Lab 1 Report

Math 391 - Introduction to Modern Physics Lab
Professor Wayne Lau
University of Michigan

Jinyan Miao

Fall 2022

Contents

Lab 1 - Statistics

Introduction

Statistics methods are important for analyzing real-world observations and making reliable interpretations of experimental data. Since many topics in physics involve analyzing experimental data and generalizing laws of physics through statistical interpretation, a good background in statistics will help us to start a career in physics, and other subjects in the STEM field. Lab 1 in Physics 391 is designed to help us to understand the statistical limitation of using sample distribution to approximate population distribution. We collect sets of data for coin tossing, dice rolling, and salt counting, then we compare our observations with the theoretically expected values. The comparisons will focus on the sample mean and sample variance, and through those comparisons, we can get a deeper understanding of how the sample mean and variance can be used to interpret the population distribution. We will also find the conditions required for approximating a probability distribution using a normal distribution.

Experimental setup and data collection

Lab 1 of Physics 391 consists of three sub-experiments:

1. The first experiment is coin tossing. 10 coins (pennies) are placed in a plastic champagne flute, we vigorously shake the cup and dump the coins into a tray, then count the number of heads that appear on top. The coin tossing procedure is performed 50 times, and a set of data is collected, indicating the number of heads each time. In this experiment, the coins are assumed to be fair, that is the probability of getting a head for each coin is exactly 1/2.

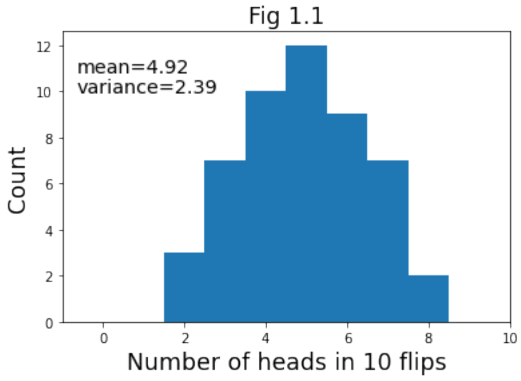
2. The second experiment is dice rolling. 2 dice, each has 6 faces and the faces are numbered from 0 to 5. The two dice are rolled 50 times. A set of data is collected, indicating the sum of the two dice for each roll. In this experiment, the dice are assumed to be fair, that is the probability of getting each face of a dice is exactly 1/6.

3. The third experiment is salt counting. We measure the number of counts of the radioactive decay event recorded by the Spectrum Techniques ST360 Geiger counter system when exposed to canisters of salt for a duration of 8 seconds. We collect two sets of data in this experiment, the first set of data contains the counts indicated by the Geiger counter when it is surrounded by 6 canisters of table salt (NaCl), and the second set of data contains the counts indicated by the Geiger counter when it is surrounded by 6 canisters of sodium-free salt substitute (KCl). 50 counts are recorded for each type of salts. In this experiment, we are interested in determining whether KCl is more radioactive than the average room background. If KCl is more radioactive, we expect to see more decay events occur in 8 seconds for KCl than that for NaCl.

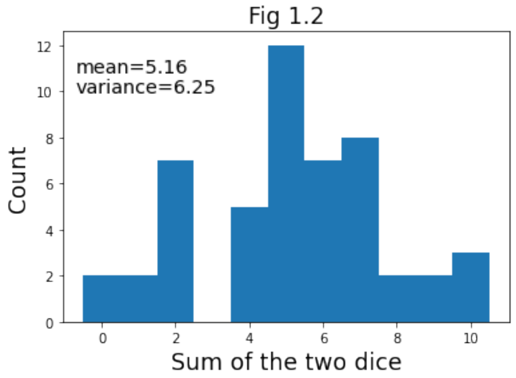
The experiment collected data is attached at the end of this text.

