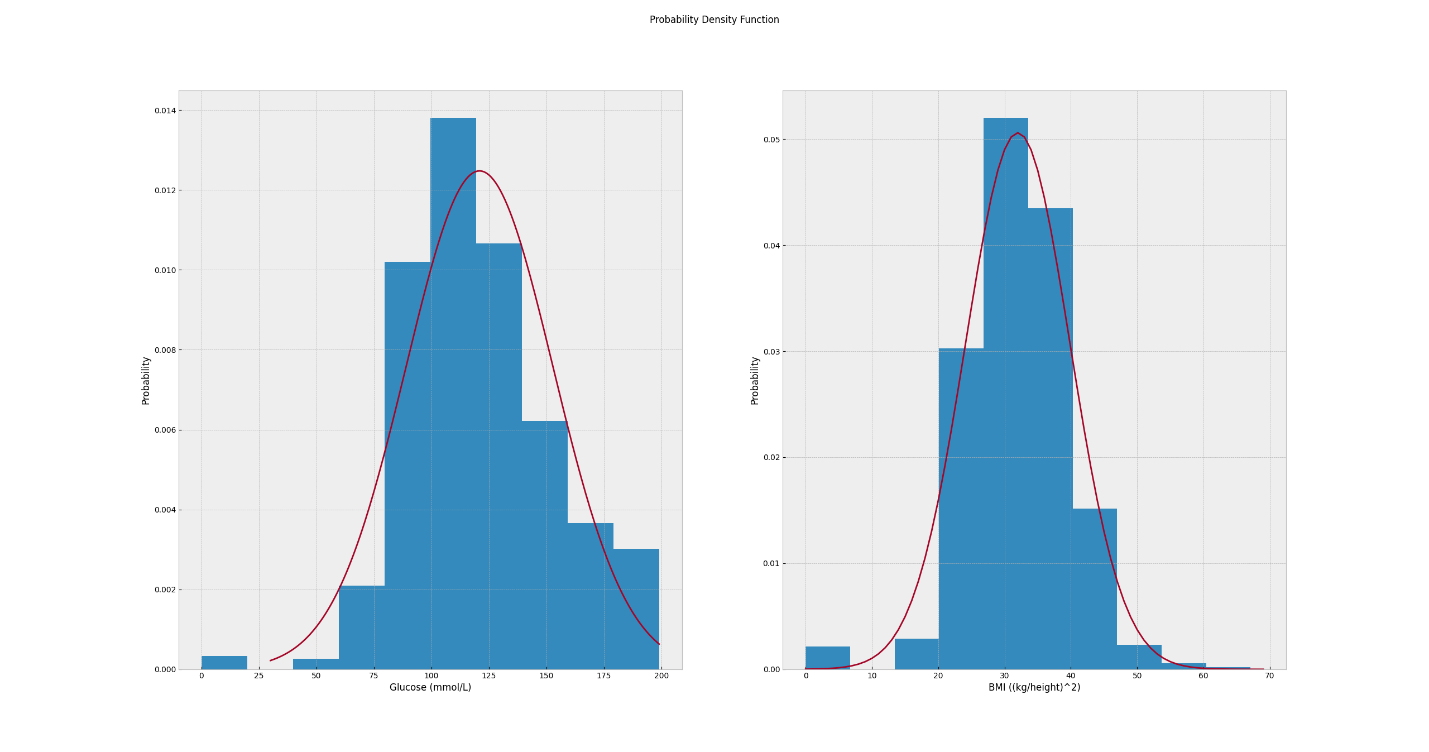
**นายสหทัศน์ ลีวัฒนา 62010922**

**Homework 3**

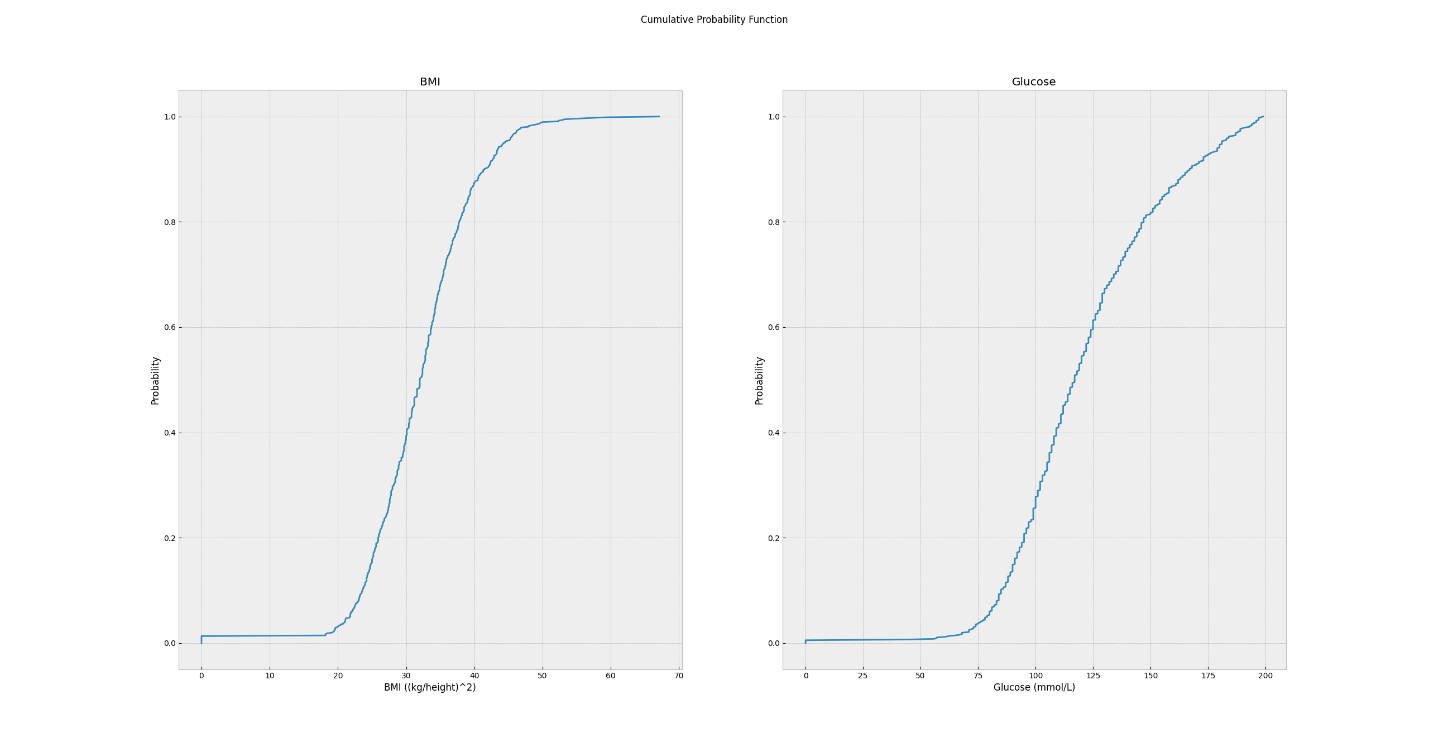
Column ที่เลือกใช้คือ

* Glucose -จำนวนน้ำตาลในเลือด หน่วย mmol/L
* BMI -ค่าดัชนีมวลกาย หน่วย (kg/height) ^2

Probability Density Function



จากกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็นของกราฟ Glucose จะเห็นได้ว่าจุดยอดของกราฟอยู่ที่ 121.0 mmol/L ซึ่งทำให้เห็นว่าจำนวนน้ำตาลในเลือดของผู้ป่วยที่เป็นโรคเบาหวานมากที่สุดอยู่ที่ 121.0 mmol/L และ จากกราฟฟังก์ชันความหนาแน่นของความน่าจะเป็นของกราฟ BMI จะเห็นได้ว่าจุดยอดของกราฟอยู่ที่ 32.0 (kg/height)^2 ซึ่งทำให้เห็นว่าดัชนีมวลกายของผู้ป่วยที่เป็นโรคเบาหวานมากที่สุดอยู่ที่ 32.0 (kg/height)^2

