Statistics 101

in English
& with Fred :D

Agenda

1. Dispersion metrics
   a. Mean
   b. Range
   c. Median
   d. Variance
   e. Standard deviation
   f. Quartiles
2. Hypothesis Testing

Hi, I'm Fred

Find me on Github

Let's start with a dataset

Dataset can be found by:

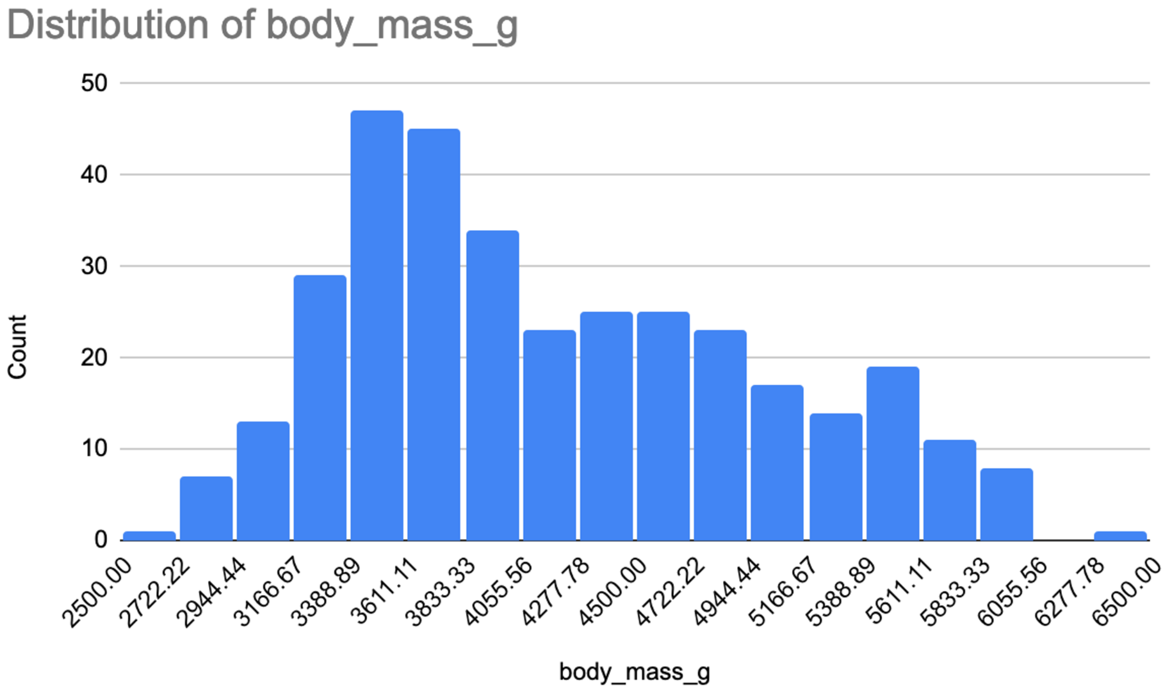
```
import pandas
import seaborn as sns
df = sns.load_dataset("penguins")
df.to_csv('penguins.csv')
```

Histograms!

Histogram

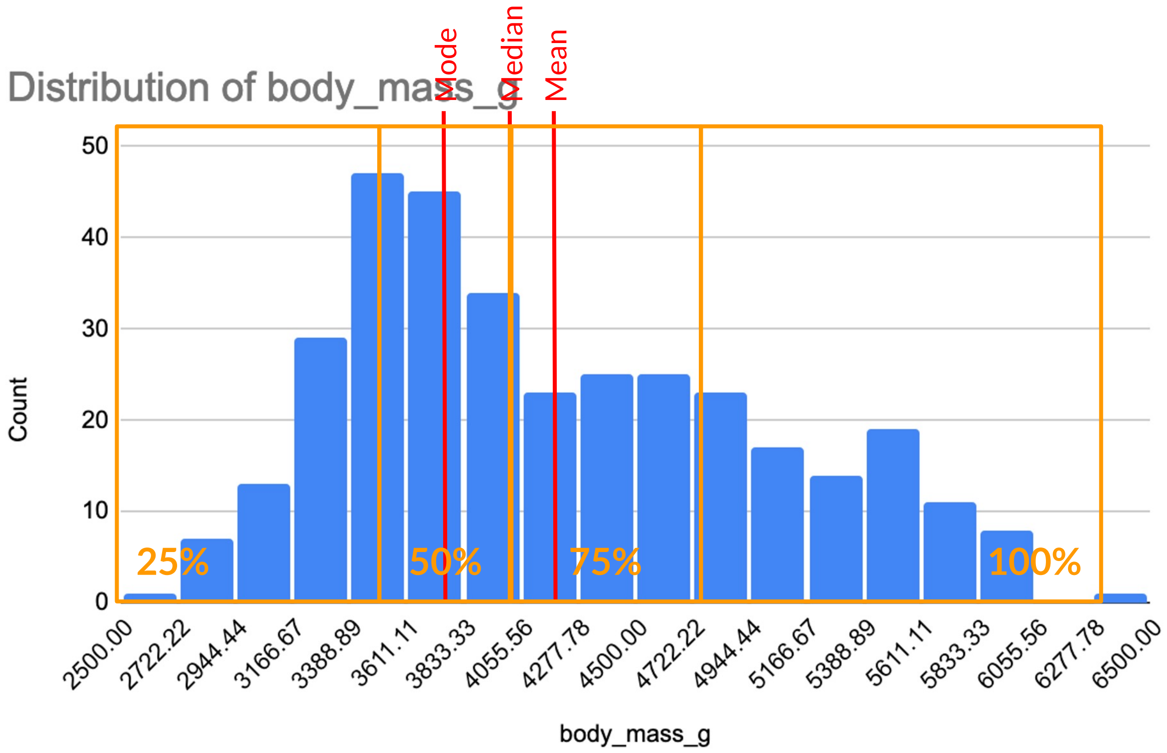
With an histogram, we can easily visualize how the data is distributed. It gives the idea of what would be the mean, standard deviation, and so on.

