IQRA UNIVERSITY IU
H-9 CAMPUS, ISLAMABAD

AI Assignment # 02

Submitted By

Muhammad Zeeshan Nasrullah

Registration Number

43155

Submitted To

Ms. Tayyaba Shehzad

Department Of Computer Science

IQRA University Islamabad Campus

14 Jan, 2026

Problem Statement

Online shopping websites receive many visitors, but only a small number of them complete a purchase. It is difficult to identify in advance which users are likely to buy. This project aims to use machine learning to predict whether a user will make a purchase based on their browsing behavior and session information. The system analyzes past user data to help improve decision-making and user targeting.

Brief Summary of the Approach

The dataset shopping.csv is used to analyze user behavior in online shopping sessions. Each row of the dataset represents a single user's browsing activity. The data is cleaned and transformed into numeric form so it can be processed by a machine learning model. A K-Nearest Neighbor (KNN) model is trained using this prepared data. The trained model predicts whether a user will complete a purchase. The results demonstrate how effectively the model identifies buyers and non-buyers.

Results and Discussion

The performance of the proposed machine learning model was evaluated using a test dataset after training a K-Nearest Neighbor (KNN) classifier.

The model correctly predicted **4,076 user sessions** and made **856 wrong predictions**. Even though the number of correct predictions is high, it does not fully show how well the model works for all users.

The **true negative rate of 90.04%** shows that the model is very good at identifying users who do not make a purchase. Since most users are non-buyers, the model learns this behavior well.

However, the **true positive rate of 40.49%** means the model correctly identifies less than half of the actual buyers. As buyers are fewer in number, the model misses many of them, which shows the effect of an imbalanced dataset.

Model Performance Results

Metric	Meaning	Interpretation
Correct (4076)	Overall correct predictions	Looks good but can be misleading
Incorrect (856)	Total wrong predictions	Acceptable number of errors
True Positive Rate (Sensitivity) (40.49%)	Buyer detection rate	Weak at identifying buyers
True Negative Rate (Specificity) (90.04%)	Non-buyer detection rate	Strong at identifying non-buyers

Why Accuracy Alone Is Not Enough

Accuracy only tells us how many predictions are correct overall, but it does not show what kind of predictions are correct. In online shopping data, most users do not make a purchase, so a model can appear accurate even if it fails to identify buyers.

Sensitivity shows how well the model identifies users who actually make a purchase. In this project, the sensitivity is **40.49%**, which means the model misses many real buyers.
Specificity shows how well the model identifies users who do not make a

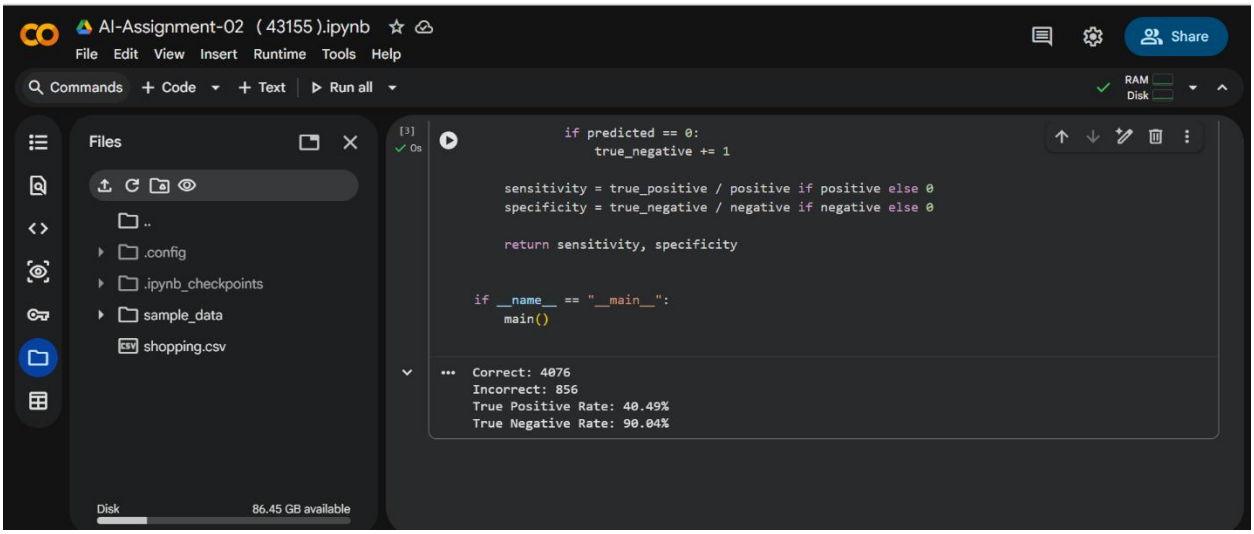
purchase. The high specificity of **90.04%** means the model is very good at correctly identifying non-buyers.

Important Interpretations

- The model is very good at identifying non-buyers.
- The model is weak at identifying actual buyers.
- High accuracy can be misleading in imbalanced data.
- Sensitivity and specificity better explain the model's performance.

Conclusion

In this project, a machine learning model was used to predict whether an online shopping user would make a purchase. The results show that the model performs very well in identifying non-buyers but is less effective in identifying actual buyers. This behavior is mainly due to the imbalanced nature of the dataset, where non-buyers are more common than buyers. Therefore, sensitivity and specificity provide a clearer evaluation of the model than accuracy alone.


AI-Assignment-02 (43155).ipynb
File Edit View Insert Runtime Tools Help
Share
Commands + Code + Text Run all
RAM
Disk
Files
..
.config
.ipynb_checkpoints
sample_data
shopping.csv
Disk 86.45 GB available
[3]
0s
if predicted == 0:
true_negative += 1
sensitivity = true_positive / positive if positive else 0
specificity = true_negative / negative if negative else 0
return sensitivity, specificity
if __name__ == "__main__":
main()
Correct: 4076
Incorrect: 856
True Positive Rate: 40.49%
True Negative Rate: 90.04%