

recall

April 7, 2017

0.0.1 This report shows applying statistical tests of the results of Multi armed bandit of pruning the parameters

0.0.2 Here, we are showing two kinds of testing ANOVA test and Nonparametric tests

1 Import needed libraries

1.1 Import libraries for manipulating the data and statistic

```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import scipy.stats as stats
from scipy.stats import ttest_1samp, wilcoxon, ttest_ind, mannwhitneyu
import scipy.special as special
import emoji
from math import pi
from statsmodels.stats.multicomp import pairwise_tukeyhsd, MultiComparison
from statsmodels.formula.api import ols
import statsmodels.stats.api as sms
```

1.2 Import libraries for static plotting

```
In [2]: import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec
%matplotlib inline
from IPython.display import set_matplotlib_formats
set_matplotlib_formats('png', 'pdf')
# some nice colors from http://colorbrewer2.org/
COLOR1 = '#7fc97f'
COLOR2 = '#beaed4'
COLOR3 = '#fdc086'
COLOR4 = '#ffff99'
COLOR5 = '#386cb0'
```

1.3 Import libraries for interactive plotting Plotly

```
In [3]: import plotly.plotly as py
from plotly.graph_objs import *
```

```

import plotly.graph_objs as go
#from plotly.tools import FigureFactory as FF
import plotly.figure_factory as FF
import cufflinks as cf
cf.go_offline()

```

<IPython.core.display.HTML object>

1.4 Import libraries for interactive plotting BOKEH

```

In [4]: from bokeh.charts import Bar, Area, defaults, Donut
        from bokeh.layouts import row, gridplot
        from bokeh.charts.attributes import cat, color
        from bokeh.charts.operations import blend
        from bokeh.plotting import figure, output_notebook, show
        from bokeh.models import Legend
        TOOLS = 'box_zoom,box_select,crosshair,resize,reset,lasso_select,pan,save,poly_select,ta
        #defaults.width = 1000
        #defaults.height = 800
        output_notebook()

```

2 Statring the test and visulize the data on small model

2.1 Load the data for pruning the weights using random expoloration

```

In [5]: datafile = "./results/result.xlsx"
        #datafileLeNet = "LecunPruningWeights.csv"
        df_recall = pd.read_excel(datafile, sheetname='accuracy')
        df_recall = pd.read_excel(datafile, sheetname='f1score')
        df_recall = pd.read_excel(datafile, sheetname='precision')
        df_recall = pd.read_excel(datafile, sheetname='recall')
        #dfLcun = pd.read_csv(datafileLeNet)
        df_recall

```

```

Out [5]:

```

	Methods	SPAM	ABALONE	ADULT	CANCER	CAR \
0	Knn	0.852601	0.542680	0.550538	0.921053	0.876665
1	LSVM	0.878613	0.557255	0.267742	0.947368	0.623619
2	SVM	0.887283	0.570627	0.432258	0.947368	0.928105
3	DT_gini	0.797688	0.473254	0.582796	0.947368	0.823275
4	DT_entorpy	0.809249	0.469521	0.566667	0.894737	0.917506
5	Bagging Knn	0.864162	0.553448	0.415054	0.947368	0.354437
6	Bagging DT	0.791908	0.550163	0.370968	0.921053	0.355466
7	Random Forest	0.745665	0.512993	0.489247	0.921053	0.704557
8	Ada Boost	0.329480	0.542570	0.416129	0.947368	0.845517
9	NB	0.933526	0.538859	0.236559	0.868421	0.750690
10	LDA	0.832370	0.556172	0.151613	0.947368	0.446854
11	QDA	0.921965	0.550510	0.274194	0.921053	0.845611

12	Log. Reg.	0.838150	0.553947	0.232258	0.947368	0.631958
13	GP Class.	0.872832	0.552310	0.510753	0.947368	0.792005
14	LightGBM	0.867052	0.528754	0.482796	0.947368	0.715657
15	Xgboost	0.595376	0.564259	0.501075	0.921053	0.794767
16	NN	0.893064	0.566747	0.551613	0.947368	0.933656
17	UCB1	0.901734	0.552544	0.575269	0.947368	0.784695
18	E Greedy	0.898844	0.552544	0.509677	0.921053	0.787362
19	Decay E Gr.	0.901734	0.552544	0.646237	0.947368	0.785101
20	Softmax	0.895954	0.552544	0.435484	0.947368	0.603720
21	Decay SM	0.901734	0.552544	0.454839	0.947368	0.760505
22	Tomp. Sampling	0.907514	0.552544	0.575269	0.947368	0.784695
23	Hedge	0.898844	0.552544	0.592473	0.947368	0.684603
24	EXP3	0.893064	0.552544	0.618280	0.947368	0.555880

	GLASS	HEART	IRIS	PIMA	POKER	TITANIC	VALLY \
0	0.513158	0.378898	1.000000	0.385965	0.241667	0.671233	0.308682
1	0.621540	0.280087	0.933333	0.122807	0.101000	0.657534	0.215434
2	0.596491	0.644105	1.000000	0.473684	0.297333	0.712329	0.241158
3	0.458577	0.636685	1.000000	0.508772	0.643000	0.726027	0.254019
4	0.504873	0.670282	0.900000	0.578947	0.482333	0.698630	0.295820
5	0.503411	0.393187	0.962963	0.175439	0.166000	0.671233	0.273312
6	0.521930	0.595246	0.866667	0.245614	0.202333	0.657534	0.221865
7	0.504873	0.679468	1.000000	0.350877	0.432667	0.630137	0.270096
8	0.324561	0.303277	0.933333	0.350877	0.106000	0.726027	0.257235
9	0.588207	0.304179	0.933333	0.210526	0.100000	0.698630	0.935691
10	0.695127	0.298266	0.966667	0.175439	0.100000	0.671233	0.176849
11	0.361306	0.470728	0.966667	0.263158	0.125000	0.808219	0.926045
12	0.538207	0.285121	0.966667	0.175439	0.100000	0.671233	0.209003
13	0.615010	0.705555	1.000000	0.368421	0.354000	0.726027	0.276527
14	0.449805	0.410668	0.933333	0.561404	0.174333	0.657534	0.266881
15	0.448830	0.616188	0.933333	0.385965	0.157333	0.643836	0.270096
16	0.569201	0.464443	0.966667	0.403509	0.320000	0.698630	0.247588
17	0.596491	0.502887	1.000000	0.578947	0.327000	0.698630	0.244373
18	0.570175	0.477680	0.933333	0.385965	0.253333	0.698630	0.260450
19	0.578947	0.414183	0.900000	0.385965	0.256333	0.698630	0.221865
20	0.569201	0.376037	0.900000	0.526316	0.285667	0.698630	0.241158
21	0.476121	0.416234	0.666667	0.228070	0.277667	0.698630	0.234727
22	0.596491	0.404421	0.666667	0.578947	0.287667	0.698630	0.244373
23	0.504386	0.407049	0.666667	0.438596	0.327000	0.698630	0.231511
24	0.384016	0.409997	0.629630	0.350877	0.254333	0.698630	0.254019

	WINE	WINE QUILTY	FACE	CHEST
0	0.979167	0.268392	0.571062	0.348958
1	1.000000	0.206591	0.000000	0.135417
2	1.000000	0.243556	0.758668	0.244461
3	0.913690	0.287854	0.257082	0.311177
4	0.881944	0.325395	0.000000	0.348793
5	0.979167	0.262090	0.000000	0.207672

6	0.916667	0.267395	0.000000	0.184689
7	0.916667	0.293228	0.000000	0.396660
8	0.916667	0.242361	0.000000	0.227679
9	0.937500	0.226134	0.000000	0.157738
10	0.958333	0.221732	0.000000	0.139385
11	1.000000	0.237772	0.000000	0.159722
12	1.000000	0.206516	0.000000	0.145337
13	0.937500	0.274211	0.000000	0.237269
14	0.902778	0.229758	0.000000	0.288360
15	0.937500	0.229811	0.000000	0.357722
16	1.000000	0.234397	0.756397	0.203125
17	1.000000	0.241319	0.766407	0.189236
18	1.000000	0.225893	0.774608	0.184524
19	1.000000	0.206392	0.759307	0.170966
20	1.000000	0.245301	0.756397	0.185268
21	1.000000	0.223855	0.756397	0.208664
22	1.000000	0.226295	0.756397	0.189236
23	1.000000	0.204154	0.756397	0.175099
24	0.979167	0.218789	0.756397	0.177745

