Project Report

On

Car Sales Prediction

Submitted in partial fulfilment of the requirements for the award of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

(Artificial Intelligence & Machine Learning)

by

Ms.K.SNEHA REDDY -22WH1A6604

Ms.P.PREETHI – 22WH1A6633

Ms.B.SOWMYA – 22WH1A6646

Ms.B.HARSHINI – 22WH1A6652

Under the esteemed guidance of

Ms. A Naga Kalyani

Assistant Professor, CSE(AI&ML)

BVRIT HYDERABAD College of Engineering for Women

(UGC Autonomous Institution | Approved by AICTE | Affiliated to JNTUH)

(NAAC Accredited - A Grade | NBA Accredited B.Tech. (EEE, ECE, CSE and IT)

Bachupally, Hyderabad – 500090

2024-25

Department of Computer Science & Engineering

(Artificial Intelligence & Machine Learning)

BVRIT HYDERABAD COLLEGE OF ENGINEERING FOR WOMEN

(Approved by AICTE, New Delhi and Affiliated to JNTUH, Hyderabad)

Accredited by NBA and NAAC with A Grade

Bachupally, Hyderabad – 500090

2024-25

CERTIFICATE

This is to certify that the major project entitled **"Classification of Academic success data using Python"** is a bonafide work carried out **by Ms.K.Sneha Reddy (22wh1a6604), Ms.P.Preethi (22wh1a6633), Ms. B.Sowmya (22wh1a6646), Ms. B.Harshini (22wh1a6652)** in partial fulfillment for the award of B. Tech degree in **Computer Science & Engineering (AI&ML), BVRIT HYDERABAD College of Engineering for Women, Bachupally, Hyderabad,** affiliated to Jawaharlal Nehru Technological University Hyderabad, Hyderabad under my guidance and supervision. The results embodied in the project work have not been submitted to any other University or Institute for the award of any degree or diploma.

Supervisor
Ms. A Naga Kalyani
Assistant Professor

Dept of CSE(AI&ML)

Head of the Department
Dr. B. Lakshmi Praveena
HOD & Professor

Dept of CSE(AI&ML)

External Examiner

DECLARATION

We hereby declare that the work presented in this project entitled **"Car Sales prediction using python"** submitted towards completion of Project work in IV Year of B.Tech of CSE(AI&ML) at **BVRIT HYDERABAD College of Engineering for Women,** Hyderabad is an authentic record of our original work carried out under the guidance of **Ms. A Naga Kalyani, Assistant Professor, Department of CSE(AI&ML).**

Sign with Date:

K.Sneha Reddy

(22wh1a6604)

Sign with Date:

P.Preethi

(22wh1a6633)

Sign with Date:

B.Sowmya

(22wh1a6646)

Sign with Date:

B.Harshini

(22wh1a6652)

ACKNOWLEDGEMENT

We would like to express our sincere thanks to **Dr. K. V. N. Sunitha, Principal, BVRIT HYDERABAD College of Engineering for Women**, for her support by providing the working facilities in the college.

Our sincere thanks and gratitude to **Dr. B. Lakshmi Praveena, Head of the Department, Department of CSE(AI&ML), BVRIT HYDERABAD College of Engineering for Women,** for all timely support and valuable suggestions during the period of our project.

We are extremely thankful to our Internal Guide**, Ms. A Naga Kalyani, Assistant Professor, CSE(AI&ML), BVRIT HYDERABAD College of Engineering for Women,** for her constant guidance and encouragement throughout the project.

Finally, we would like to thank our Major Project Coordinator, all Faculty and Staff of CSE(AI&ML) department who helped us directly or indirectly. Last but not least, we wish to acknowledge our **Parents and Friends** for giving moral strength and constant encouragement.

KSneha Reddy(22wh1a6604)

P.Preethi (22wh1a6633)

B.Sowmya(22wh1a6646)

B.Harshini(22wh1a6652)

ABSTRACT

This project analyzes a dataset using Python to develop a predictive model for binary classification. The analysis leverages data manipulation libraries like Pandas for preprocessing and feature engineering, while Matplotlib and Seaborn are utilized for exploratory data visualization. Key steps include handling missing values, examining feature distributions, and identifying correlations between variables. Classification models such as Logistic Regression and Random Forest are built and evaluated using metrics like accuracy, precision, recall, and confusion matrices. The project aims to provide data-driven insights into feature importance and classification performance, assisting stakeholders in decision-making and improving predictive capabilities.

PROBLEM STATEMENT

Academic success plays a crucial role in shaping individuals' careers and contributing to societal development. Traditional methods for predicting academic outcomes often rely on basic statistical analyses, which may overlook complex interactions between various influencing factors.

This project seeks to address the challenge of predicting academic success by employing a machine learning-based classification approach. The primary objectives include:

1. Developing classification models to predict students' academic outcomes based on features such as demographics, attendance, and prior performance.
2. Conducting exploratory data analysis to identify patterns and relationships between key factors influencing academic success.
3. Evaluating the performance of models through metrics such as accuracy, precision, recall, and confusion matrices to ensure reliability and applicability.

The ultimate aim is to provide actionable insights for educators and policymakers to enhance student performance, optimize resource allocation, and support data-driven decision-making in educational systems.