Let's try :D



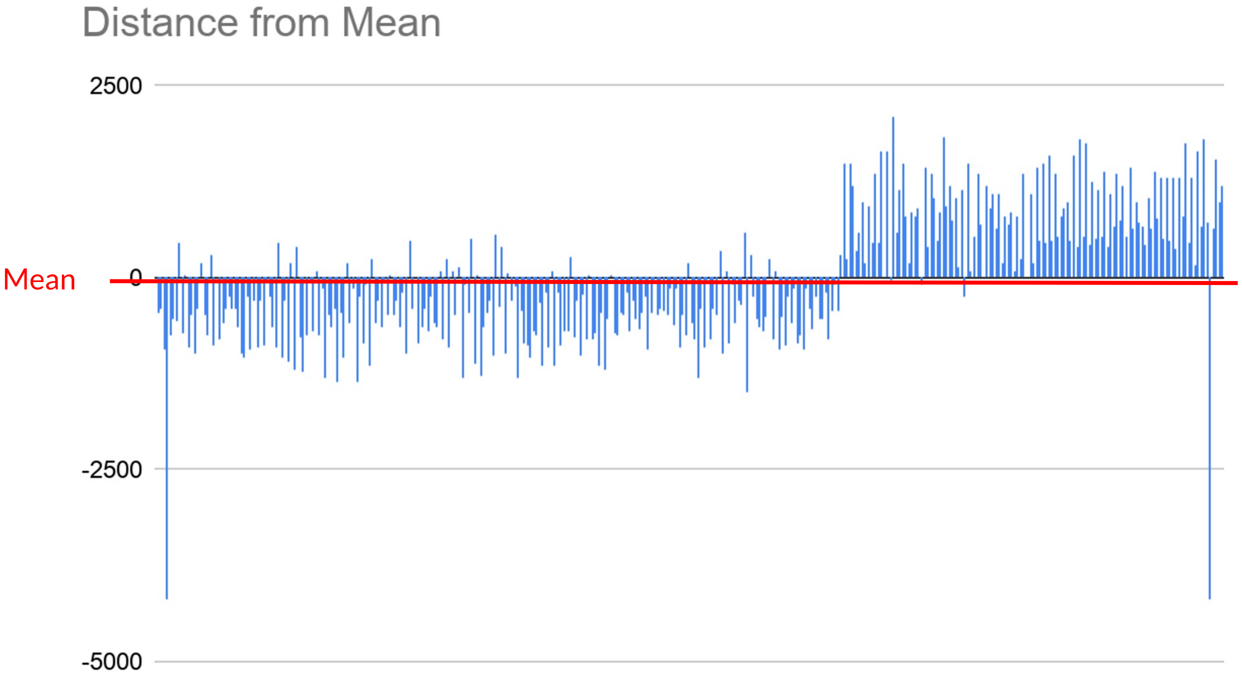
Histogram

Statistics	Value
Range	342.00
Mean	4,201.75
Median	4,050.00
Mode	3,800.00
Quartile 1	3,550.00
Quartile 2	4,050.00
Quartile 3	4,756.25
Quartile 4	6,300.00
Std Deviation	801.95
Variance	643,131.08
CV	19.80%



