question-2

March 9, 2024

1 Question 2 codes and answers

[1]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import make_pipeline
```

[7]:
```
# Load the buyer data
buyer_df = pd.read_csv('buyer.csv')

# Convert the 'Annual income' column to numeric values, assuming 'k' stands for␣
 ↪thousands
buyer_df['Annual income'] = buyer_df['Annual income'].str.replace('k', '').
 ↪astype(int) * 1000

# Preprocess the data: convert categorical variables to numeric
# Assuming the "Gender" column contains 'Male' and 'Female', and "Married"␣
 ↪column contains 'TRUE' and 'FALSE'
buyer_df['Gender'] = buyer_df['Gender'].map({'Male': 0, 'Female': 1})
buyer_df['Married'] = buyer_df['Married'].astype(int)
buyer_df['Buy'] = buyer_df['Buy'].astype(int)

# Select features and target variable
X = buyer_df[['Age', 'Gender', 'Annual income', 'Married']]
y = buyer_df['Buy']

# Split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,␣
 ↪random_state=42)
```

[15]:
```
from sklearn.linear_model import LogisticRegression

# Create a logistic regression pipeline
logistic_pipeline = make_pipeline(StandardScaler(),␣
 ↪LogisticRegression(random_state=42))
```

```
# Fit the model on the full dataset (better for feature importance analysis)
logistic_pipeline.fit(X, y)

# Get the coefficients from the logistic regression model
coefs = logistic_pipeline.named_steps['logisticregression'].coef_[0]
features = X.columns

# Now let's predict the probability of the new customer buying a house
new_customer = pd.DataFrame({
    'Age': [40],
    'Gender': [1],  # Female
    'Annual income': [310000],  # Corrected to full amount
    'Married': [0]  # Unmarried
})

# Predict the probability of the new customer buying a house using the logistic␣
 ↪regression model
probability_of_buying_logistic = logistic_pipeline.predict_proba(new_customer)[:
 ↪, 1]

# Evaluate the logistic regression model on the train and test set
logistic_pipeline.fit(X_train, y_train)
logistic_predictions = logistic_pipeline.predict(X_test)
logistic_report = classification_report(y_test, logistic_predictions,␣
 ↪zero_division=0)

print(logistic_report)
print(probability_of_buying_logistic)

# Map coefficients to features
feature_importance = pd.Series(coefs, index=features).sort_values(key=abs,␣
 ↪ascending=False)

feature_importance
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         1
           1       1.00      1.00      1.00         1

    accuracy                           1.00         2
   macro avg       1.00      1.00      1.00         2
weighted avg       1.00      1.00      1.00         2

[0.59431258]
```

[15]:
```
Annual income    1.238016
Age              0.518439
Gender           0.334001
Married          0.047491
dtype: float64
```

1.1 Logistic Regression Model Prediction

1.1.1 Probability of New Customer Buying

- The Logistic Regression model predicts a **59.4% probability** of the new customer purchasing a house.

1.1.2 Classification Report

- The test set evaluation yields perfect scores:
  - **Precision:** 100% for both classes.
  - **Recall:** 100% for both classes.
  - **F1-Score:** 100% for both classes.
  - **Accuracy:** 100% overall.

1.1.3 Decision on Customer Acquisition

- Considering the **59.4% probability** of purchase and the profile's similarity to previous buyers, if the expected revenue from the customer is greater than the high acquisition cost, acquiring the new customer could be a beneficial move.
- The decision must also take into account strategic business objectives and the capacity to onboard new customers without affecting service quality for existing ones.

1.1.4 Notes on Evaluation

- While the classification report indicates perfect metrics, the very small test set size means these scores do not necessarily indicate the model's performance in a real-world scenario.
- More data and a larger test set would be needed to accurately gauge the model's effectiveness and reliability.

1.2 Logistic Regression Analysis for Customer Acquisition

1.2.1 Feature Importance from Logistic Regression

- **Annual income** is the most significant predictor of whether a customer will buy a house.
- **Age** also positively impacts the likelihood of a customer purchasing a house.
- **Gender** has a positive, albeit smaller, effect on the prediction.
- **Married** status shows minimal importance in the logistic regression model.

1.2.2 Model Selection Rationale

- Logistic Regression was chosen for its simplicity and interpretability compared to more complex models like RandomForest.

- It outputs probabilities directly through the sigmoid function, making it suitable for binary classification problems.
- The model coefficients provide clear insights into the impact of each feature on the likelihood of a customer buying a house.

1.2.3 Limitations

- Logistic Regression assumes a linear relationship between the log-odds of the dependent variable and each independent variable, which may not capture more complex patterns.
- Feature importance is based on the scale of the data, so proper preprocessing steps such as standardization are crucial.
- It may not perform as well as more complex models when the true relationship is not linear or if there are significant interactions between features.

1.2.4 Conclusion

- The model suggests that the **annual income** is the most crucial factor in predicting house purchasing, followed by the customer's **age**.
- Given the coefficients, we can interpret that as the **annual income** and **age** increase, so does the likelihood of a customer purchasing a house.
- Since **married** status has a near-zero coefficient, it does not seem to be a decisive factor for purchasing within the model.