EC 9560 – DATA MINING

LAB 02

DARMILA.T

2020/E/027

SEMESTER 07

10th OCTOBER 2024

A comprehensive study on your data including data visualization, distribution analysis, correlation analysis.

The dataset have total 81 features. So first of all, I find the missing values count in the dataset.

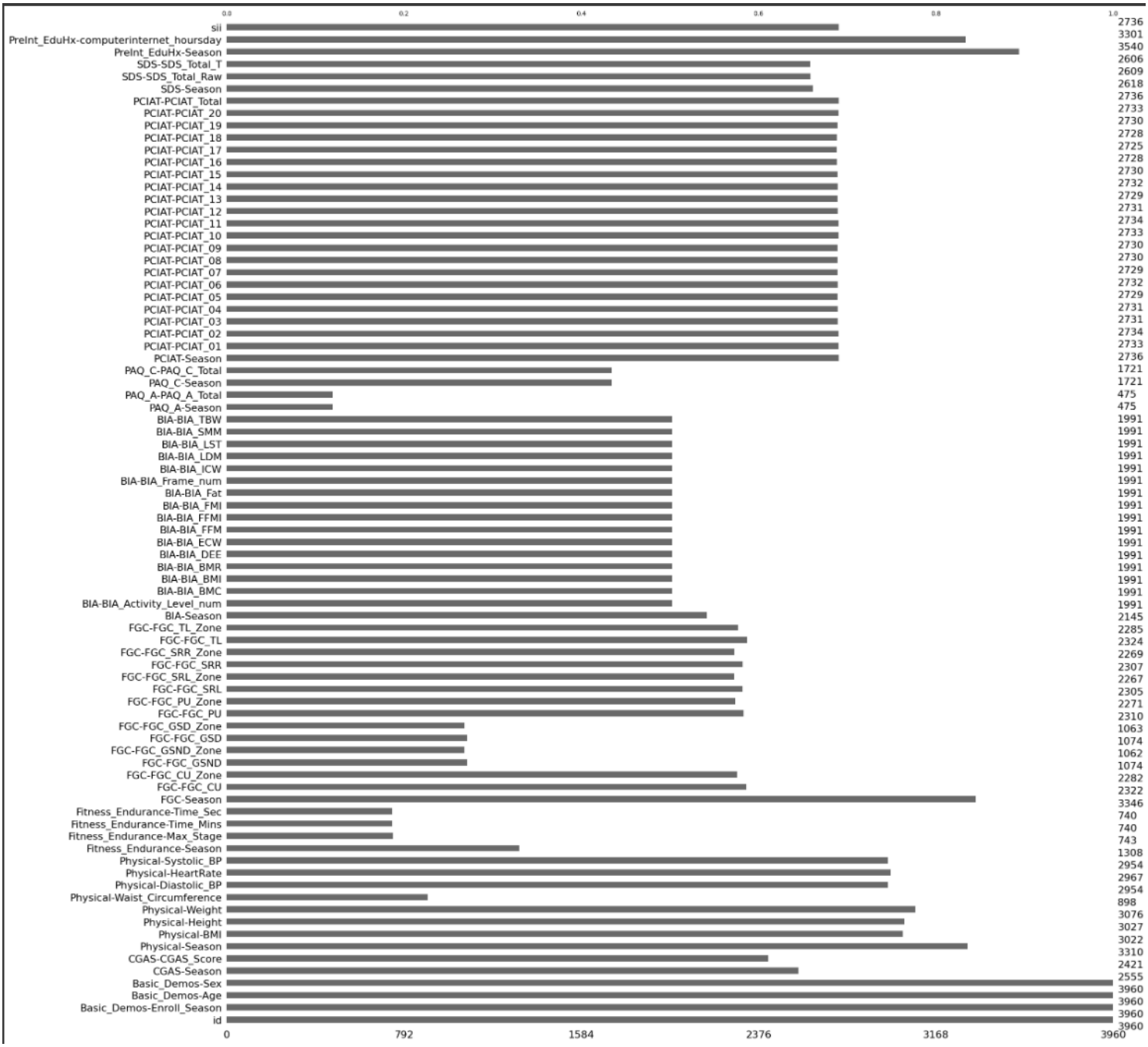
Column Name	Missing Values	Percentage (%)
id	0	0.00
Basic_Demos-Enroll_Season	0	0.00
Basic_Demos-Age	0	0.00
Basic_Demos-Sex	0	0.00
CGAS-Season	1405	35.48
CGAS-CGAS_Score	1539	38.86
Physical-Season	650	16.41
Physical-BMI	938	23.69
Physical-Height	933	23.56
Physical-Weight	884	22.32
Physical-Waist_Circumference	3062	77.32
Physical-Diastolic_BP	1006	25.40
Physical-HeartRate	993	25.08
Physical-Systolic_BP	1006	25.40
Fitness_Endurance-Season	2652	66.97
Fitness_Endurance-Max_Stage	3217	81.24
Fitness_Endurance-Time_Mins	3220	81.31
Fitness_Endurance-Time_Sec	3220	81.31
FGC-Season	614	15.51
FGC-FGC_CU	1638	41.36
FGC-FGC_CU_Zone	1678	42.37
FGC-FGC_GSND	2886	72.88
FGC-FGC_GSND_Zone	2898	73.18
FGC-FGC_GSD	2886	72.88
FGC-FGC_GSD_Zone	2897	73.16
FGC-FGC_PU	1650	41.67
FGC-FGC_PU_Zone	1689	42.65
FGC-FGC_SRL	1655	41.79
FGC-FGC_SRL_Zone	1693	42.75
FGC-FGC_SRR	1653	41.74
FGC-FGC_SRR_Zone	1691	42.70
FGC-FGC_TL	1636	41.31
FGC-FGC_TL_Zone	1675	42.30
BIA-Season	1815	45.83
BIA-BIA_Activity_Level_num	1969	49.72
BIA-BIA_BMC	1969	49.72
BIA-BIA_BMI	1969	49.72
BIA-BIA_BMR	1969	49.72
BIA-BIA_DEE	1969	49.72
BIA-BIA_ECW	1969	49.72
BIA-BIA_FFM	1969	49.72
BIA-BIA_FFMI	1969	49.72
BIA-BIA_FMI	1969	49.72
BIA-BIA_Fat	1969	49.72
BIA-BIA_Frame_num	1969	49.72
BIA-BIA_ICW	1969	49.72
BIA-BIA_LDM	1969	49.72
BIA-BIA_LST	1969	49.72
BIA-BIA_SMM	1969	49.72
BIA-BIA_TBW	1969	49.72
PAQ_A-Season	3485	88.01
PAQ_A-PAQ_A_Total	3485	88.01
PAQ_C-Season	2239	56.54
PAQ_C-PAQ_C_Total	2239	56.54
PCIAT-Season	1224	30.91
PCIAT-PCIAT_01	1227	30.98
PCIAT-PCIAT_02	1226	30.96
PCIAT-PCIAT_03	1229	31.04
PCIAT-PCIAT_04	1229	31.04
PCIAT-PCIAT_05	1231	31.09
PCIAT-PCIAT_06	1228	31.01
PCIAT-PCIAT_07	1231	31.09
PCIAT-PCIAT_08	1230	31.06
PCIAT-PCIAT_09	1230	31.06
PCIAT-PCIAT_10	1227	30.98
PCIAT-PCIAT_11	1226	30.96
PCIAT-PCIAT_12	1229	31.04
PCIAT-PCIAT_13	1231	31.09
PCIAT-PCIAT_14	1228	31.01
PCIAT-PCIAT_15	1230	31.06
PCIAT-PCIAT_16	1232	31.11
PCIAT-PCIAT_17	1235	31.19
PCIAT-PCIAT_18	1232	31.11
PCIAT-PCIAT_19	1230	31.06
PCIAT-PCIAT_20	1227	30.98
PCIAT-PCIAT_Total	1224	30.91
SDS-Season	1342	33.89
SDS-SDS_Total_Raw	1351	34.12
SDS-SDS_Total_T	1354	34.19
PreInt_EduHx-Season	420	10.61
PreInt_EduHx-computerinternet_hoursday	659	16.64
sii	1224	30.91

