Modeling

December 14, 2023

1 Data Loading and Preparation

[2]:
```
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv("data/all.csv", parse_dates=True)
```

Now, we need to preprocess the data before model fitting. (Note: this following code block is hidden in the PDF version of this notebook)

2 Predicting Night Sleep Duration (Regression)

2.1 Linear Regression

[4]:
```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# Creating the linear regression model
model = LinearRegression()

# Fitting the model on the training data
model.fit(X_train, y_train)

# Making predictions on the testing data
y_pred = model.predict(X_test)

# Evaluating the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

# Printing the model coefficients and performance metrics
print("Model Coefficients:", model.coef_)
print("Intercept:", model.intercept_)
print("Mean Squared Error:", mse)
print("R^2 Score:", r2)
```

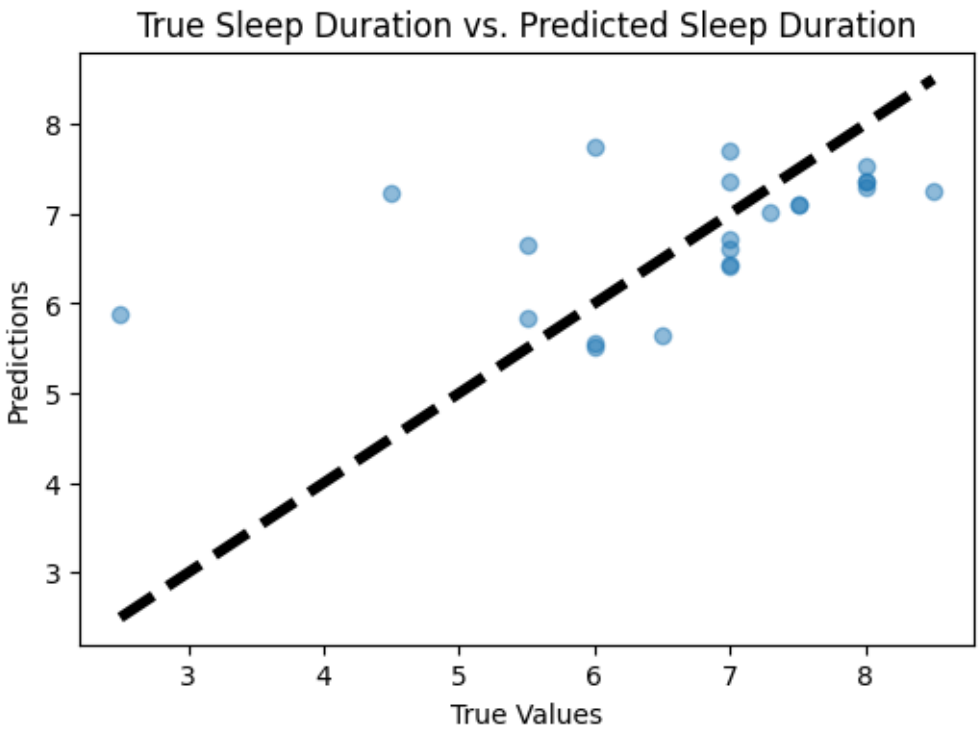
```
Model Coefficients: [-0.01698294  0.15863872 -0.0161974  -0.05333599]
Intercept: 8.824117620374018
```

```
Mean Squared Error: 1.3388887030966368
R^2 Score: 0.24568772503605874
```

- **Model Coefficients:** These values represent the estimated effect of each independent variable on the dependent variable (sleep duration), holding all other variables constant. The coefficients [-0.0169824, 0.15863872, -0.0161974, -0.05333599] correspond to the variables in the order they were input into the model. If we assume the order was ['Nap Duration Ordinal', 'Exercise Days/Week Ordinal', 'Sleep Disturbances Ordinal', 'Age Group * Ordinal'], then: For every unit increase in 'Nap Duration Ordinal', sleep duration decreases by 0.0169824 units, holding other variables constant. For every unit increase in 'Exercise Days/Week Ordinal', sleep duration increases by 0.15863872 units, holding other variables constant. For every unit increase in 'Sleep Disturbances Ordinal', sleep duration decreases by 0.0161974 units, holding other variables constant. For every unit increase in 'Age Group Ordinal', sleep duration decreases by 0.05333599 units, holding other variables constant.
- **Intercept:** The value 8.824117620374018 is the expected mean value of the dependent variable (sleep duration) when all predictors are held at zero. In the context of this model, it represents the estimated sleep duration for an individual with the baseline levels of the ordinal predictors.
- **Mean Squared Error (MSE):** The MSE value of 1.3388887030966368 is the average of the squares of the errors, i.e., the average squared difference between the estimated values and the actual value of the dependent variable. A lower MSE indicates a better fit of the model to the data.
- **R^2 Score:** The R^2 value of 0.24568772530605874 indicates that approximately 24.57% of the variability in sleep duration can be explained by the model. An R^2 score of 1 indicates a perfect fit, so a score of 0.2457 suggests that while there is some relationship, a substantial amount of the variability in sleep duration remains unexplained by the model.