Visualizing the data

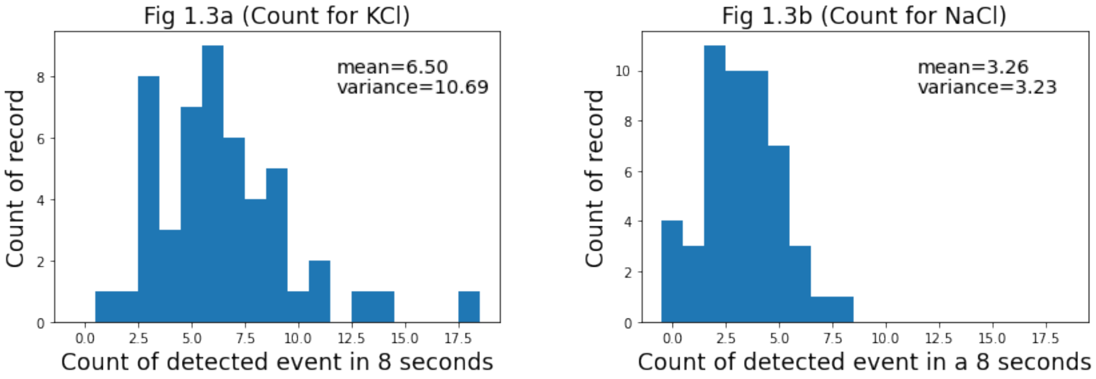
For the coin tossing experiment, the following figure (Fig 1.1) plots the total count of the number of heads in ten flips.



For the dice rolling experiment, the following figure (Fig 1.2) plots the count of the sum of the two dice.



For the salt counting experiment, the figure on the left (Fig 1.3a) plots the count recorded by the Geiger counter for the KCl in the 50 8-seconds intervals, and the figure on the right (Fig 1.3b) plots the count recorded by the Geiger counter for the background (NaCl) in the 50 8-seconds intervals.



The following table display some important statistics of the collected data. Quantities are calculated using formulas provided on the Physics 391 Lab Manual.

Experiment	Sample Mean	Sample Variance	Population Mean	Population Variance
Coin tossing	4.92	2.39	5.00	2.50
Dice rolling	5.16	6.25	5.00	5.83
Count for KCl	6.50	10.69	–	–
Count for NaCl	3.26	3.23	–	–

Here we note that the population means and population variance of the count of salts are unknown. Since, according to equation (8) and equation (16) from the Lab Manual, the calculation of the variance of sample means and variance of sample variance involves the population variance, we are not able to directly calculate those for the count of salts experiment. However, we can use the sample variance to estimate the population variance for the count of salts, in which case we can substitute the population variance by sample variance to calculate the variance of sample means and the variance of sample variance for the count of salts experiment. And we also do a similar calculation for the coins experiment and the dice experiment just for comparison.

	Variance of Sample Mean	Variance of Sample Variance
Experiment	(using population variance)	(using population variance)
Coin tossing	0.05	0.23
Dice rolling	0.12	0.96

	Variance of Sample Mean	Variance of Sample Variance
Experiment	(using sample variance)	(using sample variance)
Coin tossing	0.05	0.21
Dice rolling	0.13	1.13
Count for KCl	0.21	4.88
Count for NaCl	0.06	0.49

Analyzing the data

In this section, we will analyze the result of each experiment separately.

For the coin tossing experiment, the sample mean we obtained is 4.92 heads per 10 flips, compared to the population mean of 5.00 heads per 10 flips. The variance of sample means, calculated using population variance, is 0.05, and hence the standard deviation of the sample means is around 0.22. Here we see that the difference between the sample mean and the population mean is only 0.08, which is less than 1 standard deviation of the sample means. This shows that our sample mean is within the statistical expected dispersion from the population mean. On the other hand, the sample variance of our data is 2.39, and the population variance is 2.50, so the difference between the two is 0.11. The variance of the sample variance, calculated using the population variance, is 0.23, so the standard deviation of the sample variance is 0.48. We see that our sample variance is also less than 1 standard deviation of the sample variance, which indicates that our sample variance is within the statistical expected dispersion from the population variance. We also notice that the difference in the variance of sample means calculated using the two methods, using the sample variance and using the population variance, is small enough to be neglected, and the difference in the variance of the sample variance calculated using the two methods (0.23 versus 0.21) is also statistically insignificant. These results indicate that our sample variance is a reasonable estimation of the population variance, and also suggest that our experimental results are within statistical expectation.

