Week 4

Team Two

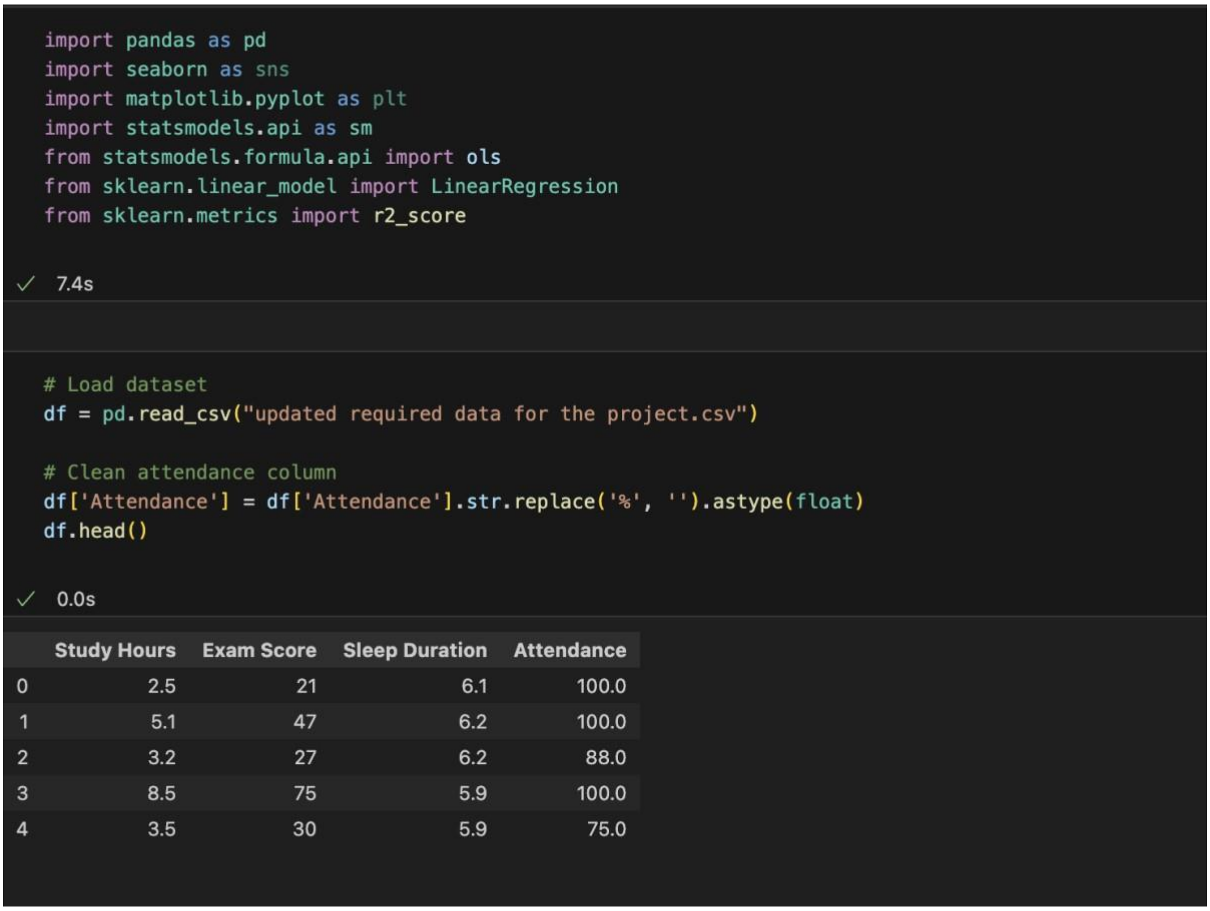
Westcliff University

Data200: Applied Statistical Analytics

Professor Alok Khatri

June 3, 2025

Week 4



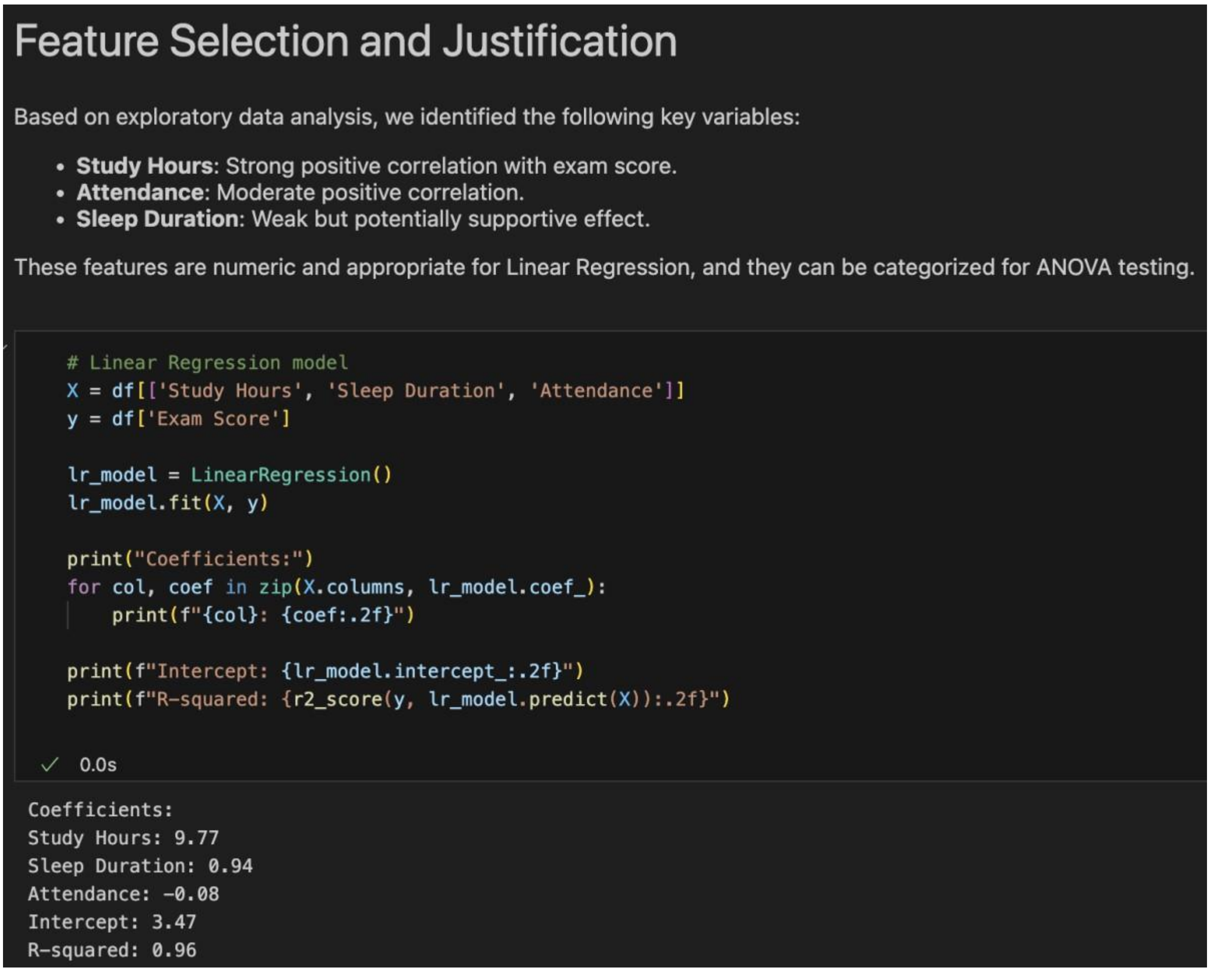
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import statsmodels.api as sm
from statsmodels.formula.api import ols
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
```

```
# Load dataset
df = pd.read_csv("updated required data for the project.csv")

