

10. Using iris data set, implement naive bayes classification for different naive Bayes classification algorithms. (i) Gaussian

- i. Find the accuracy level of the predicted value
- ii. Calculate the number of mislabeled instances
- iii. List out the class labels of mismatching records

```
import numpy as np
import pandas as pd

dataset = pd.read_csv('iris.csv')
X = dataset.iloc[:, :4].values
y = dataset['variety'].values

#split iris data to test and train
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2)

#implement Gaussian
from sklearn.naive_bayes import GaussianNB
classifier = GaussianNB()
classifier.fit(X_train, y_train)
y_pred = classifier.predict(X_test)
print("\n", y_pred, "\n")

# Find the accuracy level of the predicted value
from sklearn.metrics import accuracy_score
print ("\n Accuracy : ", accuracy_score(y_test, y_pred))

# Calculate the number of mislabeled instances
mislabeled_count = (y_test != y_pred).sum()
# Print the number of mislabeled instances
print(f"\n Number of mislabeled points: {mislabeled_count} out of {len(y_test)}")

# List out the class labels of the mismatching records
mismatches = (y_test != y_pred)
print("\n Mismatching records (Actual vs Predicted):")
```

```
for actual, predicted in zip(y_test[mismatches], y_pred[mismatches]):  
    print(f"Actual: {actual}, Predicted: {predicted}")
```