

Probability and Statistic Full Report

เสนอ

ผศ.ดร. สุรินทร์ กิตติธรกุล

จัดทำโดย

นาย สุรวิช ยอแสง รหัสนักศึกษา : 62010986  
นาย อัครวินท์ บุญเผื่อน รหัสนักศึกษา : 62011044

วิชา Probability and Statistic  
รหัสวิชา : 01076253

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
King Mongkut’s Institute of Technology Ladkrabang

สารบัญ

[Homework 1 : Data [Movie Industry] 2](#_Toc70277587)

[ชื่อชุดข้อมูล 2](#_Toc70277588)

[คอลัมน์ 2](#_Toc70277589)

[Why is it interesting? 2](#_Toc70277590)

[คำอธิบายของคอมลัมน์ที่เลือก 2](#_Toc70277591)

[วิธีการรวบรวมข้อมูล 2](#_Toc70277592)

[Homework 2 : ค่ากลางและกราฟ 3](#_Toc70277593)

[Column ที่ใช้ 3](#_Toc70277594)

[สถิติต่างๆ 3](#_Toc70277595)

[กราฟต่างๆ 3](#_Toc70277596)

[บทวิเคราะห์ 7](#_Toc70277597)

[Python code 8](#_Toc70277598)

[Homework 3 : PDF & CPF 10](#_Toc70277599)

[Probability Density Function 10](#_Toc70277600)

[Cumulative Probability Function 11](#_Toc70277601)

[Source code 12](#_Toc70277602)

[Homework 4 : Confidence Interval 14](#_Toc70277603)

[Confidence Interval 14](#_Toc70277604)

[Source code 16](#_Toc70277605)

[Homework 5 : Linear Regression 18](#_Toc70277606)

[linear regression 18](#_Toc70277607)

[Source code 20](#_Toc70277608)

# Homework 1 : Data [Movie Industry]

## ชื่อชุดข้อมูล

**Movie Industry**

## คอลัมน์

**company, gross, budget**

## Why is it interesting?

จากข้อมูลรายได้ของภาพยนตร์เทียบกับทุนสร้างจะสามารถบ่งบอกได้ว่า บริษัทที่ผลิตภาพยนตร์มีอัตราการทำกำไรของภาพยนตร์มากน้อยเพียงใด ซึ่งค่าที่ได้จะสามารถบ่งบอกได้ว่า บริษัทผลิตภาพยนตร์บริษัทใดสามารถผลิตภาพยนตร์ที่ตรงความต้องการของตลาดได้มากที่สุด และในอนาคตแนวโน้มการฉายภาพยนตร์อาจจะเป็นการออกฉายผ่านทางระบบสตรีมมิ่งออนไลน์แทนการฉายทางโรงภาพยนตร์ด้วยเหตุผลต่างๆ ซึ่งอาจจะเป็นการเปลี่ยนแปลงครั้งใหญ่ของวงการภาพยนตร์ ดังนั้นชุดข้อมูลนี้อาจตอบคำถามที่ว่าอุตสาหกรรมภาพยนตร์กำลังตายลงจริงหรือไม่

**แหล่งที่มาของชุดข้อมูล :** https://www.kaggle.com/danielgrijalvas/movies

## คำอธิบายของคอมลัมน์ที่เลือก

* **Company:** ชื่อบริษัทผู้ผลิต
* **Gross:** รายได้ของภาพยนตร์
* **Budget:** ทุนสร้างของภาพยนตร์

## วิธีการรวบรวมข้อมูล

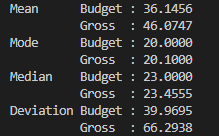
ข้อมูลจากเว็บ IMDb

# Homework 2 : ค่ากลางและกราฟ

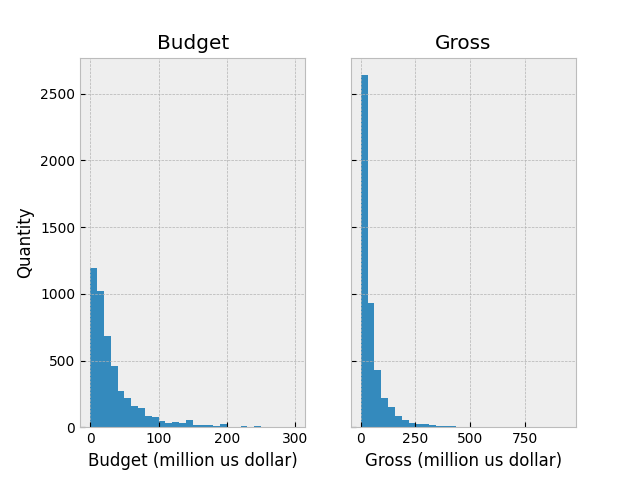
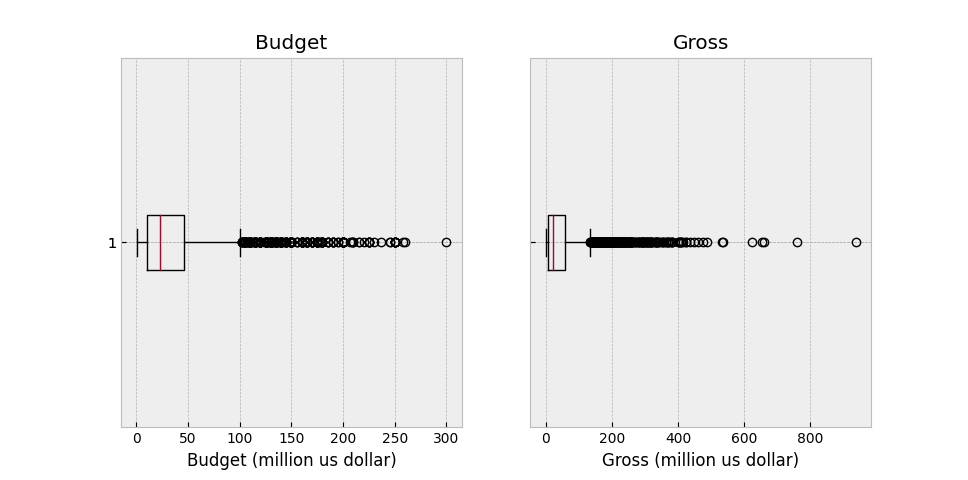
## Column ที่ใช้

