Statistics - Assignment 1

Groupname: Airport- Crew

Members: 5874903

8133405

9443009

3583266

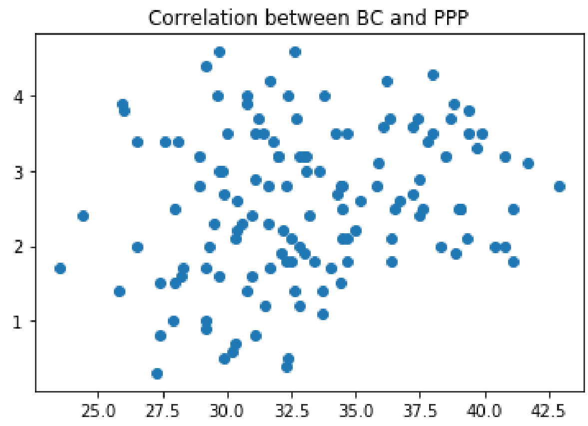
8602307

Exercise 1.1 – Coding

```
import numpy as np
from scipy.stats import norm
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import scipy as scipy
import math

# Exercise 1
data = pd.read_csv(r"C:\Users\wrigg\OneDrive\Desktop\DHBW\3. Semester\Statistik\Assignments\Sailors Assignment 1.txt", sep="\t")

# Exercise 1a
print("mean of BC=", round(np.mean(data.BC),2))
print("mean of PPP=", round(np.mean(data.PPP),2))

# Exercise 1b
print("uncorrected variance of BC=", round(np.var(data.BC),2))
print("uncorrected variance of PPP=", round(np.var(data.PPP),2))
print("corrected variance of BC=", round(np.var(data.BC, ddof=1),2))
print("corrected variance of PPP=", round(np.var(data.PPP, ddof=1),2))

# Exercise 1c
print("correlation between BC and PPP=", round(data["BC"].corr(data["PPP"]),2))


plt.scatter(data.BC, data.PPP)
plt.title("Correlation between BC and PPP")
plt.show()
```

Exercise 1.1 – Output

```
mean of BC= 33.13
mean of PPP= 2.55
uncorrected variance of BC= 16.92
uncorrected variance of PPP= 0.99
corrected variance of BC= 17.05
corrected variance of PPP= 0.99
correlation between BC and PPP= 0.21

Text(0.5, 1.0, 'Correlation between BC and PPP')
```



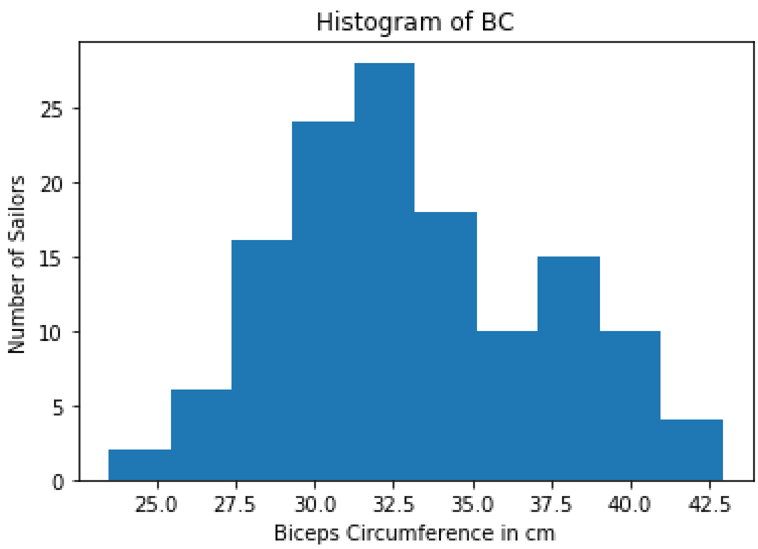
We displayed not only the corrected, but also the uncorrected variances.

We rounded the values for the mean, variance and correlation.

Exercise 1.2 a) – Coding

```
# Exercise 2a
# Histogram
plt.hist(data.BC)
plt.title("Histogram of BC")
plt.ylabel('Number of Sailors')
plt.xlabel("Biceps Circumference in cm")
```

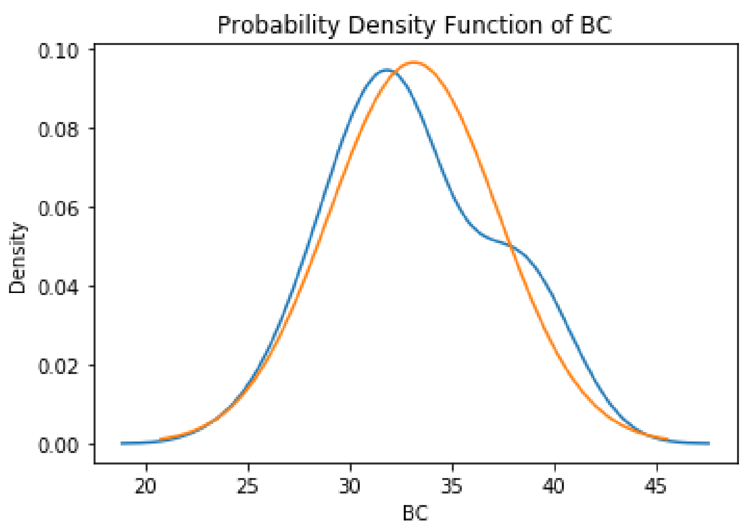
Exercise 1.2 a) – Output



Exercise 1.2 b) – Coding

```
# Exercise 2b
sns.kdeplot(data.BC)
mu = 33.13
variance = 17.05
sigma = math.sqrt(variance)
x = np.linspace(mu - 3*sigma, mu + 3*sigma, 100)
plt.plot(x, stats.norm.pdf(x, mu, sigma))

plt.title("Probability Density Function of BC")
```

Exercise 1.2 b) – Output



Exercise 1.2 c)

As to be seen from the histogram and the probability density function, BC seems to be not perfectly, but partially gaussian distributed. The maximum of the pdf is nearly at the mean of BC (μ= 33.13) and the shape of the function resembles the typical gloche.

Additionally, if you compare the pdf of BC with the normal distribution function with (mean=33.13, variance= 17.05), you can see the similarity between the two graphs.

Exercise 2 – Coding

```python
# Exercise 2

from scipy.optimize import minimize
import math


# Estimate sigma^2
mu = np.mean(data.BC)
sigma = np.var(data.BC, ddof=1)
n = data.BC.size

def loglikelihood(x):
    mu = x[0]
    sigma = x[1]
    deviations=[((i-mu)**2)for i in data.BC]
    return (n/2)*(math.log(2*np.pi) + np.log(sigma)) + (1/(2*sigma)) * sum(deviations)

print(minimize(loglikelihood, [mu, sigma]))
```

Exercise 2 – Output

```
      fun: 376.8274155749016
 hess_inv: array([[1.        , 0.        ],
       [0.        , 4.32364619]])
      jac: array([0., 0.])
  message: 'Optimization terminated successfully.'
     nfev: 20
      nit: 4
     njev: 5
   status: 0
  success: True
        x: array([33.13007519, 16.92344787])
```