Sesethu Bango **6 July 2025**

ZAIO DS-BC Assignment 1: Student Habits and Academic Performance Analysis

---

This file contains the results of programs executed. There is also a brief discussion on what these results mean, and what impact individual student habits have on their academic performance.

The execution flow followed in analysis.ipynb will be used in this document, only the most important/relevant outputs of executed code will be shown.

Phase 1: Data Loading & Preprocessing

Figure 1 shows the output where the program passed some error handling checks.

Figure 1:

```
Checking file format from file path/name
('student_habits_performance.csv', ' Correct format detected')


Checking file readibility and permissions
File read successfully
```

Figure 2 shows the output of the DataFrame Information.

Figure 2:

```
DataFrame information:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 16 columns):
 #   Column                         Non-Null Count  Dtype
---  ------                         --------------  -----
 0   student_id                     1000 non-null   object
 1   age                            1000 non-null   int64
 2   gender                         1000 non-null   object
 3   study_hours_per_day            1000 non-null   float64
 4   social_media_hours             1000 non-null   float64
 5   netflix_hours                  1000 non-null   float64
 6   part_time_job                  1000 non-null   object
 7   attendance_percentage          1000 non-null   float64
 8   sleep_hours                    1000 non-null   float64
 9   diet_quality                   1000 non-null   object
 10  exercise_frequency             1000 non-null   int64
 11  parental_education_level       909 non-null    object
 12  internet_quality               1000 non-null   object
 13  mental_health_rating           1000 non-null   int64
 14  extracurricular_participation  1000 non-null   object
 15  exam_score                     1000 non-null   float64
dtypes: float64(6), int64(3), object(7)
memory usage: 125.1+ KB
```

Figure 3 shows outputs of the DataCleaner class.

Figure 3:

```
Check for empty DataFrame:
Loaded DataFrame, hence CSV file is not empty

Check for missing values:
Number of missing values 91
Total number of values in the table 16000
Proportion of missing data to the total: 0.57% of data is missing.

Check DataFrame after replacing Null/None values:
Number of missing values 0
Total number of values in the table 16000
Proportion of missing data to the total: 0.0% of data is missing.
```

Figure 4 shows another output of the DataCleaner class.

Figure 4:

```
Column 'exam_score' contains negative values: False
Result is False, meaning there are no negetaive values. NOTE I checked all of t
he other columns too, there are no negative values
```

Phase 2: Statistical Analysis

Figures 5 and 6 show the average and median study hours per day, grouped by mental health rating. In each of these tables the average and median values don't vary in any significant amount between the different mental health ratings. One can then infer that a student's mental health has very little, if any bearing on their study hours per day.

Figure 5:

Average study hours per day, grouped by mental health rating

mental_health_rating	study_hours_per_day
1	3.62
2	3.61
3	3.58
4	3.42
5	3.36
6	3.60
7	3.68
8	3.55
9	3.60
10	3.49

Figure 6:

Median study hours per day, grouped by mental health rating

mental_health_rating	study_hours_per_day
1	3.80
2	3.65
3	3.30
4	3.40
5	3.50
6	3.60
7	3.50
8	3.60
9	3.50
10	3.60

Figure 7 shows the correlation between sleep and exam scores. There is a clear increase in mean exam scores as the sleep intervals increase. Sleep intervals 3.0 – 4.5 has a lowest mean exam score, whereas 7.5 – 9.0 and >9.0 have the highest mean exam scores. Clearly a good night's sleep does on average result in higher exam scores.

Figure 7:

```
Identify correlation between sleep and exam scores by finding exam_score groupe
d by sleep_hours.

Correlation between sleep_time and exam_scores
  sleep_intervals, hrs  mean_exam_score
0            3.0 - 4.5        61.262791
1            4.5 - 6.0        68.456333
2            6.0 - 7.5        69.955140
3            7.5 - 9.0        71.903448
4                 >9.0        72.811538
```

Figure 8 shows the detected outliers in social media usage. The outliers are values outside the range of mean +/- three standard deviations. Since there are no values below zero, there won't be any outliers less than the lower boundary value. Three outliers were found; their indices are also part of the shown result.

Figure 8:

```
Detect outliers in social media usage

Standard dev. of social media hours 1.17
Mean value: 2.5055
Upper and lower boundaries 6.02 -1.0

Outlier values: [6.2, 6.1, 7.2]
Outlier index values: [145, 361, 735]
```

Phase 3: Visualisation

Figure 9 shows a histogram of study hours per day. The distribution is approximately normal, hence the mean, median and mode will all be roughly equal. Their value will be at about the 3.5 hours mark, the approximate midpoint of the distribution. Values over 8 hours are likely to be outliers.