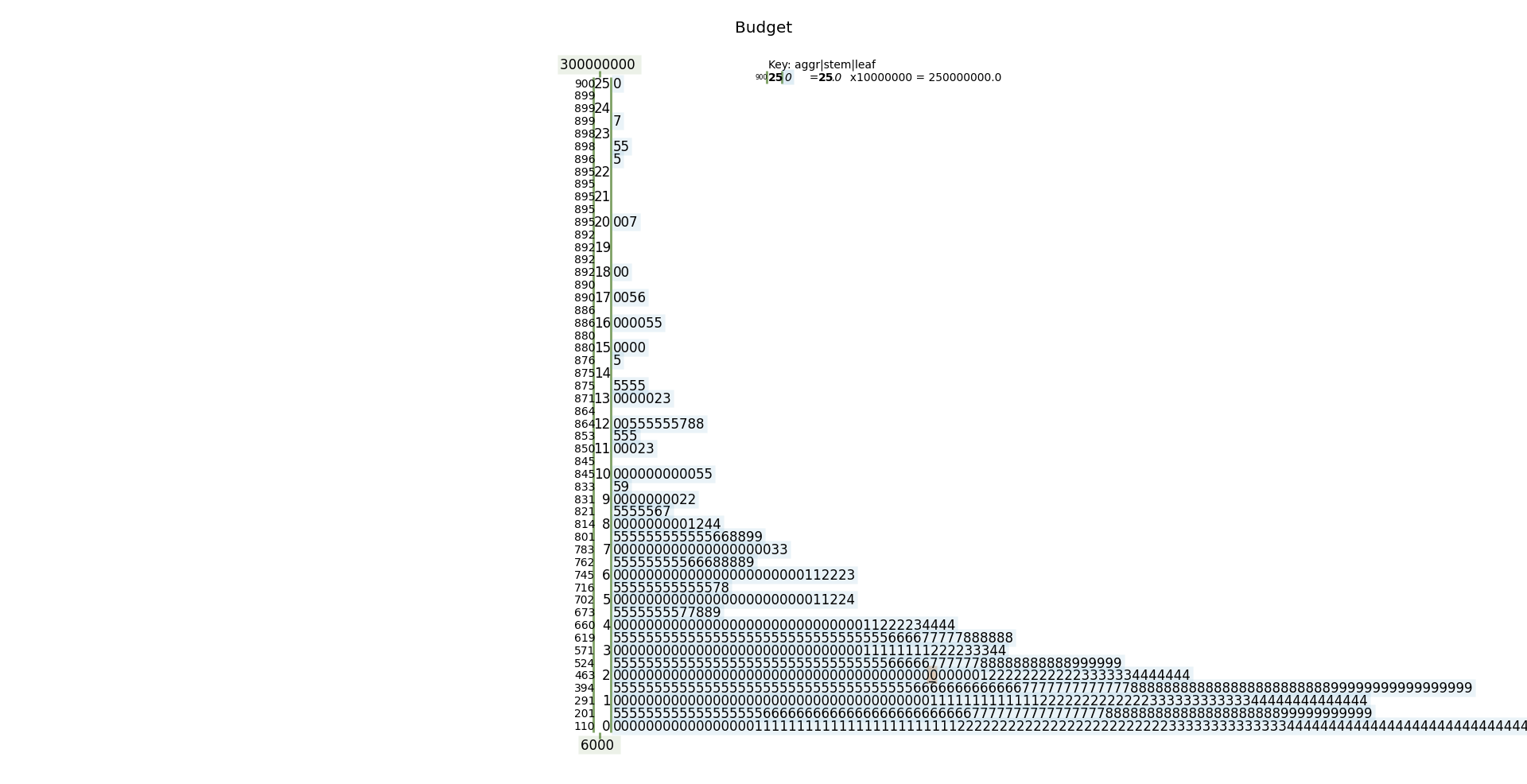
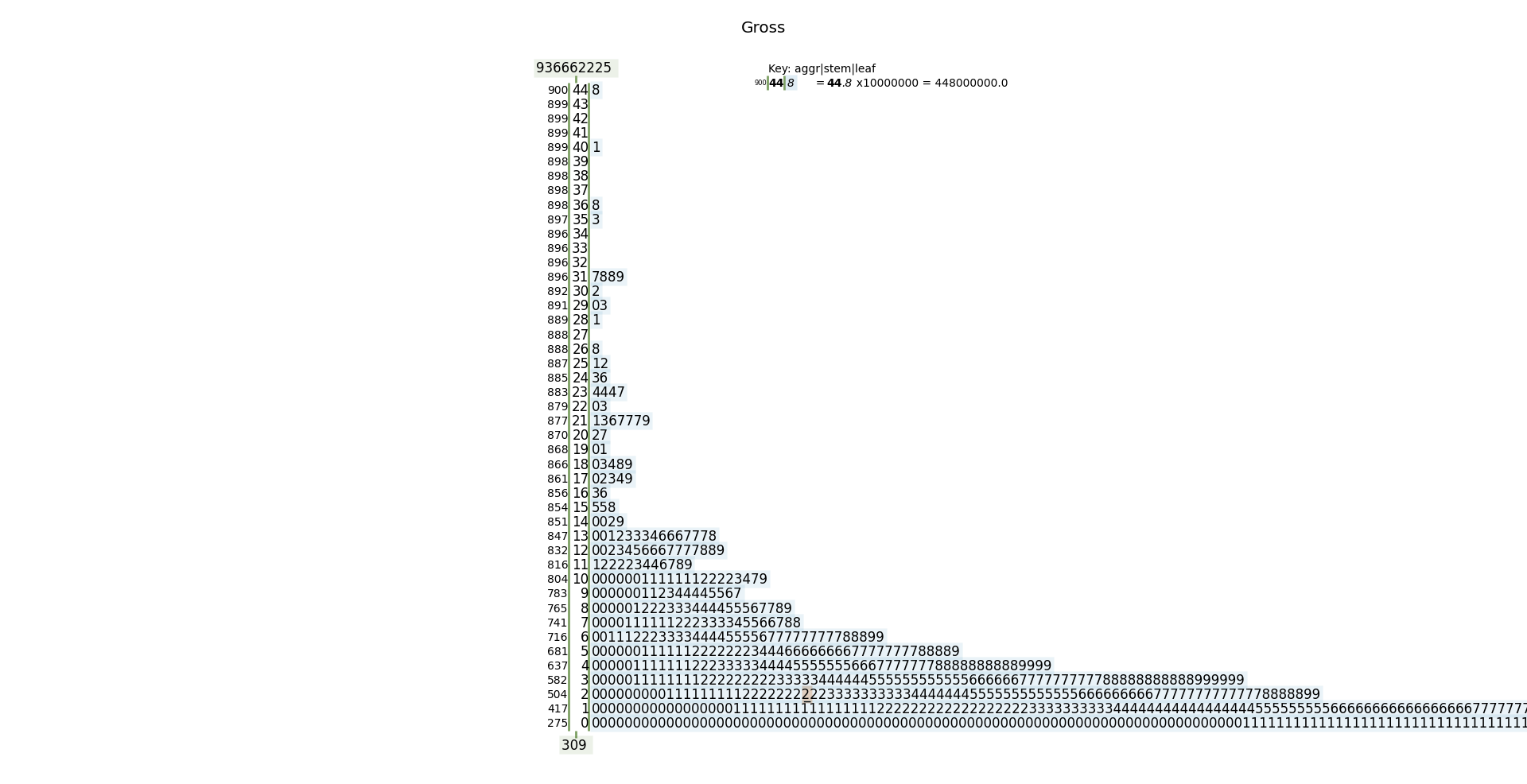
1. Budget งบประมาณในการสร้างภาพยนตร์ (million us dollar)
2. Gross รายได้ทั้งหมด (million us dollar)

