CS306: Data Analysis and Visualization

Assignment 1

Name: *Pratvi Shah* Student ID: *201801407*

1. **Probability (Question 1)**

   (a) Compute probabilities:

$$\begin{aligned} P(E) &= P(More\_than\_one\_heads) = P(HHT) + P(HTH) + P(THH) + P(HHH) \\ &= 0.1 + 0.1 + 0.17 + 0.13 \\ &= 0.5 \end{aligned}$$

$$\begin{aligned} P(F) &= P(All\_coins\_are\_the\_same) = P(HHH) + P(TTT) \\ &= 0.13 + 0.13 \\ &= 0.26 \end{aligned}$$

$$\begin{aligned} P(E \cap F) &= P(All\_coins\_are\_heads) = P(HHH) \\ &= 0.13 \end{aligned}$$

$$\begin{aligned} P(E \cup F) &= P(More\_than\_one\_heads\_or\_all\_same) \\ &= P(TTT) + P(HHT) + P(HTH) + P(THH) + P(HHH) \\ &= 0.13 + 0.1 + 0.1 + 0.17 + 0.13 \\ &= 0.63 \end{aligned}$$

   (b) Addition Rule of probability states that the probability that E or F will occur is same as the sum of probability that E will occur and the probability that B will occur minus the probability that both E and F will occur simultaneously.

$$P(E \cup F) = P(E) + P(F) - P(E \cap F)$$

$$\begin{aligned} LHS &= P(E \cup F) \\ &= 0.63 \\ RHS &= P(E) + P(F) - P(E \cap F) \\ &= 0.5 + 0.26 - 0.13 \\ &= 0.63 \end{aligned}$$

   As we can see LHS = RHS hence, verified.

(c) Compute probabilities:

$$\begin{aligned} P(G) &= P(More\_than\_and\_equal\_to\_2\_tails) \\ &= P(HTT) + P(THT) + P(TTH) + P(TTT) \\ &= 0.17 + 0.1 + 0.1 + 0.13 \\ &= 0.5 \end{aligned}$$

$$\begin{aligned} P(H) &= P(Some\_coins\_different) = 1 - P(All\_coins\_same) \\ &= 1 - (P(HHH) + P(TTT)) \\ &= 1 - (0.13 + 0.13) \\ &= 0.74 \end{aligned}$$

$$\begin{aligned} P(G \cup H) &= P(All\_are\_not\_heads) = 1 - P(HHH) \\ &= 1 - 0.13 \\ &= 0.87 \end{aligned}$$

$$\begin{aligned} P(G \cap F) &= P(All\_coins\_are\_tails) = P(TTT) \\ &= 0.13 \end{aligned}$$

(d) Rule of independence states that two events are independent if probability of occurrence of one event does not affect the probability of occurrence of the other event.

$$P(G \cap H) = P(G)P(H)$$

$$\begin{aligned} LHS &= P(G \cap H) \\ &= P(Exactly\_two\_tails) \\ &= P(G) - P(G \cap H) \\ &= 0.5 - 0.13 \\ &= 0.37 \end{aligned}$$

$$\begin{aligned} RHS &= P(G) * P(H) \\ &= 0.5 * 0.74 \\ &= 0.37 \end{aligned}$$

As we can see LHS = RHS hence, verified.

2. **Conditional Probability (Question 2)**

E = Car bought has fault
A = Mechanic certified car to be faulty
B = Mechanic certified car to be good

(a) Car has fault before taking advice from mechanic = $P(E)$

$$\begin{aligned} P(E) &= fraction\_of\_cars\_having\_fault \\ &= 0.35 \end{aligned}$$

(b) Car had fault when mechanic certified it to be faulty = $P(E \mid A)$

$$\begin{aligned} P(E \mid A) &= \frac{P(E \cap A)}{P(A)} \\ &= \frac{P(A \mid E) * P(E)}{P(A \mid E) * P(E) + P(A \mid E') * P(E')} \\ &= \frac{0.94 * 0.35}{0.94 * 0.35 + 0.12 * 0.65} \\ &= \frac{0.329}{0.407} = 0.808 \end{aligned}$$