# Clean attendance column
df['Attendance'] = df['Attendance'].str.replace('%', '').astype(float)
df.head()
```

```
   Study Hours  Exam Score  Sleep Duration  Attendance
0          2.5          21             6.1       100.0
1          5.1          47             6.2       100.0
2          3.2          27             6.2        88.0
3          8.5          75             5.9       100.0
4          3.5          30             5.9        75.0
```

In Data Loading and Cleaning, data is imported into our Python environment from a CSV file located on my personal computer, Amazon S3 or the remote FTP directory whose URL we have just extracted.

This imports all the essential Python libraries; pandas and seaborn for performing data analysis and data visualization. pandas is used to load the dataset which is named “updated required data for the project.csv”. Cleaning has to be done on the values of the Attendance column which was originally stored as a column with a "%" sign and now needs to be removed and the values converted into numeric format (float) is one important step. With this we know the data is ready for analysis. The last columns are the cleaned Attendance values and the dataset first shows the first five rows with columns Study hours, Exam Score, Sleep duration and the Attendance values. This is important as to perform statistical modeling and machine learning, you need clean properly formatted data.



Feature Selection and Justification

Based on exploratory data analysis, we identified the following key variables:

- **Study Hours**: Strong positive correlation with exam score.
- **Attendance**: Moderate positive correlation.
- **Sleep Duration**: Weak but potentially supportive effect.

These features are numeric and appropriate for Linear Regression, and they can be categorized for ANOVA testing.

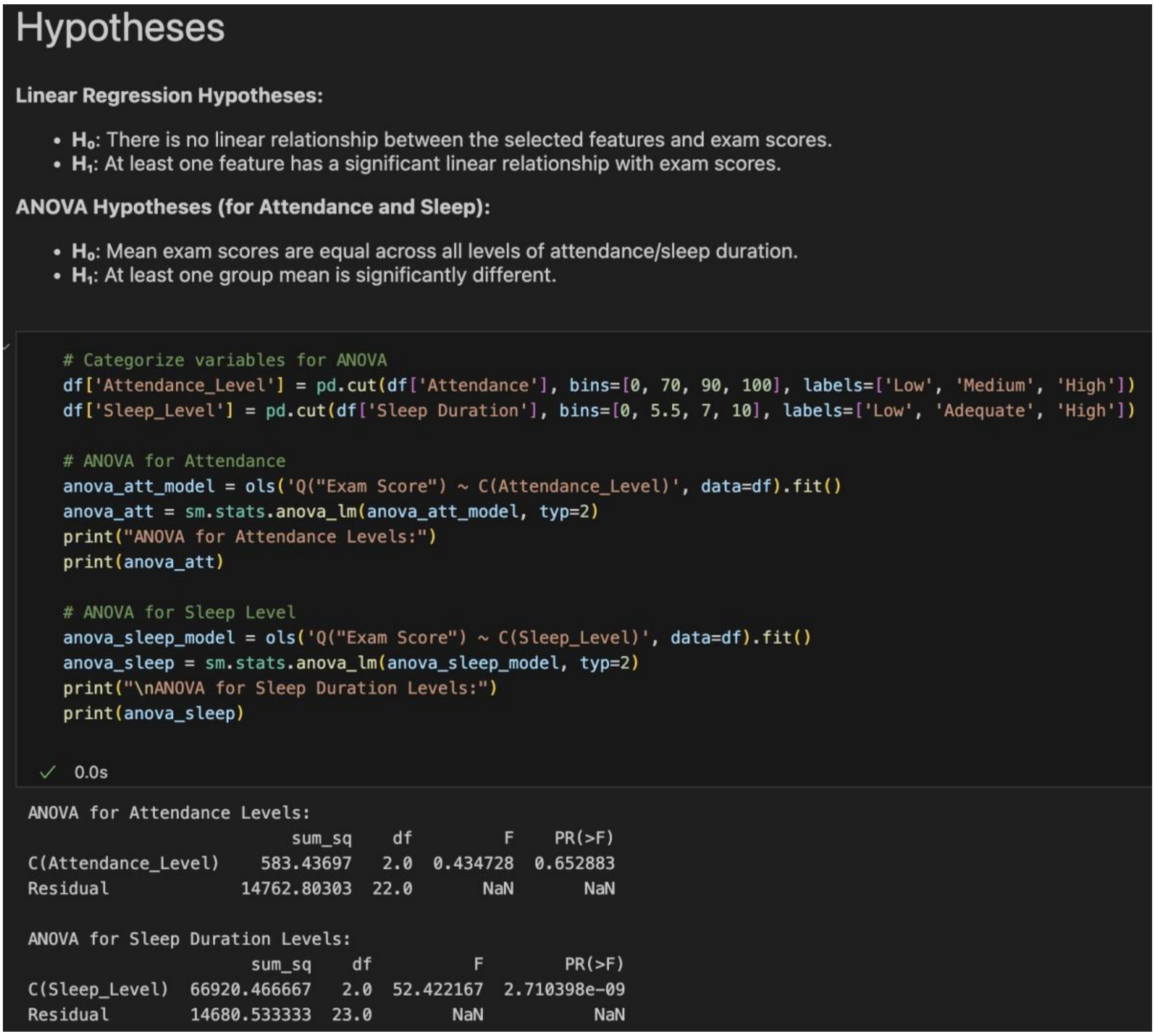
```
# Linear Regression model
X = df[['Study Hours', 'Sleep Duration', 'Attendance']]
y = df['Exam Score']