For the dice rolling experiment. The sample mean value of the sum of the two dice is 5.16, compared to the population mean of 5.00. The variance of sample means, calculated using the population variance, is 0.12, and hence the standard deviation of the sample means is around 0.35. Therefore, the difference between the sample mean and the population mean is within 1 standard deviation, indicating that our sample mean is within the statistical expected dispersion from the population mean, and hence our sample mean is a good estimation of the population mean. On the other hand, our sample variance is 6.25, and the population variance is 5.83. The variance of sample variance calculated using the population variance is 0.96, which implies the standard deviation of the sample variance is 0.97. So we conclude that our sample variance is within 1 standard deviation of the sample variance, and hence it is also statistically well-expected. Notice that the variance of the sample means and the variance of the sample variance in this experiment are much greater than that in the coin tossing experiment, and this statistical phenomenon is also illustrated by the shape of the sample distributions in Fig 1.1 and Fig 1.2. The shape of the distribution shown in Fig 1.1 for the coin tossing experiment is narrower than that in Fig 1.2 for the dice rolling experiment. This suggests that, even though we have the same population mean for the two experiments, the distribution of the two experiments is rather different, and we can get insights of the shape of the distribution by examining the population variance. We also see that the variance of sample means calculated using the sample variance is very close to that calculated using the population variance (0.12 versus 0.13), and the difference between the variance of the sample variance calculated using the two methods (0.96 versus 1.13) is also small. Therefore, we conclude that the sample variance in this experiment also serves as a good estimator for the population variance. Furthermore, both the coin tossing experiment and the dice rolling experiment suggest that one can use the sample mean and sample variance to estimate the population mean and population variance when the sample size is large enough, and as a result, the sample variance also predicts the shape of the population distribution as well as the shape of the sample distribution.

For the salts counting experiment, we are interested in whether the number of decay events for KCl is greater than that of NaCl. We start by computing the difference in the means of counts, let X denote the dataset of KCl, and let Y denote the dataset of NaCl:

$$\Delta = \overline{X} - \overline{Y} = 6.50 - 3.26 = 3.24$$

When comparing the difference between means, assuming the two datasets are independent of each other, then the variances of the two datasets add, hence we write:

$$\mathbb{V}(X - Y) = \mathbb{V}(X) + \mathbb{V}(Y) = 10.69 + 3.23 = 13.92$$

Now we can compute the standard deviation for $X - Y$:

$$\sigma = \sqrt{\mathbb{V}(X - Y)} = \sqrt{13.92} = 3.731$$

To determine whether the difference between X and Y is significant, we need to examine the variance of the sample means for $X - Y$:

$$\mathbb{V}\left(\overline{X - Y}\right) = \mathbb{V}\left(\overline{X}\right) + \mathbb{V}\left(\overline{Y}\right) = 0.21 + 0.06 = 0.27$$

Now we can compute the standard deviation of sample means for $X - Y$:

$$\sigma_d = \sqrt{\mathbb{V}\left(\overline{X - Y}\right)} = \sqrt{0.27} = 0.52$$