(c) Car had fault when mechanic certified it to be good = $P(E \mid B)$

$$\begin{aligned} P(E \mid B) &= \frac{P(E \cap B)}{P(B)} \\ &= \frac{P(B \mid E) * P(E)}{P(B \mid E) * P(E) + P(B \mid E') * P(E')} \\ &= \frac{0.06 * 0.35}{0.06 * 0.35 + 0.88 * 0.65} \\ &= \frac{0.021}{0.593} = 0.0354 \end{aligned}$$

3. **Bayes Theorem (Question 3)**

EA = Patient has disease A
EB = Patient has disease B
EC = Patient has headache
ED = Patient does not have any disease

(a) Given patient has headache probability that he has disease A = $P(EA \mid EC)$

$$\begin{aligned} P(EA \mid EC) &= \frac{P(EC \mid EA) * P(EA)}{P(EC \mid EA) * P(EA) + P(EC \mid EB) * P(EB) + P(EC \mid ED) * P(ED)} \\ &= \frac{19/20 * 1/20}{19/20 * 1/20 + 2/20 * 1/4 + 17/20 * 1/10} \\ &= \frac{0.0475}{0.0475 + 0.025 + 0.085} \\ &= \frac{0.0475}{0.1575} = 0.30159 \end{aligned}$$

(b) Given patient has headache probability that he has disease B = $P(EA \mid EC)$

$$\begin{aligned} P(EB \mid EC) &= \frac{P(EC \mid EB) * P(EB)}{P(EC \mid EA) * P(EA) + P(EC \mid EB) * P(EB) + P(EC \mid ED) * P(ED)} \\ &= \frac{2/20 * 1/4}{19/20 * 1/20 + 2/20 * 1/4 + 17/20 * 1/10} \\ &= \frac{0.025}{0.0475 + 0.025 + 0.085} \\ &= \frac{0.025}{0.1575} = 0.15873 \end{aligned}$$

(c) Given patient has headache probability that he has neither of the above disease = $P(ED \mid EC)$

$$\begin{aligned} P(ED \mid EC) &= \frac{P(EC \mid ED) * P(ED)}{P(EC \mid EA) * P(EA) + P(EC \mid EB) * P(EB) + P(EC \mid ED) * P(ED)} \\ &= \frac{17/20 * 1/10}{19/20 * 1/20 + 2/20 * 1/4 + 17/20 * 1/10} \\ &= \frac{0.085}{0.0475 + 0.025 + 0.085} \\ &= \frac{0.085}{0.1575} = 0.53968 \end{aligned}$$

(d) Given the patient has headache probability that he has both the diseases(A and B) = $P(EA \cap EB \mid EC)$

$$\begin{aligned} P(EA \cap EB \mid EC) &= \frac{P((EA \cap EB) \cap EC)}{P(EC)} \\ &= 0 \qquad (\because EA \cap EB = \phi) \end{aligned}$$

4. **Confidence interval (Question 4)**

For two independent variables the confidence interval of the difference of means uses the following formula:

$$S_p = \sqrt{\frac{(n_1 - 1) * s_1^2 + (n_2 - 1) * s_2^2}{n_1 + n_2 - 2}}$$

$$SE = S_p * \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}$$

$$Confidence\ \ Interval = (\mu_1 - \mu_2) \pm t_{n_1+n_2-2,\alpha/2} * SE$$

Here, the subscript 1 represents data of men and subscript 2 represents data of women. S_p forms the polled variance while SE is the standard error. We use t-test as the sample size is very less than the population size and α represents the significance level.

The above estimation of confidence interval is valid only when the variance of the two variables is same or has a ratio of $0.5 < \frac{\sigma_1}{\sigma_2} < 2$.