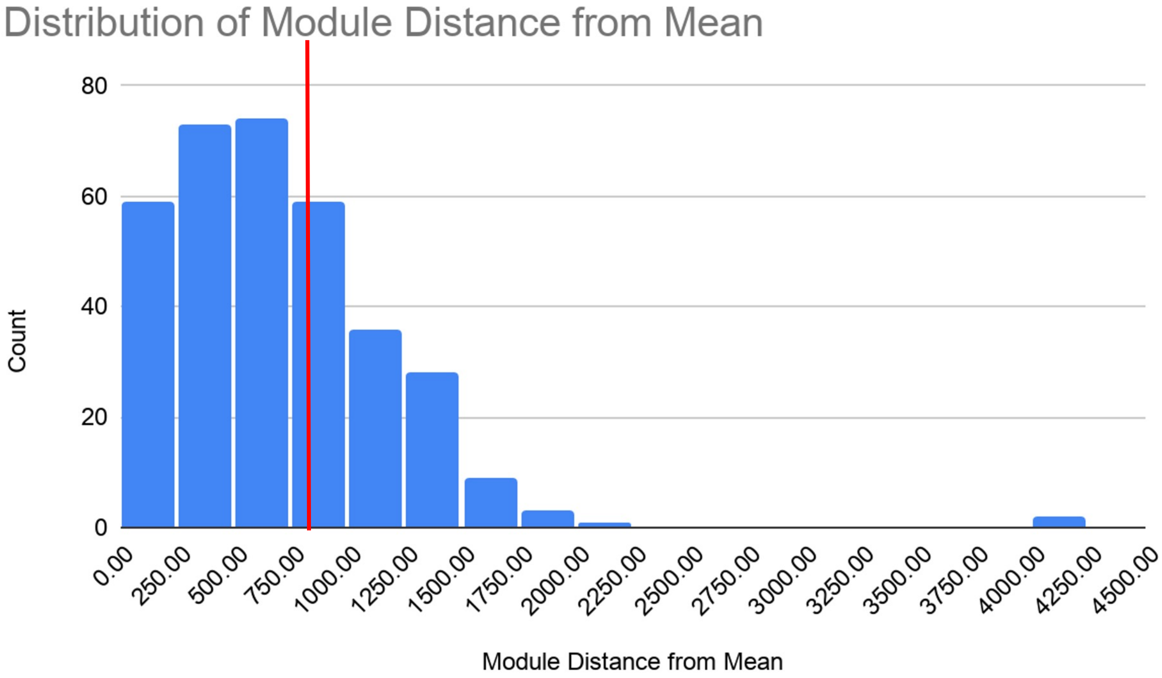
Standard Variation

Statistics	Value
Range	342.00
Mean	4,201.75
Median	4,050.00
Mode	3,800.00
Quartile 1	3,550.00
Quartile 2	4,050.00
Quartile 3	4,756.25
Quartile 4	6,300.00
Std Deviation	801.95
Variance	643,131.08
CV	19.80%



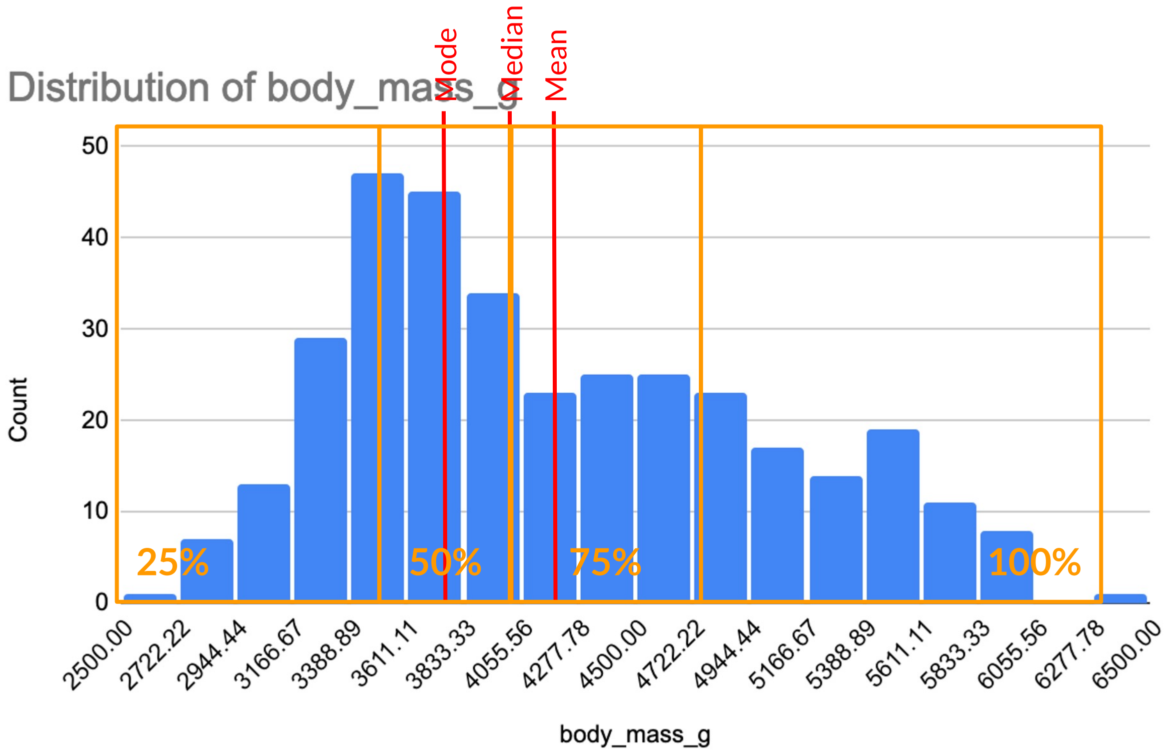
Standard Variation

Statistics	Value
Range	342.00
Mean	4,201.75
Median	4,050.00
Mode	3,800.00
Quartile 1	3,550.00
Quartile 2	4,050.00
Quartile 3	4,756.25
Quartile 4	6,300.00
Std Deviation	801.95
Variance	643,131.08
CV	19.80%



