Chapter 10.11p 170-173 : ex 7.

Twain essays = [0,225, 0.262, 0.217, 0.240, 0.230, 0.229, 0.235, 0.217] $n = 8$

Snodgrass essays = [0.209, 0.205, 0.196, 0.210, 0.202, 0.207, 0.224, 0.223, 0.220, 0.201] $n = 10$

(a) Perform a Wald test for equality of the means. Use the nonparametric plug-in estimator. Report the p-value and a 95 per cent confidence interval for the difference of means.

Wald Test: $W = \frac{\hat{\theta} - \theta}{se}$

For the Twain essays, we have:

$$\bar{X} = \frac{0,225+0,262+0,217+0,240+0,230+0,229+0,235+0,217}{8}$$

$$\bar{X} = 0,2319$$

$$Var(X)^2 = \frac{\sum (X_i - \bar{X}_1)^2}{n-1} = \frac{(0,2319-0,225)^2 + \dots}{7}$$

$$\frac{+(0,2319-0,262)^2+(0,2319-0,217)^2+(0,2319-0,240)^2}{7}$$

$$+(0,2319-0,230)^2+(0,2319-0,229)^2+(0,2319-0,235)^2,$$

$$+(0,2319-0,217)^2 \approx (0,0002121)$$

For the Snodgrass essays, we have:

$$\bar{X} = \frac{0,209+0,205+0,196+0,210+0,202+0,207+0,224+0,223}{10}$$

$$\frac{0,220+0,201}{} = 0,2097.$$

$$Var(X)^2 = 0,00009334$$

$$W = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{Var(\bar{X}_1)^2}{n_1} + \frac{Var(X_2)^2}{n_2}}} = \frac{0,2319-0,2097}{\sqrt{\frac{0,0002121}{8} + \frac{0,00009334}{10}}}$$

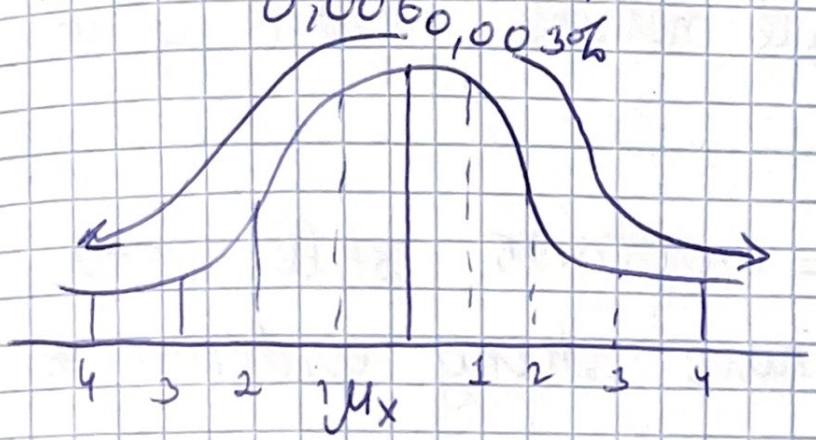
$$\approx 3,708$$

Mean difference: 0,0222

Wald statistic: 3,708

$$\delta = \sqrt{\frac{Var(X_1)^2}{n_1} + \frac{Var(X_2)^2}{n_2}} = 0,00598719 \approx 0,006.$$

$$Z = \frac{0,0222}{0,006} = 3,7 \approx 4.$$



approximately 4 std. deviations which means that only 0,003% of the data falls. which is very small and less than 0,05.

p-value: $\approx 0,00003$

So, we reject the null hypothesis - that the series follow distributions with different means.

Confidence intervals

$[0{,}0222 - 1{,}96 \cdot 0{,}006 \; ; \; 0{,}0222 + 1{,}96 \cdot 0{,}006]$

$[0{,}1044 ; 0{,}03396]$

(b) Now use a permutation test to avoid the use of large sample methods. What is your conclusion?

We got p-value = 0,0000455 still very small, and we have strong evidence to reject H_0.

