# A/B (split) testing & asymmetric distribution

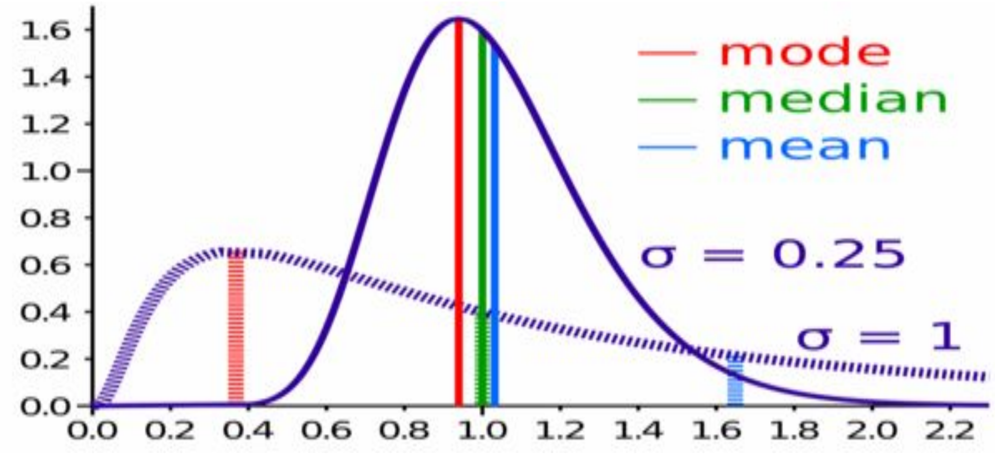
By @Drew

# What is that?

Few samples of auditory gets different experience in the same time

+ Seasonality and other influences affects on both samples equally
- We can’t just compare the averages of metrics

Coz we should take into account the distribution of our metric



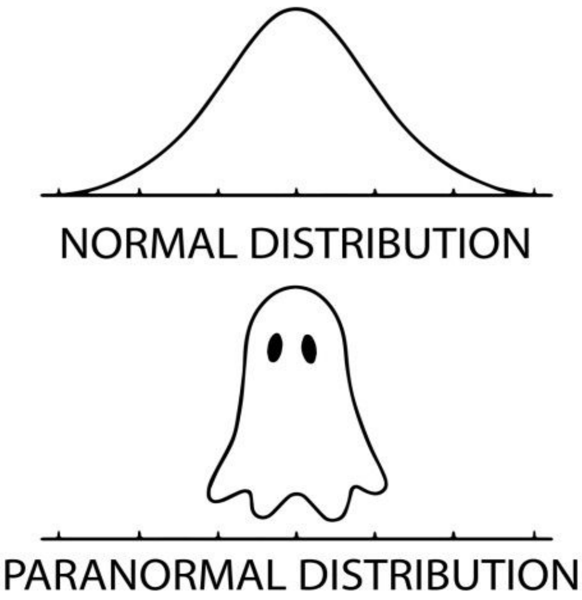
# There are two cases

In case of some **normal** distribution we use trivial well known approach

- Student's t-test

In case of some **paranormal** there is a few things we can do:

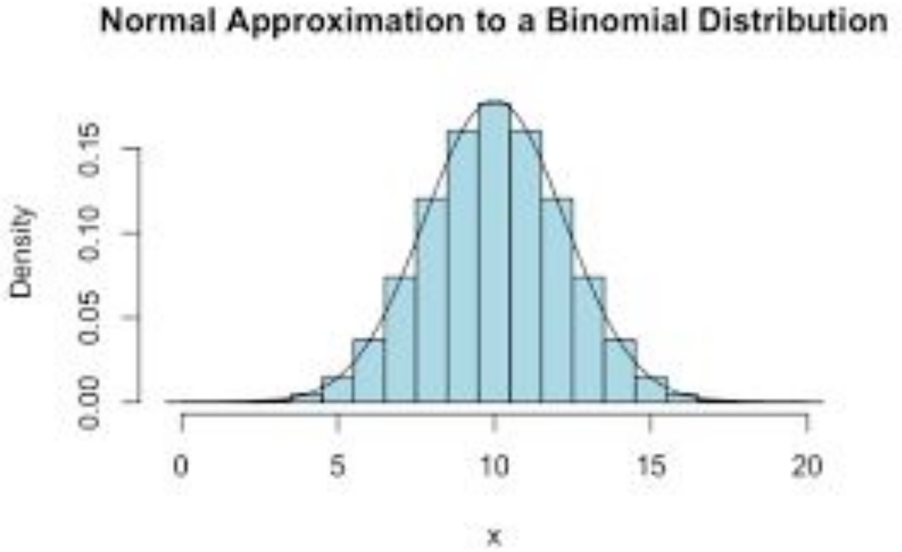
- Scale transformation
- Bootstrapping
- Nonparametric test