[55]

```python
# Get the total number of rows
total_rows = len(df_train)

# Print the header
print(f"{'Column Name':<30} {'Missing Values':<15} {'Percentage (%)':<15}")
print("-" * 60)

for i in range(len(df_train.columns)):
    missing_count = df_train[df_train.columns[i]].isnull().sum()
    missing_percentage = (missing_count / total_rows) * 100
    print(f"{df_train.columns[i]:<30} {missing_count:<15} {missing_percentage:<15.2f}")
```

Here, we can see most of the columns have much more null values because total there only 3960 samples.

```
[19]  1 df_train.shape

(3960, 82)
```



Total values of every features

Number of duplicates in the dataset

```
df_train.duplicated().sum()
```

```
0
```

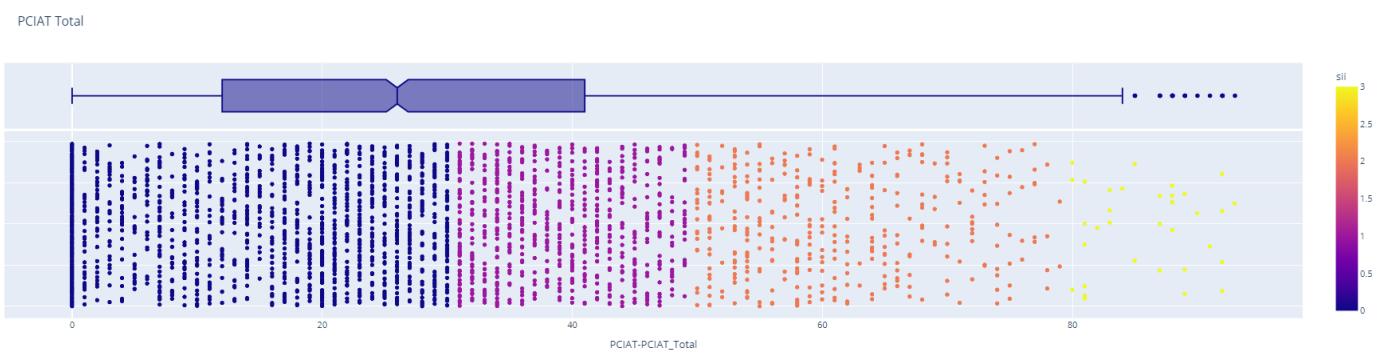
PCIAT Features:

```
PCIAT_cols = [val for val in df_train.columns[df_train.columns.str.contains('PCIAT')]]
print('Number of PCIAT features = ' , len(PCIAT_cols))
```

```
Number of PCIAT features =  22
```

- As mentioned, there are 22 PCIAT features. These comprise answers to 20 questions (each marked out of 5), the total score and 'season' when the test was carried out.
- The sii target is derived from the total PCIAT score:
  - 0-30 gives sii = 0
  - 31-49 gives sii = 1
  - 50-79 gives sii = 2
  - 80-100 gives sii = 3.

```
[86] fig = px.scatter(df_train, x = 'PCIAT-PCIAT_Total', color = 'sii', marginal_x="box", title = 'PCIAT Total')
     fig = fig.update_layout(yaxis_title="")
     fig.update_yaxes(showticklabels=False)
```

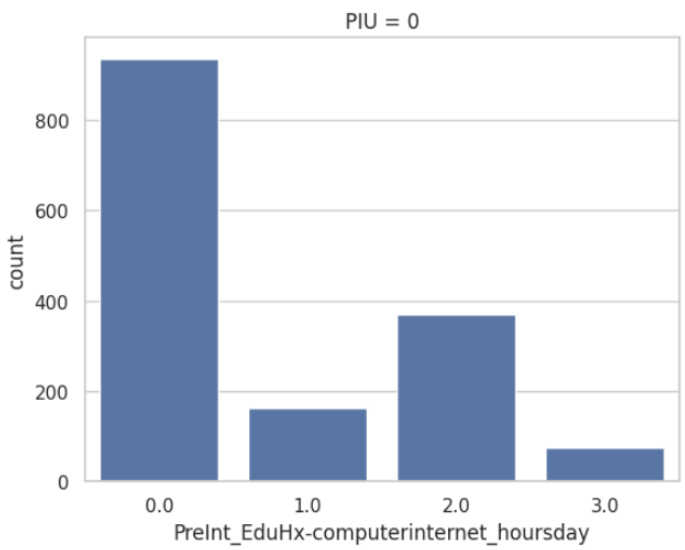


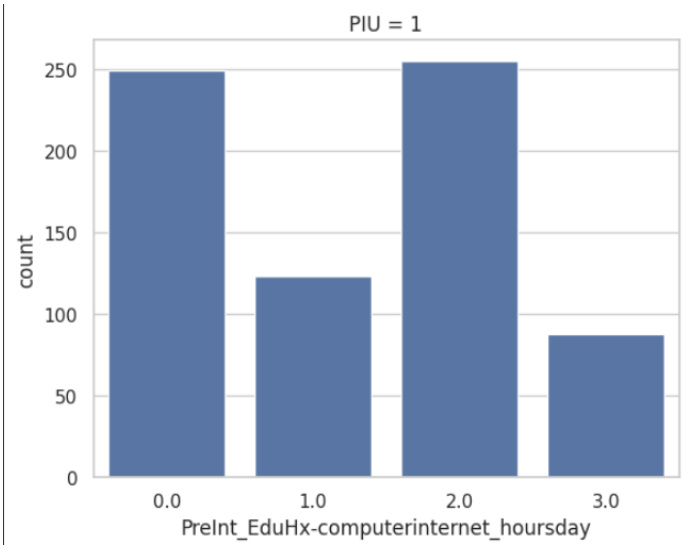
This graph shows the outliers and category the target column sii.

