*#%%*import numpy as np  
import matplotlib.pyplot as plt  
import statsmodels as stat  
import seaborn as sea  
import pandas as pd  
from pandas.plotting import scatter\_matrix  
import scipy.stats as stats  
import statsmodels.api as sm  
*#from day\_1 import mean\_CI\_model, mean\_CI\_data, mean\_PI\_model, mean\_PI\_data*import pandas.plotting as pd\_plot  
import scipy as sc  
  
*#%%*def mean\_CI\_data(data, confidence=0.95):  
 a = 1.0 \* np.array(data)  
 n = len(a)  
 m, std = np.mean(a), np.std(a)  
 *#h = se \* sc.stats.t.ppf((1 + confidence) / 2., n - 1)* h = std \* 1.96 / np.sqrt(n)  
 return m, m - h, m + h  
  
  
def mean\_CI\_model(mu, std, n, confidence=0.95):  
 m = mu  
 h = stats.norm.pdf((1 - confidence) / 2) \* std / np.sqrt(n)  
 return m, m - h, m + h  
  
  
def mean\_PI\_data(data, confidence=0.95):  
 a = 1.0 \* np.array(data)  
 n = len(a)  
 m, std = np.mean(a), np.std(a)  
 *#h = std \* sc.stats.t.ppf((1 + confidence) / 2., n - 1)* h = std \* 1.96  
 return m, m - h, m + h  
  
  
def mean\_PI\_model(mu, std, confidence=0.95):  
 m = mu  
 h = stats.norm.pdf((1 - confidence) / 2) \* std  
 return m, m - h, m + h  
  
*#%% md  
  
# Exercise 2.1*  
  
The fish oil supplement data, fishoil.dta, contains the difference in systolic blood  
pressure. In this exercise we will consider the possible effect of fish oil supplement on the  
increase in systolic blood pressure and go through an analysis similar to the one you saw  
at the lecture  
  
*#%% md*  
1) Make Q-Q plots of the difference in systolic blood pressure for each of the two  
groups. What are your comments to the plots  
  
*#%%*fish\_oil = pd.read\_csv(**'data/fishoil.csv'**, sep=**','**, na\_values=**"."**)  
print(fish\_oil)  
  
fish\_oil\_C = fish\_oil.loc[fish\_oil[**'group'**] != **'Fish oil'**]  
fish\_oil\_F = fish\_oil.loc[fish\_oil[**'group'**] == **'Fish oil'**]  
  
*## histogram*plt.rcParams.update({**'font.size'**: 10})  
fig, ax = plt.subplots(1, 1)  
  
mu, std = stats.norm.fit(fish\_oil\_C[**'systol'**])  
x = np.linspace(-80, 80)  
pdf\_data = stats.norm.pdf(x, mu, std)  
ax.plot(x, pdf\_data, color=**'blue'**, label=**'Control'**)  
ax.hist(fish\_oil\_C[**'systol'**], bins=20, density=**'True'**, color=**'blue'**, alpha=0.7)  
  
mu, std = stats.norm.fit(fish\_oil\_F[**'systol'**])  
pdf\_data = stats.norm.pdf(x, mu, std)  
ax.plot(x, pdf\_data, color=**'orange'**, label=**'Fish oil'**)  
ax.hist(fish\_oil\_F[**'systol'**], bins=20, density=**'True'**, color=**'orange'**, alpha=0.7)  
  
ax.set\_xlabel(**'Systolic pressure'**)  
ax.set\_ylabel(**'Density'**)  
ax.legend()  
ax.grid()  
plt.show()  
  
*## QQ plot*plt.rcParams.update({**'font.size'**: 10})  
sm.qqplot(fish\_oil\_F[**'systol'**], fit=True, line=**'45'**)  
plt.title(**'Fish oil group'**)  
plt.grid()  
plt.show()  
  
plt.rcParams.update({**'font.size'**: 10})  
sm.qqplot(fish\_oil\_C[**'systol'**], fit=True, line=**'45'**)  
plt.title(**'Control group'**)  
plt.grid()  
plt.show()  
  
*#%% md*  
The data seems to be well described by a normal distribution.  
The "Fish oil" group seems to have a slightly higher difference in systolic blood pressure,  
but we don't know if it's significant yet.  
  