In summary, the model indicates that exercise frequency has a positive association with sleep duration, while nap duration, sleep disturbances, and age group have negative associations. However, the model does not explain a large portion of the variability in sleep duration, suggesting that other factors not included in the model may also be important.

[5]:
```
plt.figure(figsize=(6, 4))
plt.scatter(y_test, y_pred, alpha=0.5)
plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], 'k--',␣
 ↪lw=4)
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.title('True Sleep Duration vs. Predicted Sleep Duration')
plt.show()
```

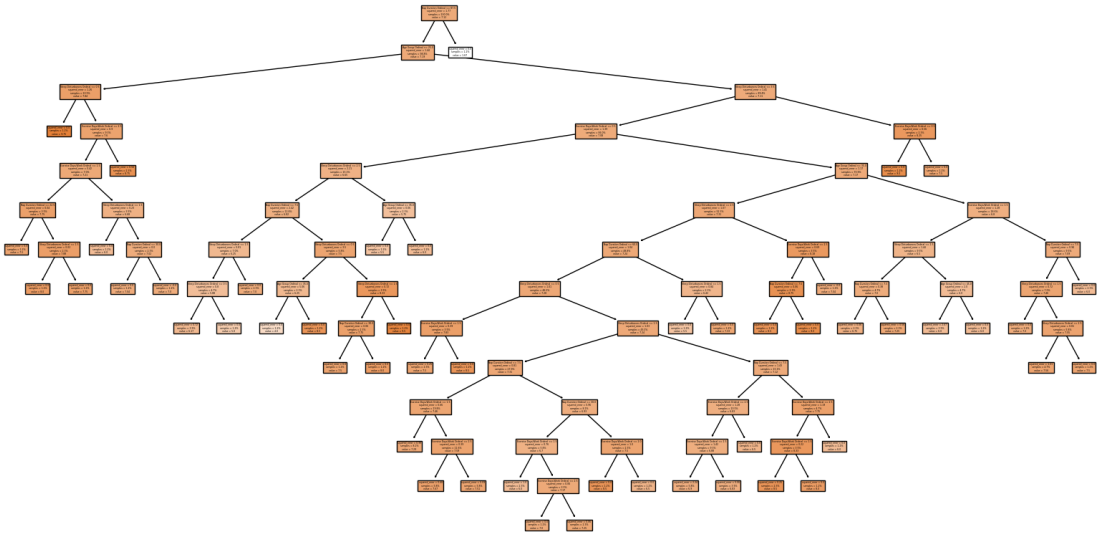


The scatter plot shows a moderate fit for the regression model, with a trend where predicted sleep duration generally aligns with true values but with noticeable prediction errors, indicating the model has room for improvement.