Histogram

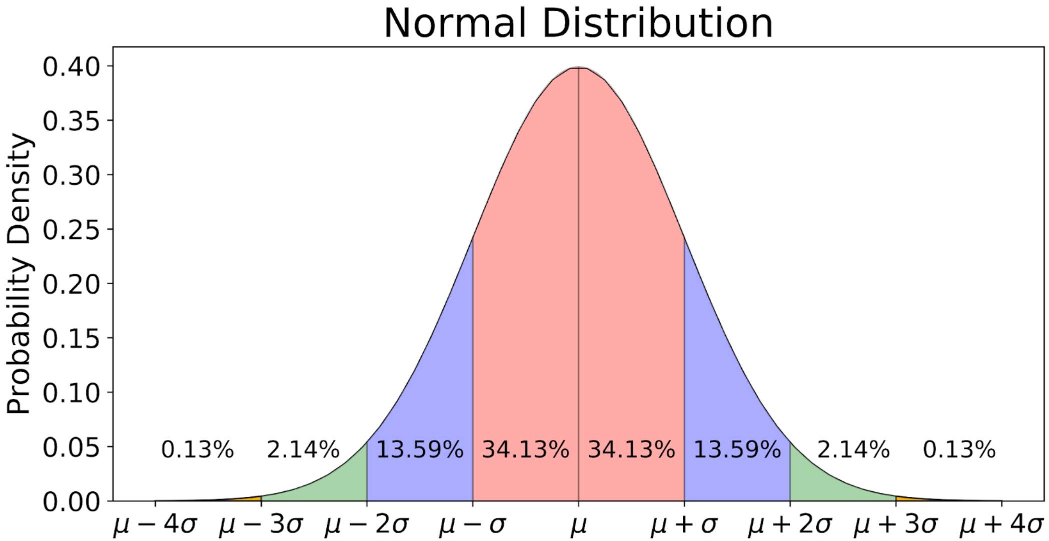
Let's first refresh our distribution, so we can now apply the normal distribution.



Normal Distribution

It is characterized by having the same number as mean, median and mode and it is also known as Gaussian distribution.

It is important due to the central limit theorem.



Normal Distribution

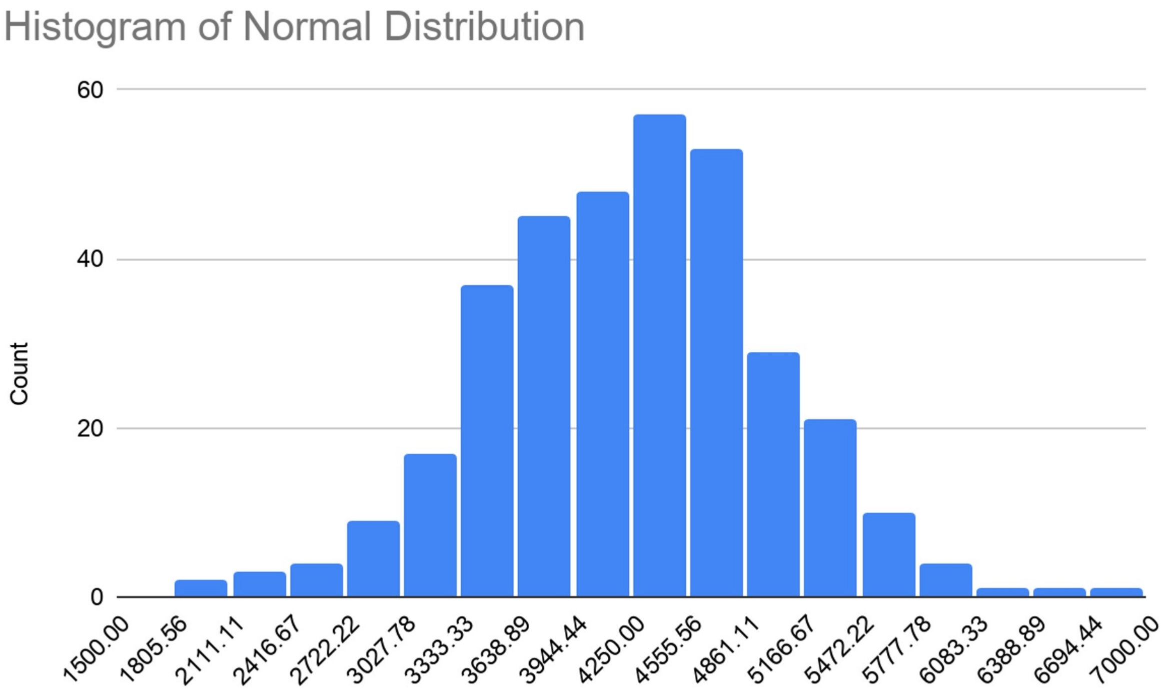
We can create a normal distribution that have the same mean that our distribution with this code:

```
import numpy
import pandas

normal = numpy.random.normal(loc=4201.754386, scale=801.954536, size=342)
normal = pandas.DataFrame(normal)
```