## สถิติต่างๆ

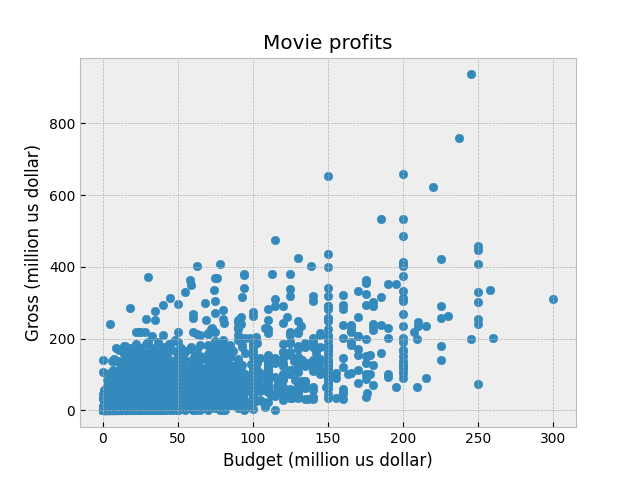


## กราฟต่างๆ

1. **Histogram**
2. **Box plots**
3. **Stem and Leaf**



1. **Scatter**



**ตัวแปรต้น :** ทุนสร้าง (Budget)

**ตัวแปรตาม :** รายได้ (Gross)

**Outlier :**

1. Budget : ทุกข้อมูลที่มากกว่า 100 ล้านดอลลาร์
2. Gross : ทุกข้อมูลที่มากกว่า 138 ล้านดอลลาร์

**เหตุผล :** เพราะในการสร้างภาพยนตร์จำเป็นต้องอาศัยทุนในการสร้าง ทางผู้ค้นคว้าจึงต้องการทราบว่าทุนในการสร้างภาพยนตร์ส่งผลต่อรายได้ของภาพยนตร์หรือไม่