# If we have some “conversions”

Which mean someone saw the offer and made action after that, or not

- It’s called “binomial distribution”
+ It is approximately normal *

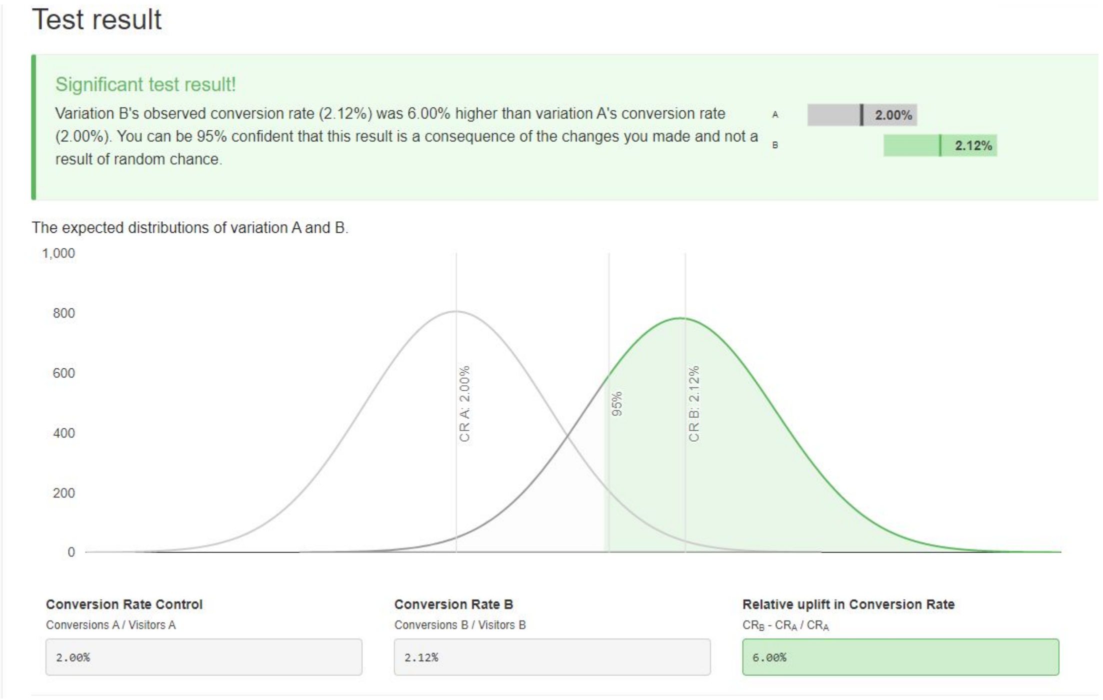



* when np>5 and n(1-p)>5

# Therefore we use

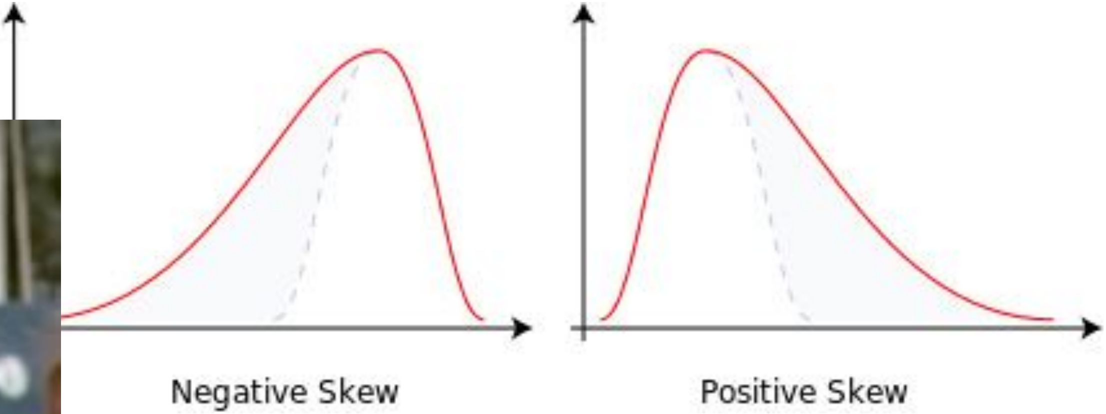