Figure 9:

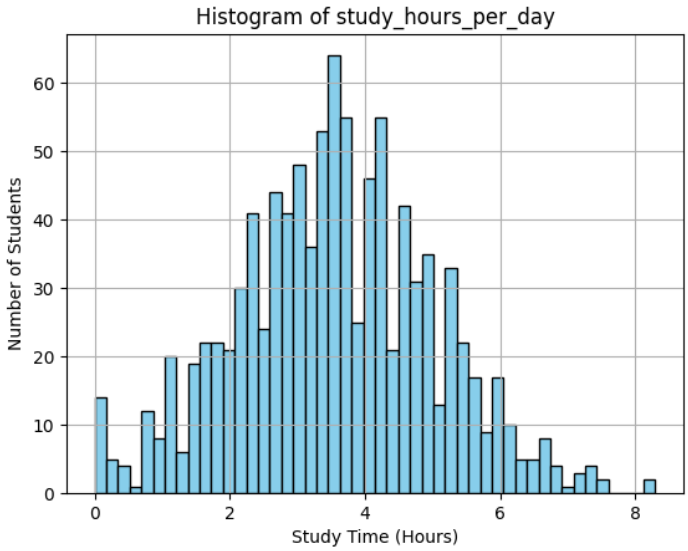


Figure 10 show a scatter plot sleep hours vs exam score. The plot doesn't indicate much in terms of a definitive trend. It is clustered around the general centre of the graph. One could say there's a subtle trend where more sleep hours correlate to higher exam scores.
The vertical and horizontal lines are the mean values for each column. On simple observation it looks like Q4 is the least populated quadrant, whereas the other 3 quadrants seem to be roughly evenly populated. We know from Phase 2 that the average exam scores steadily increased with longer durations of sleep. This scatter plot only slightly emphasises that point.

Figure 10:

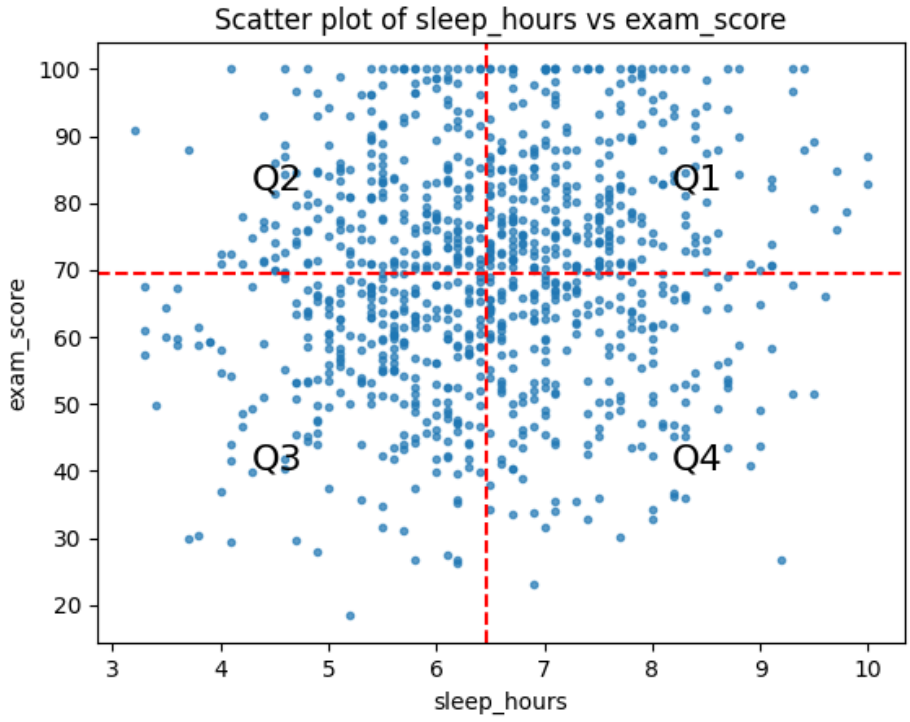
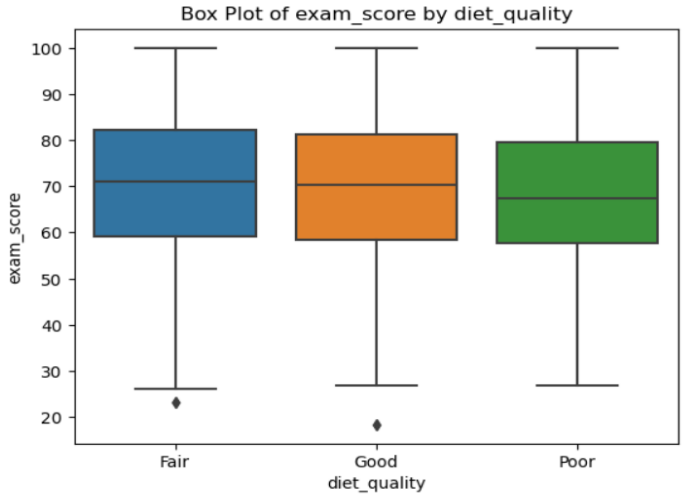


