Suraj Sawant TEB-38

DSBDA Practical No A-2: Data Wrangling II

Create an “Academic performance” dataset of students an perform the following operations using Python.

1. Scan all variables for missing values and inconsistencies. If there are missing values and/or inconsistencies,use any of the suitable techniques to deal with them.

2. Scan all numeric variables for outliers. If there are outliers, use any of the suitable techniques to deal withthem.

3. Apply data transformations on at least one of the variables. The purpose of thistransformation should be oneof the following reasons: to change the scale for better understanding of the variable, to convert a non-linear relation into a linear one, or to decrease the skewness and convert the distribution into a normal distribution.

```
import pandas as pd


import numpy as np


df=pd.DataFrame()


df['Rollo']=[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
df['Maths']=[66, 85, 78, 60, 45, 56, 70, np.nan, 80, 110]
df['Science']=[90, 83, 46, 78, 84, 57, 68, 43, 67, 58]
df['English']=[79, 83, 57, 66, 49, 87, 73, 69, 52, 68]
df['Attendance']=[90, 80, 74, 86, '93%', 88, 69, 77, 95, 96]
```

```
df
```

	Rollo	Maths	Science	English	Attendance
0	1	66.0	90	79	90
1	2	85.0	83	83	80
2	3	78.0	46	57	74
3	4	60.0	78	66	86
4	5	45.0	84	49	93%
5	6	56.0	57	87	88
6	7	70.0	68	73	69
7	8	NaN	43	69	77
8	9	80.0	67	52	95
9	10	110.0	58	68	96

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
 RangeIndex: 10 entries, 0 to 9
 Data columns (total 5 columns):
  #   Column      Non-Null Count  Dtype
 ---  ------      --------------  -----
 0   Rollo       10 non-null     int64
 1   Maths       9 non-null      float64
 2   Science     10 non-null     int64
 3   English     10 non-null     int64    4   Attendance   10 non-null     object dtypes: float64(1), int64(3), object(1) memory usage:
     532.0+ bytes
```

```
df.describe()
```

	Rollo	Maths	Science	English
count	10.00000	9.000000	10.000000	10.000000
mean	5.50000	72.222222	67.400000	68.300000
std	3.02765	18.978789	16.304055	12.798003
min	1.00000	45.000000	43.000000	49.000000

25%	3.25000	60.000000	57.250000	59.250000
50%	5.50000	70.000000	67.500000	68.500000
75%	7.75000	80.000000	81.750000	77.500000
max	10.00000	110.000000	90.000000	87.000000

```
df.isnull().sum()
```

```
Rollo        0
Maths        0
Science      0
English      0
Attendance   0
dtype: int64
```

```
df.fillna({'Maths': df['Maths'].mean()}, inplace=True)
```

```
df
```

	Rollo	Maths	Science	English	Attendance	
0	1	66.000000	90	79	90	
1	2	85.000000	83	83	80	
2	3	78.000000	46	57	74	
3	4	60.000000	78	66	86	
4	5	45.000000	84	49	93%	
5	6	56.000000	57	87	88	
6	7	70.000000	68	73	69	
7	8	72.222222	43	69	77	
8	9	80.000000	67	52	95	
9	10	110.000000		58	68	96

```
df.isnull().sum()
```

```
Rollo        0
Maths        0
Science      0
English      0
Attendance   0
dtype: int64
```

```
df
```

	Rollo	Maths	Science	English	Attendance	
0	1	66.000000	90	79	90	
1	2	85.000000	83	83	80	
2	3	78.000000	46	57	74	
3	4	60.000000	78	66	86	
4	5	45.000000	84	49	93%	
5	6	56.000000	57	87	88	
6	7	70.000000	68	73	69	
7	8	72.222222	43	69	77	
8	9	80.000000	67	52	95	
9	10	110.000000		58	68	96

```
df['Attendance'] = pd.to_numeric(df['Attendance'], errors='coerce')
```

```
df.fillna({'Attendance': df['Attendance'].mean()}, inplace=True)
```

```
df
```

	Rollo	Maths	Science	English	Attendance
0	1	66.000000	90	79	90.000000
1	2	85.000000	83	83	80.000000
2	3	78.000000	46	57	74.000000

3	4	60.000000	78	66	86.000000
4	5	45.000000	84	49	83.888889
5	6	56.000000	57	87	88.000000
6	7	70.000000	68	73	69.000000
7	8	72.222222	43	69	77.000000
8	9	80.000000	67	52	95.000000
9	10	110.000000	58	68	96.000000