Any A/B test calculator from the WEB

**p-value** is the probability of obtaining the observed results of a test, assuming that the null hypothesis is correct



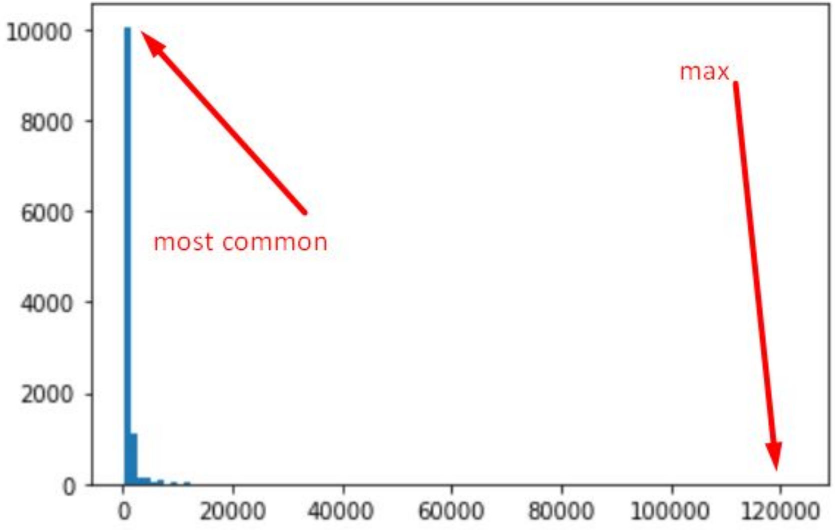
# If our metriks far from normal distribution