## บทวิเคราะห์

จากการวิเคราะห์ข้อมูลจากกราฟ จะเห็นได้ว่าเมื่อภาพยนตร์มีงบประมาณในการสร้างน้อย ก็จะมีรายได้ใกล้เคียง หรือได้กำไรใกล้เคียงกับทุนเป็นส่วนใหญ่ แล้วก็มีบางส่วนที่ได้กำไรจำนวนมาก ก็มักจะมีทุนสร้างที่มากเช่นกัน

สรุปได้ว่า ภาพยนตร์ที่มีรายได้สูง มักจะเป็นภาพยนตร์ที่มีทุนสูงเช่นกัน แต่มีภาพยนตร์จำนวนน้อยที่สามารถทำกำไรจากทุนสร้างได้หลายเท่า ดังนั้นทุนในการสร้างภาพยนตร์จะแปรผันตรงกับรายได้ของภาพยนตร์

## Python code

import statistics as stc

import matplotlib.pyplot as plt

import pandas as pd

import stemgraphic

plt.style.use('bmh')

df = pd.read\_csv('moviesfilter.csv')

# budget gross company name

x = df['budget']

y = df['gross']

z = df['company']

budget = x.to\_list()

gross = y.to\_list()

company = z.to\_list()

#format data to million dollar

for i in range(0, len(budget)):

    budget[i] = budget[i]/1000000

for i in range(0, len(gross)):

    gross[i] = gross[i]/1000000

#Print all detail

def detail():

    print("Mean      Budget :",str(stc.mean(budget)))

    print("          Gross  :",str(stc.mean(gross)))

    print("Mode      Budget :",str(stc.mode(budget)))

    print("          Gross  :",str(stc.mode(gross)))

    print("Median    Budget :",str(stc.median(budget)))

    print("          Gross  :",str(stc.median(gross)))

    print("Deviation Budget :",str(stc.stdev(budget)))

    print("          Gross  :",str(stc.stdev(gross)))

def histogram():

    fig, ax = plt.subplots(1, 2, sharey=True)

    ax[0].set\_xlabel('Budget (million us dollar)')

    ax[0].set\_ylabel('Quality')

    ax[0].set\_title('Budget')

    ax[0].hist(budget, bins=30)

    ax[1].set\_title('Gross')

    ax[1].hist(gross, bins=30)

    ax[1].set\_xlabel('Gross (million us dollar)')

    plt.show()

def boxplot():

    fig, ax = plt.subplots(1, 2, sharey=True)

    ax[0].set\_title('Budget')

    ax[0].boxplot(budget, vert=False)

    ax[0].set\_xlabel('Budget (million us dollar)')

    ax[1].set\_title('Gross')

    ax[1].boxplot(gross, vert=False)

    ax[1].set\_xlabel('Gross (million us dollar)')

    plt.show()

def stem():

    stemgraphic.stem\_graphic(df['budget'])

    plt.title('Budget')

    plt.show()

    stemgraphic.stem\_graphic(df['gross'])

    plt.title('Gross')

    plt.show()

def scatter():

    plt.xlabel('Budget (million us dollar)')

    plt.ylabel('Gross (million us dollar)')

    plt.title('Profit')

    plt.scatter(budget,gross)

    plt.show()

if \_\_name\_\_ == "\_\_main\_\_":

    detail()

    histogram()

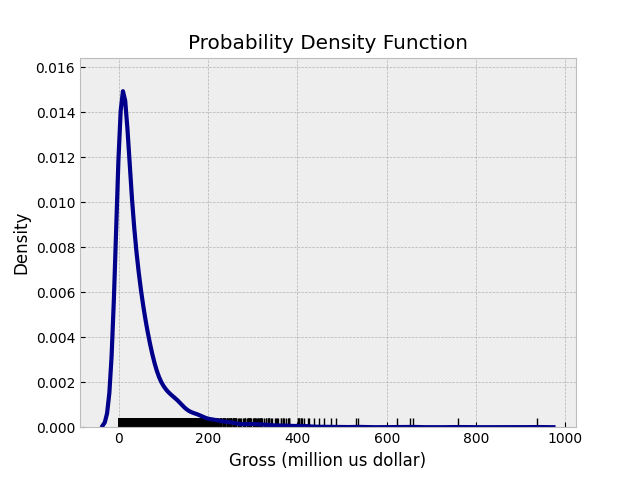
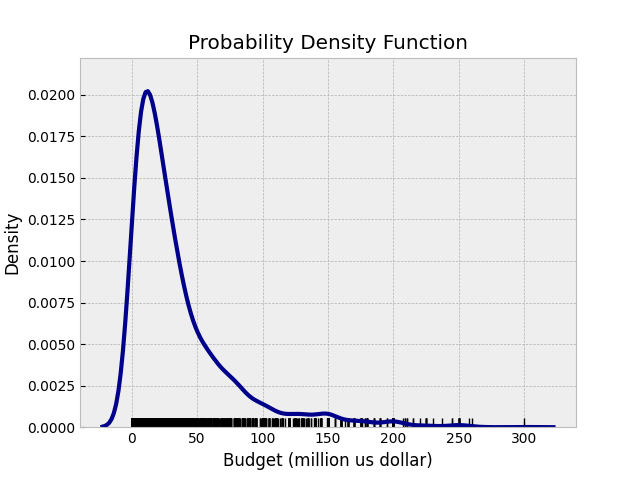
    boxplot()

    stem()

    scatter()