DATA SET

Classification of Academic dataset – Kaggle

https://www.kaggle.com/competitions/playground-series-s4e6/data

SOURCE CODE

```
from google.colab import drive
drive.mount('/content/drive')
```

```
import pandas as pd
df = pd.read_csv("/content/drive/MyDrive/data.csv")
```

```
from google.colab import drive
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix

df = pd.read_csv("/content/drive/MyDrive/data.csv")
```

```
# Drop rows with null values in the 'Target' column
df.dropna(subset=['Target'], inplace=True)
```

```
# Convert 'Target' column values to 0 and 1 (assuming binary classification)
df['Target'] = df['Target'].astype('category').cat.codes  # Encode categorical values
```

```
# EDA Analysis
print(df.info())
print(df.describe())
```

```
# Correlation Matrix
plt.figure(figsize=(12, 10))
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```

```
# Histograms for numerical features
numerical_features = df.select_dtypes(include=['number'])
for col in numerical_features.columns:
    if col != 'Target':
        plt.figure(figsize=(8, 6))
        sns.histplot(df[col], kde=True)
        plt.title(f'Distribution of {col}')
        plt.show()

# Boxplots for numerical features
for col in numerical_features.columns:
    if col != 'Target':
        plt.figure(figsize=(8, 6))
        sns.boxplot(x='Target', y=col, data=df)
        plt.title(f'Boxplot of {col} vs Target')
        plt.show()

# Countplot for categorical features (if any)
categorical_features = df.select_dtypes(include=['object'])
for col in categorical_features.columns:
    plt.figure(figsize=(10,6))
    sns.countplot(x=col, data=df)
    plt.title(f'Count of {col}')
    plt.xticks(rotation=45, ha='right')
    plt.show()

# Classification (Logistic Regression Example)
X = df.drop('Target', axis=1)
y = df['Target']

# Scale numerical features
scaler = StandardScaler()
numerical_cols = X.select_dtypes(include=['number']).columns
X[numerical_cols] = scaler.fit_transform(X[numerical_cols])

# One-hot encode categorical features
X = pd.get_dummies(X, drop_first=True)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

model = LogisticRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
```

```
print(classification_report(y_test, y_pred))
print(confusion_matrix(y_test, y_pred))
```

```
from sklearn.ensemble import RandomForestClassifier

# Initialize and train a RandomForestClassifier
rf_model = RandomForestClassifier(random_state=42)
rf_model.fit(X_train, y_train)