We can’t use previous approach

# Real case

Cart A/B test

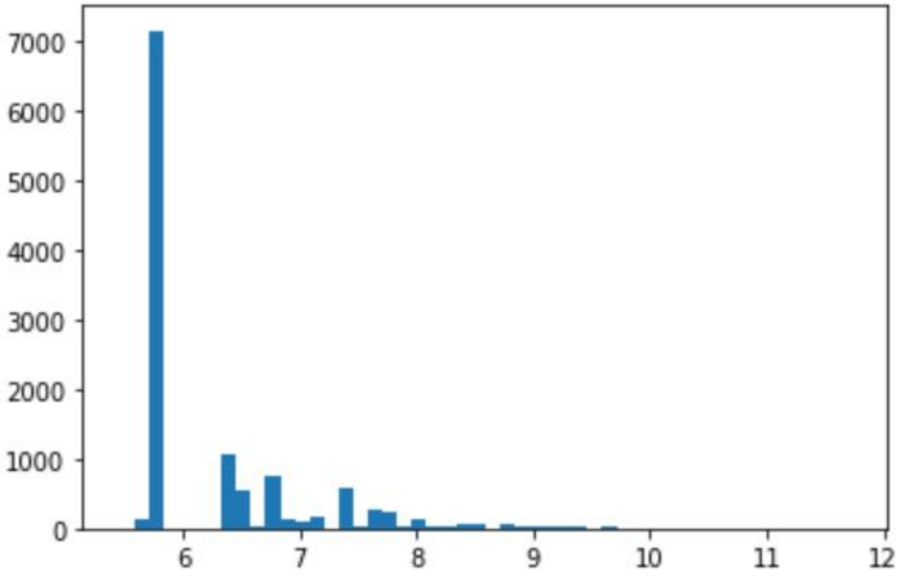



Distribution of orders by amount $

* Shapiro-Wilk test says that is definitely not a normal distribution

# Let’s try to transform data to log scale and test it

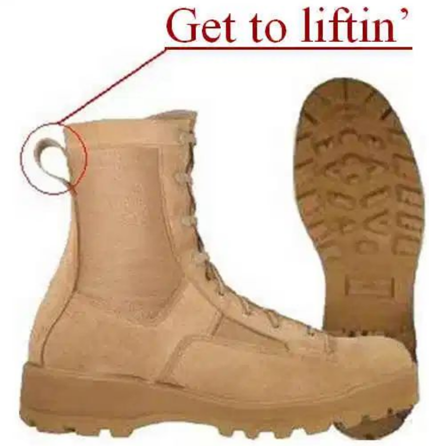
```
Shapiro-Wilk test statistic, W: nan
 p-value: 1.0
```




Unfortunately, none of "Exponential and logarithmic functions" can make this distribution normal

There is another way to figure this out...

**Bootstrap**

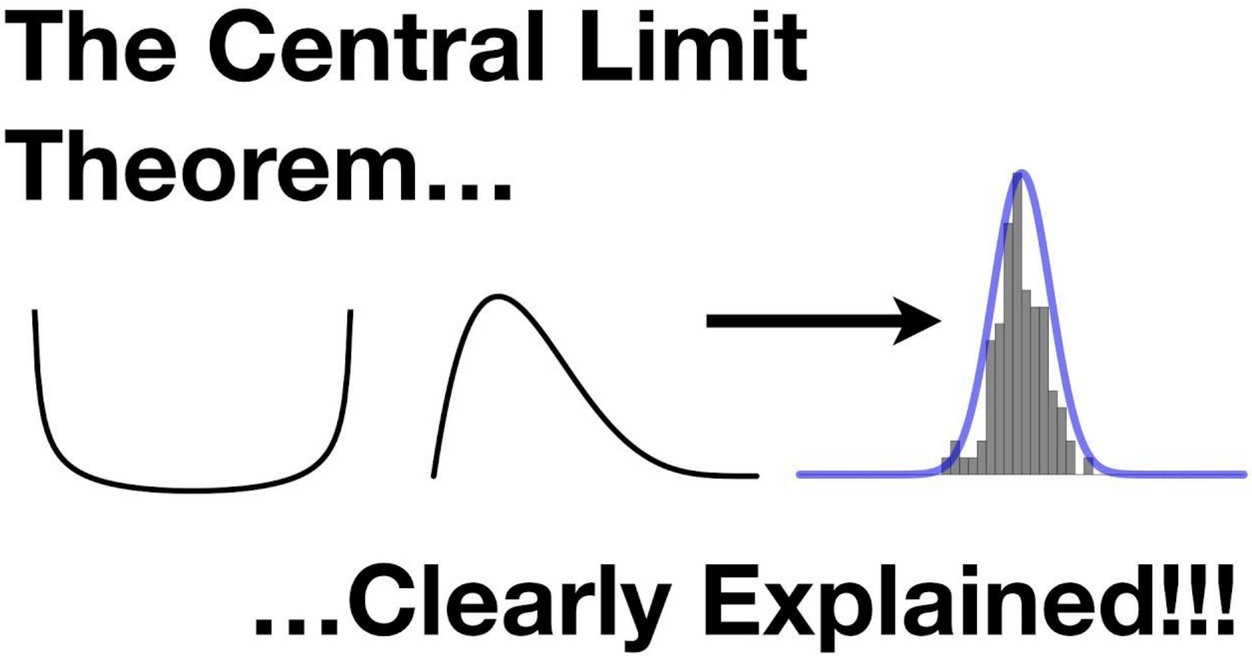



* https://en.wikipedia.org/wiki/Bootstrapping_(statistics)

