Real Data Analysis: Comparing Parametric and Nonparametric Methods for Evaluating Sleep Data

Varshitha Gudimalla

December 9, 2024

1 Introduction

This project analyzes the Sleep Data, comparing two sleep conditions to evaluate their impact on the increase in sleep hours. The objective is to investigate whether the sleep condition significantly affects the change in sleep duration using both parametric (paired t-test) and non-parametric (Wilcoxon signed-rank test) methods.

2 Data Description

2.1 Individuals in the Sample

The sample consists of 10 individuals, each measured under two sleep conditions:

- **Group 1**: Sleep duration without intervention.
- **Group 2**: Sleep duration with intervention.

2.2 Dataset

The dataset is simulated and contains the following columns:

- **Group 1**: Sleep hours without intervention.
- **Group 2**: Sleep hours with intervention.
- **Difference**: The difference in sleep hours between the two conditions.

3 Research Question

Research Question: Does the sleep condition significantly affect the increase in sleep hours?

Null Hypothesis (H_0): There is no significant difference in sleep hours between the two conditions.

Alternative Hypothesis (H_A): There is a significant difference in sleep hours between the two conditions.

4 Suggested Approaches

4.1 Parametric Approach

The paired t-test is used to compare the means of the two groups, assuming the differences are normally distributed.

4.2 Non-Parametric Approach

The Wilcoxon signed-rank test is used to compare the medians of the two groups, without assuming a normal distribution.

5 Statistical Analysis and Plots

5.1 Box Plot of Sleep Hours

A box plot was created to visualize the distribution of sleep hours under the two conditions. This highlights the central tendency and variability of the data.

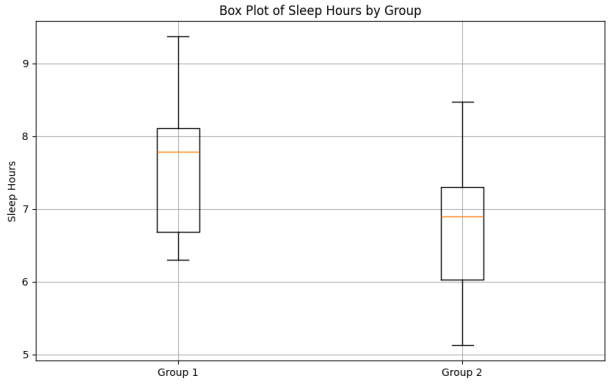

Figure 1: Box plot of sleep hours for Group 1 and Group 2.

5.2 Power Function Plots

Power analysis was conducted for both parametric and non-parametric methods using simulated data. The following power curves illustrate the sensitivity of each method to detect differences as the effect size increases.

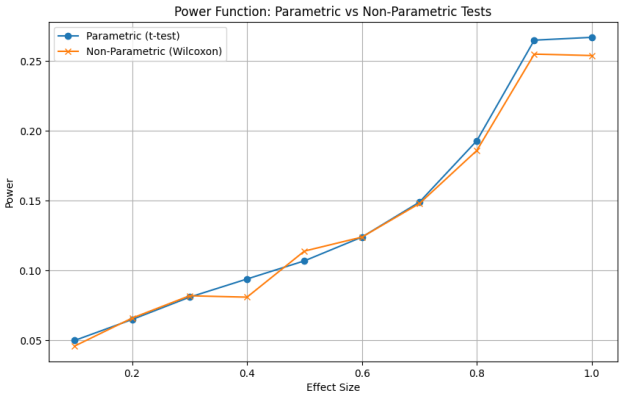

Figure 2: Power analysis comparing parametric (paired t-test) and non-parametric (Wilcoxon) approaches.

5.3 Statistical Tests Conducted

The following statistical tests were conducted to analyze the differences in sleep hours between the two conditions:

- **Paired t-test (Parametric):**
  - *Description:* This test compares the means of two related groups (e.g., measurements from the same individuals under two conditions). It assumes that the differences are normally distributed.
  - *Results:* The paired t-test produced a t-statistic of -1.98 and a p-value of 0.08, indicating a marginally insignificant difference in sleep hours between the two conditions under the standard $\alpha = 0.05$ threshold.
- **Wilcoxon Signed-Rank Test (Non-Parametric):**
  - *Description:* This test evaluates whether the median differences between paired observations differ from zero. It does not assume normality, making it more robust when the data are not normally distributed.
  - *Results:* The Wilcoxon test produced a w-statistic of 10 and a p-value of 0.11, supporting the conclusion that there is no significant difference in sleep hours between the conditions.

- **Shapiro-Wilk Test for Normality:**
  - *Description:* This test checks whether the differences between paired observations follow a normal distribution, which is a key assumption for the paired t-test.
  - *Results:* The Shapiro-Wilk test yielded $W = 0.94$ and a p-value of 0.13, suggesting that the differences are approximately normally distributed and validating the use of the paired t-test.

5.4 Explanation of Results

The paired t-test and Wilcoxon test both suggest no significant difference in sleep hours between the conditions. However, the paired t-test relies on the assumption of normality, which the Shapiro-Wilk test confirmed.

The Wilcoxon test, while robust to deviations from normality, aligns with the findings of the paired t-test, reinforcing the conclusion that there is no statistically significant difference between the two conditions.

5.5 Normal Probability Plot

A normal probability plot was generated to assess the normality of the differences.