Size of the data = $n_1 = n_2 = 10$

Standard deviation of the data: $s_1 = 11.62755$, $s_2 = 10.98408$

Mean of the data: $\mu_1 = 22$, $\mu_2 = 21.5$

$\mu_1 - \mu_2 = 0.5$

$S_p = 11.31039$

$SE = 5.05816$

(a) Confidence interval based on the above formulas used $n_1 + n_2 - 2 = 18$ as degree of freedom and $\alpha = (1 - given\%)$

$\alpha = 0.05$
Final formula = $0.5 \pm 2.10092 * 5.05816 = 0.5 \pm 10.626789$

Confidence interval 95% = [-10.12679, 11.12679]

$\alpha = 0.01$
Final formula = $0.5 \pm 2.87844 * 5.05816 = 0.5 \pm 14.55961$

Confidence interval 99% = [-14.05961, 15.05961]

The difference in means used in calculating the confidence interval shows that the average income of men is greater than women by 0.5lpa. The confidence interval shows that with 95% surity the difference between the income of women and men will not be greater than 11.12679 lpa while to be sure by around 99% the maximum difference can be 15.05961 lpa.

(b) The confidence interval includes zero for both 95% and 99% confidence level. Also, the confidence interval shows the probable difference between the average salary of men and women and as the interval has zero value so, we can not say for sure that the men's income is really different from women's income.Hence, we cannot statistically infer that there is a difference between the incomes of men and women.

5. **Random data generation and analysis (Question 5)**

Given $\mu = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$, $\Sigma = \begin{bmatrix} 4 & 4 \\ 4 & 9 \end{bmatrix}$

For, this question code has also been provided in the link at the bottom of this pdf. I have made use of seed=22 to ensure consistency of results.

```
def gen_mean_var(runs,N):
    sample_mean=np.zeros(2)
    sample_cov=[[0,0],[0,0]]
```

```
sample_cov=np.array(sample_cov)
for i in range(runs):
    #generating the random numbers for given mean and variance
    data = np.random.multivariate_normal(mean, var, N)

    #sum up the mean to calculate the average
    sample_mean =sample_mean + np.array([data[:,0].mean(),data[:,1].mean()])

    #sum up the covariance to calculate the average
    sample_cov = sample_cov + np.cov(data[:,0],data[:,1])

#required sample mean for N samples and runs=runs
sample_mean= sample_mean/runs
#required sample covariance for N samples and runs=runs
sample_cov = sample_cov/runs
return data,sample_mean,sample_cov
```

(a) Scatter plot for 10 samples using the returned *data* with $runs = 1$ and $N = 10$:

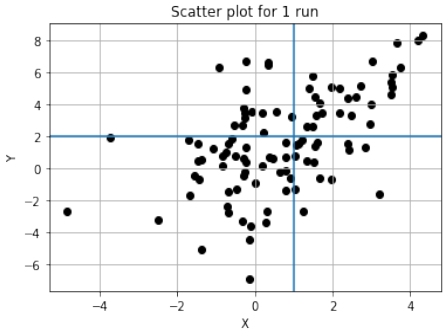


Figure 1: Scatter plot for single run in Q5 a.

(b) From part a) the mean and covariance matrix were to be calculated. The returned *sample_mean* and *sample_cov* are:

Sample mean: [0.80406214 2.06550656]

Sample covariance matrix: $\begin{bmatrix} 4.87658654 & 4.98429773 \\ 4.98429773 & 10.71705093 \end{bmatrix}$

(c) Average mean and covariance matrix were to be calculated for $runs = 10$ and $N = 10$. Following are the returned *sample_mean* and *sample_cov*:

Sample mean: [1.07206173 2.05486939]

Sample covariance matrix: $\begin{bmatrix} 4.06725384 & 3.76422331 \\ 3.76422331 & 8.24512189 \end{bmatrix}$

We could see that the sample covariance and sample mean gave better approximations to the original mean and variance when the number of runs were increased and later average was taken.