# According the CLT

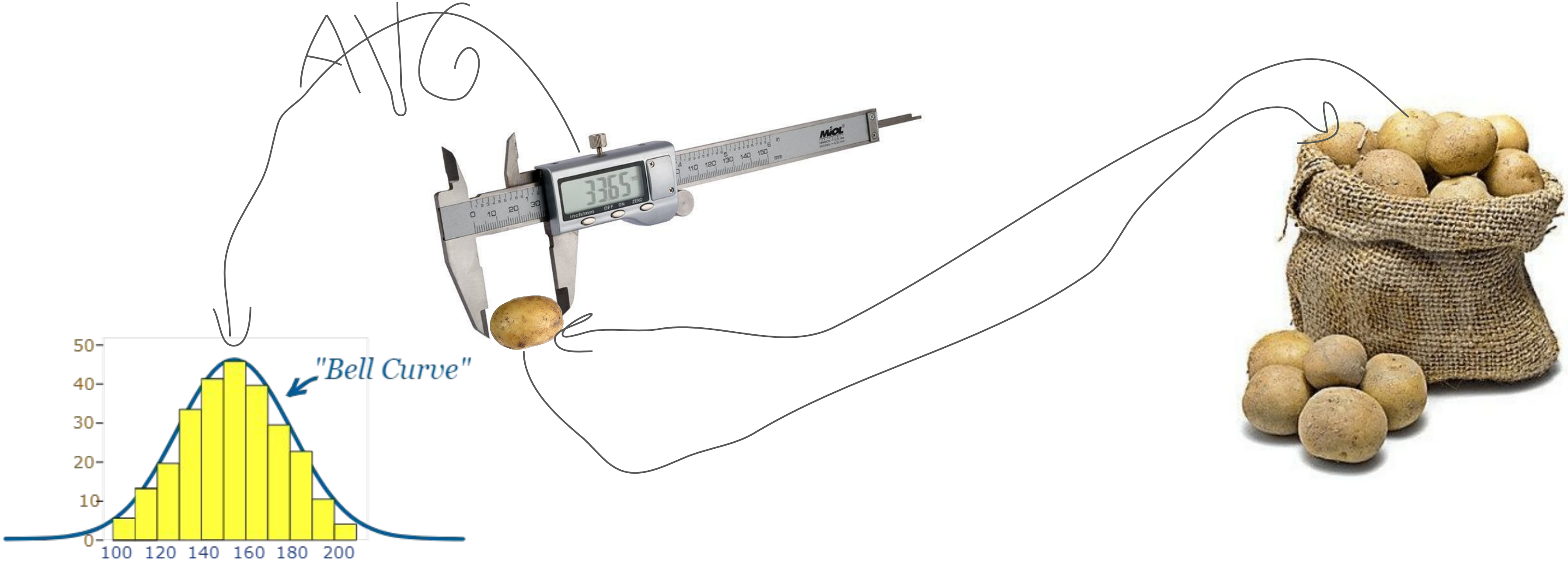
We can get normal distribution from any else if we can find averages from dozen samples or smth like this