Figure 11 shows box plots of exam score by diet quality. The Fair diet seems to generally result in infinitesimally better exam scores, especially compared to the Good diet. The Fair and Good diets have roughly the same median value. The Fair diet having slightly better Q1 and Q3 scores than the Good diet. The Poor diet's Q1, Q2 and Q3 scores are all lagging the other two diet qualities. This lag isn't large though; there looks to be less than a 5-point difference in any quartile value relative to its counterpart in a different diet quality. The maximum and minimum score for all diet qualities look to be essentially the same. There are some outliers below the minimum values In the Fair and Good diet qualities.

Figure 11:



Phase 4: Predictive Modelling

Figure 12 shows the output of the model; Predicted Score. Each predicted value corresponds to each specific student's characteristics. The actual exam scores are compared to the Predicted Scores. The difference between the two, as a percentage, is what is shown in the Percent Difference column. Also shown is the Average percent difference over the whole Percent Difference column.

Figure 12:

```
Applying the model using columns that contain only number values:
Compare actual to predicted scores, first 10 entries:
   Predicted Score  exam_score  Percent_Difference
0            53.11        56.2               -5.50
1           107.26       100.0                7.26
2            41.63        34.3               21.37
3            39.68        26.8               48.06
4            70.67        66.4                6.43
5           107.72       100.0                7.72
6            88.80        89.8               -1.11
7            76.20        72.6                4.96
8            73.63        78.9               -6.68
9            96.66       100.0               -3.34
Average percent difference: 6.59416
```

Figure 13 shows the same kind of information as in Figure 12, only now the scope of the X features of the model have been expanded.
The result however is essentially identical to that of Figure 12. Hence the addition of more X features didn't result in more accurate predictions.

Figure 13:

```
Applying the string_to_Int function to add more X features to the model, in an
attempts to make the model more accurate.
Again we compare actual to predicted scores:
   Predicted Score  exam_score  Percent_Difference
0            53.12        56.2               -5.48
1           107.10       100.0                7.10
2            42.04        34.3               22.57
3            39.57        26.8               47.65
4            70.36        66.4                5.96
5           108.09       100.0                8.09
6            88.60        89.8               -1.34
7            76.56        72.6                5.45
8            73.03        78.9               -7.44
9            96.00       100.0               -4.00
Average percent difference: 6.59525
```

Some of the predicted scores are significantly off from the actual exam scores. That make this model relatively inaccurate, specifically for the lower scores it seems. Just based off the first 10 results, exam scores lower than 35 seem to have a percent difference that is significantly larger than the rest. One would have to check the other lower scores to verify if this trend continues.
Perhaps there is a better model other than linear regression that can be applied.

Figures 14 and 15 show the input and result of the model, here it is being used to predict the exam score of a theoretical student with x-features defined in the 'x_new' DataFrame.

Figure 14:

```
x_new = pd.DataFrame([{
            'age': 25, 'gender': 2, 'study_hours_per_day': 6.0, 'social_media_hours': 3.0,
            'netflix_hours': 1.0, 'part_time_job': 1, 'attendance_percentage': 80.0,
            'sleep_hours': 7.0, 'diet_quality': 1, 'exercise_frequency': 1,
            'parental_education_level': 3, 'internet_quality': 3, 'mental_health_rating': 1,
            'extracurricular_participation': 1
```

Figure 15:

```
Applying the model to predict the exam score of a student with defined characte
ristics
New prediction:
82.1
```

Lastly figure 16 shows that the model has been saved using Pickle.

Figure 16:

```
Saving model using Pickle:
Model saved to trained_model.pkl
```