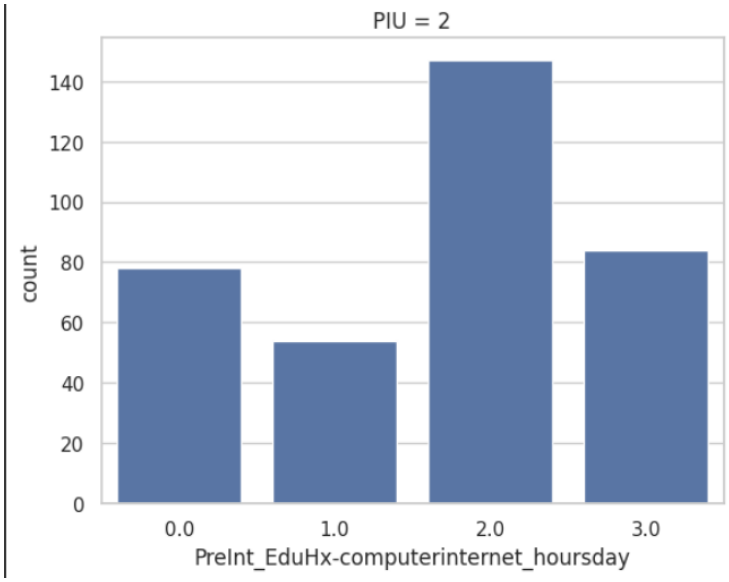
Severity Impairment Index:

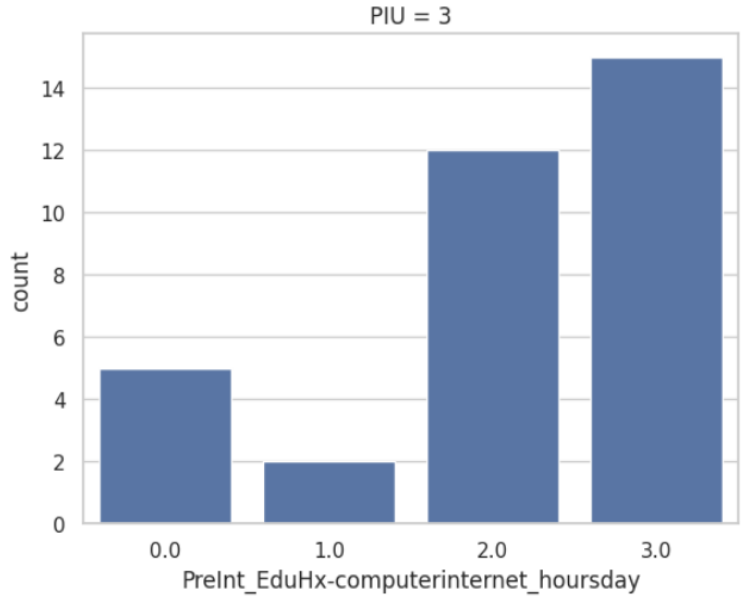
Here I plot when the sii = 0,1,2,3 for the PreInt_EduHx-computerinternet_hoursday

```
vals = ['PIU = 0', 'PIU = 1','PIU = 2', 'PIU = 3']

for i in range(4):
    plt.figure()
    plot = sns.countplot(x = df_train[df_train.sii==i]['PreInt_EduHx-computerinternet_hoursday'])
    plot.set_title(vals[i])
```

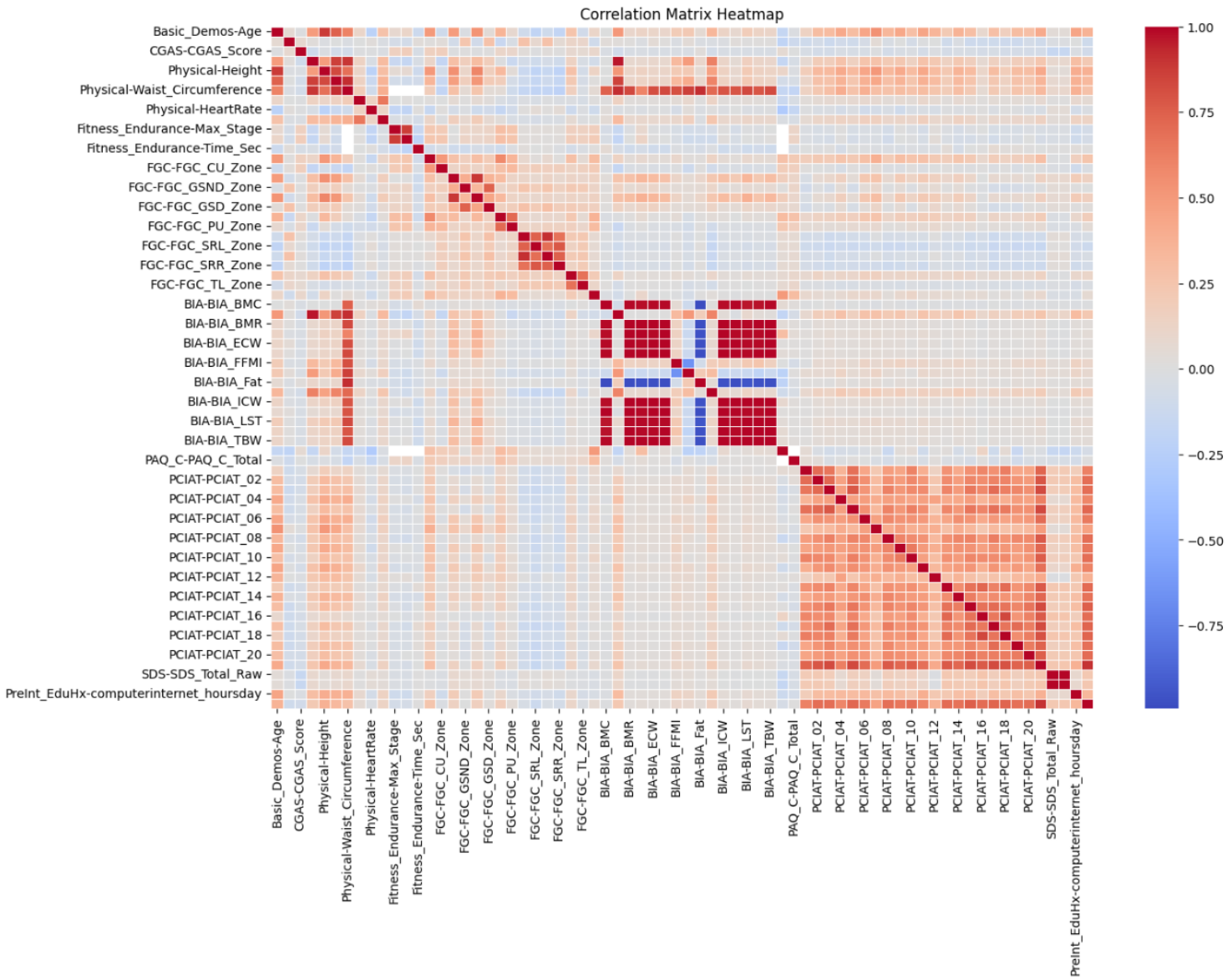








Correlation matrix



Based on this correlation find the maximum number of correlation pairs:

```
threshold = 0.8

# Find the pairs of highly correlated features
high_corr_pairs = []

# Iterate over the correlation matrix
for i in range(len(corr_matrix.columns)):
    for j in range(i+1, len(corr_matrix.columns)):
        if np.abs(corr_matrix.iloc[i, j]) > threshold:
            feature_pair = (corr_matrix.columns[i], corr_matrix.columns[j], corr_matrix.iloc[i, j])
            high_corr_pairs.append(feature_pair)

high_corr_df = pd.DataFrame(high_corr_pairs, columns=['Feature 1', 'Feature 2', 'Correlation'])
```

```
[70]
# Display the results
pd.set_option('display.max_rows', None)
high_corr_df
```

