehicles based on make, model, mileage, year, and other specifications.

Approach:

- Cleaned data and engineered new features (extracted horsepower, engine displacement, text lengths).
- Used regression algorithms: Linear Regression, Ridge, Random Forest, XGBoost, and LightGBM.
- Evaluated models using MAE, RMSE, and R^2 metrics.
- Visualized model residuals and feature importance.

Results:

Ridge Regression performed best with $R^2 = 0.847$ and RMSE ≈ 6830, indicating good generalization.

Key Learnings:

- Regularization (Ridge) can outperform complex ensemble models when data is clean and structured.
- Importance of feature scaling and preprocessing pipelines.
- How textual data (car names, engine specs) can enhance numeric prediction models.

---

3) Mobile Price Prediction

GitHub Repository: Mobile-Price-Prediction

Objective: Classify mobile phones into one of four price categories (0–3) based on their specifications.

Approach:

- Explored the dataset through EDA and correlation plots.
- Applied Logistic Regression, Random Forest, XGBoost, and LightGBM models.
- Balanced data using stratified splits and standardized feature scaling.
- Compared models using classification metrics (accuracy, precision, recall).

Results:

Logistic Regression achieved the highest performance, confirming the linear separability of the data.

Key Learnings:

- Feature importance visualization and interpretability.
- The value of simple linear models in well-structured datasets.
- Model deployment using joblib serialization.

4) ASL Image Classification
GitHub Repository: ASL-Image-Classification
Objective: Recognize hand signs (A–Z) in American Sign Language using deep learning.
Approach:
- Preprocessed images (resizing, normalization).
- Built a Convolutional Neural Network (CNN) using TensorFlow & Keras.
- Used data augmentation to prevent overfitting.
- Evaluated model performance on training and validation sets.

Results:
Achieved 94% validation accuracy on ASL dataset.
Key Learnings:
- Implementing CNNs from scratch.
- Understanding convolution, pooling, and dropout layers.
- Managing overfitting using data augmentation.

---

5) Heart Disease Detection
GitHub Repository: Detect-Heart-Disease
Objective: Predict the presence of heart disease based on medical parameters.
Approach:
- Cleaned dataset and handled missing values.
- Trained Logistic Regression, Random Forest, and XGBoost models.
- Evaluated using accuracy, recall, precision, and ROC-AUC scores.

Results:
Random Forest achieved 86% accuracy with strong recall performance.
Key Learnings:
- Handling medical datasets with care due to sensitivity and imbalance.
- Interpreting confusion matrices and ROC curves.
- Importance of recall in health-critical classification tasks.

6) Fraud Transaction Detection
GitHub Repository: Fraud-Transaction-Detection
Objective: Detect fraudulent financial transactions using transaction data.
Approach:
- Addressed class imbalance using undersampling and precision-recall analysis.
- Trained Logistic Regression, Random Forest, and XGBoost models.
- Optimized models for recall and F1-score to detect rare fraud cases.

Results:
XGBoost achieved 98% accuracy and a very high recall for fraud class.
Key Learnings:
- Managing highly imbalanced datasets.
- Using precision-recall trade-off for fraud detection.
- Understanding anomaly detection in real-world financial data.

Key Learnings

- Understood how to build complete ML pipelines from raw data to model deployment.
- Gained confidence in using advanced ML algorithms like XGBoost and LightGBM.
- Learned how to evaluate models using diverse performance metrics suited to each problem type.
- Improved problem-solving mindset and data-driven decision-making.
- Enhanced understanding of real-world data challenges, including missing values, imbalance, and overfitting.
- Developed skills in presenting ML results clearly through reports, charts, and documentation.