Statistical Analysis, Simulation, and Modeling
Mid-term project

Biligsaikhan Khurtsbayar
26002204605

1. Introduction

The Wiener process is a mathematical model used to describe the random motion of particles. As we explored stochastic processes in this course, we learned to simulate and model such random processes with the help of various statistical functions and coding libraries. This poster will demonstrate the methodologies used in generating a Wiener process simulation and analyzing the resulting data.

2. Method

2.1 Libraries/ external resources:

- numpy – Used for its random number generation, numerical calculations
- matplotlib – Used for its visualization capabilities
- scipy – Used for its statistical function
- statsmodels – Used for z-test, and q-q plot

2.2 Wiener process:

Every student has their individual parameters for creating a simulation of Wiener process determined by their birth date.
As I am born on November 23, my parameters will be the following:

```
mu = -11.0
sd = 2.3*2
```

In Wiener process, **mu** represents the drift of the process. As for **sd**, it controls the volatility/ the magnitude of its random fluctuations. In this projects, this can be seen more clearer in the following formula:

```
u = np.random.random(99)
for value in u:  # get both index and value of u
    y.append((sd * np.sqrt(2) * erfinv(2 * value - 1))+mu)
x =np.cumsum(y)
```

Moreover, I have chosen to have 100 steps for each Wiener process to make it consistent with the 100 simulations we had to create for the project.

3. Data

The simulation data obtained using the formula given in section 2.2 can be seen in the following two figures.

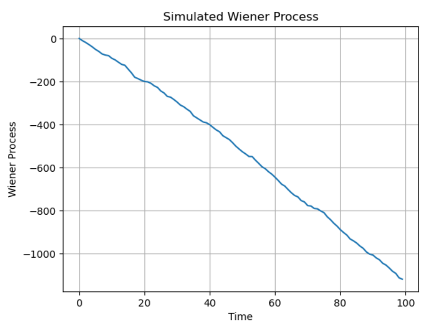


Figure 1. Singular simulated Wiener process

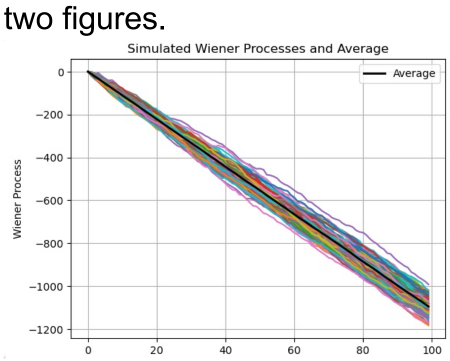


Figure 2. 100 simulated Wiener process in one graph

For further analyzing purposes the simulated data from 100 Wiener processes are saved in the following file as numpy array.

```
np.save('simulations.npy', all_simulations)
```

4. Analysis

4.1 Data reduction

To analyze the effect of **mu** and **sd** overtime on the Wiener process, we need to strip the data and take a closer look.
In this case, we take the time step 5 and 60 values from all 100 simulations and make two samples A and B ready for analysis.

4.2 Probability estimate analysis

By using the estimated **mu** and **sd** values of the samples A and B, we can create likelihood estimator for both samples. In this case, we used negative log likelihood estimator. it is considered that lower the value of negative log likelihood's value is, it is more likely to occur.

```
def neg_log_likelihood(params, data):
    mean, std_dev = params
    return -np.sum(stats.norm.logpdf(data, loc=mean, scale=std_dev))
```

4.3 Mean equivalence test

The mean equivalence test for the two samples were done by using z-test as the two samples A and B's standard deviation was accessible/known and sample sizes were over 30.
This test was also used in equivalence testing samples A and C.

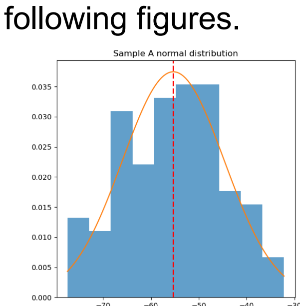
4.4 Normal distribution assessment

The sample C is created by applying data augmentation method of adding uniformly random numbers to the sample A. The resulting sample C's normality can be assessed by plotting its quantiles against the theoretical quantiles of normal distribution in q-q plot.

5. Result

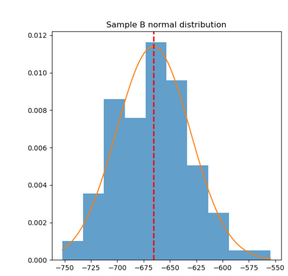
5.1 Likelihood results

In the following two distributions, as the standart deviation is lower in the sample A, we can see that the estimator estimates that the mean is more likely to occur in sample A than the mean of the sample B. We can see this result in the following figures.




```
Sample t5 - MLE Parameters (mean, std dev): -55.30956271050426 10.648349914420896
Estimator for Sample t5: 378.43434757429736
```

Figure 3. Sample A normal distribution




```
Sample t60 - MLE Parameters (mean, std dev): -665.726984243493 35.01560765785131
Estimator for Sample t60: 497.47324283769456
```

Figure 4. Sample B normal distribution

5.2 Mean equivalence test result

By comparing the resulting p value (approximately 0.011) of the z test against the confidence value of 0.05, the null hypothesis of the two means being equivalent can be rejected**.** As for testing samples A and C (0.37), the null hypothesis was not rejected.

5.3 Normality assessment result

After plotting the sample quantiles of sample C against theoretical quantiles of normal distribution, we can assess that the null hypothesis of the sample following a normal distribution can be rejected. We can see the Q-Q plot in Figure 5.

Figure 5. Sample C q-q plot

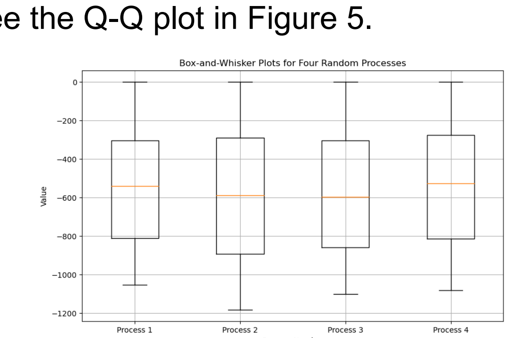


Figure 6. 4 random process whisker