	Feature 1	Feature 2	Correlation
0	Basic_Demos-Age	Physical-Height	0.880274
1	Physical-BMI	Physical-Weight	0.865662
2	Physical-BMI	Physical-Waist_Circumference	0.892149
3	Physical-BMI	BIA-BIA_BMI	0.968849
4	Physical-Height	Physical-Weight	0.833844
5	Physical-Weight	Physical-Waist_Circumference	0.916710
6	Physical-Weight	BIA-BIA_BMI	0.858036
7	Physical-Waist_Circumference	BIA-BIA_BMI	0.920539
8	Physical-Waist_Circumference	BIA-BIA_BMR	0.830669
9	Physical-Waist_Circumference	BIA-BIA_ECW	0.818122
10	Physical-Waist_Circumference	BIA-BIA_FFM	0.830669
11	Physical-Waist_Circumference	BIA-BIA_FFMI	0.809447
12	Physical-Waist_Circumference	BIA-BIA_FMI	0.861946
13	Physical-Waist_Circumference	BIA-BIA_Fat	0.915948
14	Physical-Waist_Circumference	BIA-BIA_LST	0.833994
15	Physical-Waist_Circumference	BIA-BIA_TBW	0.824784
16	Fitness_Endurance-Max_Stage	Fitness_Endurance-Time_Mins	0.873138
17	FGC-FGC_GSND	FGC-FGC_GSD	0.885205
18	FGC-FGC_SRL	FGC-FGC_SRR	0.913546
19	BIA-BIA_BMC	BIA-BIA_BMR	0.989151
20	BIA-BIA_BMC	BIA-BIA_DEE	0.978063
21	BIA-BIA_BMC	BIA-BIA_ECW	0.988967
22	BIA-BIA_BMC	BIA-BIA_FFM	0.989151
23	BIA-BIA_BMC	BIA-BIA_Fat	-0.991534
24	BIA-BIA_BMC	BIA-BIA_ICW	0.978078
25	BIA-BIA_BMC	BIA-BIA_LDM	0.993702
26	BIA-BIA_BMC	BIA-BIA_LST	0.962998
27	BIA-BIA_BMC	BIA-BIA_SMM	0.970374
28	BIA-BIA_BMC	BIA-BIA_TBW	0.985577
29	BIA-BIA_BMR	BIA-BIA_DEE	0.993108
30	BIA-BIA_BMR	BIA-BIA_ECW	0.999119
31	BIA-BIA_BMR	BIA-BIA_FFM	1.000000
32	BIA-BIA_BMR	BIA-BIA_Fat	-0.976677
33	BIA-BIA_BMR	BIA-BIA_ICW	0.997115
34	BIA-BIA_BMR	BIA-BIA_LDM	0.998667
35	BIA-BIA_BMR	BIA-BIA_LST	0.992142
36	BIA-BIA_BMR	BIA-BIA_SMM	0.992282
37	BIA-BIA_BMR	BIA-BIA_TBW	0.999607
38	BIA-BIA_DEE	BIA-BIA_ECW	0.991315
39	BIA-BIA_DEE	BIA-BIA_FFM	0.993108
40	BIA-BIA_DEE	BIA-BIA_Fat	-0.966641
41	BIA-BIA_DEE	BIA-BIA_ICW	0.992906
42	BIA-BIA_DEE	BIA-BIA_LDM	0.990607
43	BIA-BIA_DEE	BIA-BIA_LST	0.988941
44	BIA-BIA_DEE	BIA-BIA_SMM	0.987791
45	BIA-BIA_DEE	BIA-BIA_TBW	0.993356
46	BIA-BIA_ECW	BIA-BIA_FFM	0.999119
47	BIA-BIA_ECW	BIA-BIA_Fat	-0.974802
48	BIA-BIA_ECW	BIA-BIA_ICW	0.994471
49	BIA-BIA_ECW	BIA-BIA_LDM	0.997366
50	BIA-BIA_ECW	BIA-BIA_LST	0.990683
51	BIA-BIA_ECW	BIA-BIA_SMM	0.988569
52	BIA-BIA_ECW	BIA-BIA_TBW	0.998956
53	BIA-BIA_FFM	BIA-BIA_Fat	-0.976677
54	BIA-BIA_FFM	BIA-BIA_ICW	0.997115
55	BIA-BIA_FFM	BIA-BIA_LDM	0.998667
56	BIA-BIA_FFM	BIA-BIA_LST	0.992142
57	BIA-BIA_FFM	BIA-BIA_SMM	0.992282
58	BIA-BIA_FFM	BIA-BIA_TBW	0.999607
59	BIA-BIA_Fat	BIA-BIA_ICW	-0.966285
60	BIA-BIA_Fat	BIA-BIA_LDM	-0.982505
61	BIA-BIA_Fat	BIA-BIA_LST	-0.947324
62	BIA-BIA_Fat	BIA-BIA_SMM	-0.961353
63	BIA-BIA_Fat	BIA-BIA_TBW	-0.972424
64	BIA-BIA_ICW	BIA-BIA_LDM	0.993010
65	BIA-BIA_ICW	BIA-BIA_LST	0.996280
66	BIA-BIA_ICW	BIA-BIA_SMM	0.996699
67	BIA-BIA_ICW	BIA-BIA_TBW	0.998230
68	BIA-BIA_LDM	BIA-BIA_LST	0.985820
69	BIA-BIA_LDM	BIA-BIA_SMM	0.988088
70	BIA-BIA_LDM	BIA-BIA_TBW	0.996829
70	BIA-BIA_LDM	BIA-BIA_TBW	0.996829
71	BIA-BIA_LST	BIA-BIA_SMM	0.993975
72	BIA-BIA_LST	BIA-BIA_TBW	0.994466
73	BIA-BIA_SMM	BIA-BIA_TBW	0.993451
74	PCIAT-PCIAT_03	PCIAT-PCIAT_Total	0.823336
75	PCIAT-PCIAT_05	PCIAT-PCIAT_Total	0.830993
76	PCIAT-PCIAT_15	PCIAT-PCIAT_Total	0.823996
77	PCIAT-PCIAT_16	PCIAT-PCIAT_18	0.841543
78	PCIAT-PCIAT_17	PCIAT-PCIAT_Total	0.823708
79	PCIAT-PCIAT_18	PCIAT-PCIAT_Total	0.802030
80	PCIAT-PCIAT_Total	sii	0.899681
81	SDS-SDS_Total_Raw	SDS-SDS_Total_T	0.996134

