**Q1:**

1. **min**: = 37.0
2. **max**: = 100.0

np.min(mid\_score)

np.max(mid\_score)

1. **first quartile Q1**: the percentile score = 68.0

**median**: the percentile score = 77.0

**Third quartile Q3**: the percentile score = 83.0

np.percentile(mid\_score, 25)

np.percentile(mid\_score, 50)

np.percentile(mid\_score, 75)

1. **mean:**

round(np.mean(mid\_score), 3)

: = 76.715

1. **mode**: the score number that repeat most often = 77.0, 83.0

for score in mid\_score:

if count.has\_key(score):

count[score] += 1

else:

count[score] = 1

max = sorted(count.values())[len(count) - 1]

for score in count:

if count[score] == max:

print score

1. **empirical variance:**

round(np.var(mid\_score, ddof = 1), 3)

**Q2:**

1. **Compare the empirical variance before and after normalization.**

(where x is raw score to be standardized, μ is mean of the population, σ is standard deviation)

The empirical variance before normalization is 173.279, after normalization is 1.0.

mid\_score\_z = preprocessing.scale(mid\_score)

np.var(mid\_score\_z)

1. **Given original score of 90, what is the corresponding score after normalization?**

round((90 - np.mean(mid\_score)) / np.std(mid\_score, ddof = 1), 3)

1. **Pearson’s correlation coefficient between midterm scores and final scores is:**

round(np.corrcoef(mid\_score, final\_score)[1][0], 3)

= 0.544

1. **Covariance between midterm scores and final scores is:**

round(np.cov(mid\_score, final\_score)[1][0], 3)

**Q3:**

1. **the Jaccard coefficient of Citadel's Maester Library (CML) and Castle Black's library(CBL):**

round (float(58) / float(2 + 120 + 58), 3)

1. **the minkowski distance of the two vectors with regard to different h values:**
2. **h = 1 (Manhattan distance)** 6152

np.sum(np.fabs(CBL - CML))

1. **h = 2 (Euclidean)** 715.328

round(np.sqrt(np.sum((CBL - CML)\*\*2)), 3)

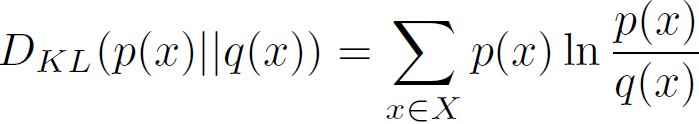
1. **h = 3 (Supremum = )** 170

np.max(np.fabs(CBL - CML))

1. **the Cosine similarity between Citadel's Maester Library (CML) and Castle Black's:**

round(np.sum(CML \* CBL) / (np.linalg.norm(CBL) \* np.linalg.norm(CML)), 3)

1. **the Kullback–Leibler divergence of these two libraries P(CML || CBL):**

 = 0.201

round(np.sum((CML / np.sum(CML)) \* np.log((CML / np.sum(CML)) / (CBL / np.sum(CBL)))), 3)

np.sum(CBL))))

**Q4:**

1. **the chi-square correlation value：**

sum = 150 + 40 + 15 + 3300 = 3505

bd = （150 + 40）\*（150 + 15）/ 3505 = 8.94436519258

bnd = (150 + 40) \* (40 + 3300) / 3505 = 181.055634807

nbd = (150 + 15) \* (15 + 3300) / 3505 = 156.055634807

bndb = (15 + 3300) \* (3300 + 40) / 3505 = 3158.94436519

(150 – 8.944) \* (150 – 8.944) / 8.944 + (40 – 181.056) \* (40 – 181.056) / 181.056 + (15 – 156.056) \* (15 – 156.056)/ 156.056 + (3300 – 3158.944) \* (3300 – 3158.944) / 3158.944 = 2468.183

bear\_diaper = 150  
bear\_nodiaper = 40  
nobear\_diaper = 15  
nobear\_nodiaper = 3300  
bear = bear\_diaper + bear\_nodiaper  
diaper = bear\_diaper + nobear\_diaper  
nobear = nobear\_diaper + nobear\_nodiaper  
nodiaper = bear\_nodiaper + nobear\_nodiaper  
sum = bear + nobear  
b\_d = float((bear \* diaper)) / float(sum)  
b\_nd = float((bear \* nodiaper)) / float(sum)  
nb\_d = float((nobear \* diaper)) / float(sum)  
nb\_nd = float((nobear \* nodiaper)) / float(sum)  
a = np.square(bear\_diaper - b\_d) / b\_d  
b = np.square(bear\_nodiaper - b\_nd) / b\_nd  
c = np.square(nobear\_diaper - nb\_d) / nb\_d  
d = np.square(nobear\_nodiaper - nb\_nd) / nb\_nd  
chi = a + b + c + d