(d) Table for different values of N was to be constructed. Following is the output for the same:

	N	Mean_1	Covariance_1	Mean_10	Covariance_10
0	20	[1.147, 2.273]	[[5.879, 5.326], [5.326, 10.212]]	[1.303, 2.466]	[[3.88, 3.897], [3.897, 10.201]]
1	40	[0.472, 1.542]	[[2.375, 1.851], [1.851, 7.167]]	[1.003, 1.981]	[[3.395, 3.581], [3.581, 9.181]]
2	60	[0.824, 1.687]	[[3.6, 4.145], [4.145, 10.758]]	[0.988, 1.908]	[[4.177, 4.127], [4.127, 9.076]]
3	80	[0.962, 1.992]	[[3.706, 3.841], [3.841, 9.111]]	[0.906, 1.883]	[[4.43, 4.608], [4.608, 9.766]]
4	100	[0.915, 1.786]	[[3.807, 3.705], [3.705, 8.597]]	[0.976, 1.943]	[[4.215, 4.167], [4.167, 8.866]]
5	200	[0.997, 1.975]	[[4.281, 4.556], [4.556, 10.193]]	[0.965, 1.981]	[[4.261, 4.349], [4.349, 9.547]]
6	300	[0.968, 1.895]	[[3.967, 4.192], [4.192, 9.417]]	[1.018, 1.953]	[[4.155, 4.145], [4.145, 9.044]]
7	400	[0.972, 1.93]	[[3.71, 3.654], [3.654, 8.613]]	[1.011, 1.996]	[[3.936, 3.894], [3.894, 8.902]]
8	500	[0.902, 1.835]	[[3.961, 4.096], [4.096, 8.625]]	[0.961, 1.876]	[[4.053, 4.176], [4.176, 9.304]]

Figure 2: Table for sample_mean and sample_cov with different N.

In the above table Mean_1 and Covariance_1 represents value for $runs = 1$ while Mean_10 and Covariance_10 is for $runs = 10$. We could in general observe that the error in mean and covariance reduced as the number of samples(N) were increased.

We found that the effect of increasing the runs and later averaging brings out better result for the same number of samples. This is observed in case of both mean and covariance. The following table shows the calculated RMSE for mean and covariance.

	N	Mean_1 RMSE	Cov_1 RMSE	Mean_10 RMSE	Cov_10 RMSE
0	20.0	0.048	2.129	0.154	0.370
1	40.0	0.244	3.810	0.000	0.188
2	60.0	0.065	0.823	0.004	0.017
3	80.0	0.001	0.037	0.011	0.377
4	100.0	0.026	0.093	0.002	0.030
5	200.0	0.000	0.530	0.001	0.153
6	300.0	0.006	0.062	0.001	0.017
7	400.0	0.003	0.118	0.000	0.009
8	500.0	0.018	0.040	0.008	0.039

Figure 3: RMSE for mean and covariance in Q5 d.

Scatter plots for varying N are available in the code file link to which has been provided at the end of this pdf. Due to large number of figures the same has not been provided here. But as the number of samples increased the density of points near the mean i.e., (1,2) increased showing the effectiveness of larger samples size to approximate the given characteristics of the data.

6. **Monte Carlo simulation (Question 6)**

In this question, using ANOVA we basically compare whether the three samples taken from a normal distribution come from the same population or not(H_0).
For this we calculate the variance within the sample(SSE) and variance between the samples(SSB). Finally we consider the degrees of freedom of SSE and SSB to calculate the f value using:

$$F = \frac{MSB}{MSE}$$

$$SSB = \Sigma n_j(\bar{x}_j - \bar{x}) \qquad MSB = \frac{SSB}{degree\_of\_freedom(SSB)}$$

$$SSE = \Sigma\Sigma(x_{ij} - \bar{x}_j)^2 \qquad MSE = \frac{SSE}{degree\_of\_freedom(SSE)}$$

where, i iterates over the rows of A,B and C and j amongst A,B and C i.e., j takes 3 values and i takes values equal to the number of observations.
Constants given $\mu = 60kg$ & $\sigma = 12kg$.
The code for generating the table from the calculated values is available in the code link at the bottom of the pdf.