*#%% md*  
2) Make a short description of the difference in systolic blood pressure using  
bysort group:summarize systol, detail and ttest systol, by(group).  
Comment on the descriptives (not the test!).  
  
*#%%*print(**'Describing the Control group:'**)  
print(fish\_oil\_C.describe(), **'**\n**'**)  
print(**'Confidence interval: '**, mean\_CI\_data(fish\_oil\_C[**'systol'**])[1:])  
print(**'Prediction interval: '**, mean\_PI\_data(fish\_oil\_C[**'systol'**])[1:])  
print(**'1-sided ttest: '**, stats.ttest\_1samp(fish\_oil\_C[**'systol'**], np.mean(fish\_oil\_C[**'systol'**])))  
  
print(**'Describing the fish oil group:'**)  
print(fish\_oil\_F.describe())  
print(**'Confidence interval: '**, mean\_CI\_data(fish\_oil\_F[**'systol'**])[1:])  
print(**'Prediction interval: '**, mean\_PI\_data(fish\_oil\_F[**'systol'**])[1:])  
print(**'1-sided ttest: '**, stats.ttest\_1samp(fish\_oil\_F[**'systol'**], np.mean(fish\_oil\_F[**'systol'**])))  
  
*#%% md*  
3) Test the hypothesis of the common standard deviation in the two groups.  
  
*#%%  
  
# scipy.stats.ttest\_ind: two-sided test for the null hypothesis that 2 independent samples  
# have identical average (expected) values. It assumes that the populations have IDENTICAL  
# variances by default  
#print(stats.ttest\_ind(fish\_oil\_C['systol'], fish\_oil\_F['systol']))  
  
# We need to test that! Testing the hypothesis std\_control = std\_fish\_oil is done by considering  
# the ratio between the two estimated standard deviations  
# F\_obs = [largest observed std/ smallest observed std]^2*std\_f = np.std(fish\_oil\_F[**'systol'**])  
std\_c = np.std(fish\_oil\_C[**'systol'**])  
print(**'Control group: std ='**, std\_c)  
print(**'Fish oil group: std ='**, std\_f)  
  
if std\_c >= std\_f:  
 F\_obs = (std\_c/std\_f)\*\*2  
else:  
 F\_obs = (std\_f/std\_c)\*\*2  
  
print(**'F\_obs ='**, F\_obs)  
p\_value = stats.f.cdf(F\_obs, len(fish\_oil\_C[**'systol'**]) - 1, len(fish\_oil\_F[**'systol'**]) - 1)  
print(**'p-value ='**, p\_value)  
  
*#%% md*  
The observed variance F\_obs is 1.6% higher for the control group. The p-value of 55%  
means that the difference may be due to sampling variance -> we can accept that std\_c = std\_f  
  
*#%% md*  
4) Return to the output from the t-test command and write a conclusion on the  
possible effect of fish oil on the change in systolic blood pressure during pregnancy.  
The conclusion should contain information on size of the possible effect, whether or  
not it is statistical significant and a discussion on the validity of the assumptions  
behind the statistical analysis.  
  
*#%%  
  
# now that we tested, and assumed correct, the hypothesis of the common standard deviation  
# in the two groups, we can do a t-test:*print(**'1-sided ttest: '**, stats.ttest\_ind(fish\_oil\_F[**'systol'**],fish\_oil\_C[**'systol'**]))  
  
*#%% md*  
We found a very high p-value when comparing the two samples, p=35%. Therefore, we can  
conclude that the difference observed is NOT statistically significant. We cannot reject the  
null hypothesis  
  
*#%% md  
  
## Exercise 2.2*  
In the experiment above two women in the control group had a decrease in the systolic  
blood pressure of more than 50 mmHg.  
  