2.2 Decision Tree Regressor

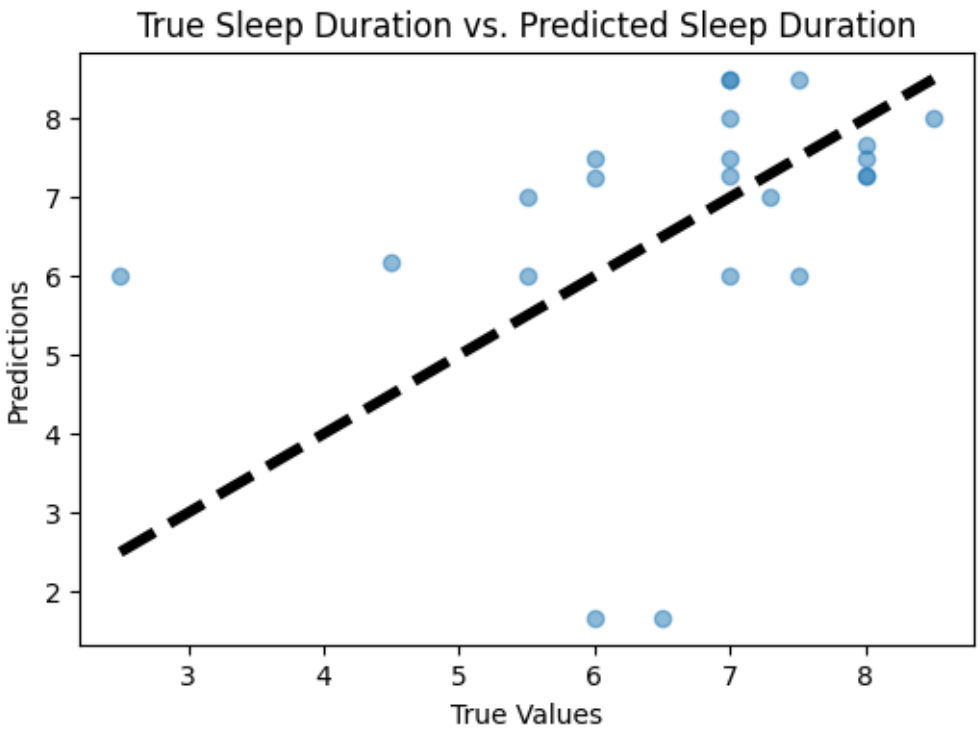
[6]:
```
from sklearn.tree import DecisionTreeRegressor
from sklearn.tree import plot_tree


# Create the Decision Tree model
decision_tree_model = DecisionTreeRegressor(random_state=42)

# Fit the model on the training data
decision_tree_model.fit(X_train, y_train)

# Make predictions on the testing data
y_pred_tree = decision_tree_model.predict(X_test)

# Evaluate the model
mse_tree = mean_squared_error(y_test, y_pred_tree)
r2_tree = r2_score(y_test, y_pred_tree)
```

```
# Visualize the decision tree
plt.figure(figsize=(20, 10))
plot_tree(decision_tree_model, filled=True, feature_names=X.columns.tolist(),␣
  ↪proportion=True, precision=2)
plt.show()

# Output the performance metrics
print("Decision Tree Mean Squared Error:", mse_tree)
print("Decision Tree R^2 Score:", r2_tree)
```



```
Decision Tree Mean Squared Error: 3.4191670270047414
Decision Tree R^2 Score: -0.9263137052815129
```

[7]:
```
plt.figure(figsize=(6, 4))
plt.scatter(y_test, y_pred_tree, alpha=0.5)
plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], 'k--',␣
  ↪lw=4)
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.title('True Sleep Duration vs. Predicted Sleep Duration')
plt.show()
```



The decision tree model has a significantly higher MSE than the linear regression model. This indicates that on average, the decision tree model's predictions are further from the true sleep duration values than the linear regression model's predictions. The R^2 score for the decision tree model is negative, which suggests that it performs worse than a simple horizontal line at the mean of the sleep duration. In contrast, the linear regression model had a positive R^2 value, indicating that it was able to explain about 24.6% of the variance in sleep duration.

Conclusion:

The linear regression model appears to be a better fit for the data compared to the decision tree model. It has a lower average error in predictions and a positive R^2 value, which means it can account for a certain proportion of the variance in sleep duration. In contrast, the decision tree model's negative R^2 score indicates that it may be overfitting the training data or not capturing the complexity of the relationship adequately.