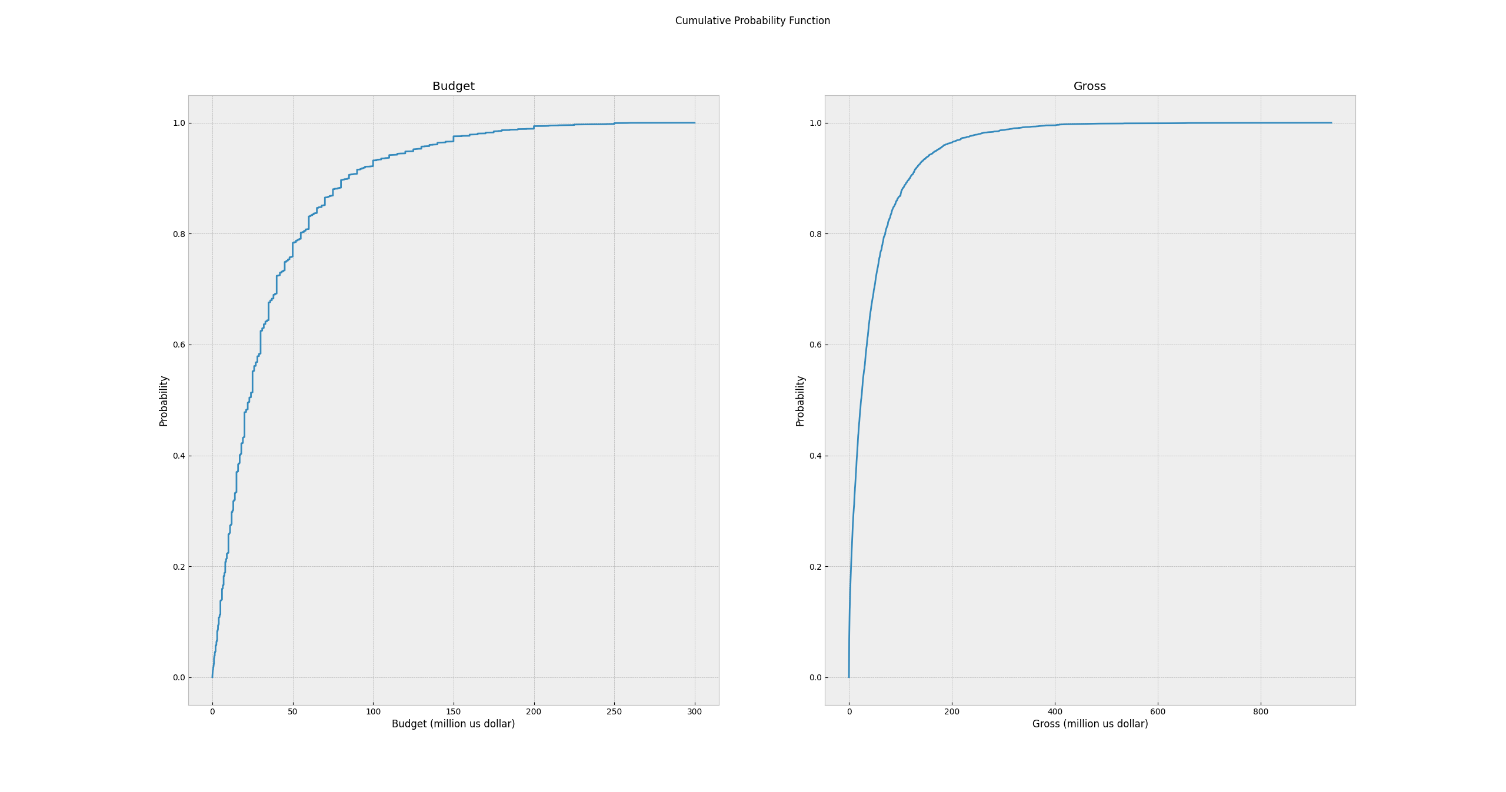
# Homework 3 : PDF & CPF

## Probability Density Function

****

ความน่าจะเป็นสูงสุดของกราฟงบประมาณอยู่ที่ 12 ล้านดอลลาร์ และรายได้ทั้งหมดอยู่ที่ประมาณ 9 ล้านดอลลาร์ จากกราฟแสดงว่า ภาพยนตร์ 20 จาก 100 เรื่อง ใช้งบประมาณที่ 12 ล้านดอลลาร์ และมีภาพยนตร์ 15 ใน 100 เรื่อง มีรายได้อยู่ที่ประมาณ 9 ล้านดอลลาร์ ดังนั้นจะสรุปได้ว่าอุตสาหกรรมภาพยนตร์ มีการใช้งบประมาณสูงกว่ารายได้ของภาพยนตร์

