Practical: Multiple Testing, FDR, and ANOVA

Objective:

To understand and implement: - The effects of multiple testing and how FDR adjustment works - Tukey's HSD for comparing group means - Permutation ANOVA in a sports context (Cricket training methods)

Section A: Multiple Testing and FDR

R Instructions:

1. Simulate 50 p-values using `runif()` between 0 and 0.1.
2. Adjust them using `p.adjust(..., method = "fdr")`.
3. Extract and print those adjusted p-values < 0.05.

Python Instructions:

1. Simulate 50 p-values using `np.random.uniform()` between 0 and 0.1.
2. Use `statsmodels.stats.multitest.multipletests()` with method `'fdr_bh'`.
3. Print adjusted p-values that are < 0.05.

Section B: Tukey's Honest Significant Difference (HSD)

R Instructions:

1. Simulate 3 groups (A, B, C), each with 10 normal values (means 5, 6, 7).
2. Perform `aov()` and extract MSE and df.
3. Use `qtukey()` to calculate critical value.
4. Calculate HSD.

Python Instructions:

1. Simulate 3 groups using `np.random.normal()`.
2. Conduct ANOVA using `scipy.stats.f_oneway()`.
3. Estimate pooled variance (MSE).
4. Use a static value or approximation for the Tukey q-statistic.
5. Compute HSD manually.

Section C: Permutation ANOVA in Sports

R Instructions:

1. Create a data frame with four cricket training types: Power, Agility, Endurance, Control (5 each).
2. Generate simulated reaction times with slightly different means.
3. Use `lmPerm::aovp()` with `perm = "Exact"`.
4. Interpret the `Pr(Prob)` output.

Python Instructions:

1. Simulate the same cricket training dataset.
2. Calculate observed group mean variance.
3. Define a permutation function to shuffle outcomes.
4. Run the permutation test (e.g., 3000 resamples).
5. Calculate the permutation p-value.