(a) The ANOVA table for the samples generated with $runs = 1$ is:

	d.o.f	sum_sq	mean_sq	F
SSB	2.0	215.501324	107.750662	0.813697
SSE	27.0	3575.369308	132.421085	0.000000

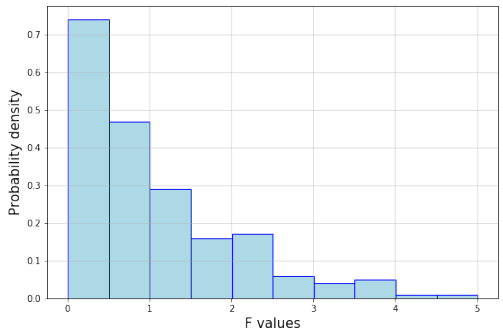
For this case **pvalue** $= 0.45379$, and the $fvalue$ is also lower than 1 hence, we can say that there is no significant difference between the three samples and therefore, **can not reject** the null hypothesis. We can conclude that the three samples are not from different populations.

(b) The F values were generated and stored for the next step in f_calc in the following code:

```
def anova_table(n,runs):
    np.random.seed(22)
    mu=60
    sigma=12
    f_calc=[]
    ans_335=0
    for i in range(runs):
        A=np.random.normal(mu,sigma,n)
        B=np.random.normal(mu,sigma,n)
        C=np.random.normal(mu,sigma,n)
        mean_a=A.mean()
        mean_b=B.mean()
        mean_c=C.mean()
        #xbar
        mu_total=(np.sum(A) + np.sum(B) + np.sum(C))/(len(A) + len(B) +
        len(C))

        #SSE and SSB
        sse = np.sum((A-mean_a)**2) + np.sum((B-mean_b)**2) +
        np.sum((C-mean_c)**2)
        ssb = len(A)*(mean_a - mu_total)**2 + len(B)*(mean_b -
        mu_total)**2 + len(C)*(mean_c - mu_total)**2

        #degree of freedom
```

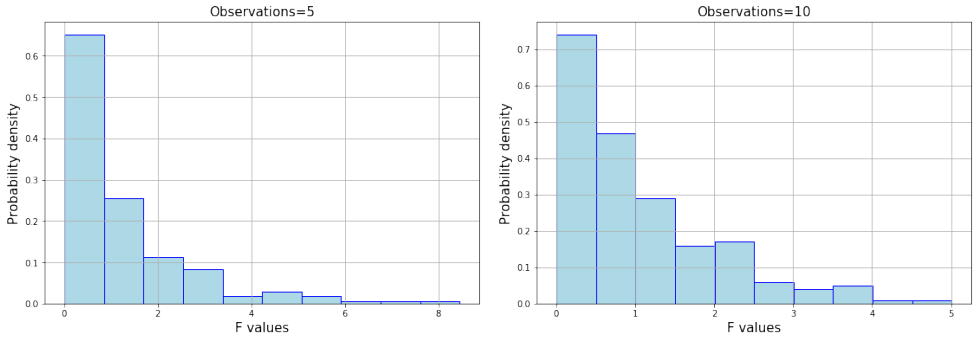
```
        dof_ssb= 3 - 1
        dof_sse = 3*(len(A)-1)

        #F calculated
        F=ssb*dof_sse/(sse*dof_ssb)

        #Array to maintain f value for all runs
        f_calc.append(F)

        #For calculating F>=3.35 proportion
        if F>=3.35:
            ans_335=ans_335+1

    return f_calc,ans_335/len(f_calc)
```