Based on this correlation,

```
[90]
selection = corr[(corr['PCIAT-PCIAT_Total']>.1) | (corr['PCIAT-PCIAT_Total']<-.1)]
selection = [val for val in selection.index]
selection.remove('PCIAT-PCIAT_Total')
selection.remove('sii')
selection.remove('Physical-BMI')
selection.remove('SDS-SDS_Total_Raw')
```

```
selection
```

```
['PCIAT-PCIAT_05',
 'PCIAT-PCIAT_15',
 'PCIAT-PCIAT_17',
 'PCIAT-PCIAT_03',
 'PCIAT-PCIAT_18',
 'PCIAT-PCIAT_02',
 'PCIAT-PCIAT_13',
 'PCIAT-PCIAT_08',
 'PCIAT-PCIAT_16',
 'PCIAT-PCIAT_10',
 'PCIAT-PCIAT_20',
 'PCIAT-PCIAT_19',
 'PCIAT-PCIAT_01',
 'PCIAT-PCIAT_14',
 'PCIAT-PCIAT_09',
 'PCIAT-PCIAT_06',
 'PCIAT-PCIAT_11',
 'PCIAT-PCIAT_04',
 'PCIAT-PCIAT_07',
 'Physical-Height',
 'PCIAT-PCIAT_12',
 'Basic_Demos-Age',
 'PreInt_EduHx-computerinternet_hoursday',
 'Physical-Weight',
 'Physical-Waist_Circumference',
 'FGC-FGC_CU',
 'BIA-BIA_BMI',
 'SDS-SDS_Total_T',
 'FGC-FGC_PU',
 'BIA-BIA_Frame_num',
 'FGC-FGC_GSD',
 'Physical-Systolic_BP',
 'FGC-FGC_GSND',
 'FGC-FGC_TL',
 'BIA-BIA_FFMI',
 'FGC-FGC_SRR_Zone',
 'FGC-FGC_SRL_Zone']
```

I select these features based on the correlation with target more than 0.1 or less than 0.1

Find the features that are have the missing values more than the half of the data samples.

```
half_missing = [val for val in df_train.columns[df_train.isnull().sum()>len(df_train)/2]]
half_missing
```

```
['Physical-Waist_Circumference',
 'Fitness_Endurance-Season',
 'Fitness_Endurance-Max_Stage',
 'Fitness_Endurance-Time_Mins',
 'Fitness_Endurance-Time_Sec',
 'FGC-FGC_GSND',
 'FGC-FGC_GSND_Zone',
 'FGC-FGC_GSD',
 'FGC-FGC_GSD_Zone',
 'PAQ_A-Season',
 'PAQ_A-PAQ_A_Total',
 'PAQ_C-Season',
 'PAQ_C-PAQ_C_Total']
```

Then I checked is this features are selected above in selection matrix

```
selection = [i for i in selection if i not in half_missing]
```

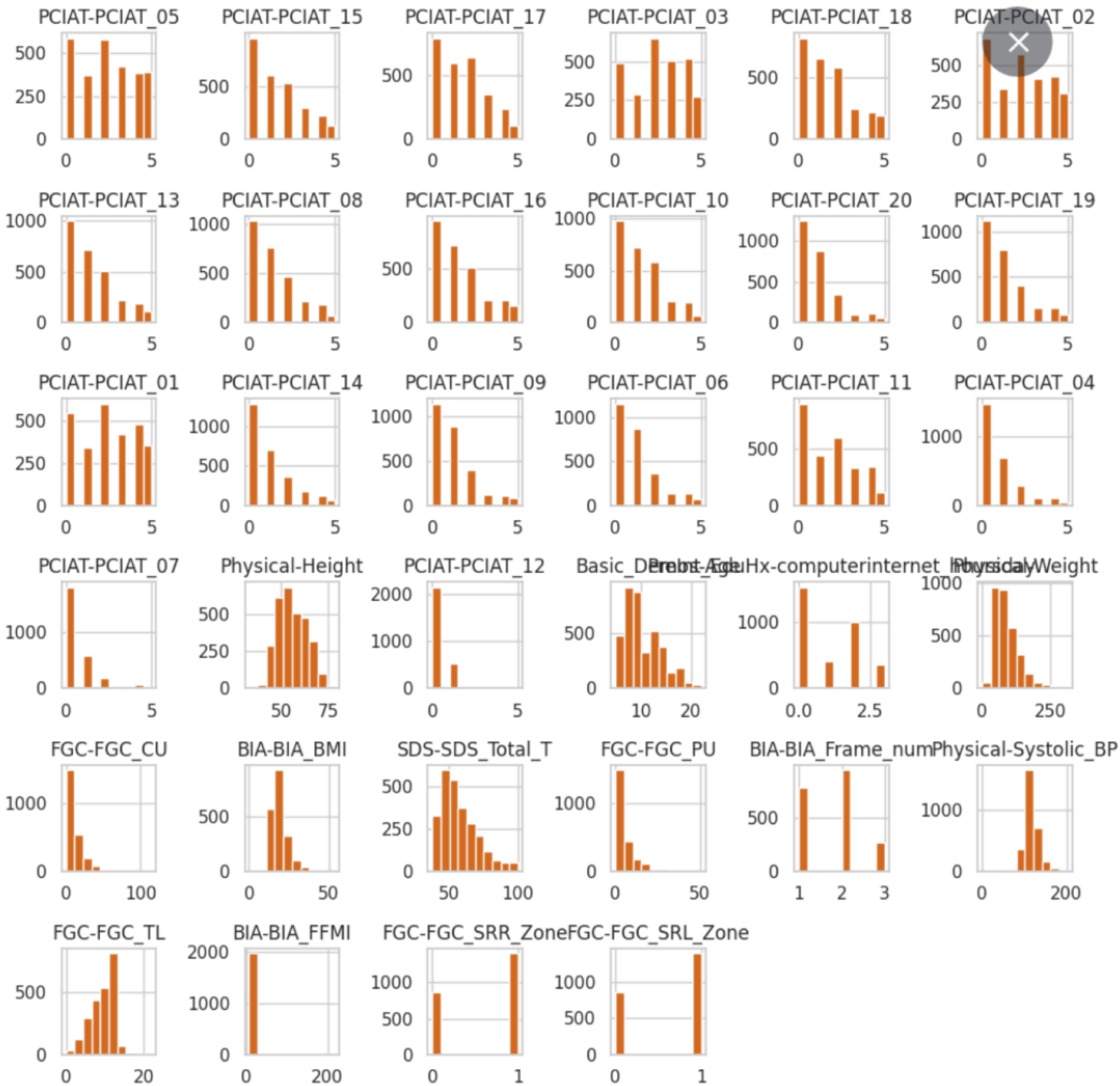
```
selection
```

```
['PCIAT-PCIAT_05',
 'PCIAT-PCIAT_15',
 'PCIAT-PCIAT_17',
 'PCIAT-PCIAT_03',
 'PCIAT-PCIAT_18',
 'PCIAT-PCIAT_02',
 'PCIAT-PCIAT_13',
 'PCIAT-PCIAT_08',
 'PCIAT-PCIAT_16',
 'PCIAT-PCIAT_10',
 'PCIAT-PCIAT_20',
 'PCIAT-PCIAT_19',
 'PCIAT-PCIAT_01',
 'PCIAT-PCIAT_14',
 'PCIAT-PCIAT_09',
 'PCIAT-PCIAT_06',
 'PCIAT-PCIAT_11',
 'PCIAT-PCIAT_04',
 'PCIAT-PCIAT_07',
 'Physical-Height',
 'PCIAT-PCIAT_12',
 'Basic_Demos-Age',
 'PreInt_EduHx-computerinternet_hoursday',
 'Physical-Weight',
 'FGC-FGC_CU',
 'BIA-BIA_BMI',
 'SDS-SDS_Total_T',
 'FGC-FGC_PU',
 'BIA-BIA_Frame_num',
 'Physical-Systolic_BP',
 'FGC-FGC_TL',
 'BIA-BIA_FFMI',
 'FGC-FGC_SRR_Zone',
 'FGC-FGC_SRL_Zone']
```

I now have 16 selected features based on a) correlation with the target and b) relatively few missing values.