3 Starting with recall

4 First, All methods

4.1 Visulize the Accuracy of all the models and methods

```
In [6]: p = Bar(df_recall, label= 'Methods',
               values = blend('SPAM', 'ABALONE', 'ADULT', 'CANCER', 'CAR', 'GLASS',
                              'HEART', 'IRIS', 'PIMA', 'POKER', 'TITANIC', 'VALLY', 'WINE',
                              'WINE QUILTY', 'FACE', 'CHEST'
                              ,name='Scores', labels_name='Score'),
               group=cat(columns='Score', sort=False),
               title="Compare the performance", legend='bottom_center',
               tools=TOOLS, plot_width=3500, plot_height=1600,
               tooltips=[('Score', '@Score'), ('Model', '@Methods')],
               xlabel='List of Models', ylabel='Score')
p.title.align = "center"
#p.yaxis.major_label_orientation = "vertical"
p.xaxis.major_label_orientation = pi/2
show(p)
```

4.1.1 Only pruning methods

```
In [7]: df_recall_pruning = df_recall.iloc[16:24]
p = Bar(df_recall_pruning, label= 'Methods',
        values = blend('SPAM', 'ABALONE', 'ADULT', 'CANCER', 'CAR', 'GLASS',
                        'HEART', 'IRIS', 'PIMA', 'POKER', 'TITANIC', 'VALLY', 'WINE',
                        'WINE QUILTY', 'FACE', 'CHEST'
```

```

        ,name='Scores', labels_name='Score'),
        group=cat(columns='Score', sort=False),
        title="Compare the performance", legend='bottom_center',
        tools=TOOLS, plot_width=2000, plot_height=1500,
        tooltips=[('Score', '@Score'), ('Model', '@Methods')],
        xlabel='List of Models', ylabel='Score')
p.title.align = "center"
#p.yaxis.major_label_orientation = "vertical"
p.xaxis.major_label_orientation = pi/2
show(p)

```

```

In [8]: df=df_recall.copy()
        df.set_index('Methods', inplace=True)
        py.iplot([
            'x': df.index,
            'y': df[col],
            'name': col
        ] for col in df.columns])

```

Out[8]: <plotly.tools.PlotlyDisplay object>

```

In [9]: df_pruning=df_recall_pruning.copy()
        df_pruning.set_index('Methods', inplace=True)
        py.iplot([
            'x': df_pruning.index,
            'y': df_pruning[col],
            'name': col
        ] for col in df_pruning.columns])

```

Out[9]: <plotly.tools.PlotlyDisplay object>

```

In [10]: df.iplot(subplots=True, shape=(16,1), shared_xaxes=True, fill=True)

<IPython.core.display.HTML object>

```

```

In [11]: df.iplot(kind='bar', barmode='stack')

<IPython.core.display.HTML object>

```

```

In [12]: df.iplot(kind='barh',barmode='stack', bargap=.2)

<IPython.core.display.HTML object>

```

```

In [13]: df.T.iplot(kind='barh',barmode='stack', bargap=.2)

<IPython.core.display.HTML object>

```

```
In [14]: df.iplot(kind='box')

<IPython.core.display.HTML object>
```

```
In [15]: df.T.iplot(kind='box')

<IPython.core.display.HTML object>
```

4.1.2 We will use alpha 0.05 to do ANOVA test. The null hypothesis there is no difference between the all methods and the alternative hypothesis there is a difference. According to p-value we see if there is a difference.

```
In [16]: df.T.columns
```

```
Out[16]: Index(['Knn', 'LSVM', 'SVM', 'DT_gini', 'DT_entropy', 'Bagging Knn',
               'Bagging DT', 'Random Forest', 'Ada Boost', 'NB', 'LDA', 'QDA',
               'Log. Reg.', 'GP Class.', 'LightGBM', 'Xgboost', 'NN', 'UCB1',
               'E Greedy', 'Decay E Gr.', 'Softmax', 'Decay SM', 'Tomp. Sampling',
               'Hedge', 'EXP3'],
              dtype='object', name='Methods')
```

```
In [17]: # Perform the ANOVA
df1 = df.T
stats.f_oneway(df1['Knn'], df1['LSVM'], df1['SVM'], df1['DT_gini'], df1['DT_entropy'],
               df1['Bagging Knn'], df1['Bagging DT'], df1['Random Forest'],
               df1['Ada Boost'], df1['NB'], df1['LDA'], df1['QDA'],
               df1['Log. Reg.'], df1['GP Class.'], df1['LightGBM'], df1['Xgboost'],
               df1['NN'], df1['UCB1'], df1['E Greedy'], df1['Decay E Gr.'],
               df1['Softmax'], df1['Decay SM'], df1['Tomp. Sampling'],
               df1['Hedge'], df1['EXP3'])
```

```
Out[17]: F_onewayResult(statistic=0.49852884857039009, pvalue=0.97851717698962248)
```

4.1.3 One post-hoc test is to perform a separate t-test for each pair of groups. We can perform a t-test between all pairs using by running each pair through the stats.ttest_ind() we covered in the following to do t-tests:

```
In [18]: # Get all models pairs
interstModel = ['NN', 'UCB1',
               'E Greedy', 'Decay E Gr.', 'Softmax', 'Decay SM', 'Tomp. Sampling',
               'Hedge', 'EXP3']
lst = list(df1.columns.values)
#lst.remove('Methods')
model_pairs = []

for m1 in range(len(df1.columns)-1):
    for m2 in range(m1+1, len(df1.columns)):
```

```

        model_pairs.append((lst[m1], lst[m2]))

    # Conduct t-test on each pair
    pvalueList = []
    new_model_pairs = []
    for m1, m2 in model_pairs:
        print('\n',m1, m2)
        pvalue = stats.ttest_ind(df1[m1], df1[m2])
        #print(pvalue[1])
        if (m1 in interstModel or m2 in interstModel):
            new_model_pairs.append((m1,m2))
            pvalueList.append(pvalue[1])
        print(pvalue)

Knn LSVM
Ttest_indResult(statistic=1.0727633173320774, pvalue=0.29193124349000515)

Knn SVM
Ttest_indResult(statistic=-0.36781225438122783, pvalue=0.71559563376600555)

Knn DT_gini
Ttest_indResult(statistic=-0.14433391333272166, pvalue=0.8862020598249345)

Knn DT_entropy
Ttest_indResult(statistic=0.04422503596114475, pvalue=0.96501808831681124)

Knn Bagging Knn
Ttest_indResult(statistic=1.0198196608333312, pvalue=0.31596816316024801)

Knn Bagging DT
Ttest_indResult(statistic=1.1074768521554712, pvalue=0.27688871857063369)

Knn Random Forest
Ttest_indResult(statistic=0.37269856550525704, pvalue=0.71199307045959093)

Knn Ada Boost
Ttest_indResult(statistic=1.1904770920367731, pvalue=0.24319417346623787)

Knn NB
Ttest_indResult(statistic=0.57001772102796855, pvalue=0.57291014719265376)

Knn LDA
Ttest_indResult(statistic=1.1956881025140573, pvalue=0.24118420529917883)

Knn QDA
Ttest_indResult(statistic=0.3276162087102753, pvalue=0.74547724352940936)

```

Knn Log. Reg.
Ttest_indResult(statistic=1.1063580981913284, pvalue=0.27736470033462785)

Knn GP Class.
Ttest_indResult(statistic=0.15078887177949582, pvalue=0.88115160812553905)

Knn LightGBM
Ttest_indResult(statistic=0.63011280860982488, pvalue=0.5333912641154317)

Knn Xgboost
Ttest_indResult(statistic=0.67901795802563347, pvalue=0.502331841234259)

Knn NN
Ttest_indResult(statistic=-0.22337732298766899, pvalue=0.82475596028480025)

Knn UCB1
Ttest_indResult(statistic=-0.32737788689758673, pvalue=0.74565567111925168)