lr_model = LinearRegression()
lr_model.fit(X, y)

print("Coefficients:")
for col, coef in zip(X.columns, lr_model.coef_):
    print(f"{col}: {coef:.2f}")

print(f"Intercept: {lr_model.intercept_:.2f}")
print(f"R-squared: {r2_score(y, lr_model.predict(X)):.2f}")
```

✓ 0.0s

```
Coefficients:
Study Hours: 9.77
Sleep Duration: 0.94
Attendance: -0.08
Intercept: 3.47
R-squared: 0.96
```

Feature Selection and Regression Analysis

It tells us why were some features selected to predict exam scores. The strongest positive effect comes from Study Hours, followed by Sleep Duration and Attendance has a small negative effect. These three features are trained into a linear regression model. We can see from the coefficients of the model that Study Hours has the biggest impact on scores (9.77 per hour) and that Sleep Duration also plays a role (0.94 per hour). Attendance negatively little affects scores by (-0.08) which is surprising though perhaps caused by other hidden factors. With an R-squared value of 0.96 the model does extremely well explaining 96% of the variation in exam scores, a near perfect fit.



Hypotheses

Linear Regression Hypotheses:

- **H_0**: There is no linear relationship between the selected features and exam scores.
- **H_1**: At least one feature has a significant linear relationship with exam scores.

ANOVA Hypotheses (for Attendance and Sleep):

- **H_0**: Mean exam scores are equal across all levels of attendance/sleep duration.
- **H_1**: At least one group mean is significantly different.

```
# Categorize variables for ANOVA
df['Attendance_Level'] = pd.cut(df['Attendance'], bins=[0, 70, 90, 100], labels=['Low', 'Medium', 'High'])
df['Sleep_Level'] = pd.cut(df['Sleep Duration'], bins=[0, 5.5, 7, 10], labels=['Low', 'Adequate', 'High'])

# ANOVA for Attendance
anova_att_model = ols('Q("Exam Score") ~ C(Attendance_Level)', data=df).fit()
anova_att = sm.stats.anova_lm(anova_att_model, typ=2)
print("ANOVA for Attendance Levels:")
print(anova_att)

# ANOVA for Sleep Level
anova_sleep_model = ols('Q("Exam Score") ~ C(Sleep_Level)', data=df).fit()
anova_sleep = sm.stats.anova_lm(anova_sleep_model, typ=2)
print("\nANOVA for Sleep Duration Levels:")
print(anova_sleep)
```

✓ 0.0s

```
ANOVA for Attendance Levels:
                          sum_sq    df         F    PR(>F)
C(Attendance_Level)    583.43697   2.0  0.434728  0.652883
Residual             14762.80303  22.0       NaN       NaN

ANOVA for Sleep Duration Levels:
                      sum_sq    df          F        PR(>F)
C(Sleep_Level)  66920.466667   2.0  52.422167  2.710398e-09
Residual        14680.533333  23.0        NaN           NaN
```

Hypothesis Testing with ANOVA (ANOVA) Hypothesis Testing : Straightforwardly, hypothesis testing consists of two steps: Null Hypothesis : This is used for reference against the alternate hypothesis and Accepting this hypothesis in ANOVA will indicate that there is no significant difference between means (H0). Unlike regression hypothesis testing, ANOVA hypothesis testing utilizes an F statistic as opposed to a t statistic.

The data is further grouped into Low, Medium and High levels of attendance and sleep levels in order to see if those variables also affect exam scores. ANOVA tests if these groups are significantly different in their scores.

The p-value here (0.65) is high, so attendance doesn't really affect scores in this dataset.

At level of sleep vs. Exam score, the p-value (2.7e-09) is very low implying that sleep duration greatly affects performance in exams.