Here we can perform the 2-sample t-test for the two datasets X and Y, and the test statistic t is calculated as the following:

$$t = \frac{\Delta}{\sigma_d} = \frac{3.24}{0.52} = 6.25$$

If the two population distributions for the two sample datasets have the same mean, then one would expect to have 0 being captured within 3 σ_d from Δ. However, the test statistic t indicates that 0 is 6.25 σ_d away from Δ, implying a low probability of equal population means for the count of decay events of the two salts. In particular, denoting means of the population distribution of KCl decay count and NaCl decay count as $\mu(X)$ and $\mu(Y)$, respectively, the one-sided 2-sample t-test statistic suggests that that p-value for $\mu(X) > \mu(Y)$ is given by:

$$p\left(\mu(X) > \mu(Y)\right) < 0.0001 < 0.05$$

hence we can reject the null hypothesis that $\mu(X) \not> \mu(Y)$, and we conclude that there is statistical evidence that the population mean of KCl decay event is greater than that of NaCl, which implies that the KCl is more radioactive than the background (NaCl). Furthermore, we want to check how good our variance for $X - Y$ is, which requires us to examine the variance of the sample variance of $X - Y$, that is, we write:

$$\mathbb{V}\left(\mathbb{V}(X - Y)\right) = \mathbb{V}(\mathbb{V}(X)) + \mathbb{V}(\mathbb{V}(Y)) = 4.88 + 0.49 = 5.37$$

so the standard deviation for the variance of $X - Y$ is given by:

$$\sigma_v = \sqrt{\mathbb{V}\left(\mathbb{V}(X - Y)\right)} = \sqrt{5.37} = 2.32$$

compare the value of $\mathbb{V}(X - Y)$ in our experiment, we see that σ_v is rather small, hence the value of $\mathbb{V}(X - Y)$ in our experiment is reliable.

Normal Approximation

The underlying probability distribution for the coin tossing experiment is in fact a binomial distribution as mentioned in the Lab Manual. Even though the probability distribution for the two-dice rolling experiment is not exactly binomial, a binomial distribution still serves as an adequate approximation for the probability distribution of multi-dice rolling. The underlying probability distribution for the salts counting experiment is the Poisson distribution. On the other hand, by the Central Limit Theorem, both the binomial distribution and the Poisson distribution can be approximated by a normal distribution in many circumstances, say, the sample size is large enough. In this section, we will investigate how normal our sample distributions are.

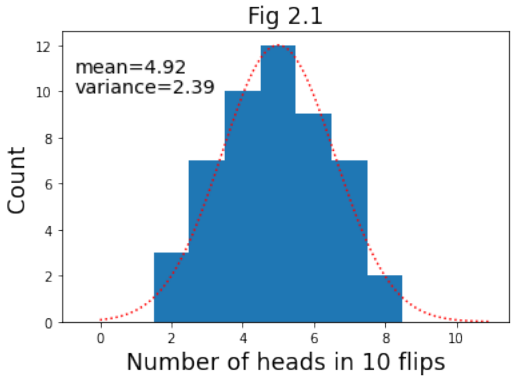


Fig 2.1 plots the sample distribution of the coin tossing experiment. The red curve is the normal distribution with the population mean and variance of the coin tossing population. The amplitude of the normal distribution curve is set to be the maximum of counts in our sample data. We see that the red curve gives an excellent approximation for the shape of our sample distribution, which suggests that normal distribution is indeed a good approximation for the binomial distribution when the sample size is large enough. The sample size in our experiment is 50.

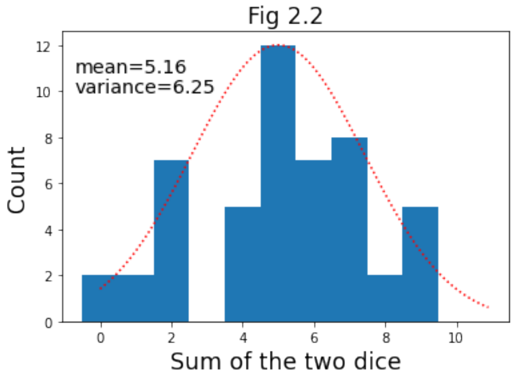
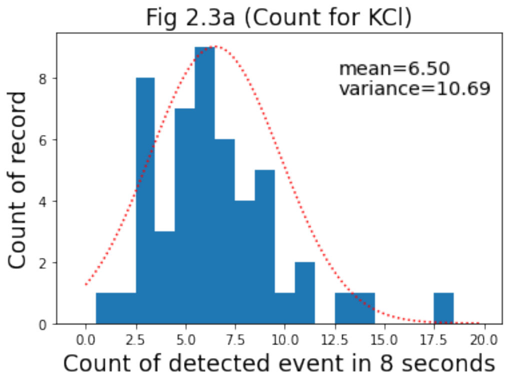