Knn E Greedy
Ttest_indResult(statistic=-0.015238416578749586, pvalue=0.98794286609423332)

Knn Decay E Gr.
Ttest_indResult(statistic=-0.0095689960223278962, pvalue=0.99242851379262098)

Knn Softmax
Ttest_indResult(statistic=0.12687294193424639, pvalue=0.89988752700725028)

Knn Decay SM
Ttest_indResult(statistic=0.39746463371222301, pvalue=0.69383885685344326)

Knn Tomp. Sampling
Ttest_indResult(statistic=-0.0039062590855205677, pvalue=0.99690912448902447)

Knn Hedge
Ttest_indResult(statistic=0.21916469958323373, pvalue=0.8280060607190588)

Knn EXP3
Ttest_indResult(statistic=0.48892686863530521, pvalue=0.62844895352152197)

LSVM SVM
Ttest_indResult(statistic=-1.3668337713445942, pvalue=0.18183380203333019)

LSVM DT_gini
Ttest_indResult(statistic=-1.2146833503778836, pvalue=0.23396149658294468)

LSVM DT_entropy
Ttest_indResult(statistic=-1.0349235015924385, pvalue=0.30897548627163168)

LSVM Bagging Knn
Ttest_indResult(statistic=-0.096483077939875539, pvalue=0.92377858554557213)

LSVM Bagging DT
Ttest_indResult(statistic=-0.067141178358859563, pvalue=0.94691479276413482)

LSVM Random Forest
Ttest_indResult(statistic=-0.74277291050774708, pvalue=0.46339614126299222)

LSVM Ada Boost
Ttest_indResult(statistic=0.042705998521787143, pvalue=0.96621887576012133)

LSVM NB
Ttest_indResult(statistic=-0.44684398020637095, pvalue=0.65819702041770212)

LSVM LDA
Ttest_indResult(statistic=0.108332833747841, pvalue=0.91445301074843699)

LSVM QDA
Ttest_indResult(statistic=-0.64954615558473383, pvalue=0.52092813388606662)

LSVM Log. Reg.
Ttest_indResult(statistic=0.024297491550743136, pvalue=0.980776215237805)

LSVM GP Class.
Ttest_indResult(statistic=-0.88757556713134644, pvalue=0.38182999280319108)

LSVM LightGBM
Ttest_indResult(statistic=-0.47978765556075736, pvalue=0.63485799007996258)

LSVM Xgboost
Ttest_indResult(statistic=-0.45271091653288337, pvalue=0.6540139832346239)

LSVM NN
Ttest_indResult(statistic=-1.2383639104619728, pvalue=0.22518434744834886)

LSVM UCB1
Ttest_indResult(statistic=-1.3435866627385509, pvalue=0.18915725956408289)

LSVM E Greedy
Ttest_indResult(statistic=-1.0652856915569149, pvalue=0.29524574654605185)

LSVM Decay E Gr.
Ttest_indResult(statistic=-1.0503034649816003, pvalue=0.30196616238522811)

LSVM Softmax
Ttest_indResult(statistic=-0.95402943572110865, pvalue=0.34769326026586511)

LSVM Decay SM
Ttest_indResult(statistic=-0.71153979813399026, pvalue=0.48224640325501289)

LSVM Tomp. Sampling
Ttest_indResult(statistic=-1.0809423873701123, pvalue=0.28833599752585953)

LSVM Hedge
Ttest_indResult(statistic=-0.88905130615187211, pvalue=0.38104916182180926)

LSVM EXP3
Ttest_indResult(statistic=-0.65232276512106935, pvalue=0.51916036347487449)

SVM DT_gini
Ttest_indResult(statistic=0.23626948934118627, pvalue=0.81482929820571504)

SVM DT_entropy
Ttest_indResult(statistic=0.41075523605717296, pvalue=0.68417123367955024)

SVM Bagging Knn
Ttest_indResult(statistic=1.3286469788764432, pvalue=0.1939829271057005)

SVM Bagging DT
Ttest_indResult(statistic=1.4268353048547333, pvalue=0.16395297554811258)

SVM Random Forest
Ttest_indResult(statistic=0.72552347892412583, pvalue=0.47375304176894784)

SVM Ada Boost
Ttest_indResult(statistic=1.4978220669436235, pvalue=0.14462855116361426)

SVM NB
Ttest_indResult(statistic=0.87539672856539785, pvalue=0.38831327866726384)

SVM LDA
Ttest_indResult(statistic=1.4872142795902308, pvalue=0.14739417256629209)

SVM QDA
Ttest_indResult(statistic=0.63390705562251148, pvalue=0.5309455216211677)

SVM Log. Reg.
Ttest_indResult(statistic=1.4011433096725241, pvalue=0.17143156915371072)

SVM GP Class.
Ttest_indResult(statistic=0.49175482203702581, pvalue=0.62647171592390283)

SVM LightGBM
Ttest_indResult(statistic=0.96177279340963662, pvalue=0.34385224175814222)

SVM Xgboost
Ttest_indResult(statistic=1.0142784505873463, pvalue=0.31856076722508914)

SVM NN
Ttest_indResult(statistic=0.13870159129516044, pvalue=0.89061284776524063)

SVM UCB1
Ttest_indResult(statistic=0.045135631308602368, pvalue=0.96429830945451145)

SVM E Greedy
Ttest_indResult(statistic=0.34409261494132642, pvalue=0.73317691085721282)

SVM Decay E Gr.
Ttest_indResult(statistic=0.34527512429196827, pvalue=0.73229683469083451)

SVM Softmax
Ttest_indResult(statistic=0.48682579680049404, pvalue=0.62991978947128546)

SVM Decay SM
Ttest_indResult(statistic=0.74596249736719222, pvalue=0.46149566732846803)

SVM Tomp. Sampling
Ttest_indResult(statistic=0.36613205236009838, pvalue=0.71683595168649461)

SVM Hedge
Ttest_indResult(statistic=0.58209811848620163, pvalue=0.56484988392299973)

SVM EXP3
Ttest_indResult(statistic=0.84166117364598936, pvalue=0.40663709488059518)

DT_gini DT_entropy
Ttest_indResult(statistic=0.18963968825852795, pvalue=0.85086877514379355)

DT_gini Bagging Knn
Ttest_indResult(statistic=1.1695403692052839, pvalue=0.25139445803851562)

DT_gini Bagging DT
Ttest_indResult(statistic=1.2677070967245505, pvalue=0.21465403571004912)

DT_gini Random Forest
Ttest_indResult(statistic=0.52400827250245285, pvalue=0.60412300257023499)

DT_gini Ada Boost
Ttest_indResult(statistic=1.3455050382368361, pvalue=0.18854439757237143)

DT_gini NB
Ttest_indResult(statistic=0.7030770723413633, pvalue=0.48742843731371344)

DT_gini LDA
Ttest_indResult(statistic=1.3399359574124503, pvalue=0.19032780754686282)

DT_gini QDA
Ttest_indResult(statistic=0.45425102823915958, pvalue=0.65291778908000198)

DT_gini Log. Reg.
Ttest_indResult(statistic=1.2498378203706109, pvalue=0.22102144722845724)

DT_gini GP Class.
Ttest_indResult(statistic=0.28835695426276592, pvalue=0.77505649951559308)

DT_gini LightGBM
Ttest_indResult(statistic=0.77961737837146994, pvalue=0.44172353383684348)

DT_gini Xgboost
Ttest_indResult(statistic=0.83246141755367364, pvalue=0.41172670959793756)

DT_gini NN
Ttest_indResult(statistic=-0.089237156659175573, pvalue=0.92948652885660032)

DT_gini UCB1
Ttest_indResult(statistic=-0.19276971362423143, pvalue=0.84843841338064263)

DT_gini E Greedy
Ttest_indResult(statistic=0.12486235517056268, pvalue=0.90146546680324913)

DT_gini Decay E Gr.
Ttest_indResult(statistic=0.1287949947209055, pvalue=0.89837945856909618)

DT_gini Softmax
Ttest_indResult(statistic=0.27236812962538998, pvalue=0.7872049098250038)

DT_gini Decay SM
Ttest_indResult(statistic=0.54746537112572813, pvalue=0.5881092447218923)