```
df.describe()
```

	Rollo	Maths	Science	English	Attendance
count	10.00000	10.000000	10.000000	10.000000	10.000000
mean	5.50000	72.222222	67.400000	68.300000	83.888889
std	3.02765	17.893374	16.304055	12.798003	8.887500
min	1.00000	45.000000	43.000000	49.000000	69.000000
25%	3.25000	61.500000	57.250000	59.250000	77.750000
50%	5.50000	71.111111	67.500000	68.500000	84.944444
75%	7.75000	79.500000	81.750000	77.500000	89.500000
max	10.00000	110.000000	90.000000	87.000000	96.000000

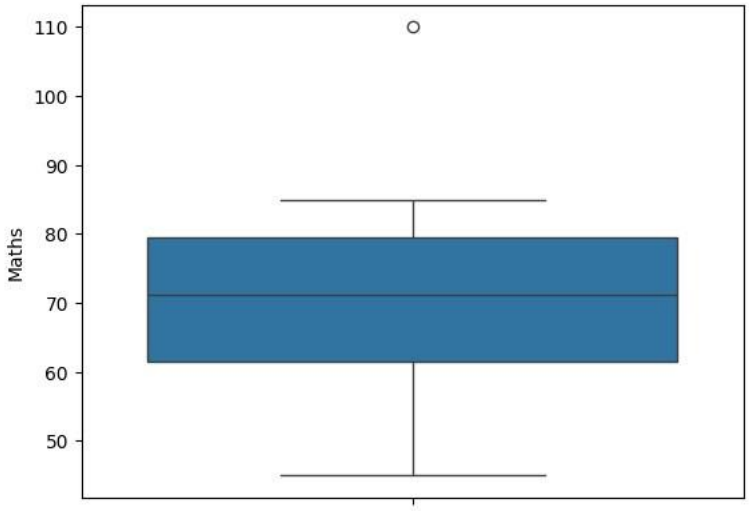
```
pip install seaborn
```

```
Collecting seaborn
  Downloading seaborn-0.13.2-py3-none-any.whl.metadata (5.4 kB)
Requirement already satisfied: numpy!=1.24.0,>=1.20 in c:\users\admin\.conda\envs\tea21\lib\site-packages (from seaborn) (2.2.1)
Requirement already satisfied: pandas>=1.2 in c:\users\admin\.conda\envs\tea21\lib\site-packages (from seaborn) (2.2.3)
Collecting matplotlib!=3.6.1,>=3.4 (from seaborn)
  Downloading matplotlib-3.10.0-cp312-cp312-win_amd64.whl.metadata (11 kB)
Collecting contourpy>=1.0.1 (from matplotlib!=3.6.1,>=3.4->seaborn)
  Downloading contourpy-1.3.1-cp312-cp312-win_amd64.whl.metadata (5.4 kB)
Collecting cycler>=0.10 (from matplotlib!=3.6.1,>=3.4->seaborn)
  Downloading cycler-0.12.1-py3-none-any.whl.metadata (3.8 kB)
Collecting fonttools>=4.22.0 (from matplotlib!=3.6.1,>=3.4->seaborn)
  Downloading fonttools-4.55.3-cp312-cp312-win_amd64.whl.metadata (168 kB)
Collecting kiwisolver>=1.3.1 (from matplotlib!=3.6.1,>=3.4->seaborn)
  Downloading kiwisolver-1.4.8-cp312-cp312-win_amd64.whl.metadata (6.3 kB)
Requirement already satisfied: packaging>=20.0 in c:\users\admin\.conda\envs\tea21\lib\site-packages (from matplotlib!=3.6.1,>=3.4->
Collecting pillow>=8 (from matplotlib!=3.6.1,>=3.4->seaborn)
  Downloading pillow-11.1.0-cp312-cp312-win_amd64.whl.metadata (9.3 kB)
Collecting pyparsing>=2.3.1 (from matplotlib!=3.6.1,>=3.4->seaborn)
  Downloading pyparsing-3.2.1-py3-none-any.whl.metadata (5.0 kB)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\admin\.conda\envs\tea21\lib\site-packages (from matplotlib!=3.6.1,>=
Requirement already satisfied: pytz>=2020.1 in c:\users\admin\.conda\envs\tea21\lib\site-packages (from pandas>=1.2->seaborn) (2024
Requirement already satisfied: tzdata>=2022.7 in c:\users\admin\.conda\envs\tea21\lib\site-packages (from pandas>=1.2->seaborn) (202
Requirement already satisfied: six>=1.5 in c:\users\admin\.conda\envs\tea21\lib\site-packages (from python-dateutil>=2.7->matplotlib
Downloading seaborn-0.13.2-py3-none-any.whl (294 kB)
Downloading matplotlib-3.10.0-cp312-cp312-win_amd64.whl (8.0 MB)
   ---------------------------------------- 0.0/8.0 MB ? eta -:--:--
   ------------ --------------------------- 2.4/8.0 MB 12.2 MB/s eta 0:00:01
   ------------------------- -------------- 5.0/8.0 MB 12.1 MB/s eta 0:00:01
   ------------------------------------- -- 7.6/8.0 MB 12.0 MB/s eta 0:00:01
   ---------------------------------------- 8.0/8.0 MB 11.3 MB/s eta 0:00:00
Downloading contourpy-1.3.1-cp312-cp312-win_amd64.whl (220 kB)
Downloading cycler-0.12.1-py3-none-any.whl (8.3 kB)
Downloading fonttools-4.55.3-cp312-cp312-win_amd64.whl (2.2 MB)
   ---------------------------------------- 0.0/2.2 MB ? eta -:--:--
   ---------------------------------------- 2.2/2.2 MB 11.3 MB/s eta 0:00:00
Downloading kiwisolver-1.4.8-cp312-cp312-win_amd64.whl (71 kB)
Downloading pillow-11.1.0-cp312-cp312-win_amd64.whl (2.6 MB)
   ---------------------------------------- 0.0/2.6 MB ? eta -:--:--
   ------------------------------------ ---- 2.4/2.6 MB 12.2 MB/s eta 0:00:01
   ---------------------------------------- 2.6/2.6 MB 11.6 MB/s eta 0:00:00
Downloading pyparsing-3.2.1-py3-none-any.whl (107 kB)
Installing collected packages: pyparsing, pillow, kiwisolver, fonttools, cycler, contourpy, matplotlib, seaborn
Successfully installed contourpy-1.3.1 cycler-0.12.1 fonttools-4.55.3 kiwisolver-1.4.8 matplotlib-3.10.0 pillow-11.1.0 pyparsing-3.2
Note: you may need to restart the kernel to use updated packages.
```

