```python
# -*- coding: utf-8 -*-
"""
Created on Tue Mar  5 19:13:14 2019
ME EN 2550
Homework 5
@author: Ryan Dalby
"""

import numpy as np
import statsmodels.stats.api as sm
import scipy.stats as stats
import matplotlib.pyplot as plt


print("B-1:")

catalyst1 = np.array([57.9, 66.2, 65.4, 65.4, 65.2, 62.6, 67.6, 63.7, 67.2, 71.
catalyst2 = np.array([66.4, 71.7, 70.3, 69.3, 64.8, 69.6, 68.6, 69.4, 65.3, 68.
catalyst1data = sm.DescrStatsW(catalyst1)
catalyst2data = sm.DescrStatsW(catalyst2)
b1statdata = catalyst1data.get_compare(catalyst2data)
_,b1pval,_ = b1statdata.ttest_ind()
b195CI = b1statdata.tconfint_diff()
print("a) 95% CI: {} \n P-value = {}".format(b195CI, b1pval))

print("b) Yes there is evidence that the mean active concentrations depend on t

b1truestd = 3 #g/L
standardizedb1effectsize = 5/b1truestd
b1power = sm.tt_ind_solve_power(effect_size=standardizedb1effectsize, nobs1=cat
print("c) power = {}".format(b1power))

figb1, (b1ax1,b1ax2) = plt.subplots(nrows=1, ncols=2,figsize = (10,5))
stats.probplot(catalyst1, plot=b1ax1)
b1ax1.set_title("Catalyst 1 Probability Plot")
stats.probplot(catalyst2, plot=b1ax2)
b1ax2.set_title("Catalyst 2 Probability Plot")
plt.show()
print("d) The sample sizes used by the experimenter to appear to be adequate to


print()
print("B-2:")

type1 = np.array([206, 188, 205, 187, 194, 193, 207, 185, 189, 213, 192, 210, 1
type2 = np.array([177, 197, 206, 201, 180, 176, 185, 200, 197, 192, 198, 188, 1
```

```
type1data = sm.DescrStatsW(type1)
type2data = sm.DescrStatsW(type2)
figb2, ((b2ax1,b2ax2),(b2ax3,b2ax4)) = plt.subplots(nrows=2, ncols=2,figsize =
stats.probplot(type1, plot=b2ax1)
b2ax1.set_title("Type 1 Probability Plot")
stats.probplot(type2, plot=b2ax2)
b2ax2.set_title("Type 2 Probability Plot")
b2ax3.set_title("Type 1 Boxplot")
b2ax3.boxplot(type1)
b2ax4.set_title("Type 2 Boxplot")
b2ax4.boxplot(type2)
plt.show()
print("a) Yes, both data sets are nearly normal since they are approximately no

b2typedata = type1data.get_compare(type2data)
_,b2pval,_ = b2typedata.ttest_ind(alternative='larger')
print("b) Fail to reject the null hypothesis(That the temperature under load fo

standardizedb2effectsize = 5/type1data.std
b2power = sm.tt_ind_solve_power(effect_size=standardizedb2effectsize , nobs1=ty
print("c) The given sample size is inadequate to have power of .90 for the give


print()
print("B-3")
low = np.array([242, 249, 235, 250, 254, 244, 258, 311, 237, 261, 314, 252])
high = np.array([302, 421, 419, 399, 317, 311, 350, 363, 392, 367, 301, 302])
lowdata = sm.DescrStatsW(low)
highdata = sm.DescrStatsW(high)
highlowcomparedata = highdata.get_compare(lowdata)
_,b3pval,_ = highlowcomparedata.ttest_ind(alternative='larger')
print("a) Yes there is evidence to support the claim that the mean grinding for
b395CI = highlowcomparedata.tconfint_diff()
print("b) 95% CI: {}".format(b395CI))
figb3, (b3ax1,b3ax2) = plt.subplots(nrows=1, ncols=2,figsize = (10,5))
stats.probplot(low, plot=b3ax1)
b3ax1.set_title("Low vibration Probability Plot")
stats.probplot(high, plot=b3ax2)
b3ax2.set_title("High vibration Probability Plot")
plt.show()
print("c) The probability plot for low vibration indicates that the data may no
```