

ex8

August 16, 2024

```
[ ]: import pandas as pd
import numpy as np
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
```

```
[ ]: df = pd.read_csv(r'datasets\Breast_Cancer.csv')
df.shape
```

```
[ ]: (4024, 16)
```

```
[ ]: le = LabelEncoder()
df['Race'] = le.fit_transform(df['Race'])
df['Marital Status'] = le.fit_transform(df['Marital Status'])
df['T Stage ' ] = le.fit_transform(df['T Stage '])
df['N Stage'] = le.fit_transform(df['N Stage'])
df['6th Stage'] = le.fit_transform(df['6th Stage'])
df['differentiate'] = le.fit_transform(df['differentiate'])
df['Grade'] = le.fit_transform(df['Grade'])
df['A Stage'] = le.fit_transform(df['A Stage'])
df['Estrogen Status'] = le.fit_transform(df['Estrogen Status'])
df['Progesterone Status'] = le.fit_transform(df['Progesterone Status'])
```

```
[ ]: X = df.drop('Status',axis=1)
y = df['Status']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
↳random_state=42)
```

```
[ ]: def evaluate_model_with_lda(n_components):
    lda = LinearDiscriminantAnalysis(n_components=n_components)
    X_train_lda = lda.fit_transform(X_train, y_train)
    X_test_lda = lda.transform(X_test)

    model = LogisticRegression(max_iter=10000)
    model.fit(X_train_lda, y_train)
```

```
y_pred = model.predict(X_test_lda)
accuracy = accuracy_score(y_test, y_pred)

return accuracy, lda
```

```
[ ]: components_list = [1]
      results = {}

      for n in components_list:
          accuracy, lda = evaluate_model_with_lda(n)
          results[n] = accuracy
          print(f"Number of LDA components: {n}, Accuracy: {accuracy:.4f}")
```

Number of LDA components: 1, Accuracy: 0.8990