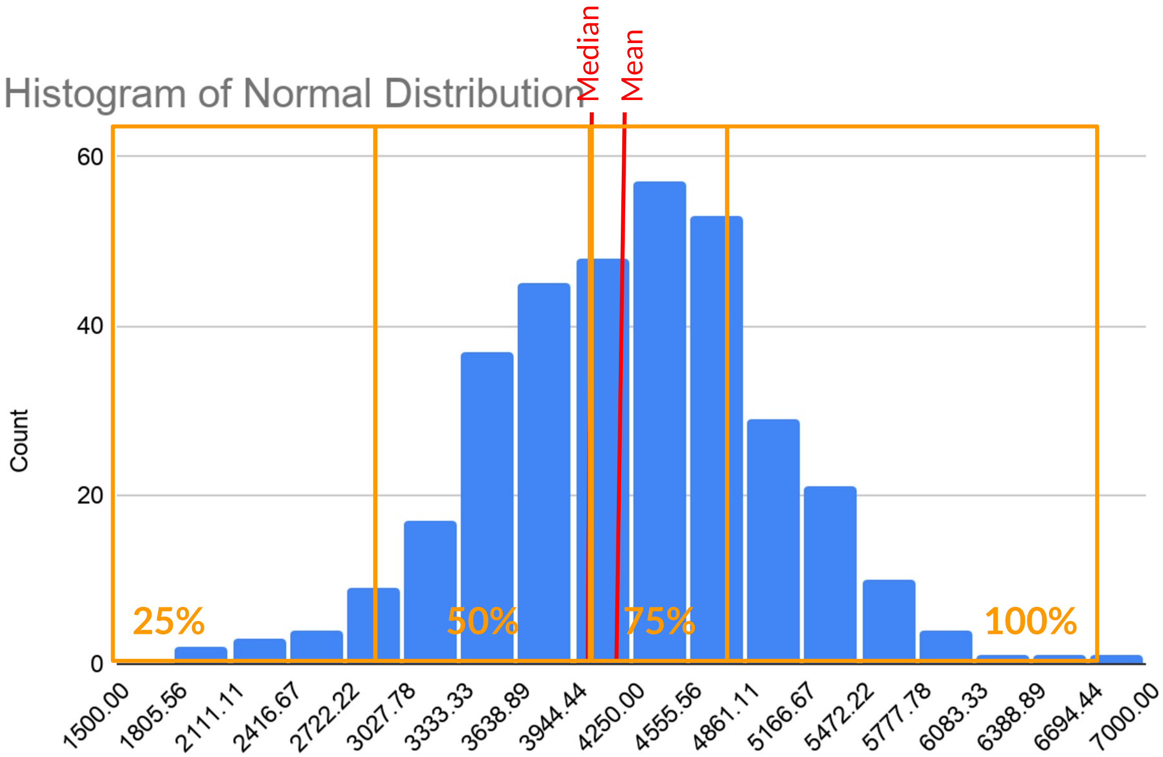
Normal Distribution

Statistics	Value
Range	342.00
Mean	4,239.45
Median	4,281.34
Mode	#N/A
Quartile 1	3,758.85
Quartile 2	4,281.34
Quartile 3	4,716.73
Quartile 4	6,913.67
Std Deviation	763.57
Variance	583,044.40
CV	17.83%



Normal Distribution

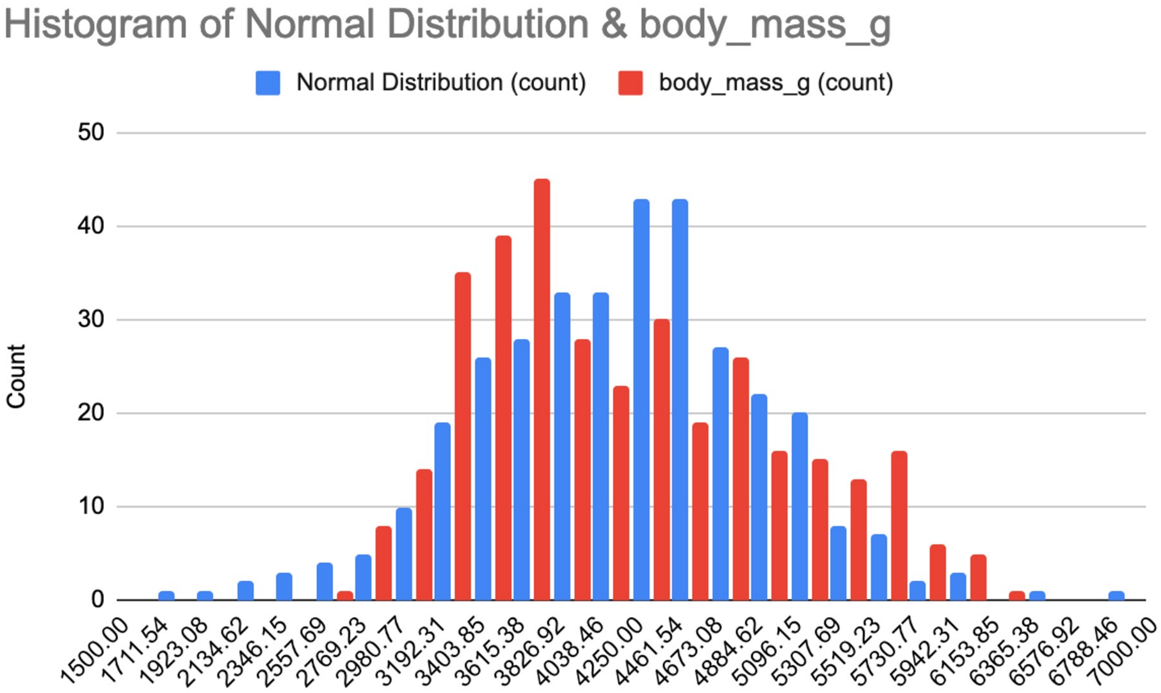
Statistics	Value
Range	342.00
Mean	4,239.45
Median	4,281.34
Mode	#N/A
Quartile 1	3,758.85
Quartile 2	4,281.34
Quartile 3	4,716.73
Quartile 4	6,913.67
Std Deviation	763.57
Variance	583,044.40
CV	17.83%