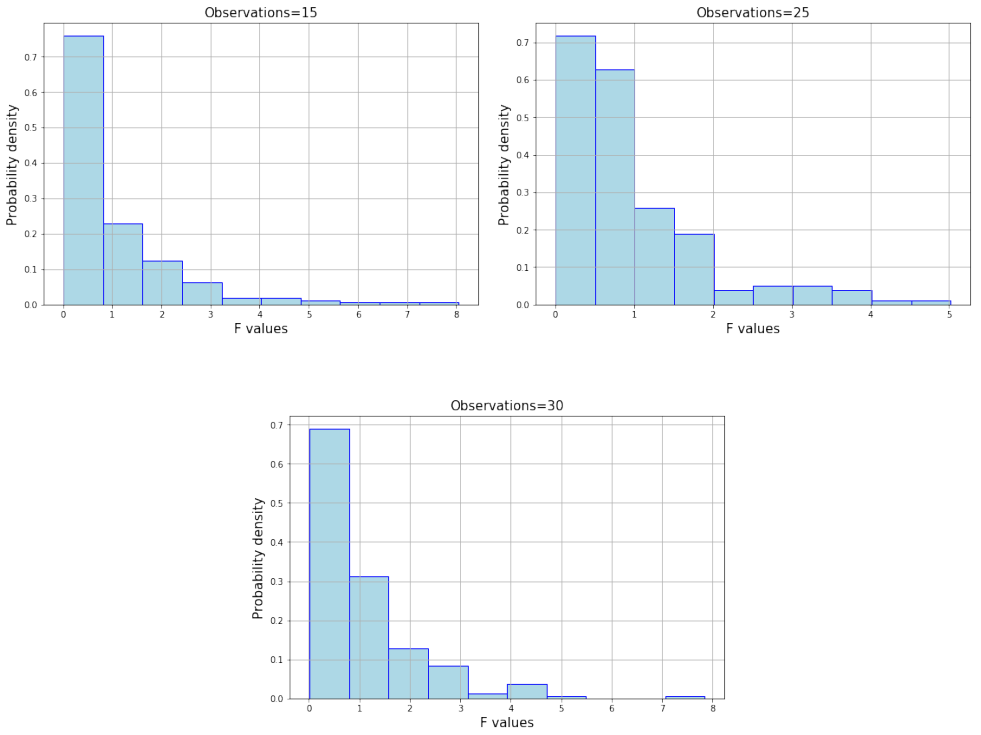
(c) The following plot was obtained for the F-values calculated in part b):



We can see from the plot that a very small portion of the observations have $fvalue$>3.35 and hence the calculated proportion that exceeds $F_{0.05} = 3.35$ is **0.035**. This shows that only 3.5% of the times the three samples are statistically different with 95% surety (as $\alpha = 0.05$ and hence 95%).

(d) We could get the following observations by changing the number of observations(n) in the code:





The F-value and p-value(for 1 run) and the ratio of observations $F_{0.05} > 3.35$ (for 200 runs) are given as follows:

$Observations(N)$	$p-value$	F_{value}	$F_{0.05} > 3.35$
5	0.872	0.139	0.075
10	0.454	0.814	0.035
15	0.415	0.898	0.055
25	0.600	0.515	0.030
30	0.497	0.705	0.040

We can see that the proportion is quite low also the F-values themselves are quite low. Furthermore, the pvalue calculated by using inbuilt function(for 1 run) also gave pvalues>0.05 hence, we cannot reject the null hypothesis.