DT_gini Tomp. Sampling
Ttest_indResult(statistic=0.14125633437830132, pvalue=0.88861172596075289)

DT_gini Hedge
Ttest_indResult(statistic=0.36959016512367604, pvalue=0.71428404610274643)

DT_gini EXP3
Ttest_indResult(statistic=0.64480715871099614, pvalue=0.52395279503317471)

DT_entropy Bagging Knn
Ttest_indResult(statistic=0.97998371042577814, pvalue=0.33493148988370736)

DT_entorpy Bagging DT
Ttest_indResult(statistic=1.0657228928793689, pvalue=0.29505122831606856)

DT_entorpy Random Forest
Ttest_indResult(statistic=0.32903760646672287, pvalue=0.74441336480182829)

DT_entorpy Ada Boost
Ttest_indResult(statistic=1.1502359471311923, pvalue=0.25913335842927882)

DT_entorpy NB
Ttest_indResult(statistic=0.53212997507075221, pvalue=0.59855517988058216)

DT_entorpy LDA
Ttest_indResult(statistic=1.1578257872692217, pvalue=0.25607023860845707)

DT_entorpy QDA
Ttest_indResult(statistic=0.29029660255347278, pvalue=0.77358660088143116)

DT_entorpy Log. Reg.
Ttest_indResult(statistic=1.0683038375345291, pvalue=0.29390476061588111)

DT_entorpy GP Class.
Ttest_indResult(statistic=0.10949358593505126, pvalue=0.91354015965133206)

DT_entorpy LightGBM
Ttest_indResult(statistic=0.58840186613074819, pvalue=0.56066677250363406)

DT_entorpy Xgboost
Ttest_indResult(statistic=0.63661440442227302, pvalue=0.52920404665929333)

DT_entorpy NN
Ttest_indResult(statistic=-0.2660985305563664, pvalue=0.79198367164403805)

DT_entorpy UCB1
Ttest_indResult(statistic=-0.37102500145715839, pvalue=0.71322619489690453)

DT_entorpy E Greedy
Ttest_indResult(statistic=-0.0583259729721382, pvalue=0.95387575996632057)

DT_entorpy Decay E Gr.
Ttest_indResult(statistic=-0.052105859171499819, pvalue=0.95878975992497195)

DT_entorpy Softmax
Ttest_indResult(statistic=0.083193556041282549, pvalue=0.93425027948919781)

DT_entorpy Decay SM
Ttest_indResult(statistic=0.35429705599846506, pvalue=0.72559456297329916)

DT_entorpy Tomp. Sampling
Ttest_indResult(statistic=-0.048401915128963025, pvalue=0.96171673564643345)

DT_entorpy Hedge
Ttest_indResult(statistic=0.17474220005749688, pvalue=0.86245619211032665)

DT_entorpy EXP3
Ttest_indResult(statistic=0.44481821087754503, pvalue=0.65964398725966933)

Bagging Knn Bagging DT
Ttest_indResult(statistic=0.034540605222754567, pvalue=0.97267486432873707)

Bagging Knn Random Forest
Ttest_indResult(statistic=-0.67272173607779828, pvalue=0.50627345893847031)

Bagging Knn Ada Boost
Ttest_indResult(statistic=0.14641448190564746, pvalue=0.88457365211893457)

Bagging Knn NB
Ttest_indResult(statistic=-0.36881396995166849, pvalue=0.71485654649927477)

Bagging Knn LDA
Ttest_indResult(statistic=0.20938310026123902, pvalue=0.83556455657890594)

Bagging Knn QDA
Ttest_indResult(statistic=-0.58046315183659136, pvalue=0.56593740399338077)

Bagging Knn Log. Reg.
Ttest_indResult(statistic=0.12227624214462898, pvalue=0.90349568929760071)

Bagging Knn GP Class.
Ttest_indResult(statistic=-0.82598638186235496, pvalue=0.4153326205686324)

Bagging Knn LightGBM
Ttest_indResult(statistic=-0.39830459308229993, pvalue=0.69322629446567297)

Bagging Knn Xgboost
Ttest_indResult(statistic=-0.36905324139352613, pvalue=0.71468004841592914)

Bagging Knn NN
Ttest_indResult(statistic=-1.1936836501237287, pvalue=0.24195589504668624)

Bagging Knn UCB1
Ttest_indResult(statistic=-1.3045847900914158, pvalue=0.20195406997382845)

Bagging Knn E Greedy
Ttest_indResult(statistic=-1.0119695792834869, pvalue=0.31964534298421665)

Bagging Knn Decay E Gr.
Ttest_indResult(statistic=-0.99627222439945085, pvalue=0.32708629446221849)

Bagging Knn Softmax
Ttest_indResult(statistic=-0.89493761088224599, pvalue=0.37794490608018938)

Bagging Knn Decay SM
Ttest_indResult(statistic=-0.64021043739986372, pvalue=0.52689566341724248)

Bagging Knn Tomp. Sampling
Ttest_indResult(statistic=-1.0284394700982176, pvalue=0.31196415280519724)

Bagging Knn Hedge
Ttest_indResult(statistic=-0.82633599266512614, pvalue=0.41513742493235051)

Bagging Knn EXP3
Ttest_indResult(statistic=-0.57720378848028786, pvalue=0.56810855999957521)

Bagging DT Random Forest
Ttest_indResult(statistic=-0.74242750870830165, pvalue=0.46360221906613341)

Bagging DT Ada Boost
Ttest_indResult(statistic=0.1176383318501224, pvalue=0.90713832283496332)

Bagging DT NB
Ttest_indResult(statistic=-0.41545507545233223, pvalue=0.68076542618336955)

Bagging DT LDA
Ttest_indResult(statistic=0.18423476940365108, pvalue=0.85506900983892442)

Bagging DT QDA
Ttest_indResult(statistic=-0.63386090394442318, pvalue=0.53097523468738217)

Bagging DT Log. Reg.
Ttest_indResult(statistic=0.093767009479664154, pvalue=0.92591769042636285)

Bagging DT GP Class.
Ttest_indResult(statistic=-0.89734045178585553, pvalue=0.3766824389671013)

Bagging DT LightGBM
Ttest_indResult(statistic=-0.45192297877799908, pvalue=0.65457511153761549)

Bagging DT Xgboost
Ttest_indResult(statistic=-0.42242455444672034, pvalue=0.67572747537800137)

Bagging DT NN
Ttest_indResult(statistic=-1.2850560445888053, pvalue=0.20860590204764889)

Bagging DT UCB1
Ttest_indResult(statistic=-1.4039950730036361, pvalue=0.17058844148212679)

Bagging DT E Greedy
Ttest_indResult(statistic=-1.0961284022855671, pvalue=0.28174416127087132)

Bagging DT Decay E Gr.
Ttest_indResult(statistic=-1.0782243811996963, pvalue=0.28952724758311216)

Bagging DT Softmax
Ttest_indResult(statistic=-0.9751983610774887, pvalue=0.33726032312416931)

Bagging DT Decay SM
Ttest_indResult(statistic=-0.70737319842498092, pvalue=0.48479382348767885)

Bagging DT Tomp. Sampling
Ttest_indResult(statistic=-1.1172662215920766, pvalue=0.27274870223831826)

Bagging DT Hedge
Ttest_indResult(statistic=-0.90513574380382644, pvalue=0.37260558881823991)

Bagging DT EXP3
Ttest_indResult(statistic=-0.64323360417754238, pvalue=0.52495920337731938)

Random Forest Ada Boost
Ttest_indResult(statistic=0.83795484656493557, pvalue=0.40868279903970983)

Random Forest NB
Ttest_indResult(statistic=0.2444054738073683, pvalue=0.80858070794494008)

Random Forest LDA
Ttest_indResult(statistic=0.86431894709105628, pvalue=0.39427124267924529)

Random Forest QDA
Ttest_indResult(statistic=0.009120156160979118, pvalue=0.9927836487569941)

Random Forest Log. Reg.
Ttest_indResult(statistic=0.77414467035545109, pvalue=0.44490363856056592)

Random Forest GP Class.
Ttest_indResult(statistic=-0.19938968614062091, pvalue=0.84330323503154103)