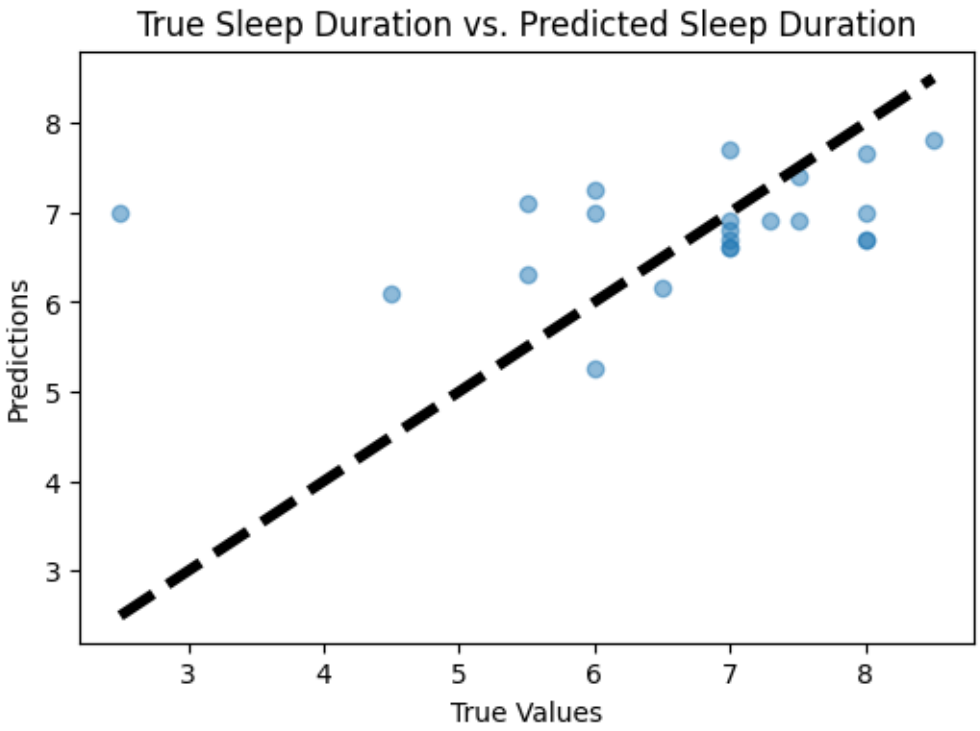
2.3 K-Nearest Neighbor Classification

[8]:
```
from sklearn.neighbors import KNeighborsRegressor

# Create the KNN regressor model, using n_neighbors=5 as a starting point
knn_model = KNeighborsRegressor(n_neighbors=5)

# Fit the model on the training data
knn_model.fit(X_train, y_train)
```

```
# Make predictions on the testing data
y_pred_knn = knn_model.predict(X_test)

# Evaluate the model
mse_knn = mean_squared_error(y_test, y_pred_knn)
r2_knn = r2_score(y_test, y_pred_knn)

# Output the performance metrics
print("KNN Regressor Mean Squared Error:", mse_knn)
print("KNN Regressor R^2 Score:", r2_knn)
```

```
KNN Regressor Mean Squared Error: 1.6240662402638626
KNN Regressor R^2 Score: 0.08502245365711447
```

[9]:
```
plt.figure(figsize=(6, 4))
plt.scatter(y_test, y_pred_knn, alpha=0.5)
plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], 'k--',␣
↪lw=4)
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.title('True Sleep Duration vs. Predicted Sleep Duration')
plt.show()
```



- **Mean Squared Error (MSE):** The MSE is approximately 1.624. This value indicates the average squared difference between the actual and predicted sleep durations. Compared to the decision tree model, which had a much higher MSE, the KNN regressor has reduced the average error in predictions, but it is still higher than what we observed with the linear regression model.
- **R^2 Score:** The R^2 score is approximately 0.085. This value indicates that the KNN model explains about 8.5% of the variance in sleep duration. It is a positive score, which is an improvement over the decision tree model (which had a negative R^2 score), but it is significantly lower than the R^2 score from the linear regression model, which was approximately 0.246.
- **Interpretation:** The KNN regressor's performance is not as strong as the linear regression model but is better than the decision tree model based on these metrics. The relatively low R^2 score from the KNN regressor suggests that while the model captures some of the variability in sleep duration, a large portion of the variance remains unexplained by the model.

Conclusion:

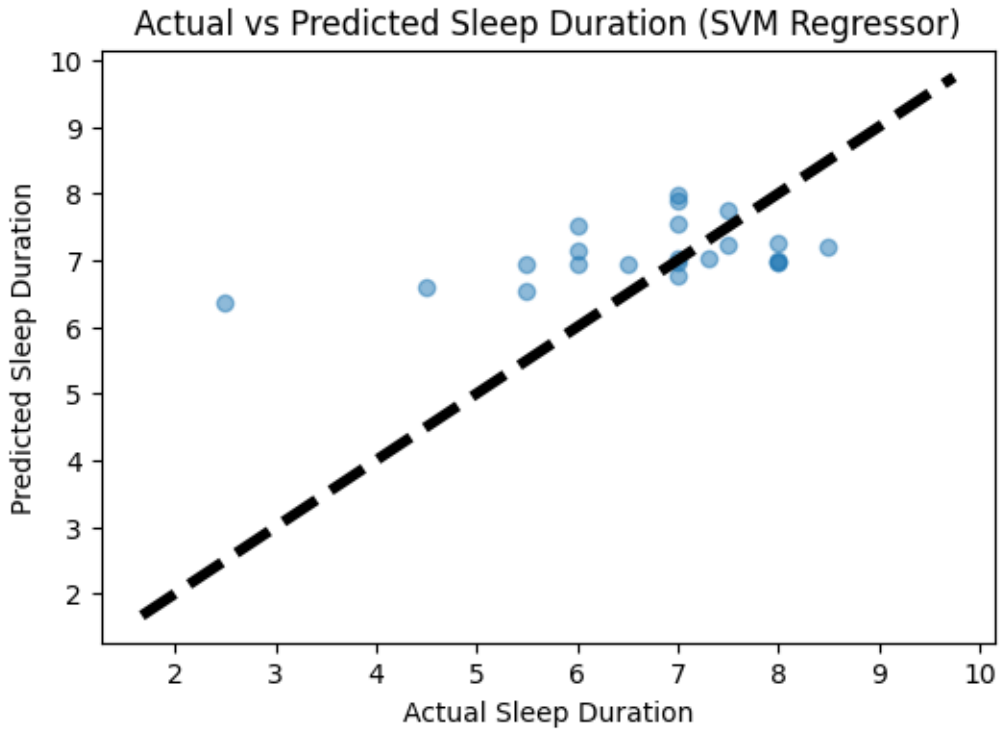
In summary, while the KNN regressor has shown some predictive capabilities, its performance is not yet optimal compared to the linear regression model for this particular dataset.

2.4 Support Vector Machine

[10]:
```
from sklearn.svm import SVR
from sklearn.preprocessing import StandardScaler

# Standardize the features (important for SVM)
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# Create the SVM regressor model
svm_model = SVR()

# Fit the model on the training data
svm_model.fit(X_train_scaled, y_train)

# Make predictions on the testing data
y_pred_svm = svm_model.predict(X_test_scaled)

# Evaluate the model
mse_svm = mean_squared_error(y_test, y_pred_svm)
r2_svm = r2_score(y_test, y_pred_svm)

# Output the performance metrics
print("SVM Regressor Mean Squared Error:", mse_svm)
print("SVM Regressor R^2 Score:", r2_svm)
```

```
SVM Regressor Mean Squared Error: 1.5909012929685098
SVM Regressor R^2 Score: 0.10370714849810991
```

[11]:
```
# Visualization
plt.figure(figsize=(6, 4))
plt.scatter(y_test, y_pred_svm, alpha=0.5)
plt.plot([y.min(), y.max()], [y.min(), y.max()], 'k--', lw=4)
plt.xlabel('Actual Sleep Duration')
plt.ylabel('Predicted Sleep Duration')
plt.title('Actual vs Predicted Sleep Duration (SVM Regressor)')
plt.show()
```



- **Mean Squared Error (MSE):** The MSE is approximately 1.591, indicating the average squared difference between the observed actual and the model's predicted sleep durations. This value reflects the model's prediction error; the closer the MSE is to 0, the better the model's predictive accuracy.
- **R^2 Score:** The R^2 score is approximately 0.104, which means that the SVM model explains about 10.37% of the variance in sleep duration. The R^2 score is a measure of how well the observed outcomes are replicated by the model, based on the proportion of total variation of outcomes explained by the model.
- **Interpretation:** The SVM model's MSE is slightly higher than the linear regression model's MSE but lower than the decision tree's MSE, placing its predictive accuracy between the two. The positive R^2 score indicates that the SVM model has captured some of the variance in the target variable, albeit a small portion.

Conclusion:

While the SVM model shows some predictive ability, its performance is modest, as it explains just over 10% of the variance in sleep duration. This suggests that the model, with its current configuration, captures only a fraction of the factors affecting sleep duration. There may be room for improvement through hyperparameter tuning, feature engineering, or trying different kernel functions within the SVM framework to potentially enhance model performance.

3 Predicting Sleep Quality (Classification)

[ ]:

[ ]:

4 Conclusion