Unfortunately we have no access to the whole **general population** data

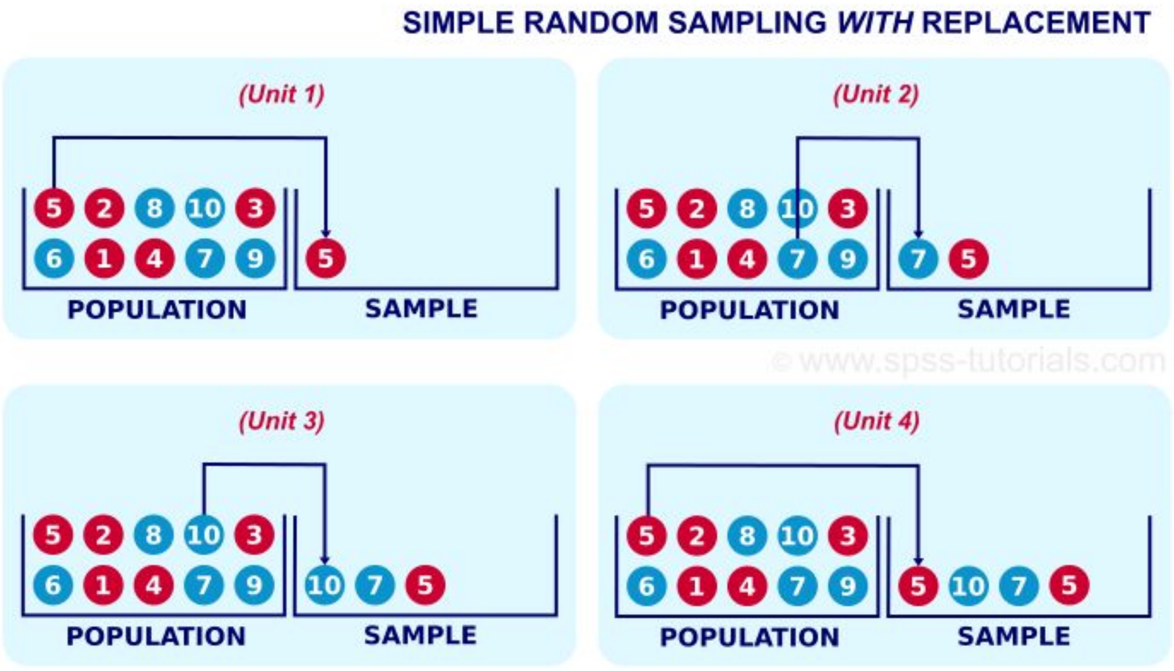


# Bootstrapping

Sampling With Replacement and calculating the statistics



# Sampling With Replacement



# Some code for bootstrapping

```python
def get_bootstrap_samples(data, n_samples):
    indices = np.random.randint(0, len(data), size=(n_samples, len(data)))
    samples = np.array(data)[indices]
    return samples
```

```python
def stat_intervals(stat, alpha=0.05):
    boundaries = np.percentile(stat, [100 * alpha / 2., 100 * (1 - alpha / 2.)])
    return boundaries
```

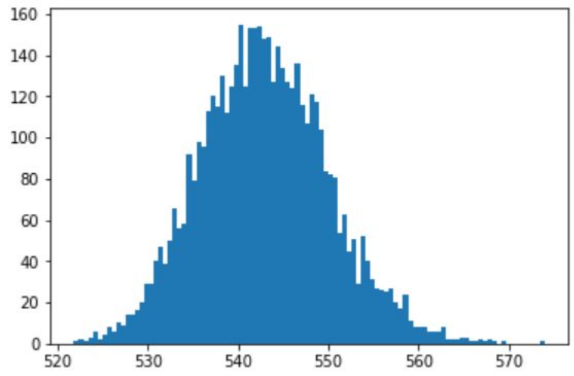
```python
def statistic_func(samples):
    return np.array([np.mean(sample) for sample in samples])
```

```python
def pipline(data):
    samples = get_bootstrap_samples(data, 5000)
    statistic = statistic_func(samples)
    intervals = stat_intervals(statistic)
    return {"intervals":intervals, "statistic":statistic}
```