## Cumulative Probability Function

****

ความชันของกราฟต้นทุนของภาพยนตร์ถึง 80% อยู่ที่ 50 ล้านดอลลาร์จะเริ่มมีความชันที่น้อยลงอย่างเห็นได้ชัด กราฟรายได้ของภาพยนตร์ที่ 80% เห็นได้ชัดว่าความชันของต้นทุนจะเริ่มน้อยลงที่ประมาณ 70 ล้านดอลลาร์ สรุปได้ว่าภาพยนตร์ส่วนใหญ่ถึง 80% ใช้งบประมาณที่ 50 ล้านดอลลาร์ และมีรายได้อยู่ที่ 70 ล้านดอลลาร์

## Source code

import statistics as stc

import matplotlib.pyplot as plt

import pandas as pd

import seaborn as sns

import numpy as np

plt.style.use('bmh')

df = pd.read\_csv('moviesfilter.csv')

# budget gross company name

x = df['budget']

y = df['gross']

z = df['company']

budget = x.to\_list()

gross = y.to\_list()

company = z.to\_list()

#format data to million dollar

for i in range(0, len(budget)):

    budget[i] = budget[i]/1000000

for i in range(0, len(gross)):

    gross[i] = gross[i]/1000000

#Print all detail

def detail():

    print("Mean      Budget :",str(stc.mean(budget)))

    print("          Gross  :",str(stc.mean(gross)))

    print("Mode      Budget :",str(stc.mode(budget)))

    print("          Gross  :",str(stc.mode(gross)))

    print("Median    Budget :",str(stc.median(budget)))

    print("          Gross  :",str(stc.median(gross)))

    print("Deviation Budget :",str(stc.stdev(budget)))

    print("          Gross  :",str(stc.stdev(gross)))

def densityplot():

    sns.distplot(budget, hist = False, kde = True, rug = True,color = 'darkblue', kde\_kws={'linewidth': 3},rug\_kws={'color': 'black'})

    # Plot formatting

    plt.title('Probability Density Function')

    plt.xlabel('Budget (million us dollar)')

    plt.ylabel('Density')

    plt.show()

    sns.distplot(gross, hist = False, kde = True, rug = True,color = 'darkblue', kde\_kws={'linewidth': 3},rug\_kws={'color': 'black'})

    # Plot formatting

    plt.title('Probability Density Function')

    plt.xlabel('Gross (million us dollar)')

    plt.ylabel('Density')

    plt.show()

def cumulative():

    budgetData = sorted(np.array(budget))

    grossData = sorted(np.array(gross))

    budgetProb = 1. \* np.arange(len(budgetData)) / (len(budgetData)-1)

    grossProb = 1. \* np.arange(len(grossData)) / (len(grossData)-1)

    fig, ax = plt.subplots(1, 2)

    fig.suptitle('Cumulative Probability Function')

    ax[0].set\_title('Budget')

    ax[0].plot(budgetData, budgetProb)

    ax[0].set\_xlabel('Budget (million us dollar)')

    ax[0].set\_ylabel('Probability')

    ax[1].set\_title('Gross')

    ax[1].plot(grossData, grossProb)

    ax[1].set\_xlabel('Gross (million us dollar)')

    ax[1].set\_ylabel('Probability')

    plt.show()

if \_\_name\_\_ == "\_\_main\_\_":

    detail()

    densityplot()

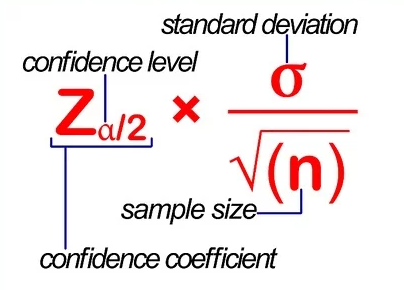
    cumulative()

# Homework 4 : Confidence Interval

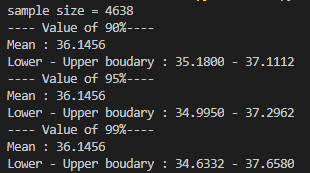
## Confidence Interval

กราฟแสดงช่วงของความเชื่อมั่นในระดับต่างๆ เทียบกับกราฟ PDF

ช่วงความเชื่อมั่น (confidence interval) หมายถึง ช่วงของค่าประมาณที่ประกอบไปด้วยค่าต่ำสุด และค่าสูงสุด ที่คำนวณขึ้นมาจากสูตรข้างต้น ที่บ่งบอกค่าเฉลี่ยที่บอกระดับความเชื่อมั่น โดยอิงจากกลุ่มตัวอย่าง (Sample) ที่สามารถใช้อ้างอิงถึงข้อมูลทั้งหมดได้ (Population)

****

สูตรการคำนวณหา Confidence Interval (CI)



ค่าที่ได้จากการคำนวณ

จากการคำนวณโดยใช้ช่วงระดับความเชื่อมั่น 3 ช่วง ได้แก่ 90% 95& และ 99% ได้ค่าดังนี้

