

UCS2612 MACHINE LEARNING LABORATORY

EXERCISE-8

Applications of Random Forest and AdaBoost Ensemble Techniques

Code:

Original file is located at

<https://colab.research.google.com/drive/1LAPC5EoUgsr9D4Dt5Af6VyOBoNbcLNRo>

Breast Cancer Wisconsin

<https://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic>

"""

import pandas as pd

import numpy as np

import sklearn as sk

import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split

from sklearn.ensemble import AdaBoostClassifier

from sklearn.metrics import accuracy_score

from sklearn.feature_selection import SelectKBest

from sklearn.feature_selection import chi2

"""## Loading the dataset."""

df = pd.read_csv('wdbc.data')

df.head()

"""## Pre-Processing the data (Handling missing values, Encoding, Normalization, Standardization). & Exploratory Data Analysis."""

import pandas as pd

```
from sklearn.preprocessing import StandardScaler

import matplotlib.pyplot as plt

selected_features = ['radius1', 'texture1', 'perimeter1', 'area1', 'perimeter2', 'area2', 'radius3', 'texture3', 'perimeter3', 'area3']

X = df[selected_features]

scaler = StandardScaler()

X_scaled = scaler.fit_transform(X)

X_scaled_df = pd.DataFrame(X_scaled, columns=X.columns)

plt.figure(figsize=(12, 6))

for feature in selected_features:
    plt.plot(X[feature], label=f'{feature} original')
    plt.plot(X_scaled_df[feature], label=f'{feature} scaled')

plt.title('Difference Graph: Original vs Scaled')

plt.xlabel('Sample Index')

plt.ylabel('Feature Value')

plt.legend()

plt.show()

X = df.iloc[:, 2:]

y = df.iloc[:, 1]

"""## Feature Engineering techniques."""

k = 10

select_k_best = SelectKBest(score_func=chi2, k=k)

X_new = select_k_best.fit_transform(X, y)
```

```
selected_features = X.columns[select_k_best.get_support(indices=True)]
```

```
print("Selected features:", selected_features)
```

```
"""## Split the data into training, testing and validation sets."""
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
"""## Model Selection."""
```

```
ada_boost = AdaBoostClassifier(n_estimators=50, random_state=42) # You can adjust  
hyperparameters as needed
```

```
ada_boost
```

```
"""## Train the model."""
```

```
ada_boost.fit(X_train, y_train)
```

```
y_pred = ada_boost.predict(X_test)
```

```
"""## Test the model."""
```

```
accuracy = accuracy_score(y_test, y_pred)
```

```
print("Accuracy:", accuracy)
```

GitHub link:

<https://github.com/rohith18111407/Machine-Learning-Lab/tree/main/Exercise-8>