Customer Behavior Prediction Using Naive Bayes

Objective

The goal of this project is to classify customers as either 'Bargain Hunters' or 'Premium Buyers' based on their purchase history using a supervised machine learning approach.

Dataset

For demonstration purposes, synthetic classification data was generated using make_classification() from scikit-learn. In a real-world scenario, features might include:

- Average Purchase Value
- Discount Usage Rate
- Brand Loyalty Score
- Number of Purchases per Month
- Preferred Product Category

Methodology

Data Preprocessing:

- Generated synthetic dataset with 200 samples and 5 features.
- Split data into training (70%) and testing (30%).

Model Used:

- Gaussian Naive Bayes classifier (suitable for continuous-valued features assuming a normal distribution).

Evaluation Metrics:

- Confusion Matrix (visualized as a heatmap)
- Accuracy, Precision, Recall, F1-score

Code Summary

Import libraries and create data

Customer Behavior Prediction Using Naive Bayes

```
X, y = make_classification(n_samples=200, n_features=5, n_classes=2)


# Train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)


# Train model
model = GaussianNB()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)


# Evaluation
cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot=True, cmap='Blues')
print(classification_report(y_test, y_pred))
```

Results

Confusion Matrix:

	Predicted Bargain	Predicted Premium
Actual Bargain	26	4
Actual Premium	3	17

Classification Report:

	precision	recall	f1-score	support
Bargain Hunter	0.90	0.87	0.88	30
Premium Buyer	0.81	0.85	0.83	20

Customer Behavior Prediction Using Naive Bayes

Accuracy: 0.86

Conclusion

The Naive Bayes classifier achieved an accuracy of 86%, showing good performance in distinguishing

between customer types.

The model is efficient, interpretable, and suitable for fast deployment in customer analytics applications.

Future Scope

- Apply the model on real customer datasets.
- Experiment with advanced models like Random Forest or SVM.
- Perform feature selection for better accuracy.