Random Forest LightGBM
Ttest_indResult(statistic=0.27096935697302482, pvalue=0.78827034865700463)

Random Forest Xgboost
Ttest_indResult(statistic=0.31332708909001933, pvalue=0.75620029045202741)

Random Forest NN
Ttest_indResult(statistic=-0.5810108934700674, pvalue=0.56557294864567398)

Random Forest UCB1
Ttest_indResult(statistic=-0.69125738964141714, pvalue=0.49471863742289579)

Random Forest E Greedy
Ttest_indResult(statistic=-0.37836775082631285, pvalue=0.70782176584666989)

Random Forest Decay E Gr.
Ttest_indResult(statistic=-0.36822086758914147, pvalue=0.71529411629216533)

Random Forest Softmax
Ttest_indResult(statistic=-0.24290641527177592, pvalue=0.80973106314427312)

Random Forest Decay SM
Ttest_indResult(statistic=0.028637368012752473, pvalue=0.9773434710570017)

Random Forest Temp. Sampling
Ttest_indResult(statistic=-0.37878484642528742, pvalue=0.7075152345251734)

Random Forest Hedge
Ttest_indResult(statistic=-0.15798678308923969, pvalue=0.87552585177285247)

Random Forest EXP3
Ttest_indResult(statistic=0.11095349341291755, pvalue=0.91239221395232639)

Ada Boost NB
Ttest_indResult(statistic=-0.51185230811149873, pvalue=0.61250186000828655)

Ada Boost LDA
Ttest_indResult(statistic=0.071021400010044131, pvalue=0.94385207391031001)

Ada Boost QDA
Ttest_indResult(statistic=-0.72319215272571602, pvalue=0.47516300751147811)

Ada Boost Log. Reg.
Ttest_indResult(statistic=-0.017397562935197123, pvalue=0.98623464588184884)

Ada Boost GP Class.
Ttest_indResult(statistic=-0.98461769315374192, pvalue=0.33268671795343596)

Ada Boost LightGBM
Ttest_indResult(statistic=-0.55440373938026655, pvalue=0.58341232517339003)

Ada Boost Xgboost
Ttest_indResult(statistic=-0.52714949086997409, pvalue=0.60196664627420793)

Ada Boost NN
Ttest_indResult(statistic=-1.3607671828268417, pvalue=0.18372330220674352)

Ada Boost UCB1
Ttest_indResult(statistic=-1.4760608860482116, pvalue=0.15034767873159582)

Ada Boost E Greedy
Ttest_indResult(statistic=-1.1785933542516112, pvalue=0.24782412178170049)

Ada Boost Decay E Gr.
Ttest_indResult(statistic=-1.1608190265567189, pvalue=0.25486951523694318)

Ada Boost Softmax
Ttest_indResult(statistic=-1.0624560382353923, pvalue=0.29650688897759103)

Ada Boost Decay SM
Ttest_indResult(statistic=-0.80346923773380319, pvalue=0.42802412796969069)

Ada Boost Tomp. Sampling
Ttest_indResult(statistic=-1.2001214421171955, pvalue=0.23948389295954292)

Ada Boost Hedge
Ttest_indResult(statistic=-0.99569087396931821, pvalue=0.32736412163034201)

Ada Boost EXP3
Ttest_indResult(statistic=-0.74295730815796013, pvalue=0.46328614566576143)

NB LDA
Ttest_indResult(statistic=0.5554996780874375, pvalue=0.58267211364636262)

NB QDA
Ttest_indResult(statistic=-0.20827715901923505, pvalue=0.83642016547531828)

NB Log. Reg.
Ttest_indResult(statistic=0.47318885308198955, pvalue=0.63950349819202956)

NB GP Class.
Ttest_indResult(statistic=-0.41000796016248631, pvalue=0.68471338285964789)

NB LightGBM
Ttest_indResult(statistic=0.0019395469860828407, pvalue=0.99846530657066879)

NB Xgboost
Ttest_indResult(statistic=0.035141241663040557, pvalue=0.97219990009550961)

NB NN
Ttest_indResult(statistic=-0.74869028585760766, pvalue=0.45987398452687533)

NB UCB1
Ttest_indResult(statistic=-0.8460837155804829, pvalue=0.40420449353993848)

NB E Greedy
Ttest_indResult(statistic=-0.5722377235552677, pvalue=0.57142462733272392)

NB Decay E Gr.
Ttest_indResult(statistic=-0.56200575659358698, pvalue=0.57828732944695804)

NB Softmax
Ttest_indResult(statistic=-0.45576864665863942, pvalue=0.65183837440893666)

NB Decay SM
Ttest_indResult(statistic=-0.21736889436650408, pvalue=0.8293924955094174)

NB Tomp. Sampling
Ttest_indResult(statistic=-0.57592590138855293, pvalue=0.56896093856705254)

NB Hedge
Ttest_indResult(statistic=-0.3844052385803689, pvalue=0.70338957927943424)

NB EXP3
Ttest_indResult(statistic=-0.15002815171436654, pvalue=0.8817465467624892)

LDA QDA
Ttest_indResult(statistic=-0.75697974660677769, pvalue=0.45496647766490839)

LDA Log. Reg.
Ttest_indResult(statistic=-0.084515819110785192, pvalue=0.93320781363163596)

LDA GP Class.
Ttest_indResult(statistic=-1.0041190022285222, pvalue=0.32335206267992656)

LDA LightGBM
Ttest_indResult(statistic=-0.59701582514273932, pvalue=0.5549762248530532)

LDA Xgboost
Ttest_indResult(statistic=-0.57141808114283488, pvalue=0.57197286873654452)

LDA NN
Ttest_indResult(statistic=-1.3581958518603419, pvalue=0.18452876491033476)

LDA UCB1
Ttest_indResult(statistic=-1.4654540575006512, pvalue=0.15320024518050987)

LDA E Greedy
Ttest_indResult(statistic=-1.185803492678305, pvalue=0.24500735064257073)

LDA Decay E Gr.
Ttest_indResult(statistic=-1.1696332232399687, pvalue=0.25135764765831459)

LDA Softmax
Ttest_indResult(statistic=-1.0757379821716313, pvalue=0.29062003465021008)

LDA Decay SM
Ttest_indResult(statistic=-0.83202462091783891, pvalue=0.4119693438291494)

LDA Tomp. Sampling
Ttest_indResult(statistic=-1.2044369770057308, pvalue=0.23783730730180369)

LDA Hedge
Ttest_indResult(statistic=-1.0122784817783961, pvalue=0.31950009136424407)

LDA EXP3
Ttest_indResult(statistic=-0.7747541208359443, pvalue=0.44454881899849974)

QDA Log. Reg.
Ttest_indResult(statistic=0.67643657940107704, pvalue=0.50394579400496897)

QDA GP Class.
Ttest_indResult(statistic=-0.18203630411057617, pvalue=0.85677870924178701)

QDA LightGBM
Ttest_indResult(statistic=0.2259516653802868, pvalue=0.82277136636963455)

QDA Xgboost
Ttest_indResult(statistic=0.26141839622051843, pvalue=0.79555629683260465)

QDA NN
Ttest_indResult(statistic=-0.50995868483656914, pvalue=0.61381194173499209)

QDA UCB1
Ttest_indResult(statistic=-0.601999889059461, pvalue=0.55169725154252114)

QDA E Greedy
Ttest_indResult(statistic=-0.3344880292016586, pvalue=0.74033863170867753)

QDA Decay E Gr.
Ttest_indResult(statistic=-0.3267034750695782, pvalue=0.74616066972047201)

QDA Softmax
Ttest_indResult(statistic=-0.21728598430946483, pvalue=0.82945651897203265)

QDA Decay SM
Ttest_indResult(statistic=0.015561300265179613, pvalue=0.98768741104152924)

QDA Tomp. Sampling
Ttest_indResult(statistic=-0.33233469383956604, pvalue=0.74194755712408456)

QDA Hedge
Ttest_indResult(statistic=-0.14400533665597387, pvalue=0.8864592747293446)

QDA EXP3
Ttest_indResult(statistic=0.085602536284712497, pvalue=0.93235114104225336)

Log. Reg. GP Class.
Ttest_indResult(statistic=-0.91820568796791147, pvalue=0.36583486431497148)

