ALGO PYTHON TEST

The assignment must be submitted back in 24 hours. Incomplete solutions will still be considered but late submissions will not. **Please submit the solution in public git repo. All the questions need to be coded in python.**

|  |  |  |
| --- | --- | --- |
| Data Description | | |
| Column1 | Customer | This is the customer location code |
| Column2 | 48806 | This is the key product against which all other titles will be measured. |
| Column3 | 47106 | Non-key product that should be compared to the key title |
| Column4 | 47287 | Non-key product that should be compared to the key title |
| Column5 | 48020 | Non-key product that should be compared to the key title |
| Column6 | 48863 | Non-key product that should be compared to the key title |
| Column7 | 50546 | Non-key product that should be compared to the key title |

**Question 1:** Lookup up the algorithm for the Wilcoxon Signed Rank Test. This can be found in text books or on Wikipedia.

Code the Wilcoxon Signed Rank Test in Python using the data provided in the tab "data" as your test case. Do not use any pre-packaged statistic functions, you must code the algorithm yourself.

Your code should calculate the signed rank test for each pair of products using the Key title each time. You do not need to calculate p-values or make hypothesis decisions, just calculate the test statistic for each pair. So for example the code should calculate the signed rank test for 48806 and 47106, then it should calculate the signed rank test for 48806 and 47287, ……, and finally for 48806 and 50546.

**The code should work for any data supplied to it.**  **Any data supplied to it implies that there will only be one key product (the left most column) and as many other products as the user deems necessary. Supply all code as well as the Signed Rank Test statistics for each pair of products. *Additional Note: The signed rank test has a special way that it handles ties of rank. Any form of rank tie handling is acceptable so don't get caught up in how ranks are handled by your chosen rank function.***

**Question 2:** Calculate all permutations of the non-key columns. Using 100% equal weighting calculate the weighted combination of the non-key columns and the run through the Wilkoxon Signed Rank test. Note: This must be coded in Python and should handle any number of columns. The current example is 1 key product and 5 non-key, but another example could be 1 key product and 3 non-key. Or 8 non-key etc… *Additional Note: Please see the tab titled, "Question2 - Permut & Weights" for additional help with this question.*

**Question 3:** Considering question number one, which non-key product had the best fit to the key product?

**Question 4:** Considering question number two, which non-key product permutation had the best fit to the key product?

**NOTE: External Libraries such as Pandas, Numpy or any other statistic libraries are not allowed.**