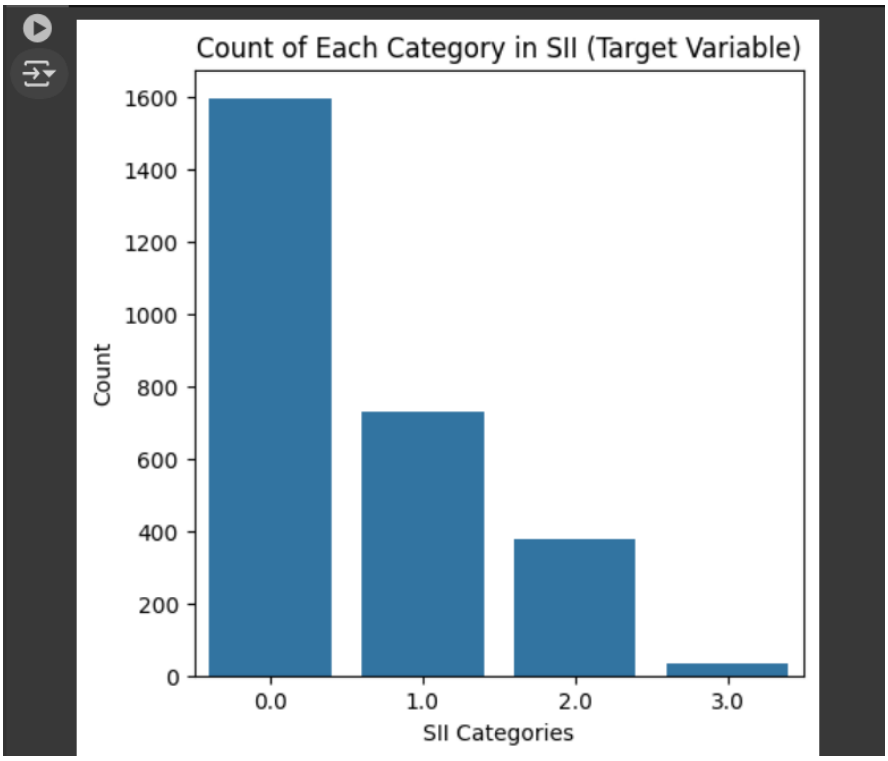
```
df_train[selection].hist(figsize=(10,10), grid = True, color = 'chocolate')
plt.tight_layout()
```

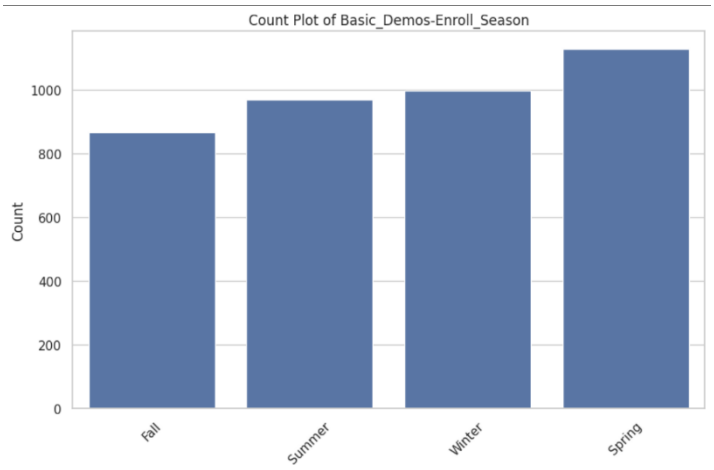
histograms for a selection of numerical features in a DataFrame named df_train