Comparing body_mass_g to Normal Distribution

As we can see, our distribution (in red) tends a bit to the left.

A distribution that doesn't follow the normal distribution may need a different way of testing for hypothesis.



Hypothesis Testing

Nonparametric Testing

Since we are not sure about our distribution, it would be unwise to simply do the tests in the parametric way.

We will use the frequentism method, which defines an event's probability as the limit of its relative frequency in many trials.

How? Bootstrap Resampling

Consider the array `body_mass_g`:

1. We will take any value randomly, write it down and put back to the array.
2. We will do the previous step N times, where N is the length of the array.
3. Once we finish writing down the values, we will calculate the mean of this new array.

Since computer power is not a problem, we will do the 3 steps 100.000 times.

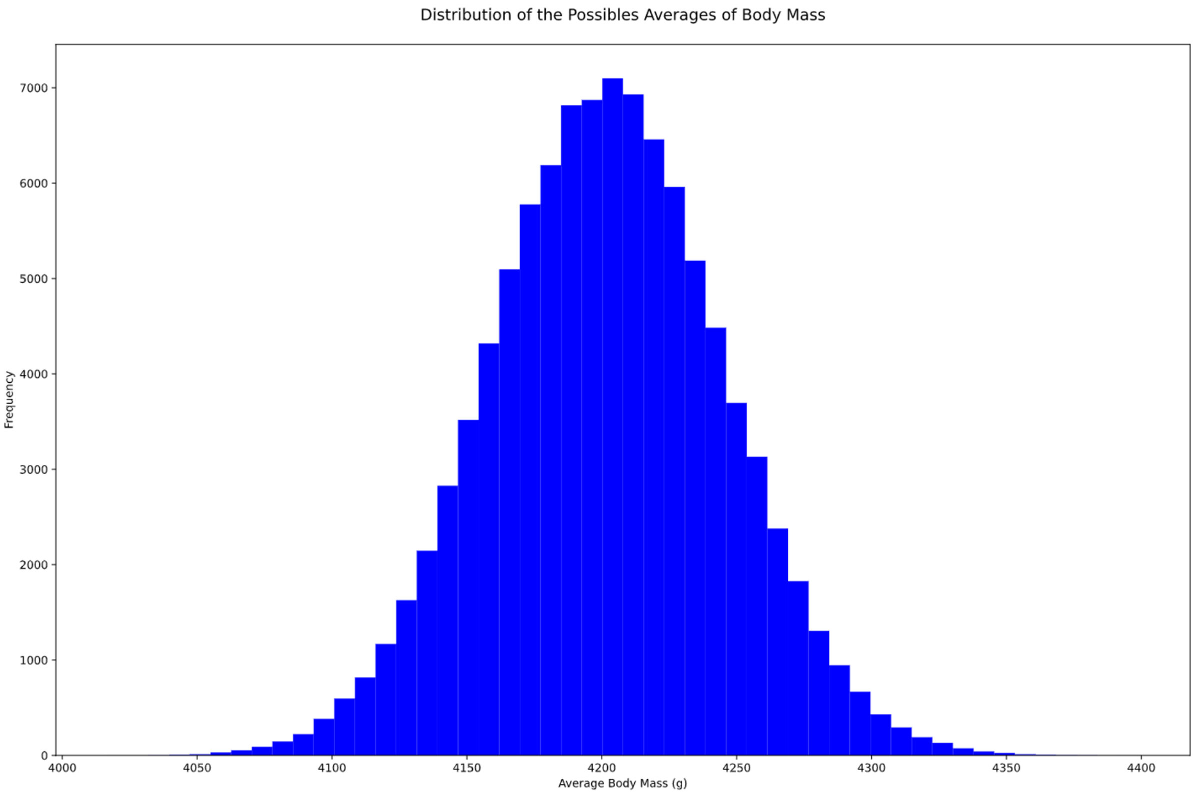
You can do this with this simple function:

```
import numpy as np
# bootstrap replicas
def bootstrap_replicate_1d(data, func):
    """Generate bootstrap replicate of 1D data."""
    bs_sample = np.random.choice(data, len(data))
    return func(bs_sample)
# many bootstraps replicas
def draw_bs_reps(data, func, size=1):
    """Draw bootstrap replicates."""
    # Initialize array of replicates: bs_replicates
    bs_replicates = np.empty(size)
    # Generate replicates
    for i in range(size):
        bs_replicates[i] = bootstrap_replicate_1d(data, func)
    return bs_replicates
```