# Make predictions
rf_y_pred = rf_model.predict(X_test)

# Evaluate the model
print(classification_report(y_test, rf_y_pred))
print(confusion_matrix(y_test, rf_y_pred))
```

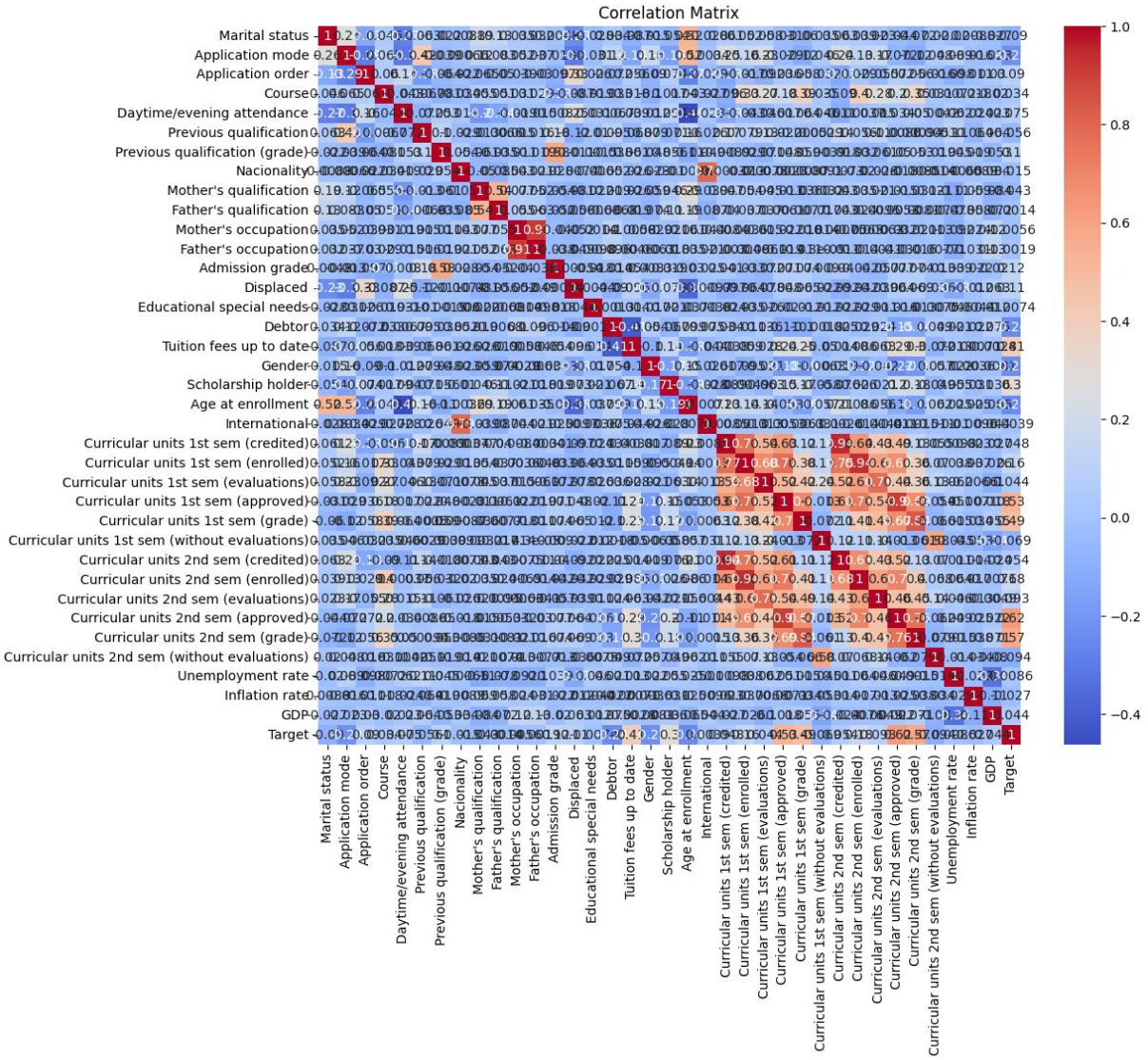
OUTPUT

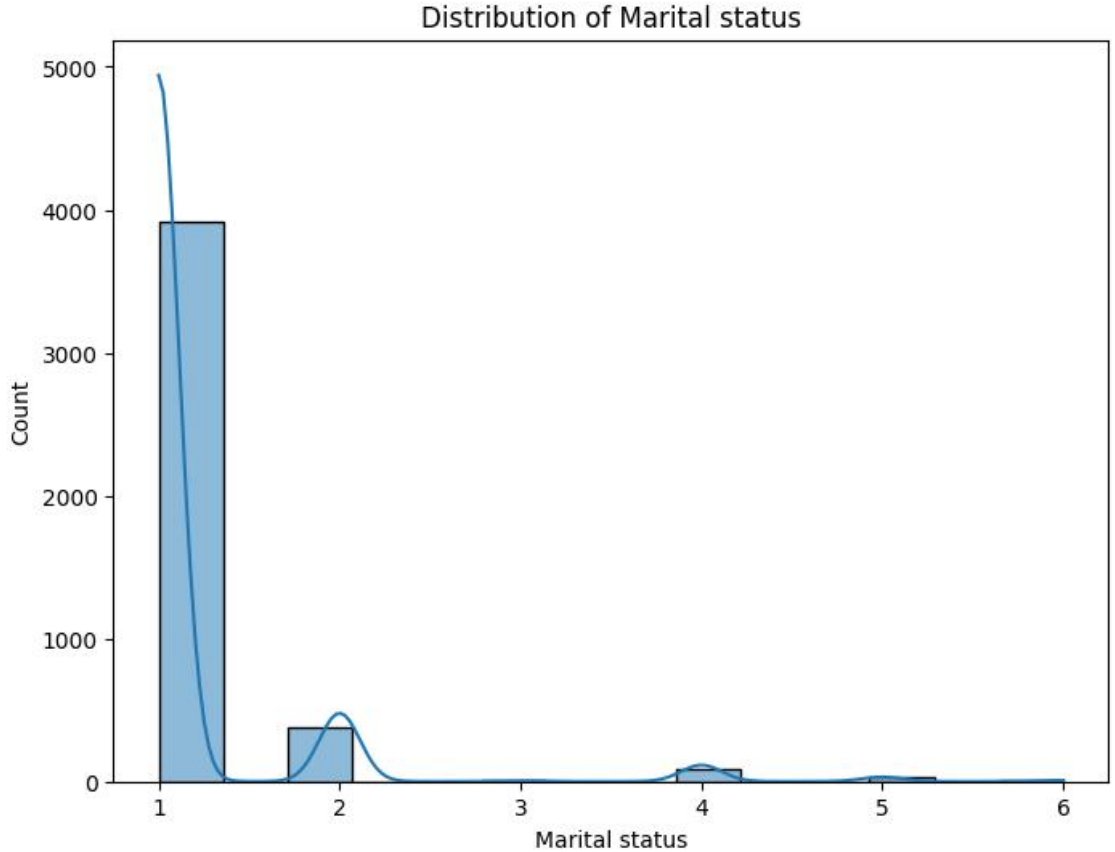
	Marital status	Application mode	Application order	Course
count	4424.000000	4424.000000	4424.000000	4424.000000
mean	1.178571	18.669078	1.727848	8856.642631
std	0.605747	17.484682	1.313793	2063.566416
min	1.000000	1.000000	0.000000	33.000000
25%	1.000000	1.000000	1.000000	9085.000000
50%	1.000000	17.000000	1.000000	9238.000000
75%	1.000000	39.000000	2.000000	9556.000000
max	6.000000	57.000000	9.000000	9991.000000

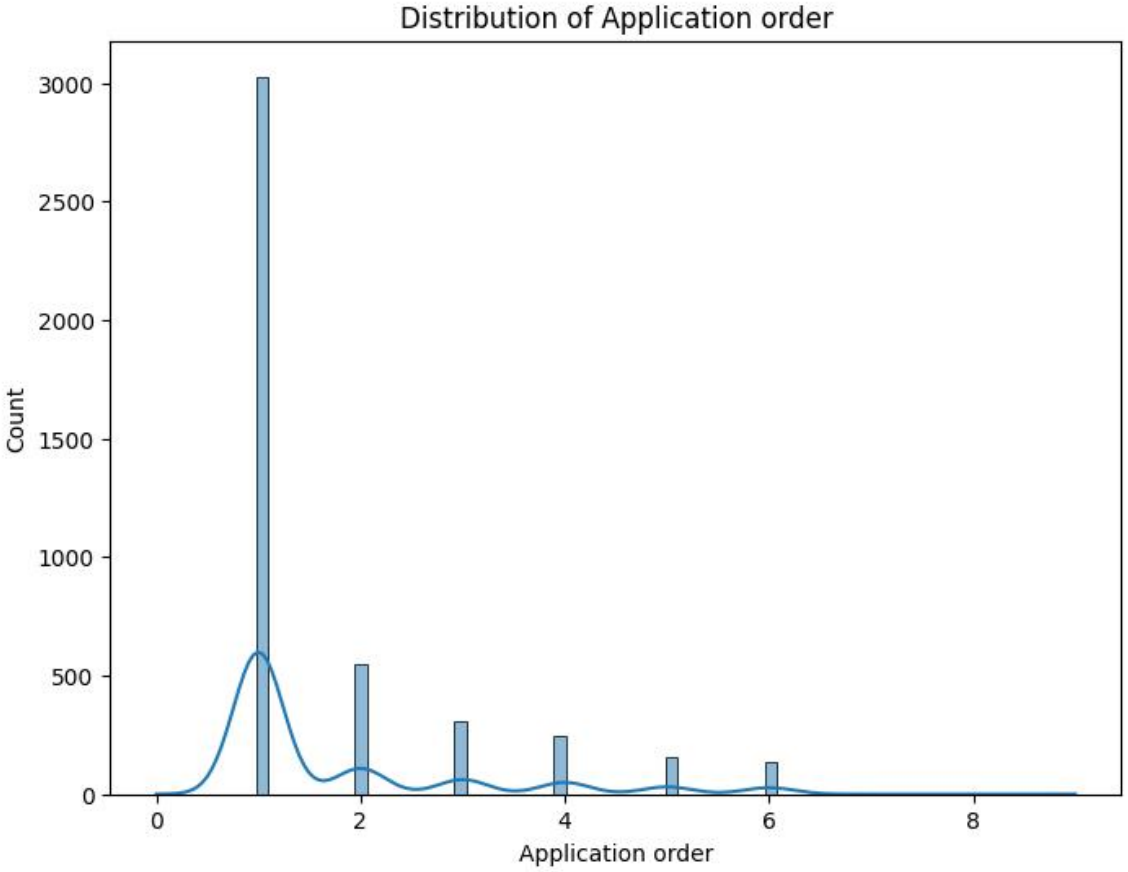
