A.S. TRITTHIK THILAGAR

Reg No: 3122237001057

Sri Sivasubramaniya Nadar College of Engineering
Chennai
(An autonomous Institution affiliated to Anna University)

Degree & Branch: B.E. Computer Science & Engineering
Semester: V — Academic Year: 2025–2026 (Odd)
Subject Code & Name: ICS1512 - Machine Learning Algorithms Laboratory
Batch: 2023–2028

Experiment 1: ML Task Analysis and Feature Selection

Aim

To explore datasets from public repositories and identify appropriate machine learning tasks, feature selection techniques, and suitable ML algorithms.

Libraries Used

- Pandas
- Seaborn
- Matplotlib
- sklearn
- Numpy
- Scipy
- statsmodels

Mathematical/Theoretical Description

- Algorithms: Logistic Regression, Decision Trees, Random Forests, Random Forest Regressor
- Feature Selection: Pearson Correlation, Chi-square Test, ANOVA
- Evaluation Metrics: RMSE, MAE, R2 Score, Accuracy, Precision, Recall, F1 Score

Dataset Analysis Table

Dataset	Type of ML Task	Feature Selection Technique	Suitable ML Algorithm
Iris Dataset	Supervised - Classification	ANOVA, Correlation Matrix	Logistic Regression, KNN
Loan Amount Prediction	Supervised - Regression	Pearson Correlation, VIF	Random Forest Regressor
Predicting Diabetes	Supervised - Classification	Chi-square Test, ANOVA	Logistic Regression, Random Forest
Classification of Email Spam	Supervised - Classification	Chi-square, Mutual Info	Naive Bayes, SVM
Handwritten Character Recognition (MNIST)	Supervised - Classification	PCA, Variance Threshold	CNN, SVM

Loan Amount Prediction Model Code

```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import mean_squared_error, r2_score, mean_absolute_error
import numpy as np

df = pd.read_csv('data.csv')
df.drop(columns=['Loan_ID', 'Loan_Status'])

for col in df.columns:
    if df[col].dtype == 'object':
        df[col].fillna(df[col].mode()[0], inplace=True)
    else:
        df[col].fillna(df[col].median(), inplace=True)

for col in df.select_dtypes(include='object'):
    le = LabelEncoder()
    df[col] = le.fit_transform(df[col].astype(str))

X = df.drop('LoanAmount', axis=1)
y = df['LoanAmount']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

y_pred = model.predict(X_test)
rmse = np.sqrt(mean_squared_error(y_test, y_pred))
mae = mean_absolute_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

```

```python
print('MODEL PERFORMANCE')
print(f"RMSE: {rmse}\nMAE: {mae}\nR2: {r2}")
```

Code Execution Outputs & Visualizations

After Label Encoding:

Correlation Analysis for Feature Selection:

Regression Model Performance:

Results and Discussions

The regression model for loan amount prediction showed poor performance with low R^2 score, indicating weak correlation between features and the target. The correlation map confirmed that most input features lacked strong linear relationships with LoanAmount, and important real-world factors like credit score or employment stability were missing. Classification models for tasks like Diabetes and Email Spam performed well with accuracy and F1-score metrics using Random Forest and Logistic Regression.

Learning Outcomes

- Distinguished between classification and regression tasks.
- Applied correlation and statistical tests for feature selection.
- Gained hands-on practice with encoding, data preprocessing, and performance evaluation.
- Understood workflow from data ingestion to model interpretation.