```
import seaborn as sns
```

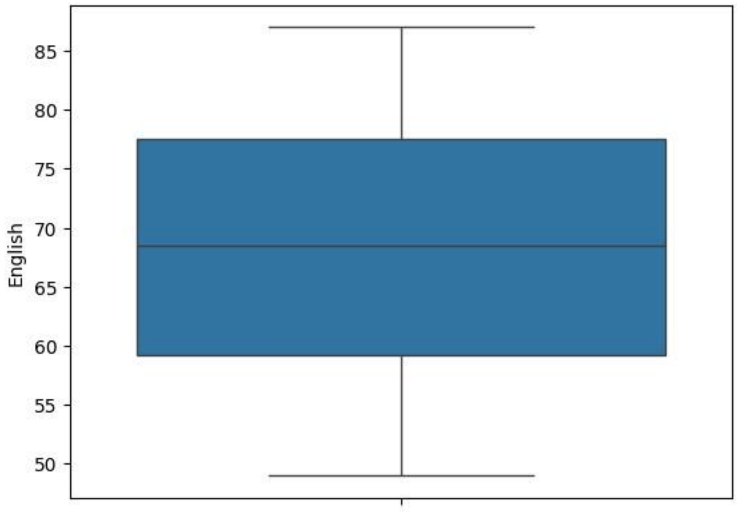
```
sns.boxplot(y=df['Maths'])
```

```
<Axes: ylabel='Maths'>
```



```
sns.boxplot(y=df['English'])
```

```
<Axes: ylabel='English'>
```



```
Q1 = df['Maths'].quantile(0.25)
Q3 =
df['Maths'].quantile(0.75) IQR
= Q3 - Q1 lower_bound = Q1 -
1.1 * IQR upper_bound = Q3 +
1.1 * IQR
```

```
lower_bound
```

```
np.float64(41.7)
```

```
upper_bound
```

```
np.float64(99.3)
```

```
df
```

	Rollo	Maths	Science	English	Attendance
0	1	66.000000	90	79	90.000000
1	2	85.000000	83	83	80.000000
2	3	78.000000	46	57	74.000000
3	4	60.000000	78	66	86.000000
4	5	45.000000	84	49	83.888889
5	6	56.000000	57	87	88.000000
6	7	70.000000	68	73	69.000000

7	8	72.222222	43	69	77.000000
8	9	80.000000	67	52	95.000000
9	10	110.000000	58	68	96.000000