Bootstrap Resampling

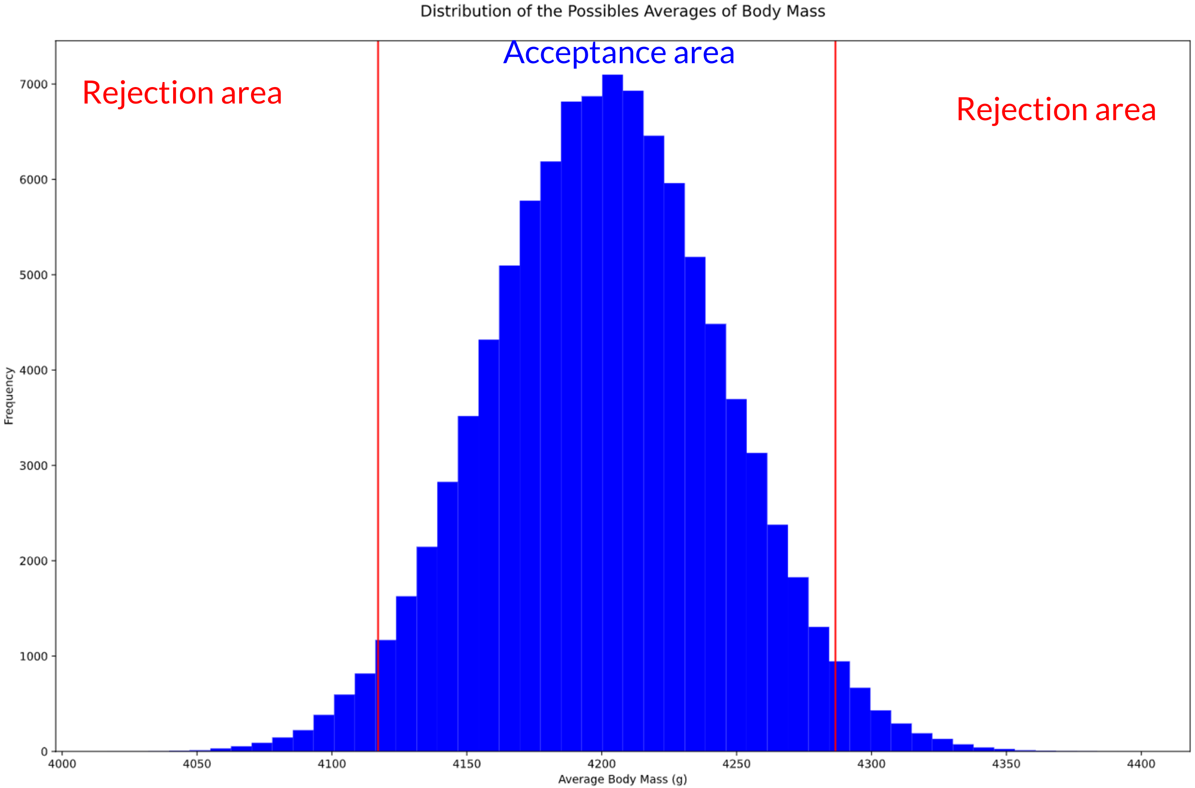
Applying that function to the `body_mass_g` array will produce an array of results. Since the array for the calculation of the mean was done randomly, we now have a normal distribution of the possible values of the mean.

Let's test it!



Confidence Interval

By plotting the percentiles 2.5 and 97.5, we can now say that with 95% confidence the mean is between 4117.1 and 4286.62.

