Machine Learning（SS24）

Assignment 01: Preprocessing and K-Nearest Neighbors

Qu Wang

M.Sc, Matriculation Number 3700666, Study Program: Integrative Technologies and Architectural Design Research (ITECH)

st190363@stud.uni-stuttgart.de

Lianhan Huang

M.Sc, Matriculation Number 3700459, Study Program: Integrative Technologies and Architectural Design Research(ITECH)

St188954@stud.uni-stuttgart.de

1. Preprocessing

Final result:

ID	Age	Income	Owns_Car
1.0	25.0	0.3333333333333335	1.0
2.0	33.75	0.0	0.0
3.0	35.0	0.5	1.0
4.0	45.0	1.0000000000000002	1.0
5.0	30.0	0.6666666666666667	0.0

Number of Vehicles	Preferred Transport Mode_Bike	Preferred Transport Mode_Car	Preferred Transport Mode_Public Transport
2.0	0.0	1.0	0.0
0.0	0.0	0.0	1.0
1.0	0.0	1.0	0.0
0.0	0.0	1.0	0.0
0.0	1.0	0.0	0.0

Codes:

assignment1.py

```python
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
import matplotlib.pyplot as plt


# Specify the file path
file_path = "D:\\A1_ITECH\\12_ML\Assignments_github\ML_assignment\\assignment1\\transportation_preferance.csv"

# Read the CSV file into a DataFrame
df = pd.read_csv(file_path)

print(df.head())

#question_a
# Identify columns with missing values
missing_cols = df.columns[df.isnull().any()]

# Impute missing values
for col in missing_cols:
    if col == 'Age':
        mean_age = df['Age'].mean()
        df[col].fillna(mean_age, inplace=True)
    elif col == 'Income':
        median_income = df['Income'].median()
        df[col].fillna(median_income, inplace=True)
    elif col == 'Number of Vehicles':
        mode_vehicles = df['Number of Vehicles'].mode()[0]
        df[col].fillna(mode_vehicles, inplace=True)

print(df)
```

```python

#question_b
# Initialize the MinMaxScaler
scaler = MinMaxScaler()

# Apply Min-Max scaling to the "Income (K$)" column
df['Income'] = scaler.fit_transform(df[['Income']])

print(df['Income'])

#question_c
# Binary encoding for "Owns_Car" column
df['Owns_Car'] = df['Owns_Car'].map({'Yes': 1, 'No': 0})

# One-hot encoding for "Preferred Transport Mode" column
df = pd.get_dummies(df, columns=['Preferred Transport Mode'], dtype=int)

print(df['Preferred Transport Mode_Car'], df['Preferred Transport Mode_Bike'],
      df['Preferred Transport Mode_Public Transport'])

# Plotting the DataFrame
plt.figure(figsize=(40, 30))  # Adjust the figure size as needed
plt.table(cellText=df.values,
          colLabels=df.columns,
          loc='center')
plt.axis('off')  # Turn off the axis
plt.savefig( *args: 'table_image.png', bbox_inches='tight', pad_inches=0.05)  # Save as image
plt.show()

```