	Daytime/evening attendance	Previous qualification
count	4424.000000	4424.000000
mean	0.890823	4.577758
std	0.311897	10.216592
min	0.000000	1.000000
25%	1.000000	1.000000
50%	1.000000	1.000000
75%	1.000000	1.000000
max	1.000000	43.000000

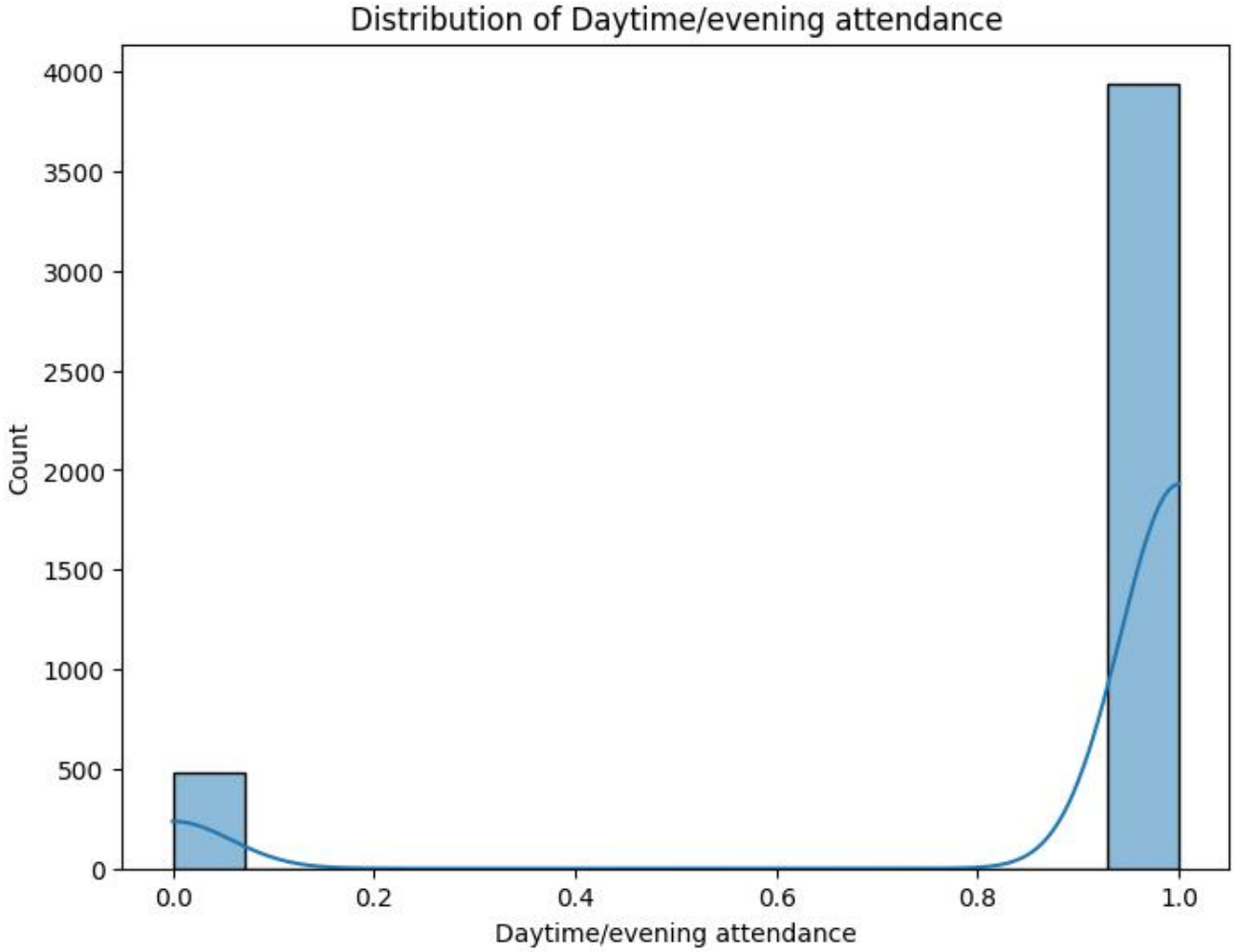
	Previous qualification (grade)	Nacionality	Mother's qualification
count	4424.000000	4424.000000	4424.000000
mean	132.613314	1.873192	19.561935
std	13.188332	6.914514	15.603186
min	95.000000	1.000000	1.000000
25%	125.000000	1.000000	2.000000
50%	133.100000	1.000000	19.000000
75%	140.000000	1.000000	37.000000
max	190.000000	109.000000	44.000000

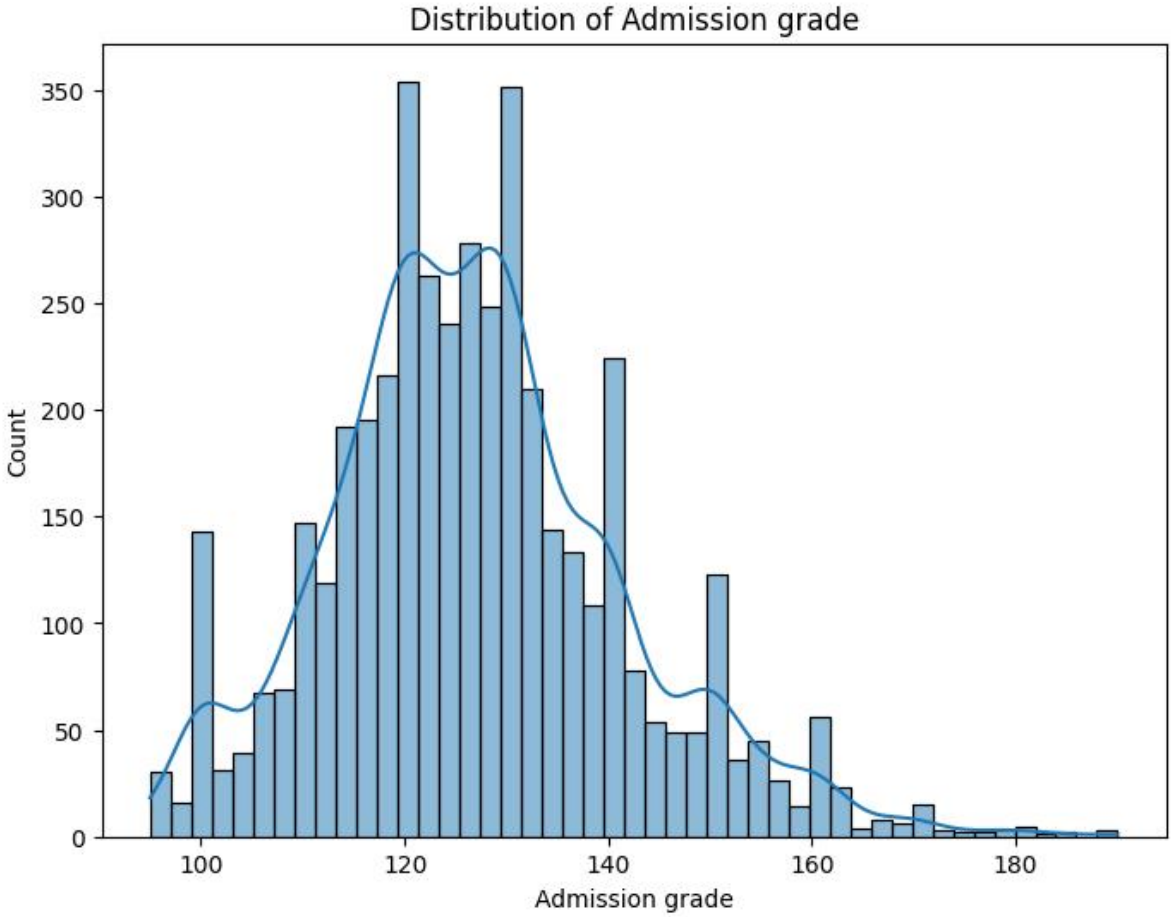