Fig 2.2 plots the sample distribution of the dice rolling experiment. The red curve is the normal distribution with the population mean and variance of the sum of the two-dice rolling population. The amplitude of the normal distribution curve is set to be the maximum of sums in our sample data. In our sample distribution, there is no count record at a sum equal to 3, which makes our model appears to be a little bimodel. Nevertheless, the overall shape of our sample distribution is still outlined by the red curve, and we expect that the red curve will give an even better approximation for the sample distribution if we collect more sample data.



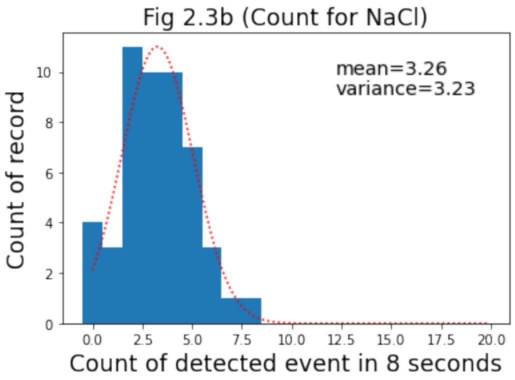


Fig 2.3a and 2.3b plot the sample distribution for the salts counting experiment. The red curves in both plots are normal distribution curves generated by using the sample mean and variance, with amplitude defined by the maximum of the count of record. Note that one should have used the population mean and variance to generate the normal distribution, but in our case the population distribution is unknown, and as mentioned above, the sample means and variance give a good approximation for the population means and variance when the sample size is large enough, then using the sample mean and variance to generate the normal distribution curve is adequate in our case. We observe that both sample distributions appear to be skewed to the right, which is one of the natures of our model as we do not have negative counts. The red curves do give an approximation for the shape of our sample distribution, except it does not demonstrate the skewness nature of our model. This result verifies the statistical fact that normal distribution is a good approximation for the Poisson distribution only when the population mean of the underlying Poisson distribution is sufficiently large, in which case the skewness of the sample distribution is not significant.

Lastly, we see that the sample variance is approximately equal to the sample mean for the NaCl sample distribution, and this is predicted by the nature of the underlying Poison distribution, which has equal population mean and variance. However, for the KCl sample distribution, the difference between the sample mean and sample variance is significant. This is probably caused by the few rare events occurring at the right-hand tail of the distribution, which increases the variance of the distribution, but it also suggests that our KCl sample distribution is not a good approximation of the population distribution for the count of KCl decay events.

Summary

In this text, we have justified, by using our experiment data, that sample means and sample variance give a good approximation for the population distribution when the sample size is sufficiently large. We have also verified that KCl is more radioactive than the background by performing a 2-sample t-test for the KCl and the NaCl decay-event-counting datasets. Lastly, we have also evaluated whether the normal approximation is suitable for the 3 probability distributions in our experiment, and concluded that normal approximation is adequate for binomial distribution when the sample size is sufficiently large, and adequate for Poisson distribution when the population mean of the Poisson distribution, and the sample size, is sufficiently large. These results, as one should keep in mind, will play important roles when interpreting the phenomena observed in our future modern physics experiments.