*#%% md*  
1) Exclude these two women and repeat the analysis above. How does this affect your  
conclusions?  
  
*#%%*fish\_oil\_C\_rem\_outl = []  
for i in range(0, len(fish\_oil\_C[**'systol'**])):  
 if fish\_oil\_C[**'systol'**][i] < -50:  
 fish\_oil\_C\_rem\_outl.append(fish\_oil\_C[**'systol'**][i])  
  
*# testing the hypothesis of common std*std\_f = np.std(fish\_oil\_F[**'systol'**])  
std\_c = np.std(fish\_oil\_C\_rem\_outl)  
print(**'Control group: std ='**, std\_c)  
print(**'Fish oil group: std ='**, std\_f)  
  
if std\_c >= std\_f:  
 F\_obs = (std\_c/std\_f)\*\*2  
else:  
 F\_obs = (std\_f/std\_c)\*\*2  
  
print(**'F\_obs ='**, F\_obs)  
p\_value = stats.f.cdf(F\_obs, len(fish\_oil\_C\_rem\_outl) - 1, len(fish\_oil\_F[**'systol'**]) - 1)  
print(**'p-value ='**, p\_value)  
  
*# High p-value -> we accept that the difference is due to sampling variance  
  
#Doing a t-test*print(**'1-sided ttest: '**, stats.ttest\_ind(fish\_oil\_F[**'systol'**],fish\_oil\_C\_rem\_outl))  
  
*#%% md*  
Removing the two outliers we get a very small p-value. This would mean that the difference  
observed IS statistically significant. But we are removing data points, so let's continue  
the analysis...  
  
*#%% md*  
2) The approach of excluding the potential ourliers is easy, but we are then no longer  
analyzing the complete data. Another approach is to estimate the standard error using a  
resampling technique called bootstrap. Here we create similar studies by sampling data  
from each of the two groups with replacement and compute estimated differences in  
2 means in each study. The variation between these estimated means is an estimate of the  
standard error. The bootstrap technique does not require that the observations follow a  
normal distribution. The command bootstrap dif=(r(mu\_2)-r(mu\_1)), reps(1000): ttest systol , by(group)  
Compare the results of the bootstrap to the t-test used in Exercise 2.1.  
  
  
*#%%*from bootstrap\_stat import bootstrap\_stat as bp  
comb = [fish\_oil\_F[**'systol'**], fish\_oil\_C[**'systol'**]]  
dist = bp.EmpiricalDistribution(comb)  
  
def statistic(comb):  
 return np.mean(comb[0]) - np.mean(comb[1])  
  
print(**'Standard error = '**, bp.standard\_error(dist, statistic, B=1000))  
print(bp.t\_interval(dist, statistic, statistic(comb)))  
*# the confidence intervals are not working - see why  
#ci\_low, ci\_high = bp.bcanon\_interval(dist, statistic, comb, alpha=0.025)  
#print(ci\_low, ci\_high)  
  
#%% md  
  
## Exercise 2.3*  
  
The data set hp.dta contains data on the heart period (the average time in ms between two  
consecutive heart beats) during night and day for a group of healthy persons divided into  
physically active and passive persons.  
  
*#%% md*  
1) Describe the heart period during day for the ‘active’ and the ‘passive’ persons  
  
*#%%*Heart = pd.read\_csv(**'data/hp.csv'**, sep=**','**, na\_values=**"."**)  
print(Heart)  
Heart\_active = Heart.loc[Heart[**'group'**] == **'active'**]  
Heart\_passive = Heart.loc[Heart[**'group'**] == **'passive'**]  
  
print(**'Active group**\n**'**, Heart\_active.describe(), **'**\n \n**'**)  
print(**'Passive group**\n**'**, Heart\_passive.describe())  
  