	Father's qualification	...	Curricular units 2nd sem (credited)
count	4424.000000	...	4424.000000
mean	22.275316	...	0.541817
std	15.343108	...	1.918546
min	1.000000	...	0.000000
25%	3.000000	...	0.000000
50%	19.000000	...	0.000000
75%	37.000000	...	0.000000
max	44.000000	...	19.000000

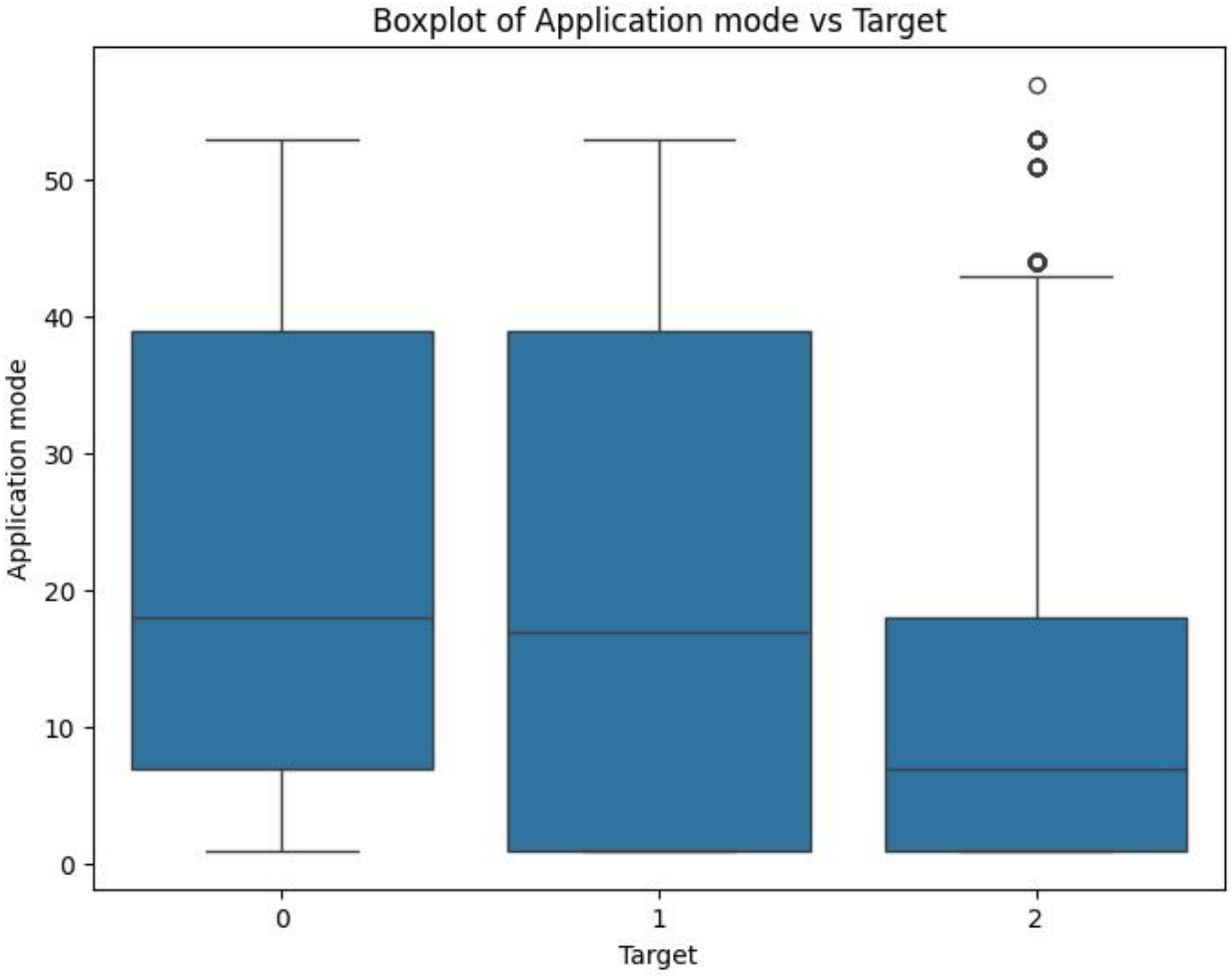

Correlation Matrix
Marital status
Application mode
Application order
Course
Daytime/evening attendance
Previous qualification
Previous qualification (grade)
Nacionality
Mother's qualification
Father's qualification
Mother's occupation
Father's occupation
Admission grade
Displaced
Educational special needs
Debtor
Tuition fees up to date
Gender
Scholarship holder
Age at enrollment
International