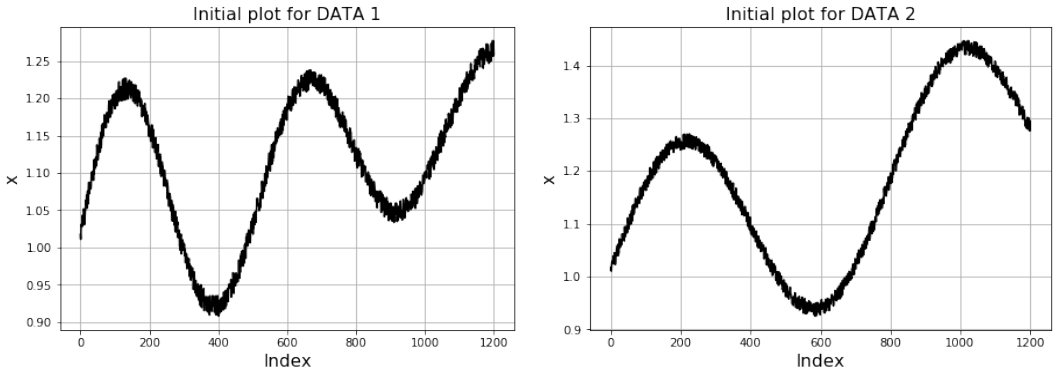
(e) F value is calculated by considering the variance within the samples(SSE) and between the samples(SSB). And the corresponding F-values obtained can be found from f-statistic table that corresponds to a given α value.

The value of α is the desired significance level. The critical F-value is calculated by taking the dof(degrees of freedom) of SSE and SSB, and finding the corresponding value in the f-table. Like for our case in part d) $F_c = 3.35$ when $\alpha = 0.05$ and dof(SSB)=2 and dof(SSE)=27. So, for any F value greater than F_c the null hypothesis can be rejected as the variance between the samples (SSB) comes out to be significantly greater than the variance within the sample. Hence, the samples can not come from the same population which is equivalent to rejecting the null hypothesis.

When we **decrease the α value** then the area under the curve reduces and hence, the F_c **increases**. This will allow greater proportion of data to be within the confidence interval and thereby increase the percentage of surety(=(1-α)*100) associated with the observation.

7. **Data Analysis : periodicity in data (Question 7)**

(a) From the below plot we can see that the data has a linear component and in the direction of that linear component there is a repetitive behaviour which can account for the periodicity in the data.



(b) The data appears to be periodic and to analyze the periodicity we can shift the data with a certain lag value and then perform the auto-correlation with the original signal. This auto-correlation will give us peaks(troughs and crests) when after one period the data overlaps with itself. This will only happen if the data is periodic and first cycle overlaps significantly with the next cycle and so on. So, the maximum difference between the two observed peaks in auto-correlation values will give us the periodicity of the data. This can be the difference between two crests or two troughs. While finding the peaks it is possible that due to noise in the data there are some local peaks centered around the prominent peaks, in that case we consider the maximum(in case of crests) and the minimum(in case of troughs) valued peaks as the prominent peaks and then proceed with the calculation of periodicity.

(c) The pseudo code for evaluating the periodicity of data (y) is:

- Normalize the data
- Evaluate the auto-correlation function (y_{corr}) of the two data with lag k (y_n and y_{n-k}) or shifting the data by k and then evaluating the correlation between the original and the shifted version.
- As the data is limited the max value of lag is equal to the size of the data (taken care by mode='full' in auto-correlation) as beyond that auto-correlation will be zero.
- The auto-correlation is symmetrical about zero so discard one half of the auto-correlated data
- Now the first value of y_{corr} represents the complete direct overlap of the data with itself i.e., lag=0, so we ignore the first peak and divide the rest to normalize the auto-correlation values
- Let the indices of the troughs be y_{neg_peaks}, evaluate the difference between the peaks to get the possible values of periodicity
- Let the indices of the crests be y_{pos_peaks}, evaluate the difference between the peaks to get the possible values of periodicity
- As periodicity can be due to the repetitive troughs or crests, so maximum of the periodicity calculated from the above two steps will be labelled as the periodicity of the data