Log. Reg. LightGBM
Ttest_indResult(statistic=-0.50856349002081525, pvalue=0.6147780212373567)

Log. Reg. Xgboost
Ttest_indResult(statistic=-0.48174280124866742, pvalue=0.63348446666402602)

Log. Reg. NN
Ttest_indResult(statistic=-1.2718204803616877, pvalue=0.21320815989169214)

Log. Reg. UCB1
Ttest_indResult(statistic=-1.3782891642115165, pvalue=0.17830724696071232)

Log. Reg. E Greedy
Ttest_indResult(statistic=-1.0980641773905411, pvalue=0.28091167185082794)

Log. Reg. Decay E Gr.
Ttest_indResult(statistic=-1.0826236429317213, pvalue=0.28760087554536712)

Log. Reg. Softmax
Ttest_indResult(statistic=-0.98659240610111909, pvalue=0.33173324518372027)

Log. Reg. Decay SM
Ttest_indResult(statistic=-0.74243372021514487, pvalue=0.46359851260814722)

Log. Reg. Tomp. Sampling
Ttest_indResult(statistic=-1.1147669648292633, pvalue=0.27380140933068919)

Log. Reg. Hedge
Ttest_indResult(statistic=-0.92173017220328124, pvalue=0.36402294041491468)

Log. Reg. EXP3
Ttest_indResult(statistic=-0.68345655385014936, pvalue=0.49956343059883956)

GP Class. LightGBM
Ttest_indResult(statistic=0.4491386760211139, pvalue=0.65655958830979122)

GP Class. Xgboost
Ttest_indResult(statistic=0.4917522113192106, pvalue=0.62647353997847866)

GP Class. NN
Ttest_indResult(statistic=-0.35579863589426719, pvalue=0.72448118281467311)

GP Class. UCB1
Ttest_indResult(statistic=-0.4553423383657938, pvalue=0.65214151131733578)

GP Class. E Greedy
Ttest_indResult(statistic=-0.16166295824142304, pvalue=0.87265514853975279)

GP Class. Decay E Gr.
Ttest_indResult(statistic=-0.15470671761862703, pvalue=0.87808869011520896)

GP Class. Softmax
Ttest_indResult(statistic=-0.030513318731804712, pvalue=0.97585976898770688)

GP Class. Decay SM
Ttest_indResult(statistic=0.22454658510707484, pvalue=0.82385441421843053)

GP Class. Tomp. Sampling
Ttest_indResult(statistic=-0.15523809372426031, pvalue=0.87767341410769295)

GP Class. Hedge
Ttest_indResult(statistic=0.053126680184355894, pvalue=0.95798317515830056)

GP Class. EXP3
Ttest_indResult(statistic=0.30610395511927518, pvalue=0.76163976389644739)

LightGBM Xgboost
Ttest_indResult(statistic=0.036330097427954407, pvalue=0.97125982140328793)

LightGBM NN
Ttest_indResult(statistic=-0.82232531186351598, pvalue=0.41738009902118223)

LightGBM UCB1
Ttest_indResult(statistic=-0.93200655235667851, pvalue=0.35877361036574973)

LightGBM E Greedy
Ttest_indResult(statistic=-0.62996133454393011, pvalue=0.53348902720725788)

LightGBM Decay E Gr.
Ttest_indResult(statistic=-0.61751809205438213, pvalue=0.54155241009074362)

LightGBM Softmax
Ttest_indResult(statistic=-0.50332372728240138, pvalue=0.61841247017757084)

LightGBM Decay SM
Ttest_indResult(statistic=-0.24080179580401762, pvalue=0.81134684535853685)

LightGBM Temp. Sampling
Ttest_indResult(statistic=-0.63724247286096958, pvalue=0.52880048459465723)

LightGBM Hedge
Ttest_indResult(statistic=-0.4260483330426395, pvalue=0.67311399287689966)

LightGBM EXP3
Ttest_indResult(statistic=-0.16748810501898195, pvalue=0.86810994741702174)

Xgboost NN
Ttest_indResult(statistic=-0.87234612056034067, pvalue=0.38994821294546167)

Xgboost UCB1
Ttest_indResult(statistic=-0.98506690529298235, pvalue=0.33246965661581196)

Xgboost E Greedy
Ttest_indResult(statistic=-0.67744751461287223, pvalue=0.50331338568430439)

Xgboost Decay E Gr.
Ttest_indResult(statistic=-0.66412776612824109, pvalue=0.51168100702631514)

Xgboost Softmax
Ttest_indResult(statistic=-0.54948908829873022, pvalue=0.58673739577166728)

Xgboost Decay SM
Ttest_indResult(statistic=-0.28218391275697052, pvalue=0.7797401509490105)

Xgboost Temp. Sampling
Ttest_indResult(statistic=-0.68660489665039515, pvalue=0.49760493833623798)

Xgboost Hedge
Ttest_indResult(statistic=-0.4717774836890366, pvalue=0.64049903160533939)

Xgboost EXP3
Ttest_indResult(statistic=-0.20856844042398481, pvalue=0.83619479651578932)

NN UCB1
Ttest_indResult(statistic=-0.095904847049291447, pvalue=0.92423393719264446)

NN E Greedy
Ttest_indResult(statistic=0.20327150895462609, pvalue=0.84029530350917858)

NN Decay E Gr.
Ttest_indResult(statistic=0.20614488300789804, pvalue=0.83807036908058008)

NN Softmax
 Ttest_indResult(statistic=0.34346525960631008, pvalue=0.73364396640281049)

NN Decay SM
 Ttest_indResult(statistic=0.60290484350534423, pvalue=0.55110296407348847)

NN Temp. Sampling
 Ttest_indResult(statistic=0.22087642718857728, pvalue=0.82668506113821727)

NN Hedge
 Ttest_indResult(statistic=0.43593990961668366, pvalue=0.66600131312503785)

NN EXP3
 Ttest_indResult(statistic=0.69513674728798902, pvalue=0.4923191774940765)

UCB1 E Greedy
 Ttest_indResult(statistic=0.30415082414691436, pvalue=0.76311273546266389)

UCB1 Decay E Gr.
 Ttest_indResult(statistic=0.30578116809952655, pvalue=0.76188313426223186)

UCB1 Softmax
 Ttest_indResult(statistic=0.44871079852923279, pvalue=0.65686477886647809)

UCB1 Decay SM
 Ttest_indResult(statistic=0.71226893021136528, pvalue=0.48180140594874132)

UCB1 Temp. Sampling
 Ttest_indResult(statistic=0.32546608127016646, pvalue=0.74708752623669006)

UCB1 Hedge
 Ttest_indResult(statistic=0.54506565657565609, pvalue=0.58973799298344098)

UCB1 EXP3
 Ttest_indResult(statistic=0.80908183722065152, pvalue=0.42483864790236936)

E Greedy Decay E Gr.
 Ttest_indResult(statistic=0.0053442305516179268, pvalue=0.99577132158061787)

E Greedy Softmax
 Ttest_indResult(statistic=0.13874386267548908, pvalue=0.89057973073757624)

E Greedy Decay SM
 Ttest_indResult(statistic=0.40256293973130414, pvalue=0.69012403175573245)

E Greedy Temp. Sampling
 Ttest_indResult(statistic=0.011522173895478683, pvalue=0.9908831227852819)

E Greedy Hedge
Ttest_indResult(statistic=0.22877146201702028, pvalue=0.82059891522523642)

E Greedy EXP3
Ttest_indResult(statistic=0.49155574020307952, pvalue=0.62661081717350797)

Decay E Gr. Softmax
Ttest_indResult(statistic=0.13159771023529093, pvalue=0.89618110279101648)

Decay E Gr. Decay SM
Ttest_indResult(statistic=0.39227452829696768, pvalue=0.69762852646500995)

Decay E Gr. Tomp. Sampling
Ttest_indResult(statistic=0.0058675983009874467, pvalue=0.99535720605713673)

Decay E Gr. Hedge
Ttest_indResult(statistic=0.22032097476187781, pvalue=0.82711366676939102)

Decay E Gr. EXP3
Ttest_indResult(statistic=0.4798146684352439, pvalue=0.63483900405692306)