Curricular units 1st sem (credited)
Curricular units 1st sem (enrolled)
Curricular units 1st sem (evaluations)
Curricular units 1st sem (approved)
Curricular units 1st sem (grade)
Curricular units 1st sem (without evaluations)
Curricular units 2nd sem (credited)
Curricular units 2nd sem (enrolled)
Curricular units 2nd sem (evaluations)
Curricular units 2nd sem (approved)
Curricular units 2nd sem (grade)
Curricular units 2nd sem (without evaluations)
Unemployment rate
Inflation rate
GDP
Target
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4

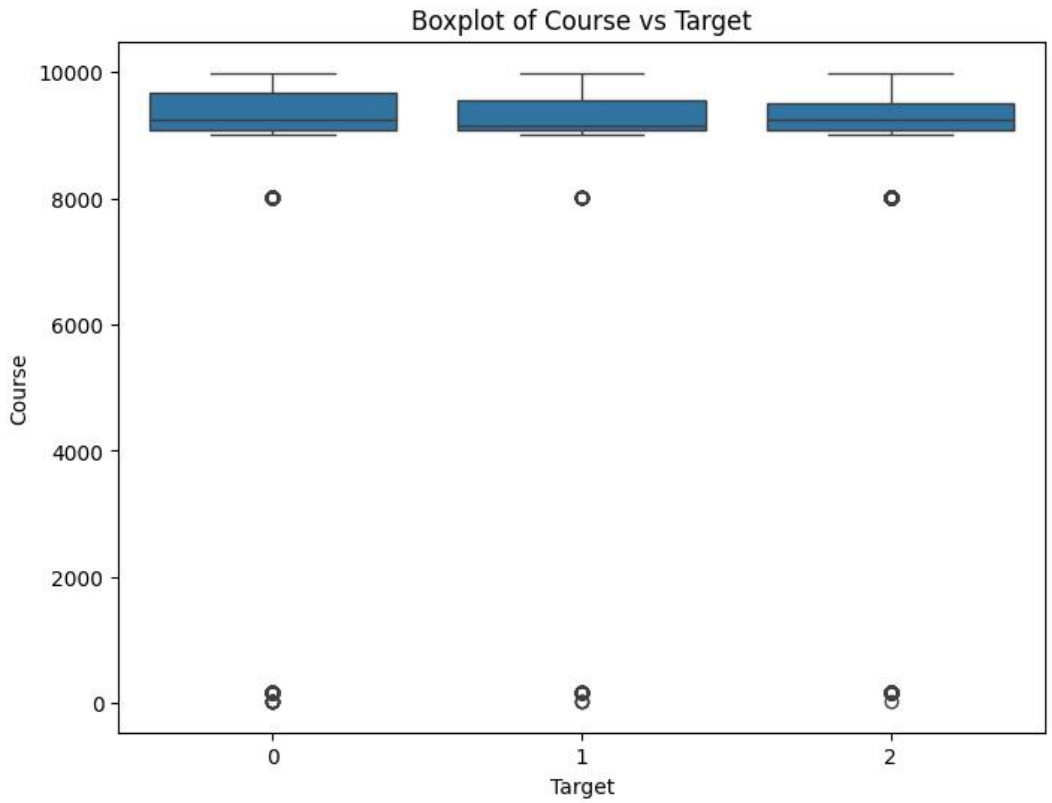

Distribution of Marital status
Count
Marital status
5000
4000
3000
2000
1000
0
1
2
3
4
5
6