(d) Following functions were used in the given order to perform the required analysis and the periodicity was calculated based on the peaks of the data:

```
#to produce plots of the given data
def initial_plots(x,name):
    plt.figure(figsize=[7,5])
    plt.plot(x,'k')
    plt.xlabel('Index',fontsize=15)
    plt.ylabel('x',fontsize=15)
    plt.title('Initial plot for '+str(name),fontsize=15)
    plt.grid(True)
    plt.show()


#To calculate the auto-correlation of the data
def get_autocorrelation(x):
    xmean = np.mean(x)
    x -= xmean
    autocorr = np.correlate(x, x,mode='full')
    n=autocorr.size

    #Filtering the positive part as it is symmetrical
    #Dividing by the first value after that to get autocorrelation
    #between 0 and 1
    temp = autocorr[int(n/2):]/autocorr[int(n/2)]
    return temp

#To get the plot for auto-correlation v/s lag to notice the peaks
def corr_plot(x,name):
    data_corr=get_autocorrelation(x)
    plt.figure(figsize=[7,5])
    plt.plot(data_corr,'k')
    plt.xlabel('Index',fontsize=15)
    plt.ylabel('Autocorrelation',fontsize=15)
    plt.title('For understanding periodicity of '+str(name),fontsize=15)
    plt.grid(True)
    return data_corr

#To calculate periodicity
neg_peak=[]
pos_peak=[]
for i in range(1,len(data_corr)-1):
    if(data_corr[i]>0 and data_corr[i]>data_corr[i-1] and
                            data_corr[i]>data1_corr[i+1]):
        pos_peak.append(i)
    elif(data_corr[i]<0 and data_corr[i]<data_corr[i-1] and
                            data_corr[i]<data_corr[i+1]):
        neg_peak.append(i)

#Periodicity
#There were a few steps involved before calculating periodicity
#but they are hard coded for the two data sets. (code in the .ipynb file)
```

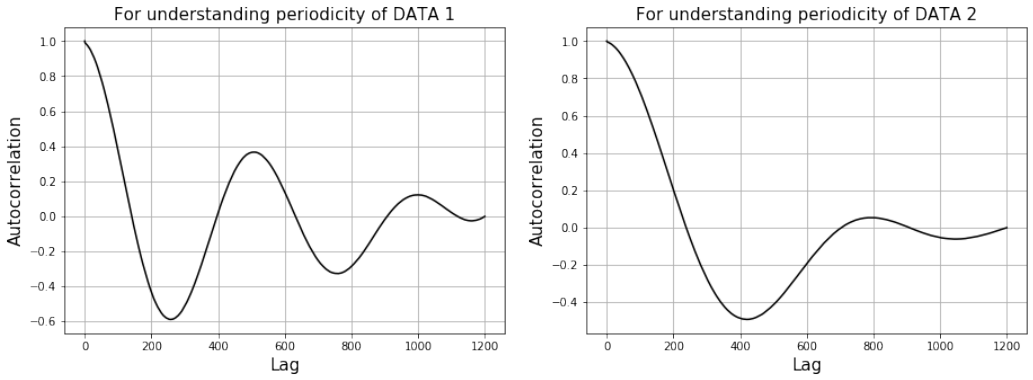
```
#neg_period= max periodicity calculated using the troughs
#pos_period= max periodicity calculated using the crests
period= max(neg_period, pos_period)
```

.

The steps mentioned above were implemented for both the data files and the plots for the same are attached below. Also, we could calculate the periodicity of the two data.

DATA 1
Trough: [254, 257, 264, 745, 755, 758, 760, 1153, 1156, 1162]
Crest: [500, 507, 510, 513, 992, 999, 1001, 1003, 1011]
Prominent maxima: [507, 1001]
Prominent minima: [257, 755]
Periodicity of DATA 1: 498

DATA 2
Trough: [413, 417, 422, 1038, 1040, 1043, 1047, 1054, 1058]
Crest: [789, 791, 796, 801]
Prominent maxima: 796
Prominent minima: [422, 1047]
Periodicity of DATA 2: 796



The periodicity calculated is the longest period after which the data will repeat. There can be other short periods within a given cycle.

To view the codes for Q5-Q7 open the given link in google colab:
`https://colab.research.google.com/drive/1ZzCpXPTn5eVG7rI2LqZeWqJn08udtekl?usp=sharing`