```
pip install scikit-learn
```

```
Collecting scikit-learn
  Downloading scikit_learn-1.6.1-cp312-cp312-win_amd64.whl.metadata (15 kB)
Requirement already satisfied: numpy>=1.19.5 in c:\users\admin\.conda\envs\tea21\lib\site-packages (from scikit-learn) (2.2.1)
Collecting scipy>=1.6.0 (from scikit-learn)
  Downloading scipy-1.15.1-cp312-cp312-win_amd64.whl.metadata (60 kB)
Collecting joblib>=1.2.0 (from scikit-learn)
  Downloading joblib-1.4.2-py3-none-any.whl.metadata (5.4 kB)
Collecting threadpoolctl>=3.1.0 (from scikit-learn)
  Downloading threadpoolctl-3.5.0-py3-none-any.whl.metadata (13 kB)
Downloading scikit_learn-1.6.1-cp312-cp312-win_amd64.whl (11.1 MB)
   ---------------------------------------- 0.0/11.1 MB ? eta -:--:--
   -------- ------------------------------- 2.4/11.1 MB 11.2 MB/s eta 0:00:01
   ---------------- ----------------------- 4.7/11.1 MB 11.4 MB/s eta 0:00:01
   -------------------------- ------------- 7.3/11.1 MB 11.3 MB/s eta 0:00:01
   ------------------------------------ --- 10.0/11.1 MB 11.7 MB/s eta 0:00:01
   ---------------------------------------- 11.1/11.1 MB 11.2 MB/s eta 0:00:00
Downloading joblib-1.4.2-py3-none-any.whl (301 kB)
Downloading scipy-1.15.1-cp312-cp312-win_amd64.whl (43.6 MB)
   ---------------------------------------- 0.0/43.6 MB ? eta -:--:--
   -- ------------------------------------- 2.6/43.6 MB 11.6 MB/s eta 0:00:04
   ---- ----------------------------------- 4.7/43.6 MB 11.4 MB/s eta 0:00:04
   ------ --------------------------------- 7.6/43.6 MB 11.7 MB/s eta 0:00:04
   --------- ------------------------------ 10.0/43.6 MB 11.7 MB/s eta 0:00:03
   ----------- ---------------------------- 12.6/43.6 MB 11.6 MB/s eta 0:00:03
   ------------- -------------------------- 15.2/43.6 MB 11.7 MB/s eta 0:00:03
   ---------------- ----------------------- 17.8/43.6 MB 11.8 MB/s eta 0:00:03
   ------------------ --------------------- 20.2/43.6 MB 11.7 MB/s eta 0:00:03
   -------------------- ------------------- 22.8/43.6 MB 11.7 MB/s eta 0:00:02
   ----------------------- ---------------- 25.4/43.6 MB 11.8 MB/s eta 0:00:02
   ------------------------- -------------- 27.8/43.6 MB 11.7 MB/s eta 0:00:02
   ---------------------------- ----------- 30.7/43.6 MB 11.8 MB/s eta 0:00:02
   ------------------------------ --------- 33.0/43.6 MB 11.7 MB/s eta 0:00:01
   -------------------------------- ------- 35.7/43.6 MB 11.7 MB/s eta 0:00:01
   ---------------------------------- ----- 38.0/43.6 MB 11.8 MB/s eta 0:00:01
   ------------------------------------- -- 40.6/43.6 MB 11.8 MB/s eta 0:00:01
   ---------------------------------------  43.5/43.6 MB 11.8 MB/s eta 0:00:01
   ---------------------------------------- 43.6/43.6 MB 11.5 MB/s eta 0:00:00
Downloading threadpoolctl-3.5.0-py3-none-any.whl (18 kB)
Installing collected packages: threadpoolctl, scipy, joblib, scikit-learn
Successfully installed joblib-1.4.2 scikit-learn-1.6.1 scipy-1.15.1 threadpoolctl-3.5.0
Note: you may need to restart the kernel to use updated packages.
```

```
from sklearn.preprocessing import MinMaxScaler
```

```
scaler=MinMaxScaler()
df[['Attendance']]=scaler.fit_transform(df[['Attendance']])
```

```
df
```

	Rollo	Maths	Science	English	Attendance
0	1	66.000000	90	79	0.777778
1	2	85.000000	83	83	0.407407
2	3	78.000000	46	57	0.185185
3	4	60.000000	78	66	0.629630
4	5	45.000000	84	49	0.551440
5	6	56.000000	57	87	0.703704
6	7	70.000000	68	73	0.000000
7	8	72.222222	43	69	0.296296
8	9	80.000000	67	52	0.962963
9	10	110.000000	58	68	1.000000

```
df.describe()
```

	Rollo	Maths	Science	English	Attendance

count	10.00000	10.000000	10.000000	10.000000	10.000000
mean	5.50000	72.222222	67.400000	68.300000	0.551440
std	3.02765	17.893374	16.304055	12.798003	0.329167
min	1.00000	45.000000	43.000000	49.000000	0.000000
25%	3.25000	61.500000	57.250000	59.250000	0.324074
50%	5.50000	71.111111	67.500000	68.500000	0.590535
75%	7.75000	79.500000	81.750000	77.500000	0.759259
max	10.00000	110.000000	90.000000	87.000000	1.000000

```
df
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-1-00cf07b74dcd> in <cell line: 0>() ---
-> 1 df

NameError: name 'df' is not defined
```

Start coding or generate with AI.