Cumulative Probability Function

จากกราฟ CPF ของ BMI จะเห็นได้ว่าช่วงแรกของกราฟยังไม่ชันมาก จะชันขึ้นอย่างมากในช่วง 19 ถึง 50 (kg/height)^2 เพราะข้อมูลส่วนใหญ่จะสะสมอยู่ในช่วงนี้ และในช่วง 50 ขึ้นไปจะเห็นได้ว่าความชันลดลงเรื่อย ๆ และในส่วนกราฟ CPF ของ Glucose จะเห็นได้ว่าช่วงแรกของกราฟยังไม่ชันมาก จะชันขึ้นอย่างมากในช่วง 60 mmol/L ขึ้นไปและหยุดในจุด 200 mmol/L

Source code

PDF

*import* matplotlib.pyplot *as* plt

*import* pandas *as* pd

*from* scipy.stats *import* norm

*import* statistics *as* stc

plt.style.use('bmh')

df = pd.read\_csv('diabetes.csv')

*# age glucose BMI*

x = df['Age']

y = df['Glucose']

z = df['BMI']

*# convert to list*

age = x.to\_list()

glucose = y.to\_list()

bmi = z.to\_list()

data = glucose

data2 = bmi

*# calculate parameters*

sample\_mean = stc.mean(data)

sample\_std = stc.stdev(data)

print('Mean=%.3f, Standard Deviation=%.3f' % (sample\_mean, sample\_std))

sample\_mean2 = stc.mean(data2)

sample\_std2 = stc.stdev(data2)

*# define the distribution*

dist = norm(sample\_mean, sample\_std)

dist2 = norm(sample\_mean2, sample\_std2)

*# sample probabilities for a range of outcomes*

values = [value *for* value *in* range(30, 200)]

probabilities = [dist.pdf(value) *for* value *in* values]

values2 = [value2 *for* value2 *in* range(0, 70)]

probabilities2 = [dist2.pdf(value2) *for* value2 *in* values2]

fig, ax = plt.subplots(1, 2)

fig.suptitle('Probability Density Function')

*# plot the histogram and pdf*

ax[0].hist(data, bins=10, density=True)

ax[0].plot(values, probabilities)

ax[0].set\_xlabel('Glucose (mmol/L)')

ax[0].set\_ylabel('Probability')

ax[1].hist(data2, bins=10, density=True)

ax[1].plot(values2, probabilities2)

ax[1].set\_xlabel('BMI ((kg/height)^2)')

ax[1].set\_ylabel('Probability')

plt.show()

CFP

*import* matplotlib.pyplot *as* plt

*import* numpy *as* np

*import* pandas *as* pd

plt.style.use('bmh')

df = pd.read\_csv('diabetes.csv')

*# read data*

x = df['Age']

y = df['Glucose']

z = df['BMI']

*# to list*

age = x.to\_list()

glucose = y.to\_list()

bmi = z.to\_list()

data = np.array(bmi)

data2 = np.array(glucose)

data.sort()

data2.sort()

*# https://www.youtube.com/watch?v=fQ0Iy0Sew\_U*

*# yvals = np.zeros(len(data))*

*# for i in range(len(data)):*

*#     yvals[i] = (i+1)/len(yvals)*

*# plt.plot(data, yvals, 'k')*

*# https://stackoverflow.com/questions/24788200/calculate-the-cumulative-distribution-function-cdf-in-python*

p = 1. \* np.arange(len(data)) / (len(data) - 1)

p2 = 1. \* np.arange(len(data2)) / (len(data) - 1)

fig, ax = plt.subplots(1, 2)

fig.suptitle('Cumulative Probability Function')

ax[0].plot(data, p)

ax[0].set\_title('BMI')

ax[0].set\_xlabel('BMI ((kg/height)^2)')

ax[0].set\_ylabel('Probability')

ax[1].plot(data2, p2)

ax[1].set\_title('Glucose')

ax[1].set\_xlabel('Glucose (mmol/L)')

ax[1].set\_ylabel('Probability')

plt.show()