*## QQ plot*plt.rcParams.update({**'font.size'**: 10})  
sm.qqplot(Heart\_active[**'day'**], fit=True, line=**'45'**)  
plt.title(**'Active person, day'**)  
plt.grid()  
plt.show()  
  
plt.rcParams.update({**'font.size'**: 10})  
sm.qqplot(Heart\_active[**'night'**], fit=True, line=**'45'**)  
plt.title(**'Active person, night'**)  
plt.grid()  
plt.show()  
  
plt.rcParams.update({**'font.size'**: 10})  
sm.qqplot(Heart\_passive[**'day'**], fit=True, line=**'45'**)  
plt.title(**'Passive person, day'**)  
plt.grid()  
plt.show()  
  
plt.rcParams.update({**'font.size'**: 10})  
sm.qqplot(Heart\_passive[**'night'**], fit=True, line=**'45'**)  
plt.title(**'Passive person, night'**)  
plt.grid()  
plt.show()  
  
  
*#%% md*  
2) Calculate a 95% prediction interval for the heart period for a passive person  
during day. Do the same for the active group. Compare the intervals and comment  
on whether these are valid 95%-prediction intervals.  
  
*#%%  
  
# Log transform*Heart\_active[**'log day'**] = np.log(Heart\_active[**'day'**])  
Heart\_passive[**'log day'**] = np.log(Heart\_passive[**'day'**])  
  
plt.rcParams.update({**'font.size'**: 10})  
fig, ax = plt.subplots(1, 2)  
mu, std = stats.norm.fit(Heart\_passive[**'day'**])  
x = np.linspace(500, 1200)  
pdf\_data = stats.norm.pdf(x, mu, std)  
ax[0].plot(x, pdf\_data, color=**'blue'**, label=**'Passive'**)  
ax[0].hist(Heart\_passive[**'day'**], bins=10, density=**'True'**, color=**'blue'**, alpha=0.7)  
  
mu, std = stats.norm.fit(Heart\_active[**'day'**])  
pdf\_data = stats.norm.pdf(x, mu, std)  
ax[0].plot(x, pdf\_data, color=**'orange'**, label=**'Active'**)  
ax[0].hist(Heart\_active[**'day'**], bins=10, density=**'True'**, color=**'orange'**, alpha=0.7)  
ax[0].legend()  
ax[0].grid()  
  
mu, std = stats.norm.fit(Heart\_passive[**'log day'**])  
x = np.linspace(np.log(500), np.log(1200))  
pdf\_data = stats.norm.pdf(x, mu, std)  
ax[1].plot(x, pdf\_data, color=**'blue'**, label=**'Passive'**)  
ax[1].hist(Heart\_passive[**'log day'**], bins=10, density=**'True'**, color=**'blue'**, alpha=0.7)  
  
mu, std = stats.norm.fit(Heart\_active[**'log day'**])  
pdf\_data = stats.norm.pdf(x, mu, std)  
ax[1].plot(x, pdf\_data, color=**'orange'**, label=**'Active'**)  
ax[1].hist(Heart\_active[**'log day'**], bins=10, density=**'True'**, color=**'orange'**, alpha=0.7)  
  
ax[1].set\_xlabel(**'log Heart rate'**)  
ax[1].set\_ylabel(**'Density'**)  
ax[1].legend()  
ax[1].grid()  
plt.show()  
  
print(**'Passive person, daytime: 95% PI = '**, mean\_PI\_data(Heart\_passive[**'day'**]))  
print(**'Active person, daytime: 95% PI = '**, mean\_PI\_data(Heart\_active[**'day'**]))  
print(**'Log transform Passive person, daytime: 95% PI = '**, np.exp(mean\_PI\_data(Heart\_passive[**'log day'**])))  
print(**'Log transform Active person, daytime: 95% PI = '**, np.exp(mean\_PI\_data(Heart\_active[**'log day'**])))  
  
*#%% md*  
The 95% PI is not valid for the non-transformed sample, as the data cannot be well approximated by a normal  
distribution - especially the "passive" group. The log transformed analysis  
is a little better, but it still doesn't look completely correct  
  
*#%% md*  
3) Compare the heart period during day in the two groups. The comparison should  
(among other things) include a non-parametric test of no difference between the two  
groups.  
  