(7b)

```
import numpy as np

X = [0.225, 0.262, 0.217, 0.240, 0.230, 0.229, 0.235, 0.217]
Y = [0.209, 0.205, 0.196, 0.210, 0.202, 0.207, 0.224, 0.223, 0.220, 0.201]

X = np.array(X)
Y = np.array(Y)

N = 1000000
all_data = np.concatenate([X, Y])
nx = len(X)

# by calculations in part a
diff_hat = 0.0222
cnt = 0

for i in range(1000000):
    np.random.shuffle(all_data)
    x = all_data[:nx]
    y = all_data[nx:]
    diff = x.mean() - y.mean()
    if diff > diff_hat:
        cnt += 1

p_value = cnt / N

print(p_value)
```

Chapter 13.10 p. 226-229: ex. 6, data source:

6. Get the passenger car mileage data from

http://lib.stat.cmu.edu/DASL/Datafiles/carmpgdat.html

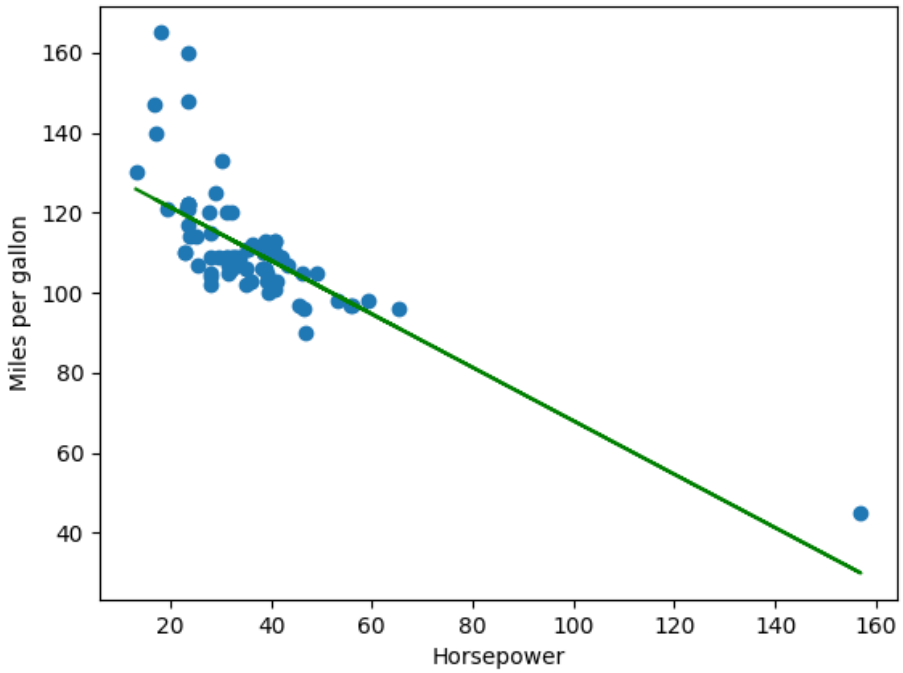
(a) Fit a simple linear regression model to predict MPG (miles per gallon) from HP (horsepower). Summarize your analysis including a plot of the data with the fitted line.

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.linear_model import LinearRegression

data = pd.read_csv('data.csv')

X = data['HP'].values.reshape(-1, 1)
Y = data['MPG'].values

model = LinearRegression()
model.fit(X, Y)
```

```
print(model.intercept_, model.coef_)
plt.scatter(X, Y)
plt.plot(X, model.predict(X), color='green')
plt.xlabel("Horsepower")
plt.ylabel("Miles per gallon")
plt.show()
```

intercept_ - 134.7307189183848

coef_ - [-0.66723438]



(b) Repeat the analysis but use log(MPG) as the response. Compare the analyses.

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.linear_model import LinearRegression

data = pd.read_csv('data.csv')

X = data['HP'].values.reshape(-1, 1)
Y = np.log(data['MPG']).values
model = LinearRegression()
model.fit(X, Y)
print(model.intercept_, model.coef_)
plt.scatter(X, Y)
plt.plot(X, model.predict(X), color='green')
plt.xlabel("Horsepower")
```

```
plt.ylabel("Miles per gallon")
plt.show()
```

intercept_ - 4.965981466781696

coef_ - [-0.00749009]