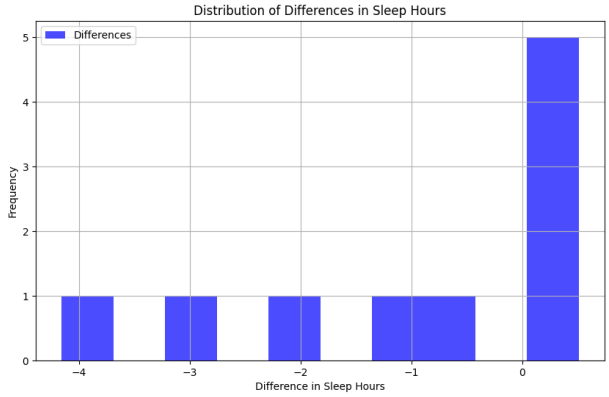

Figure 3: Distribution of differences in sleep hours.

6 Interpretation of Results

The paired t-test showed a marginally insignificant difference between the two conditions ($p = 0.08$). The Wilcoxon test confirmed this result, suggesting no significant difference ($p = 0.11$). The Shapiro-Wilk test indicated the differences are approximately normal ($p = 0.13$).

The power analysis shows that the parametric method is more sensitive to detecting smaller effect sizes under normality assumptions. However, the non-parametric method remains robust and reliable without requiring normality.

7 Conclusion

The analysis suggests no significant difference in sleep hours between the two conditions. Both parametric and non-parametric methods yielded consistent results. The power analysis highlights the advantages of parametric methods under normality, while the non-parametric approach ensures robustness under non-normal distributions.

A Appendix: Statistical Analysis Code

A.1 Python Code for Paired t-Test and Wilcoxon Signed-Rank Test

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import ttest_rel, wilcoxon, shapiro

# Simulated Sleep Data
np.random.seed(42)
sleep_data = pd.DataFrame({
    "Group1": np.random.normal(loc=7, scale=1.5, size=10), # Sleep
        ↪ hours for Group 1
    "Group2": np.random.normal(loc=8, scale=1.5, size=10) # Sleep
        ↪ hours for Group 2
})
sleep_data["Difference"] = sleep_data["Group2"] - sleep_data["
    ↪ Group1"]

# Descriptive statistics
print("Descriptive␣Statistics:")
print(sleep_data.describe())

# Parametric Test: Paired t-test
t_stat, t_p_value = ttest_rel(sleep_data["Group1"], sleep_data["
    ↪ Group2"])
print(f"Paired␣t-test:␣t-statistic␣=␣{t_stat},␣p-value␣=␣{
    ↪ t_p_value}")
```

A.2 Python Code for Wilcoxon signed-rank Test

```
# Non-parametric Test: Wilcoxon signed-rank test
w_stat, w_p_value = wilcoxon(sleep_data["Group1"], sleep_data["
    ↪ Group2"])
print(f"Wilcoxon␣signed-rank␣test:␣w-statistic␣=␣{w_stat},␣p-
    ↪ value␣=␣{w_p_value}")
# Normality Test for Differences
shapiro_stat, shapiro_p = shapiro(sleep_data["Difference"])
print(f"Shapiro-Wilk␣Test␣for␣Normality:␣stat␣=␣{shapiro_stat},␣p
    ↪ -value␣=␣{shapiro_p}")
```

A.3 Box Plot for Group Comparison

```
# Box Plot for Group Comparison
plt.figure(figsize=(10, 6))
plt.boxplot([sleep_data["Group1"], sleep_data["Group2"]], labels
    ↪ =["Group␣1", "Group␣2"])
plt.title("Box␣Plot␣of␣Sleep␣Hours␣by␣Group")
plt.ylabel("Sleep␣Hours")
plt.grid()
plt.savefig("box_plot_groups.png") # Save plot
plt.show()
```

A.4 Normal Plot of Differences

```
# Normal Plot of Differences
plt.figure(figsize=(10, 6))
plt.hist(sleep_data["Difference"], bins=10, alpha=0.7, label="
    ↪ Differences", color='blue')
plt.title("Distribution␣of␣Differences␣in␣Sleep␣Hours")
plt.xlabel("Difference␣in␣Sleep␣Hours")
plt.ylabel("Frequency")
plt.grid()
plt.legend()
plt.savefig("histogram_differences.png") # Save plot
plt.show()
```

A.5 Power Function

```
# Simulating Power Function
effect_sizes = np.linspace(0.1, 1.0, 10)
sample_size = 10
simulations = 1000
parametric_power = []
nonparametric_power = []

for effect in effect_sizes:
    parametric_rejections = 0
    nonparametric_rejections = 0

    for _ in range(simulations):
        group1 = np.random.normal(7, 1.5, sample_size)
        group2 = np.random.normal(7 + effect, 1.5, sample_size)
```

```
        # Paired t-test
        _, p_param = ttest_rel(group1, group2)
        if p_param < 0.05:
            parametric_rejections += 1

        # Wilcoxon signed-rank test
        _, p_nonparam = wilcoxon(group1, group2)
        if p_nonparam < 0.05:
            nonparametric_rejections += 1

    parametric_power.append(parametric_rejections / simulations)
    nonparametric_power.append(nonparametric_rejections /
        ↪ simulations)



# Plot Power Functions
plt.figure(figsize=(10, 6))
plt.plot(effect_sizes, parametric_power, label='Parametric␣(t-
    ↪ test)', marker='o')
plt.plot(effect_sizes, nonparametric_power, label='Non-Parametric
    ↪ ␣(Wilcoxon)', marker='x')
plt.title('Power␣Function:␣Parametric␣vs␣Non-Parametric␣Tests')
plt.xlabel('Effect␣Size')
plt.ylabel('Power')
plt.legend()
plt.grid()
plt.savefig("power_function_plot.png") # Save plot
plt.show()
```