*#%%  
#First, we test the std\_p = std\_a hypothesis for the log transformed data:*std\_p = np.std(np.log(Heart\_passive[**'day'**]))  
std\_a = np.std(np.log(Heart\_active[**'day'**]))  
  
if std\_a >= std\_p:  
 F\_obs = (std\_a/std\_p)\*\*2  
else:  
 F\_obs = (std\_p/std\_a)\*\*2  
  
print(**'F\_obs ='**, F\_obs)  
p\_value = stats.f.cdf(F\_obs, len(Heart\_active[**'day'**]) - 1, len(Heart\_passive[**'day'**]) - 1)  
print(**'(std\_p==std\_a) p-value ='**, p\_value)  
*# High p-value -> we accept that the difference is due to sampling variance  
  
#Doing a t-test*print(**'1-sided ttest: '**, stats.ttest\_ind(np.log(Heart\_active[**'day'**]),  
 np.log(Heart\_passive[**'day'**])))  
  
*# we will use the Mann-Whitney U test: a non-parametric test for the  
# null hypothesis*mann\_whitney = stats.mannwhitneyu(Heart\_active[**'day'**], Heart\_passive[**'day'**])  
print(**'Mann-Whitney test: '**, mann\_whitney)  
  
*#%% md*First we tested to see if the variance of the two groups is the same; this has a high  
p-value, so we can accept that the difference is due to sampling variance. Additionally,  
when testing for the null hypothesis (of the two data sets being the same) we get a  
small p-value, indicating that the difference observed is statistically significant (with  
parametric t-test AND the non-parametrix Mann-Whitney test)  
*#%% md*  
4) Write a conclusion on the possible difference between the heart period during the  
day for physical active and passive persons. The conclusion should contain  
information on size of the possible difference, whether or not it is statistical  
significant and a discussion on the validity of the assumptions behind the statistical  
analysis.  
  
*#%% md*We found a statistically significant difference between the heart rates of active and  
not active people during the day time. Assuming a normal distribution of the log-transformed  
data, the mean value for the passive group was 772 (95% PI = 613, 971). For the active  
group, this was 847 (95% PI = 654, 1096). The difference was statistically significant  
for both a parametric (p = 0.0006) and non-parametric tests (p = 0.0002).  
  
*#%% md  
  
## Exercise 2.4*  
This exercise focuses on power and sample size calculations. Consider the study on  
diastolic blood pressure (DBP) used in the lecture on sample size calculation.  
  
*#%% md*  
1) Suppose that you want to design a study comparing two groups, with the same  
standard deviation (sd=8) and a hypothesised difference in means of 5:  
How many participants should you have in each group to obtain a power of 90%.  
  
*#%%*import statsmodels  
  
power = []  
for n\_obs in range(10, 500):  
 pow = statsmodels.stats.power.TTestIndPower.power(self=pow, effect\_size=5/9, nobs1=n\_obs,  
 alpha=0.05, ratio=1)  
 power.append(100\*pow)  
 if pow >= 0.899 and pow<=0.901:  
 print(**'# participants for a power of 90% = '**, n\_obs)  
  
plt.rcParams.update({**'font.size'**: 10})  
fig, ax = plt.subplots(1, 1)  
ax.plot(power)  
ax.set\_xlabel(**'# participants'**)  
ax.set\_ylabel(**'Power [%]'**)  
plt.show()  
  
*#%% md*  
2) Now, suppose that you have planned a study with one intervention group and one  
control group both with 200 participants. Your plan is to compare the systolic blood  
pressure in the two groups and you expect the mean SBP in the control group to be  
130 mmHg and in the intervention group to be 125 mmHg. Furthermore, your best  
guess is that the standard deviation in both groups will be 25 mmHg. Determine the  
statistical power of this study.  
  
*#%%*pow = statsmodels.stats.power.TTestIndPower.power(self=pow, effect\_size=(130-125)/25,  
 nobs1=200, alpha=0.05, ratio=1)  
print(**'The statistical power of this study is: '**, pow\*100, **'%'**)