Experiment Data

Index	**Coins**	**Dice**	**NaCl**	**KCl**
0	5	2	5	4
1	3	2	5	3
2	5	5	4	5
3	7	7	3	7
4	3	6	5	6
5	7	4	2	6
6	4	5	2	7
7	7	6	0	3
8	6	2	2	7
9	7	4	3	8
10	5	10	2	1
11	6	4	4	3
12	6	6	7	5
13	3	5	3	7
14	2	5	2	3
15	4	5	6	8
16	2	10	8	6
17	7	5	4	11
18	5	7	0	5
19	5	6	6	2
20	4	9	3	3
21	5	2	0	7
22	5	5	4	5
23	5	7	2	9
24	5	0	4	9
25	2	8	3	6
26	6	7	4	6
27	7	4	1	9
28	5	5	0	5
29	4	1	2	8
30	3	6	1	9
31	3	7	4	5
32	6	7	5	6
33	3	8	3	8
34	7	5	1	3
35	4	2	6	11
36	5	1	3	13
37	6	6	4	9
38	8	5	3	6
39	4	10	3	10
40	4	9	2	4
41	4	0	2	14
42	5	2	3	19
43	6	2	4	6
44	6	5	4	6
45	3	5	2	3
46	6	7	2	7
47	8	7	5	4
48	4	4	5	5
49	4	6	5	3