1. เปอร์เซ็นต์ความเชื่อมั่น 90% จะมีต้นทุนอยู่ที่ 35.18 – 37.11 ล้านดอลลาร์
2. เปอร์เซ็นต์ความเชื่อมั่น 95% จะมีต้นทุนอยู่ที่ 39.99 – 37.29 ล้านดอลลาร์
3. เปอร์เซ็นต์ความเชื่อมั่น 99% จะมีต้นทุนอยู่ที่ 34.63 – 37.65 ล้านดอลลาร์

จากการคำนวณช่วงความเชื่อมั่นที่ระดับความเชื่อมั่น ทั้ง 3 ระดับ ได้แก่ 90%, 95% และ 99% จะได้ค่าของความเชื่อมั่นตามข้อมูลข้างต้น จึงสรุปได้ว่า ในช่วงต้นทุนที่ 35.18 – 37.11 ล้านดอลลาร์ จะคลอบคลุมช่วงข้อมูลทั้งหมดที่ 90% ในช่วงต้นทุนที่ 39.99 – 37.29 ล้านดอลลาร์ จะคลอบคลุมช่วงข้อมูลทั้งหมดที่ 95% และถ้าใช้ข้อมูลในช่วง 34.63 – 37.65 ล้านดอลลาร์ จะคลอบคลุมข้อมูลทั้งหมดถึง 99%

สรุปได้ว่า หากมีการผลิตภาพยนตร์เพิ่มขึ้น แล้วมีการสุ่มกลุ่มตัวอย่าง (sample size) ใหม่อีกครั้ง แล้วนำมาคำนวณค่าความเชื่อมั่นอีกครั้ง จะพบว่าค่าเฉลี่ยที่ได้จะต้องยังคงอยู่ในช่วงความเชื่อมั่นที่คำนวณได้ข้างต้นตามระดับความเชื่อมั่นข้างต้น เช่น ถ้าใช้ช่วงความเชื่อมั่นที่ 95% โอกาสที่สุ่มใหม่แล้วได้ค่าอยู่ในช่วงเดิมก็จะอยู่ที่ 95% โดยถ้าใช้ช่วงข้อมูลที่ 34.63 – 37.65 ล้านดอลลาร์ กลุ่มข้อมูลตัวอย่างใหม่ที่สุ่มมาต้องอยู่ในช่วงนี้อย่างแน่นอน เนื่องจากมีค่าความเชื่อมั่นถึง 99%

## Source code

import pandas

import matplotlib.pyplot as plt

import numpy as np

import scipy.stats

plt.style.use('bmh')

columns = pandas.read\_csv('../Lab2/moviesfilter.csv')

# budget

x = columns['budget']

budget = x.to\_list()

#format data to million dollar

for i in range(0, len(budget)):

    budget[i] = budget[i]/1000000

print('sample size =',len(budget))

def mean\_confidence\_interval(data, confidence=0.95):

    a = 1.0 \* np.array(data)

    n = len(a)

    m, se = np.mean(a), scipy.stats.sem(a)

    h = se \* scipy.stats.t.ppf((1 + confidence) / 2., n-1)

    print('---- Value of {:.0f}%----'.format(confidence\*100))

    print('Mean :',m)

    print('Lower - Upper boudary :{:.4f} - {:.4f}'.format(m-h,m+h))

    return m, m-h, m+h

m1, lB1, uB1 = mean\_confidence\_interval(budget,0.90)

m2, lB2, uB2 = mean\_confidence\_interval(budget,0.95)

m3, lB3, uB3 = mean\_confidence\_interval(budget,0.99)

count, bins\_count = np.histogram(budget, bins=18)

pdf = count / sum(count)

figure, func = plt.subplots(3, 1, figsize=(8, 10))

plt.tight\_layout(pad=5,h\_pad=5.0)

y = np.linspace(0,1)

title1,title2 = 'Confidence Interval (CI) Lv.','% of Movies budget.'

xlabel = "Budget (million us dollar)"

ylabel = "Probability"

func[0].set\_title(title1+'90'+title2)

func[0].set\_xlabel(xlabel)

func[0].set\_ylabel(ylabel)

func[0].plot(bins\_count[1:], pdf, color="green", label="PDF" )

x1,x2 = np.linspace(lB1,lB1),np.linspace(uB1,uB1)

func[0].plot(x1,y, label="Lower Boudary = {:.4f}".format(lB1))

func[0].plot(x2,y, label="Upper Boudary = {:.4f}".format(uB1))

func[0].legend()

func[0].axis(ymax=1)

func[1].set\_title(title1+'95'+title2)

func[1].set\_xlabel(xlabel)

func[1].set\_ylabel(ylabel)

func[1].plot(bins\_count[1:], pdf, color="green", label="PDF" )

x1,x2 = np.linspace(lB2,lB2),np.linspace(uB2,uB2)

func[1].plot(x1,y, label="Lower Boudary = {:.4f}".format(lB2))

func[1].plot(x2,y, label="Upper Boudary = {:.4f}".format(uB2))

func[1].legend()

func[1].axis(ymax=1)

func[2].set\_title(title1+'99'+title2)

func[2].set\_xlabel(xlabel)

func[2].set\_ylabel(ylabel)

func[2].plot(bins\_count[1:], pdf, color="green", label="PDF" )

x1,x2 = np.linspace(lB3,lB3),np.linspace(uB3,uB3)

func[2].plot(x1,y, label="Lower Boudary = {:.4f}".format(lB3))

func[2].plot(x2,y, label="Upper Boudary = {:.4f}".format(uB3))

func[2].legend()

func[2].axis(ymax=1)