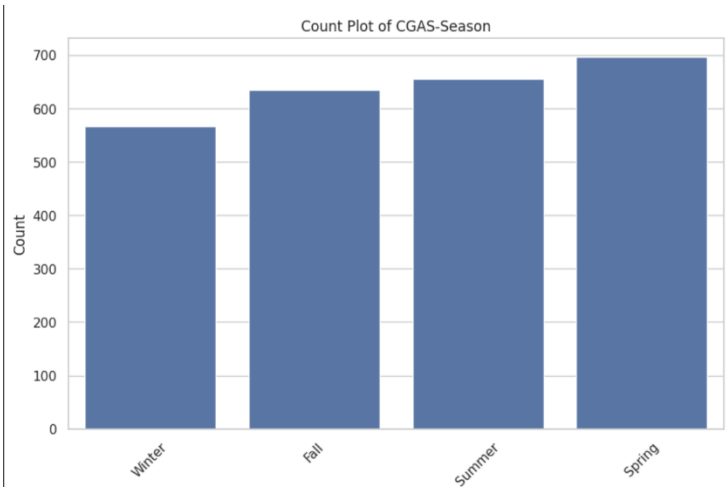


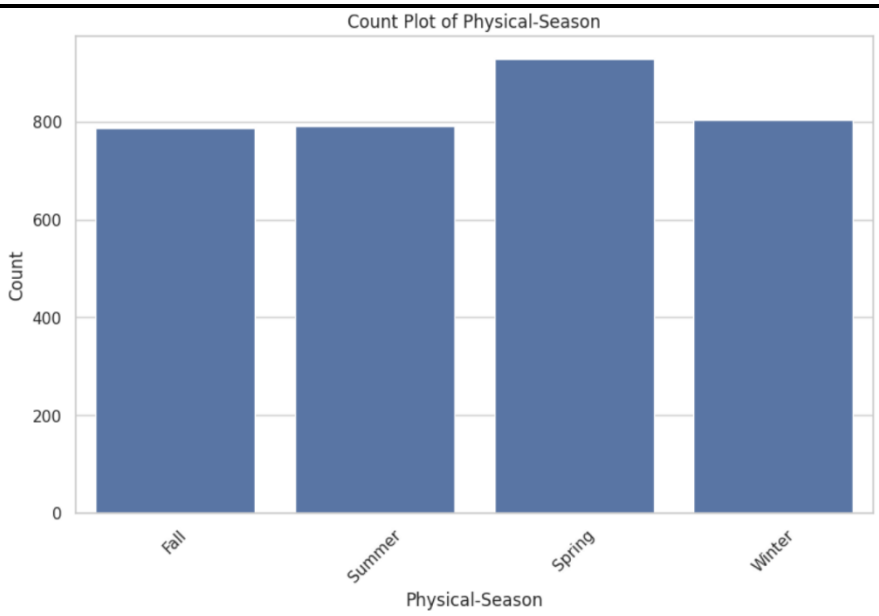
Visualization

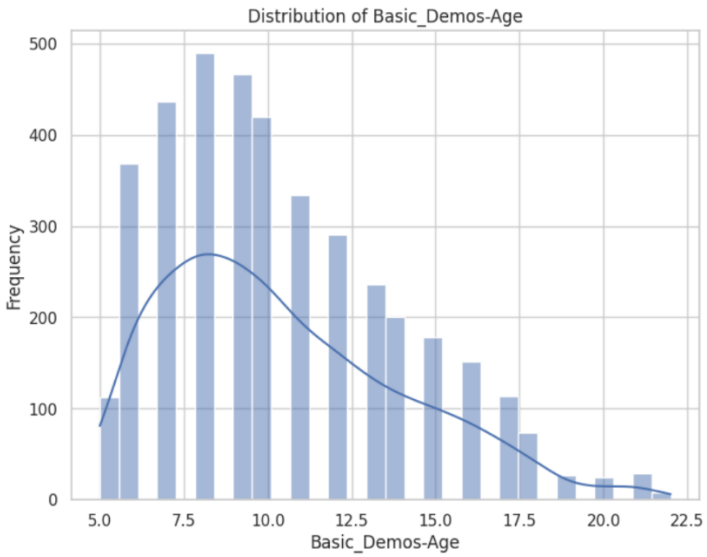
SII – Target

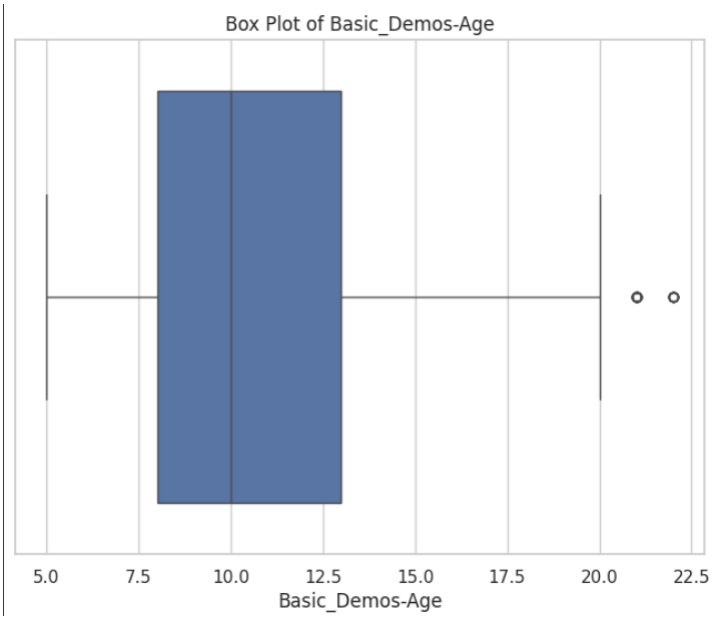


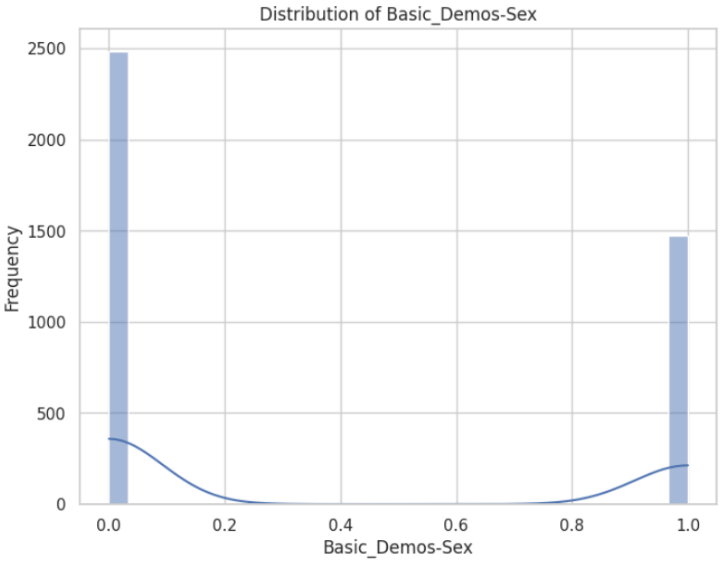


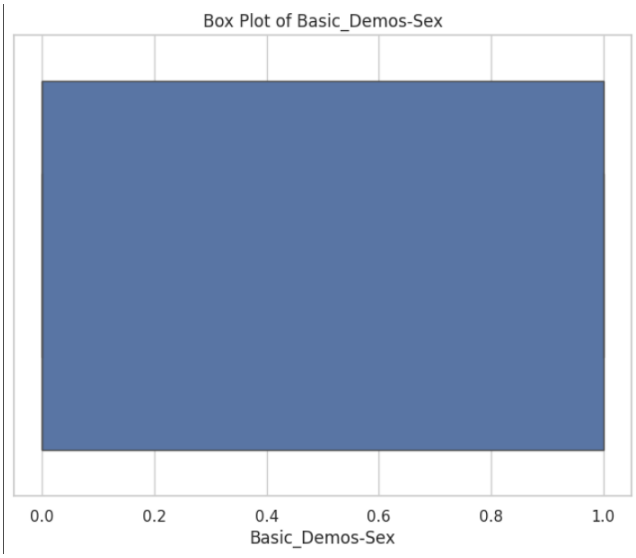


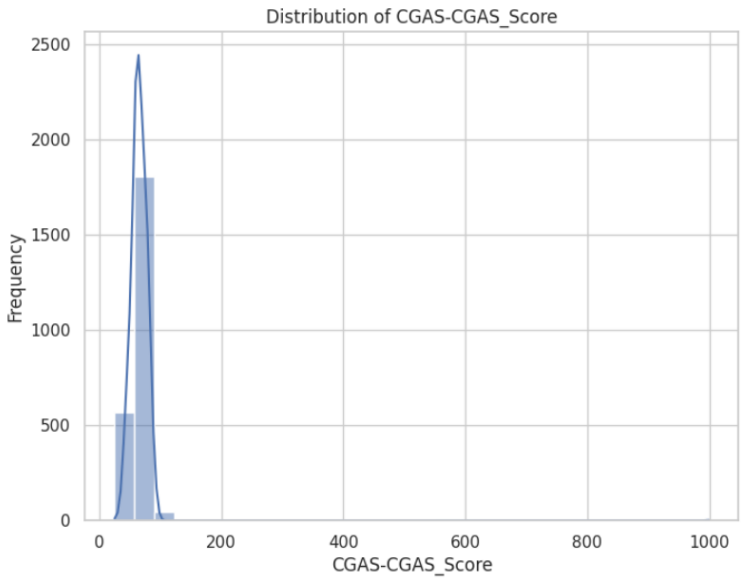

Count Plot of Physical-Season
Count
0
200
400
600
800
Fall
Summer
Spring
Winter
Physical-Season

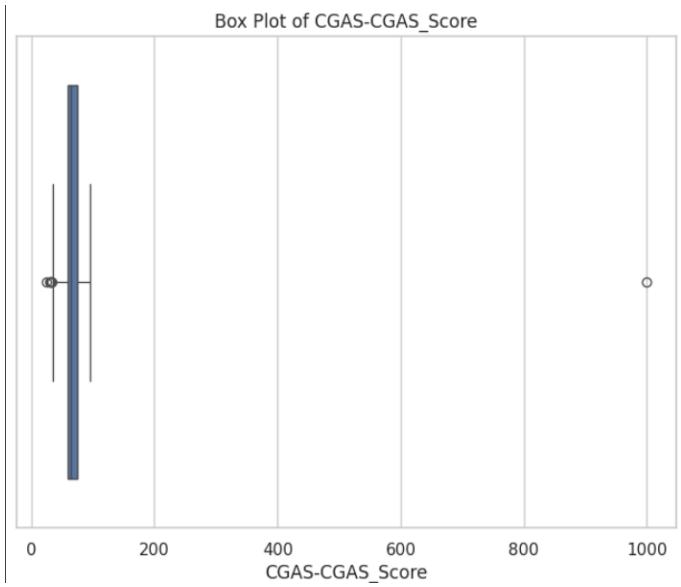

Distribution of Basic_Demos-Age
Frequency
0
100
200
300
400
500
5.0
7.5
10.0
12.5
15.0
17.5
20.0
22.5
Basic_Demos-Age