Code

The code for computing statistics of the data sets is attached.

```python
import pandas as pd
import numpy as np
import os

data_df = pd.read_csv('data/StatisticsLabCollectedData.csv',header=0, index_col=0)
anal_data_df = pd.read_csv('data/StatisticsLabCollectedData.csv',header=0, index_col=0)

means = [np.mean(data_df[column_name].values) for column_name in data_df.columns]
stdev = [np.std(data_df[column_name].values) for column_name in data_df.columns]
vars = [np.var(data_df[column_name].values) for column_name in data_df.columns]
anal_data_df.loc['means'] = means
anal_data_df.loc['stdev'] = stdev
anal_data_df.loc['vars'] = vars

def varSampleMean(N, sig2):
  """
  N: integer, sample of means
  sig2: float, population variance
  """
  return (1/N)*sig2

def coinTheoMean(p, n):
  """
  n: integer, number of trial
  p: float, probability of head
  """
  return p*n

def dieTheoMean(n_f):
  """
  n_f: integer, number of faces on each die
  """
  return n_f-1*1.0

def coinTheoVar(p, n):
  """
  p: float, probability of head
  n: integer, number of trial
  """
  return p*(1-p)*n

def dieTheoVar(n_f):
  """
  n_f: integer, number of faces on each die
  """
  return (n_f*n_f - 1)/6

def coinVarOVar(sig2, N, p):
  """
  sig2: float, population variance
  N: integer, number of trials
  p: float, probability of tossing a head
  """
  return (2*(sig2**2))/(N-1)+((1-6*p*(1-p))*sig2)/N


def dieVarOVar(sig2, n_f, N):
  """
  n_f: integer, number of faces on each die
  sig2: float, population variance
  N: integer, number of trials
  """
  return (2*sig2*sig2)/(N-1) - ((n_f*n_f+1)*sig2)/(10*N)


def saltVarOVar(sig2, N):
  """
  sig2: float, population variance
  N: integer, number of trials
  """
  return (2*sig2*sig2)/(N-1) + (sig2/N)


diffMeansKCLsNACL = (anal_data_df.loc['means']['KCl']-anal_data_df.loc['means']['NaCl'])

def plotHist(x_name, bins_arr, xlabel, ylabel, tit, pos,
             Gau=False, GauTMean=0, GauTVar=0, GauRange=np.arange(0, 10, 0.1) ):
  """
  x_name: string, name of data to be plot
  bins_arr: array, bins numbers
  xlabel: string, x-axis label
  ylabel: string, y-axis label
  tit: string, title of the plot
  pos: float, horizontal position (in fraction) of the annotation
  Gau: boolean, wheter overplot the gaussian distribution
  """
  x = data_df[x_name]
  n, bins, patches = plt.hist(x,bins=bins_arr,histtype='bar', align='left')
  plt.xlabel(xlabel,fontsize='xx-large')
  plt.ylabel(ylabel, fontsize='xx-large')
  plt.title(tit, fontsize='xx-large')
  plt.annotate('mean=%.2f'%anal_data_df.loc['means'][x_name], (pos,0.8),
               xycoords='figure fraction', fontsize='x-large')
  plt.annotate('variance=%.2f'%anal_data_df.loc['vars'][x_name], (pos,0.75),
               xycoords='figure fraction', fontsize='x-large')
  if Gau==True:
    x_data = bins[0:len(bins)-1]
    x_data = x_data + 0.5*(x_data[1] - x_data[0])
    x_gaussian = GauRange
    sigma_gaussian_curve = np.sqrt(GauTVar)
    mean_gaussian_curve = GauTMean
    amplitude = np.max(n)
    params_histogram, params_covariance = opt.curve_fit(gaussian_model, x_data, n, p0=[5,4,10])
    plt.plot(x_gaussian, gaussian_model(x_gaussian, mean_gaussian_curve, sigma_gaussian_curve,amplitude),'r',ls=':')
    plt.show()


from prettytable import PrettyTable

data_Tab = PrettyTable(["Experiment", "Sample Mean",
                        "Sample Variance", "Population Mean",
                        "Population Variance"],
                       digits=3, round=True)
data_Tab.add_row(["Number of head in 10 flips",
                  anal_data_df.loc['means']['Coins'].round(2),
                  anal_data_df.loc['vars']['Coins'].round(2),
                  round(coinTheoMean(0.5, 10),2),
                  round(coinTheoVar(0.5,10),2)])
data_Tab.add_row(["Sum of two dice",
                 anal_data_df.loc['means']['Dice'].round(2),
                 anal_data_df.loc['vars']['Dice'].round(2),
                 round(dieTheoMean(6),2),
                 round(dieTheoVar(6),2)])
data_Tab.add_row(["Count for KCl",
                  anal_data_df.loc['means']['KCl'].round(2),
                  anal_data_df.loc['vars']['KCl'].round(2),'-','-'])
data_Tab.add_row(["Count for NaCl",
                  anal_data_df.loc['means']['NaCl'].round(2),
                  anal_data_df.loc['vars']['NaCl'].round(2),'-','-'])

data_var_Tab = PrettyTable(["Experiment",
                            "Variance of Sample Mean (using population variance)",
                            "Variance of Sample Variance (using population variance)",
                            "Variance of Sample Mean (using sample variance)",
                            "Variance of Sample Variance (using sample variance)", ], digits=3, round=True)
data_var_Tab.add_row(
        ["Number of head in 10 flips",
         round(varSampleMean(50,coinTheoVar(0.5,10)),2),
         round(coinVarOVar(coinTheoVar(0.5,10), 50, 0.5),2),
         round(varSampleMean(50,anal_data_df.loc['vars']['Coins']),2),
         round(coinVarOVar(anal_data_df.loc['vars']['Coins'], 50, 0.5),2),])
data_var_Tab.add_row(
        ["Sum of two dice",
         round(varSampleMean(50,dieTheoVar(6)),2),
         round(dieVarOVar(dieTheoVar(6), 6, 50),2),
         round(varSampleMean(50,anal_data_df.loc['vars']['Dice']),2),
         round(dieVarOVar(anal_data_df.loc['vars']['Dice'], 6, 50),2)])
data_var_Tab.add_row(
        ["Count for KCl",
         '-',
         '-',
         round(varSampleMean(50,anal_data_df.loc['vars']['KCl']),2),
         round(saltVarOVar(anal_data_df.loc['vars']['KCl'], 50),2)])
data_var_Tab.add_row(
        ["Count for NaCl",
         '-',
         '-',
         round(varSampleMean(50,anal_data_df.loc['vars']['NaCl']),2),
         round(saltVarOVar(anal_data_df.loc['vars']['NaCl'], 50),2)])
```