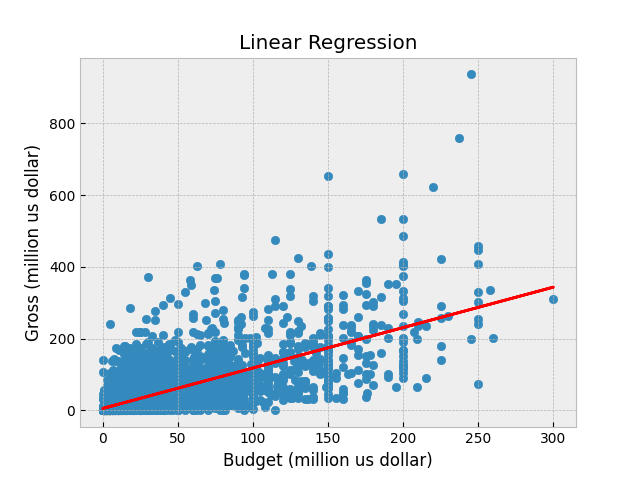
plt.show()

# Homework 5 : Linear Regression

## linear regression

Linear Regression หรือ การวิเคราะห์การถดถอยเป็นการศึกษาความสัมพันธ์ระหว่างตัวแปรตั้งแต่ 2 ตัวขึ้นไป ซึ่งได้แก่ตัว ประมาณการ (Predictor, X) และตัวตอบสนอง (Response, y) โดยเป็นความสัมพันธ์แบบเชิงเส้น (Linear)

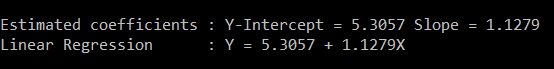
ชุดข้อมูลที่ผู้ศึกษาได้เลือกมาเป็นชุดข้อมูลของต้นทุนและรายได้ของภาพยนตร์ โดยได้กำหนดให้ ตัวประมาณการ (Predictor, X) เป็นค่าของงบประมาณในการสร้างภาพยนตร์ และตัวตอบสนอง (Response, y) เป็นรายได้ของภาพยนตร์ เมื่อค่าของชุดข้อมูลดังกล่าวไปคำนวณหา Linear Regression หรือ การวิเคราะห์การถดถอย แล้วนำมาวาดลงบนกราฟ Scatter plot จึงได้กราฟดังนี้



Linear Regression Graph

แกน X ของกราฟ : Budget (งบประมาณที่ใช้ผลิตภาพยนตร์) หน่วย ล้านดอลลาร์

แกน Y ของกราฟ : Gross (รายได้จากภาพยนตร์) หน่วย ล้านดอลลาร์



ค่าที่ได้จากกวิเคราะห์การถดถอย

จากกราฟ linear regression ที่ได้จากการคำนวณ จากชุดข้อมูลงบประมาณและรายได้จากการสร้างภาพยนตร์ โดยการกำหนดแกน X เป็นค่าประมาณการ คือ งบประมาณที่ใช้ผลิตภาพยนตร์ (Budget) และ แกน Y เป็นตัวตอบสนอง คือ รายได้จากภาพยนตร์ (Gross) จากการวิเคราะห์การถดถอย จะได้ค่าความชันอยู่ที่ 1.1279 ซึ่งเป็นค่าบวก จะสามารถสรุปได้ว่าค่าตอบสนองมีความสัมพันธ์กับค่าประมาณการสูง เช่น ถ้าภาพยนตร์มีงบประมาณที่ X จะมี รายได้ประมาณ y และเมื่อ ภาพยนตร์มีงบประมาณที่ x + 1 รายได้ภาพยนตร์ก็จะอยู่ที่ y + 1.1279 ตามสมการที่ได้จากรูปข้างต้น

## Source code

import numpy as np

from matplotlib import pyplot as plt

import pandas as pd

plt.style.use('bmh')

df = pd.read\_csv('moviesfilter.csv')

# budget gross company name

x = df['budget']

y = df['gross']

z = df['company']

budget = x.to\_list()

gross = y.to\_list()

company = z.to\_list()

# format data to million dollar

for i in range(0, len(budget)):

    budget[i] = budget[i]/1000000

for i in range(0, len(gross)):

    gross[i] = gross[i]/1000000

def regression\_line():

    # W is regression coefficients

    X = np.vstack((budget,np.ones(len(budget)))).T

    W = np.linalg.inv(X.T @ X) @ X.T @ gross

    plt.xlabel('Budget (million us dollar)')

    plt.ylabel('Gross (million us dollar)')

    plt.title('Linear Regression')

    plt.scatter(budget, gross)

    # z is Predicted vector

    z = X @ W

    plt.plot(budget, z, color='r')

    print()

    print('Estimated coefficients : Y-Intercept = {:.4f} Slope = {:.4f}'.format(W[1],W[0]))

    print('Linear Regression      : Y = {:.4f} + {:.4f}X'.format(W[1],W[0]))

    plt.show()

if \_\_name\_\_ == "\_\_main\_\_":

    regression\_line()