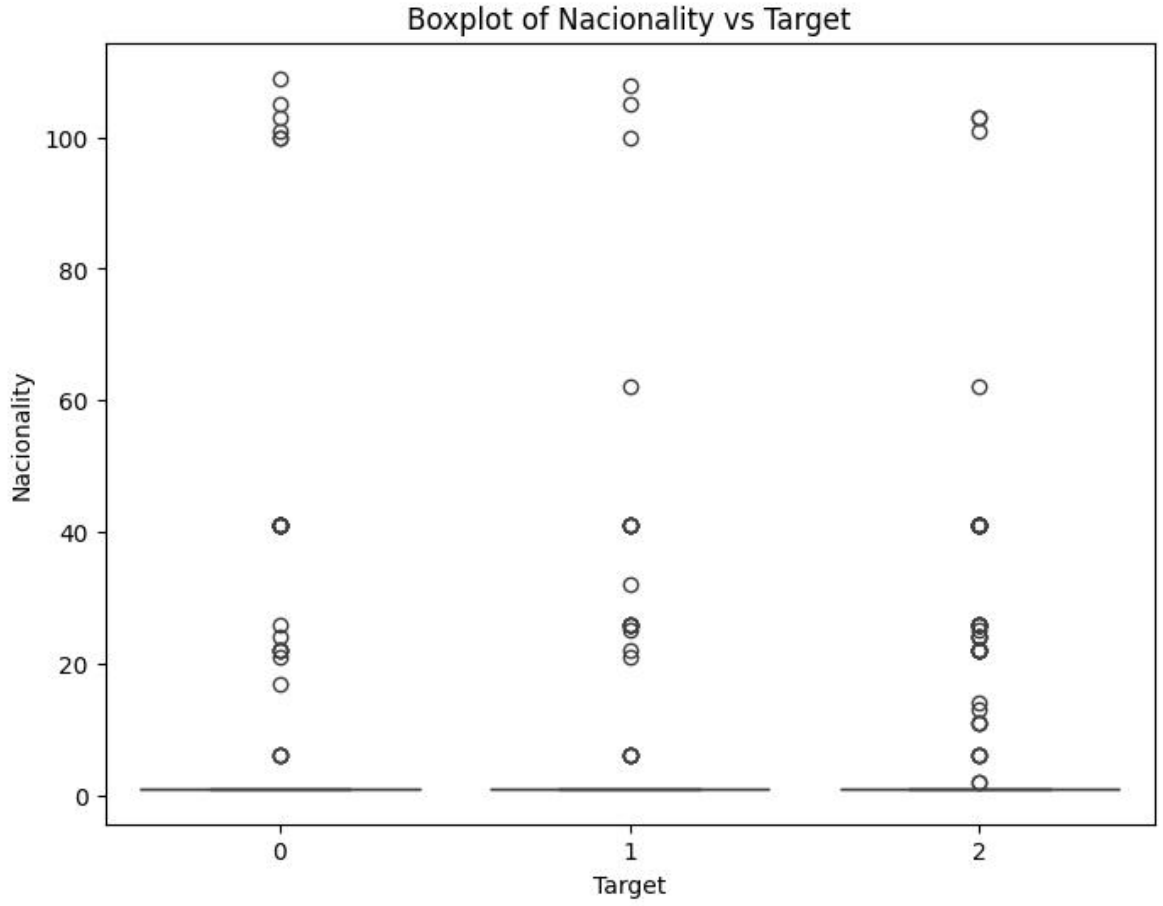

Distribution of Application order
Count
Application order
0
500
1000
1500
2000
2500
3000
0
2
4
6
8

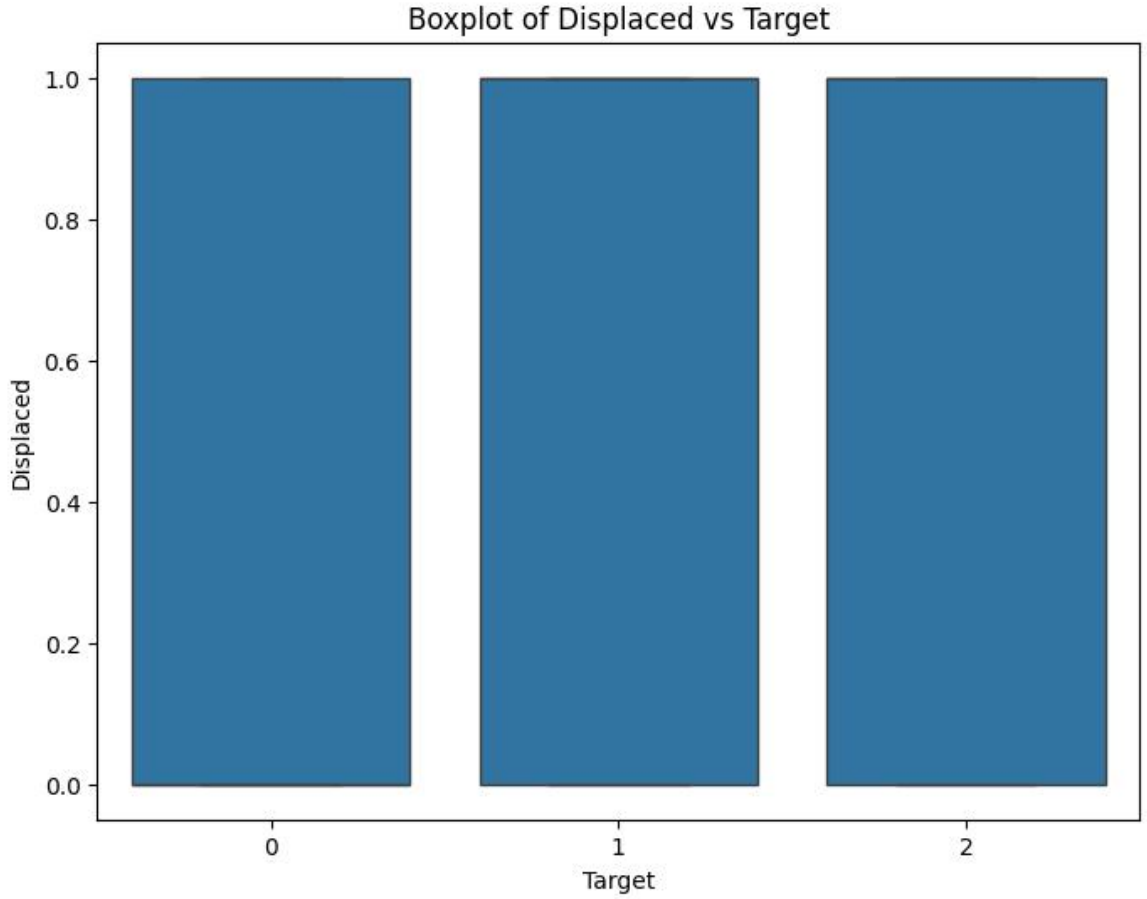

Distribution of Daytime/evening attendance
Count
Daytime/evening attendance
0
500
1000
1500
2000
2500
3000
3500
4000
0.0
0.2
0.4
0.6
0.8
1.0

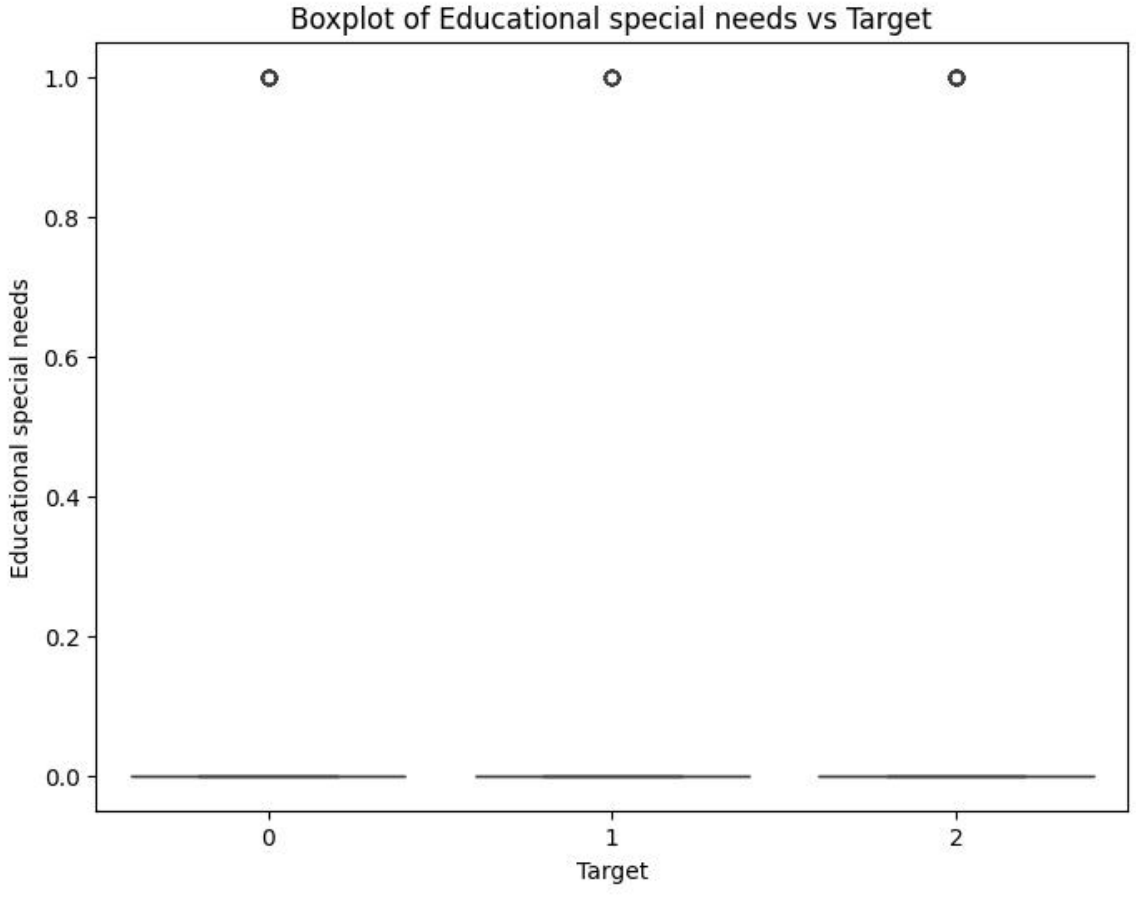

Distribution of Admission grade
Count
Admission grade
0
50
100
150
200
250
300
350
100
120
140
160
180


Boxplot of Application mode vs Target
Application mode
Target
0
10
20
30
40
50
0
1
2


Boxplot of Course vs Target
Course
Target
10000
8000
6000
4000
2000
0
0
1
2


Boxplot of Nacionality vs Target
Nacionality
Target
100
80
60
40
20
0
0
1
2


Boxplot of Displaced vs Target
Displaced
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
Target


Boxplot of Educational special needs vs Target
Educational special needs
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
Target

	precision	recall	f1-score	support
0	0.83	0.77	0.80	316
1	0.45	0.28	0.34	151
2	0.76	0.91	0.83	418
accuracy			0.75	885
macro avg	0.68	0.65	0.66	885
weighted avg	0.73	0.75	0.74	885

```
[[244  27  45]
 [ 35  42  74]
 [ 14  24 380]]
```

	precision	recall	f1-score	support
0	0.84	0.77	0.81	316
1	0.49	0.29	0.37	151
2	0.76	0.92	0.83	418
accuracy			0.76	885
macro avg	0.70	0.66	0.67	885
weighted avg	0.74	0.76	0.74	885

```
[[244  21  51]
 [ 36  44  71]
 [  9  25 384]]
```