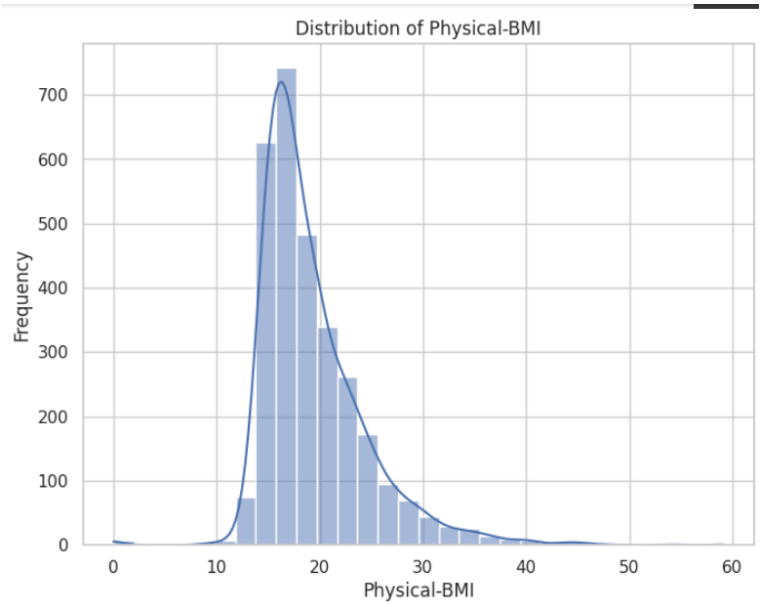

Box Plot of Basic_Demos-Age
5.0
7.5
10.0
12.5
15.0
17.5
20.0
22.5
Basic_Demos-Age

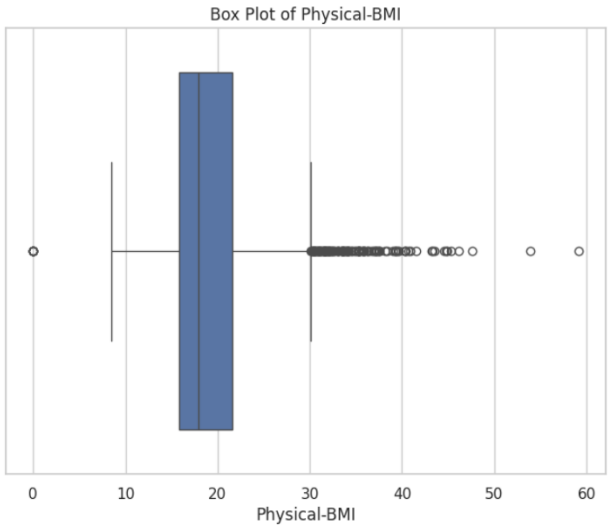

Distribution of Basic_Demos-Sex
Frequency
0
500
1000
1500
2000
2500
0.0
0.2
0.4
0.6
0.8
1.0
Basic_Demos-Sex


Box Plot of Basic_Demos-Sex
0.0
0.2
0.4
0.6
0.8
1.0
Basic_Demos-Sex










```python
# Set the style for seaborn
sns.set(style="whitegrid")

# List of categorical features
categorical_features = [
    "Basic_Demos-Enroll_Season",
    "CGAS-Season",
    "Physical-Season"
]

# List of numerical features
numerical_features = [
    "Basic_Demos-Age",
    "Basic_Demos-Sex",
    "CGAS-CGAS_Score",
    "Physical-BMI"
]
```

```
# Plotting categorical features
for feature in categorical_features:
    plt.figure(figsize=(10, 6))
    sns.countplot(data=df_train, x=feature)
    plt.title(f'Count Plot of {feature}')
    plt.xlabel(feature)
    plt.ylabel('Count')
    plt.xticks(rotation=45)
    plt.show()

# Plotting numerical features
for feature in numerical_features:
    plt.figure(figsize=(8, 6))
    # Histogram
    sns.histplot(df_train[feature], bins=30, kde=True)  # Add kde=True for a kernel
density estimate
    plt.title(f'Distribution of {feature}')
    plt.xlabel(feature)
    plt.ylabel('Frequency')
    plt.show()

    # Box Plot
    plt.figure(figsize=(8, 6))
    sns.boxplot(x=df_train[feature])
    plt.title(f'Box Plot of {feature}')
    plt.xlabel(feature)
    plt.show()
```

Here, I used bar chart and box plot for visualize the categorical and numerical values.

Summary Statistics

```
1 df_train.describe()
```

	Basic_Demos-Age	Basic_Demos-Sex	CGAS-CGAS_Score	Physical-BMI	Physical-Diastolic_BP	Physical-HeartRate	Physical-Systolic_BP	Fitness_Endurance-Max_Stage	Fitness_Endurance-Time_Sec	FGC-FGC_CU	...	PCIAT-PCIAT_11	PCIAT-PCIAT_12	PCIAT-PCIAT_13	PCIAT-PCIAT_14
count	3960.000000	3960.000000	2421.000000	3022.000000	2954.000000	2967.000000	2954.000000	743.000000	740.000000	2322.000000	...	2734.000000	2731.000000	2729.000000	2732.000000
mean	10.433586	0.372727	65.454771	19.331929	69.648951	81.597236	116.983074	4.989233	27.581081	11.259690	...	1.685443	0.244599	1.340051	1.035505
std	3.574648	0.483591	22.341862	5.113934	13.611226	13.665196	17.061225	2.014072	17.707751	11.807781	...	1.543074	0.522956	1.411156	1.301712
min	5.000000	0.000000	25.000000	0.000000	0.000000	27.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000	0.000000
25%	8.000000	0.000000	59.000000	15.869350	61.000000	72.000000	107.000000	4.000000	12.750000	3.000000	...	0.000000	0.000000	0.000000	0.000000
50%	10.000000	0.000000	65.000000	17.937682	68.000000	81.000000	114.000000	5.000000	28.000000	9.000000	...	2.000000	0.000000	1.000000	1.000000
75%	13.000000	1.000000	75.000000	21.571244	76.000000	90.500000	125.000000	6.000000	43.000000	15.750000	...	3.000000	0.000000	2.000000	2.000000
max	22.000000	1.000000	999.000000	59.132048	179.000000	138.000000	203.000000	28.000000	59.000000	115.000000	...	5.000000	5.000000	5.000000	5.000000

8 rows × 46 columns

PCIAT-PCIAT_18	PCIAT-PCIAT_19	PCIAT-PCIAT_20	SDS-SDS_Total_T	PreInt_EduHx-computerinternet_hoursday	sii
2728.000000	2730.000000	2733.000000	2606.000000	3301.000000	2736.000000
1.613636	1.158974	0.943652	57.763622	1.060588	0.580409
1.529178	1.343661	1.185460	13.196091	1.094875	0.771122
0.000000	0.000000	0.000000	38.000000	0.000000	0.000000
0.000000	0.000000	0.000000	47.000000	0.000000	0.000000
1.000000	1.000000	1.000000	55.000000	1.000000	0.000000
2.000000	2.000000	1.000000	64.000000	2.000000	1.000000
5.000000	5.000000	5.000000	100.000000	3.000000	3.000000