Softmax Decay SM
Ttest_indResult(statistic=0.26886207312766597, pvalue=0.78987624274343915)

Softmax Tomp. Sampling
Ttest_indResult(statistic=-0.131488295738727, pvalue=0.89626690825175048)

Softmax Hedge
Ttest_indResult(statistic=0.089005688385162401, pvalue=0.92966893149159169)

Softmax EXP3
Ttest_indResult(statistic=0.35634937857828608, pvalue=0.72407297553261274)

Decay SM Tomp. Sampling
Ttest_indResult(statistic=-0.40360169172865346, pvalue=0.68936811369293127)

Decay SM Hedge
Ttest_indResult(statistic=-0.18529244124364513, pvalue=0.85424673517863203)

Decay SM EXP3
Ttest_indResult(statistic=0.080775507356725909, pvalue=0.93615696256797098)

Tomp. Sampling Hedge
Ttest_indResult(statistic=0.22444449731522798, pvalue=0.82393311812201653)

Tomp. Sampling EXP3
Ttest_indResult(statistic=0.49581359542551828, pvalue=0.6236388435240402)

```
Hedge EXP3
Ttest_indResult(statistic=0.27252755168395165, pvalue=0.78708350527957582)
```

```
In [19]: for pair, p in zip(new_model_pairs, pvalueList):
        if p < 0.05:
            print('The pvalue between',pair, 'is', p, '< 0.05 then',
                  emoji.emojize('REJECT the NULL Hypothesis :thumbs_up_sign:'))
        else:
            print('The pvalue between',pair, 'is', p, '> 0.05 then',
                  emoji.emojize('FAIL to REJECT the NULL Hypothesis :thumbs_down_sign:'))
```

```
The pvalue between ('Knn', 'NN') is 0.824755960285 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('Knn', 'UCB1') is 0.745655671119 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('Knn', 'E Greedy') is 0.987942866094 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('Knn', 'Decay E Gr.') is 0.992428513793 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('Knn', 'Softmax') is 0.899887527007 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('Knn', 'Decay SM') is 0.693838856853 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('Knn', 'Tomp. Sampling') is 0.996909124489 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('Knn', 'Hedge') is 0.828006060719 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('Knn', 'EXP3') is 0.628448953522 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('LSVM', 'NN') is 0.225184347448 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('LSVM', 'UCB1') is 0.189157259564 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('LSVM', 'E Greedy') is 0.295245746546 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('LSVM', 'Decay E Gr.') is 0.301966162385 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('LSVM', 'Softmax') is 0.347693260266 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('LSVM', 'Decay SM') is 0.482246403255 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('LSVM', 'Tomp. Sampling') is 0.288335997526 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('LSVM', 'Hedge') is 0.381049161822 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('LSVM', 'EXP3') is 0.519160363475 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('SVM', 'NN') is 0.890612847765 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('SVM', 'UCB1') is 0.964298309455 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('SVM', 'E Greedy') is 0.733176910857 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('SVM', 'Decay E Gr.') is 0.732296834691 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('SVM', 'Softmax') is 0.629919789471 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('SVM', 'Decay SM') is 0.461495667328 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('SVM', 'Tomp. Sampling') is 0.716835951686 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('SVM', 'Hedge') is 0.564849883923 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('SVM', 'EXP3') is 0.406637094881 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('DT_gini', 'NN') is 0.929486528857 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('DT_gini', 'UCB1') is 0.848438413381 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('DT_gini', 'E Greedy') is 0.901465466803 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('DT_gini', 'Decay E Gr.') is 0.898379458569 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('DT_gini', 'Softmax') is 0.787204909825 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('DT_gini', 'Decay SM') is 0.588109244722 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('DT_gini', 'Tomp. Sampling') is 0.888611725961 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('DT_gini', 'Hedge') is 0.714284046103 > 0.05 then FAIL to REJECT the NULL Hypothesis
The pvalue between ('DT_gini', 'EXP3') is 0.523952795033 > 0.05 then FAIL to REJECT the NULL Hypothesis
```

The pvalue between ('DT_entorpy', 'NN') is 0.791983671644 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('DT_entorpy', 'UCB1') is 0.713226194897 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('DT_entorpy', 'E Greedy') is 0.953875759966 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('DT_entorpy', 'Decay E Gr.') is 0.958789759925 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('DT_entorpy', 'Softmax') is 0.934250279489 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('DT_entorpy', 'Decay SM') is 0.725594562973 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('DT_entorpy', 'Tomp. Sampling') is 0.961716735646 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('DT_entorpy', 'Hedge') is 0.86245619211 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('DT_entorpy', 'EXP3') is 0.65964398726 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging Knn', 'NN') is 0.241955895047 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging Knn', 'UCB1') is 0.201954069974 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging Knn', 'E Greedy') is 0.319645342984 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging Knn', 'Decay E Gr.') is 0.327086294462 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging Knn', 'Softmax') is 0.37794490608 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging Knn', 'Decay SM') is 0.526895663417 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging Knn', 'Tomp. Sampling') is 0.311964152805 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging Knn', 'Hedge') is 0.415137424932 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging Knn', 'EXP3') is 0.56810856 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging DT', 'NN') is 0.208605902048 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging DT', 'UCB1') is 0.170588441482 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging DT', 'E Greedy') is 0.281744161271 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging DT', 'Decay E Gr.') is 0.289527247583 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging DT', 'Softmax') is 0.337260323124 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging DT', 'Decay SM') is 0.484793823488 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging DT', 'Tomp. Sampling') is 0.272748702238 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging DT', 'Hedge') is 0.372605588818 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Bagging DT', 'EXP3') is 0.524959203377 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Random Forest', 'NN') is 0.565572948646 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Random Forest', 'UCB1') is 0.494718637423 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Random Forest', 'E Greedy') is 0.707821765847 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Random Forest', 'Decay E Gr.') is 0.715294116292 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Random Forest', 'Softmax') is 0.809731063144 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Random Forest', 'Decay SM') is 0.977343471057 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Random Forest', 'Tomp. Sampling') is 0.707515234525 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Random Forest', 'Hedge') is 0.875525851773 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Random Forest', 'EXP3') is 0.912392213952 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Ada Boost', 'NN') is 0.183723302207 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Ada Boost', 'UCB1') is 0.150347678732 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Ada Boost', 'E Greedy') is 0.247824121782 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Ada Boost', 'Decay E Gr.') is 0.254869515237 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Ada Boost', 'Softmax') is 0.296506888978 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Ada Boost', 'Decay SM') is 0.42802412797 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Ada Boost', 'Tomp. Sampling') is 0.23948389296 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Ada Boost', 'Hedge') is 0.32736412163 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Ada Boost', 'EXP3') is 0.463286145666 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('NB', 'NN') is 0.459873984527 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('NB', 'UCB1') is 0.40420449354 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('NB', 'E Greedy') is 0.571424627333 > 0.05 then FAIL to REJECT the NULL Hypothesis