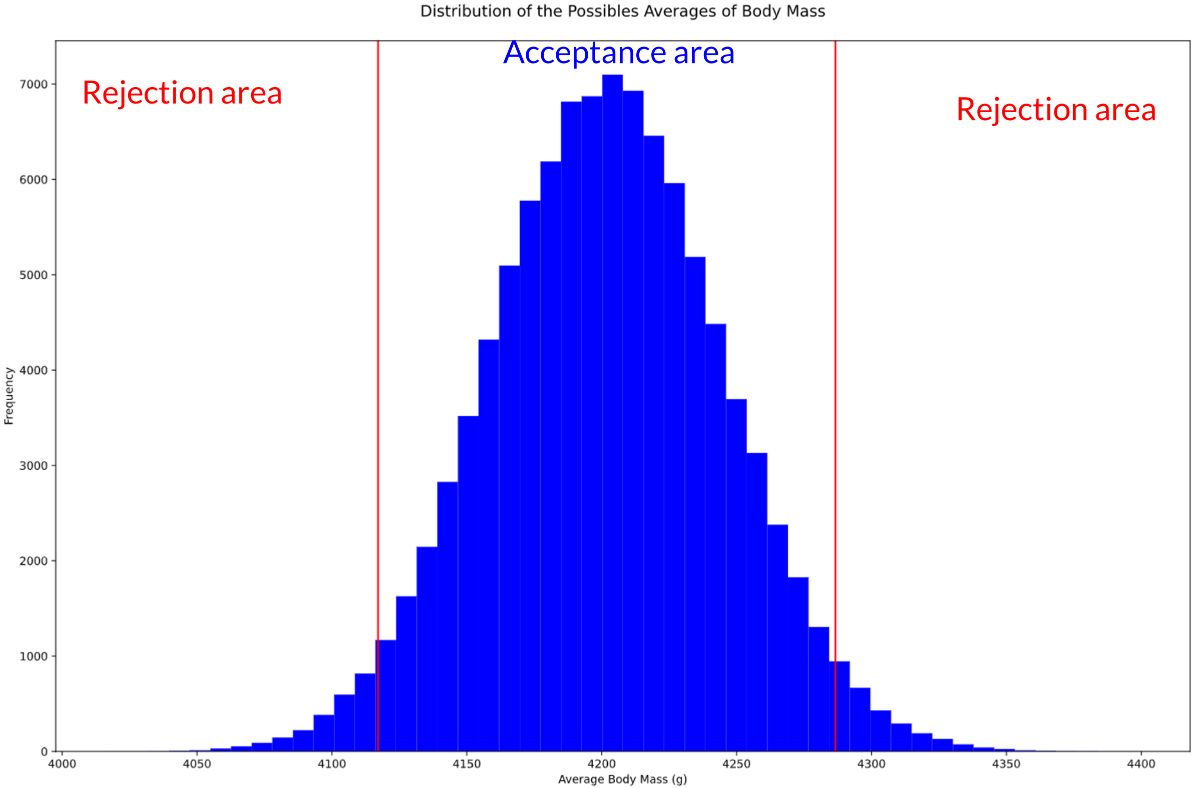


Hypothesis Testing

We will now test if there is a possibility to the calculated average be zero:

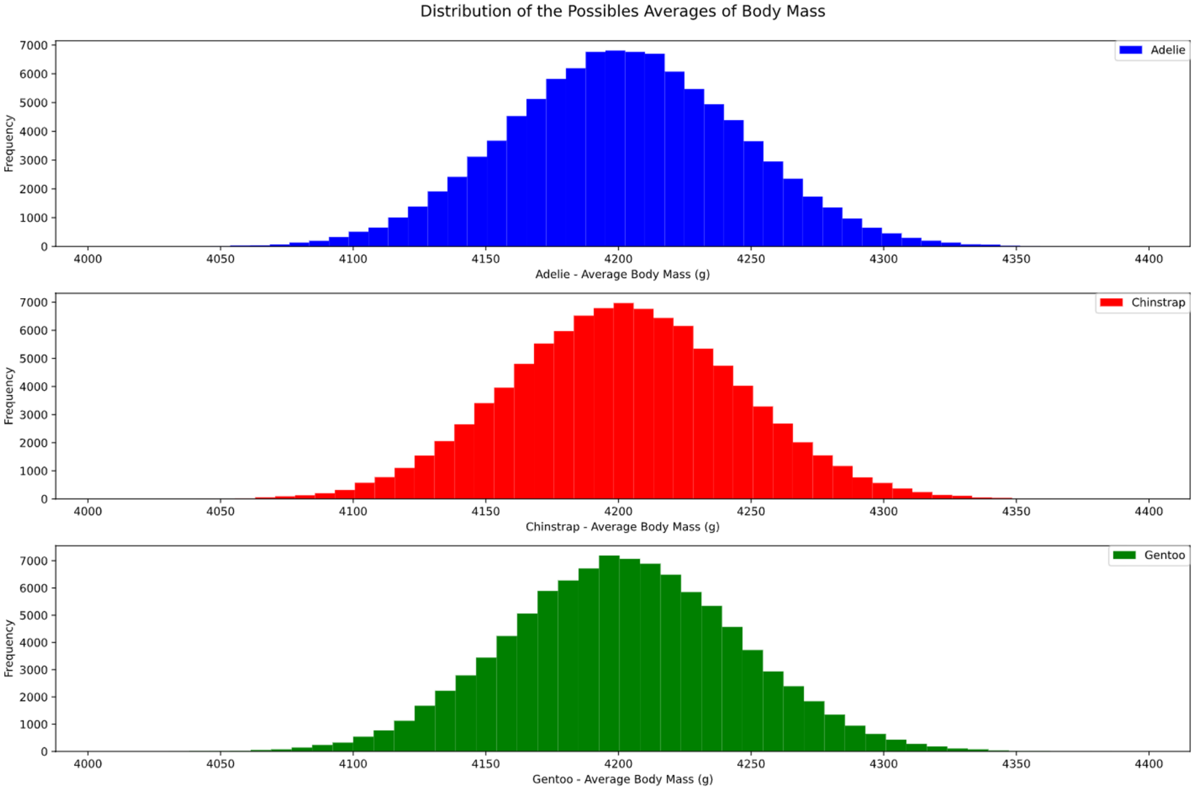
- H0: mean = 0
- H1: mean != 0

As we can see on the graph, averages below the percentile 2.5 (4117.1) are very rare and out of our confidence interval. Therefore we do not have enough proof to reject the null hypothesis.



Hypothesis Testing

By plotting the the graph for each of the three species, we can graphically see that there's is no difference between them concerning the body mass averages.



That's all :D

All codes used can be found here. It's an HTML file, so download it and open in your browser.