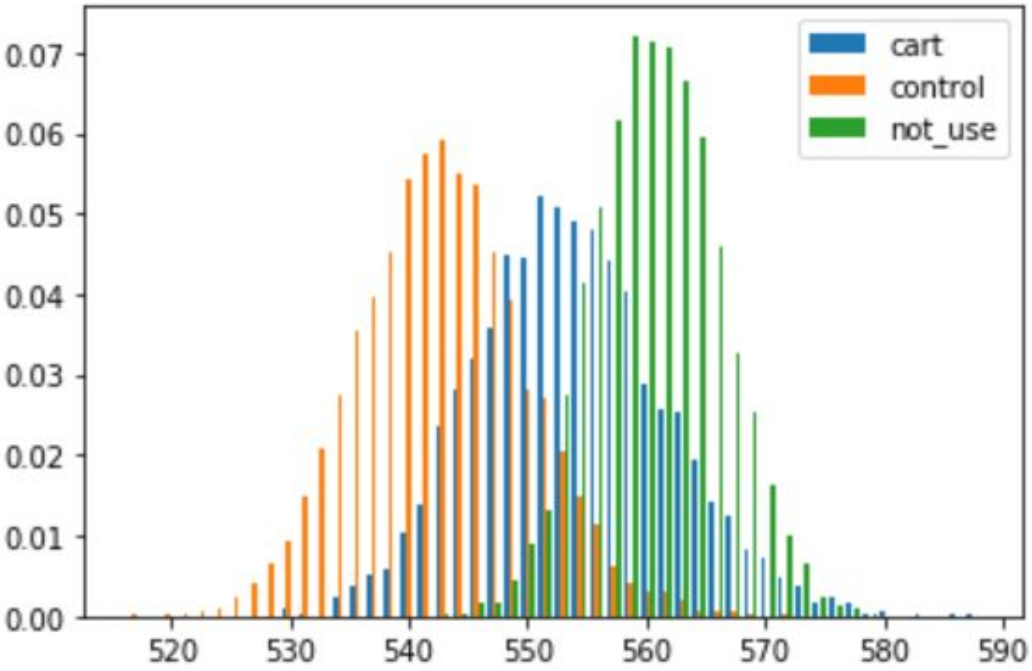
# And we gets

Shapiro-Wilk test says that is kinda  normal distribution

```
Shapiro-Wilk test statistic, W: 0.9971110224723816
 p-value: 3.349401467289681e-08
```




And now we can compare it

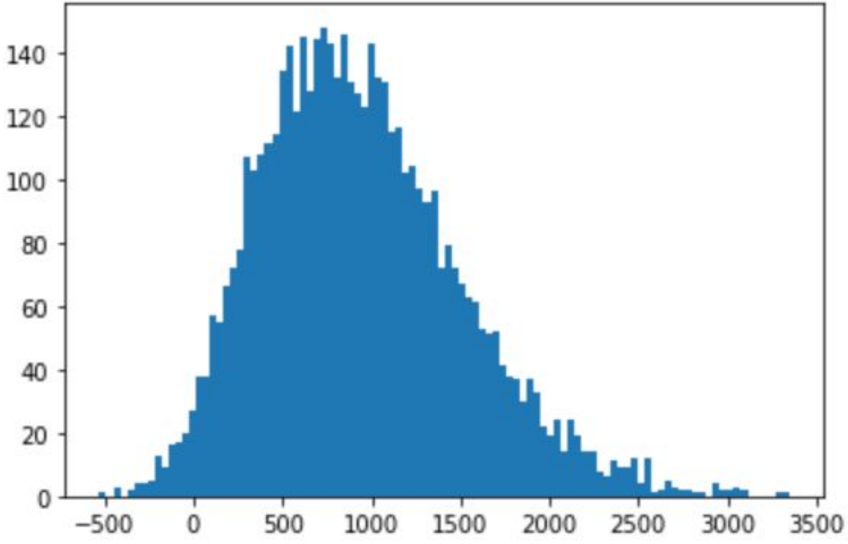


# Estimate the results

We can use the difference between averages as a metric, in order to estimate its distribution

In this case, obviously, some variant much more profitable than the other

```
mean diff 940.305045522435
mean diff 36.82695737263019 % higher than the control group
confidence intervals [  20.54013613 2204.64748901]
```



# Nonparametric methods

It’s very simple to use it

But I’m not sure how can I interpret the results

```
from scipy.stats import mannwhitneyu
from scipy.stats import wilcoxon

stat, p = mannwhitneyu(cart, control)
print("stat:", stat, "  p:", p)
stat, p = wilcoxon(cart, control)
print("stat: ", stat, "  p:", p)
```

```
stat: 38624436.0   p: 0.08958661725343231
stat:  4403712.0   p: 0.09373286986450771
```

# Conclusion

Bootstrapping can help us in case of:

- Asymmetric distribution of observed metrics
- Small samples

to estimate amount of difference between samples

Nonparametric methods help us to test statistical hypothesis of difference between samples

Thx u 4 attention :)

@Drew