The pvalue between ('NB', 'Decay E Gr.') is 0.578287329447 > 0.05 then FAIL to REJECT the NULL H
 The pvalue between ('NB', 'Softmax') is 0.651838374409 > 0.05 then FAIL to REJECT the NULL Hypot
 The pvalue between ('NB', 'Decay SM') is 0.829392495509 > 0.05 then FAIL to REJECT the NULL Hypo
 The pvalue between ('NB', 'Tomp. Sampling') is 0.568960938567 > 0.05 then FAIL to REJECT the NUL
 The pvalue between ('NB', 'Hedge') is 0.703389579279 > 0.05 then FAIL to REJECT the NULL Hypothe
 The pvalue between ('NB', 'EXP3') is 0.881746546762 > 0.05 then FAIL to REJECT the NULL Hypothes
 The pvalue between ('LDA', 'NN') is 0.18452876491 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('LDA', 'UCB1') is 0.153200245181 > 0.05 then FAIL to REJECT the NULL Hypothe
 The pvalue between ('LDA', 'E Greedy') is 0.245007350643 > 0.05 then FAIL to REJECT the NULL Hyp
 The pvalue between ('LDA', 'Decay E Gr.') is 0.251357647658 > 0.05 then FAIL to REJECT the NULL
 The pvalue between ('LDA', 'Softmax') is 0.29062003465 > 0.05 then FAIL to REJECT the NULL Hypot
 The pvalue between ('LDA', 'Decay SM') is 0.411969343829 > 0.05 then FAIL to REJECT the NULL Hyp
 The pvalue between ('LDA', 'Tomp. Sampling') is 0.237837307302 > 0.05 then FAIL to REJECT the NU
 The pvalue between ('LDA', 'Hedge') is 0.319500091364 > 0.05 then FAIL to REJECT the NULL Hypoth
 The pvalue between ('LDA', 'EXP3') is 0.444548818998 > 0.05 then FAIL to REJECT the NULL Hypothe
 The pvalue between ('QDA', 'NN') is 0.613811941735 > 0.05 then FAIL to REJECT the NULL Hypothesi
 The pvalue between ('QDA', 'UCB1') is 0.551697251543 > 0.05 then FAIL to REJECT the NULL Hypothe
 The pvalue between ('QDA', 'E Greedy') is 0.740338631709 > 0.05 then FAIL to REJECT the NULL Hyp
 The pvalue between ('QDA', 'Decay E Gr.') is 0.74616066972 > 0.05 then FAIL to REJECT the NULL H
 The pvalue between ('QDA', 'Softmax') is 0.829456518972 > 0.05 then FAIL to REJECT the NULL Hypo
 The pvalue between ('QDA', 'Decay SM') is 0.987687411042 > 0.05 then FAIL to REJECT the NULL Hyp
 The pvalue between ('QDA', 'Tomp. Sampling') is 0.741947557124 > 0.05 then FAIL to REJECT the NU
 The pvalue between ('QDA', 'Hedge') is 0.886459274729 > 0.05 then FAIL to REJECT the NULL Hypoth
 The pvalue between ('QDA', 'EXP3') is 0.932351141042 > 0.05 then FAIL to REJECT the NULL Hypothe
 The pvalue between ('Log. Reg.', 'NN') is 0.213208159892 > 0.05 then FAIL to REJECT the NULL Hyp
 The pvalue between ('Log. Reg.', 'UCB1') is 0.178307246961 > 0.05 then FAIL to REJECT the NULL H
 The pvalue between ('Log. Reg.', 'E Greedy') is 0.280911671851 > 0.05 then FAIL to REJECT the NU
 The pvalue between ('Log. Reg.', 'Decay E Gr.') is 0.287600875545 > 0.05 then FAIL to REJECT the
 The pvalue between ('Log. Reg.', 'Softmax') is 0.331733245184 > 0.05 then FAIL to REJECT the NUL
 The pvalue between ('Log. Reg.', 'Decay SM') is 0.463598512608 > 0.05 then FAIL to REJECT the NU
 The pvalue between ('Log. Reg.', 'Tomp. Sampling') is 0.273801409331 > 0.05 then FAIL to REJECT
 The pvalue between ('Log. Reg.', 'Hedge') is 0.364022940415 > 0.05 then FAIL to REJECT the NULL
 The pvalue between ('Log. Reg.', 'EXP3') is 0.499563430599 > 0.05 then FAIL to REJECT the NULL H
 The pvalue between ('GP Class.', 'NN') is 0.724481182815 > 0.05 then FAIL to REJECT the NULL Hyp
 The pvalue between ('GP Class.', 'UCB1') is 0.652141511317 > 0.05 then FAIL to REJECT the NULL H
 The pvalue between ('GP Class.', 'E Greedy') is 0.87265514854 > 0.05 then FAIL to REJECT the NUL
 The pvalue between ('GP Class.', 'Decay E Gr.') is 0.878088690115 > 0.05 then FAIL to REJECT the
 The pvalue between ('GP Class.', 'Softmax') is 0.975859768988 > 0.05 then FAIL to REJECT the NUL
 The pvalue between ('GP Class.', 'Decay SM') is 0.823854414218 > 0.05 then FAIL to REJECT the NU
 The pvalue between ('GP Class.', 'Tomp. Sampling') is 0.877673414108 > 0.05 then FAIL to REJECT
 The pvalue between ('GP Class.', 'Hedge') is 0.957983175158 > 0.05 then FAIL to REJECT the NULL
 The pvalue between ('GP Class.', 'EXP3') is 0.761639763896 > 0.05 then FAIL to REJECT the NULL H
 The pvalue between ('LightGBM', 'NN') is 0.417380099021 > 0.05 then FAIL to REJECT the NULL Hypo
 The pvalue between ('LightGBM', 'UCB1') is 0.358773610366 > 0.05 then FAIL to REJECT the NULL Hy
 The pvalue between ('LightGBM', 'E Greedy') is 0.533489027207 > 0.05 then FAIL to REJECT the NUL
 The pvalue between ('LightGBM', 'Decay E Gr.') is 0.541552410091 > 0.05 then FAIL to REJECT the
 The pvalue between ('LightGBM', 'Softmax') is 0.618412470178 > 0.05 then FAIL to REJECT the NULL
 The pvalue between ('LightGBM', 'Decay SM') is 0.811346845359 > 0.05 then FAIL to REJECT the NUL

The pvalue between ('LightGBM', 'Tomp. Sampling') is 0.528800484595 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('LightGBM', 'Hedge') is 0.673113992877 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('LightGBM', 'EXP3') is 0.868109947417 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Xgboost', 'NN') is 0.389948212945 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Xgboost', 'UCB1') is 0.332469656616 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Xgboost', 'E Greedy') is 0.503313385684 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Xgboost', 'Decay E Gr.') is 0.511681007026 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Xgboost', 'Softmax') is 0.586737395772 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Xgboost', 'Decay SM') is 0.779740150949 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Xgboost', 'Tomp. Sampling') is 0.497604938336 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Xgboost', 'Hedge') is 0.640499031605 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Xgboost', 'EXP3') is 0.836194796516 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('NN', 'UCB1') is 0.924233937193 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('NN', 'E Greedy') is 0.840295303509 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('NN', 'Decay E Gr.') is 0.838070369081 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('NN', 'Softmax') is 0.733643966403 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('NN', 'Decay SM') is 0.551102964073 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('NN', 'Tomp. Sampling') is 0.826685061138 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('NN', 'Hedge') is 0.666001313125 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('NN', 'EXP3') is 0.492319177494 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('UCB1', 'E Greedy') is 0.763112735463 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('UCB1', 'Decay E Gr.') is 0.761883134262 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('UCB1', 'Softmax') is 0.656864778866 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('UCB1', 'Decay SM') is 0.481801405949 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('UCB1', 'Tomp. Sampling') is 0.747087526237 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('UCB1', 'Hedge') is 0.589737992983 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('UCB1', 'EXP3') is 0.424838647902 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('E Greedy', 'Decay E Gr.') is 0.995771321581 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('E Greedy', 'Softmax') is 0.890579730738 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('E Greedy', 'Decay SM') is 0.690124031756 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('E Greedy', 'Tomp. Sampling') is 0.990883122785 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('E Greedy', 'Hedge') is 0.820598915225 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('E Greedy', 'EXP3') is 0.626610817174 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Decay E Gr.', 'Softmax') is 0.896181102791 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Decay E Gr.', 'Decay SM') is 0.697628526465 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Decay E Gr.', 'Tomp. Sampling') is 0.995357206057 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Decay E Gr.', 'Hedge') is 0.827113666769 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Decay E Gr.', 'EXP3') is 0.634839004057 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Softmax', 'Decay SM') is 0.789876242743 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Softmax', 'Tomp. Sampling') is 0.896266908252 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Softmax', 'Hedge') is 0.929668931492 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Softmax', 'EXP3') is 0.724072975533 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Decay SM', 'Tomp. Sampling') is 0.689368113693 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Decay SM', 'Hedge') is 0.854246735179 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Decay SM', 'EXP3') is 0.936156962568 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Tomp. Sampling', 'Hedge') is 0.823933118122 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Tomp. Sampling', 'EXP3') is 0.623638843524 > 0.05 then FAIL to REJECT the NULL Hypothesis
 The pvalue between ('Hedge', 'EXP3') is 0.78708350528 > 0.05 then FAIL to REJECT the NULL Hypothesis