Statistics By Default

Lecture 1: SE, z-scores, False Positives, FDR-corrections

Notes and slides can be found here:
https://github.com/drewrl3v/StatsByDefault

Andrew Lizarraga

Standard Error

- What is standard error?

Standard Error

- What is standard error?

$$SE = \frac{\sigma}{\sqrt{N}}$$

Standard Error

- **What is standard error?**

$$SE = \frac{\sigma}{\sqrt{N}}$$

We should be kosher and know where $\sqrt{N}$ comes from.

Standard Error

- **What is standard error?**

1. $Var(X) = Std(X)$
2. $Var(X + y) = Var(X) + Var(Y)$ (assuming $X, Y \sim i.i.d$ from their distribution)
3. For any $a \neq 0$, we have $Var(\frac{X}{a}) = \frac{Var(X)}{a^2}$

Standard Error

- **What is standard error?**

1. $Var(X) = Std(X)$
2. $Var(X + y) = Var(X) + Var(Y)$ (assuming $X, Y \sim i.i.d$ from their distribution)
3. For any $a \neq 0$, we have $Var(\frac{X}{a}) = \frac{Var(X)}{a^2}$

We take N measurements (i.i.d) to get our aggregate measurement:

$$agg = \frac{x_1 + \cdots + x_N}{N}$$

Standard Error

- **What is standard error?**

1. $Var(X) = Std(X)$
2. $Var(X + y) = Var(X) + Var(Y)$ (assuming $X, Y \sim i.i.d$ from their distribution)
3. For any $a \neq 0$, we have $Var(\frac{X}{a}) = \frac{Var(X)}{a^2}$

We take N measurements (i.i.d) to get our aggregate measurement:

$$agg = \frac{x_1 + \cdots + x_N}{N}$$

The variance fo the aggregate measurement is, by 1. and 3. given by

$$Var(agg) = \frac{Var(x_1) + \cdots + Var(X_N)}{N^2}$$

Standard Error

- **What is standard error?**

All the individual measurements are taken from the same distribution, so they all have the same standard deviation, call it $\sigma = Std(X)$. So $Var(x_i) = \sigma^2$. And now we have:

$$Var(agg) = \frac{\sigma^2 + \cdots + \sigma^2}{N^2} = \frac{N\sigma^2}{N^2}$$

Standard Error

- **What is standard error?**

All the individual measurements are taken from the same distribution, so they all have the same standard deviation, call it $\sigma = Std(X)$. So $Var(x_i) = \sigma^2$. And now we have:

$$Var(agg) = \frac{\sigma^2 + \cdots + \sigma^2}{N^2} = \frac{N\sigma^2}{N^2}$$

Therefore:

$$SE = \sqrt{Var(agg)} = \frac{\sigma}{\sqrt{N}}$$

Standard Error

```python
def trading_system(exchange: str) -> float:
    if exchange == "ASDAQ":
        execution_cost = 12.0
    elif exchange == "BYSE":
        execution_cost = 10.0
    else:
        raise ValueError("Exchange Not supported")
    execution_cost += np.random.normal()
    return execution_cost
```

✓ 0.0s

```python
def aggregate_measurement_with_se(exchange: str, num_individual_measurements: int):
    individual_measurements = np.array(
        [trading_system(exchange) for _ in range(num_individual_measurements)]
    )
    aggregate_measurement = individual_measurements.mean()
    sd_1                  = individual_measurements.std()
    se = sd_1 / np.sqrt(num_individual_measurements)
    return aggregate_measurement, se
```

0.0s

Standard Error

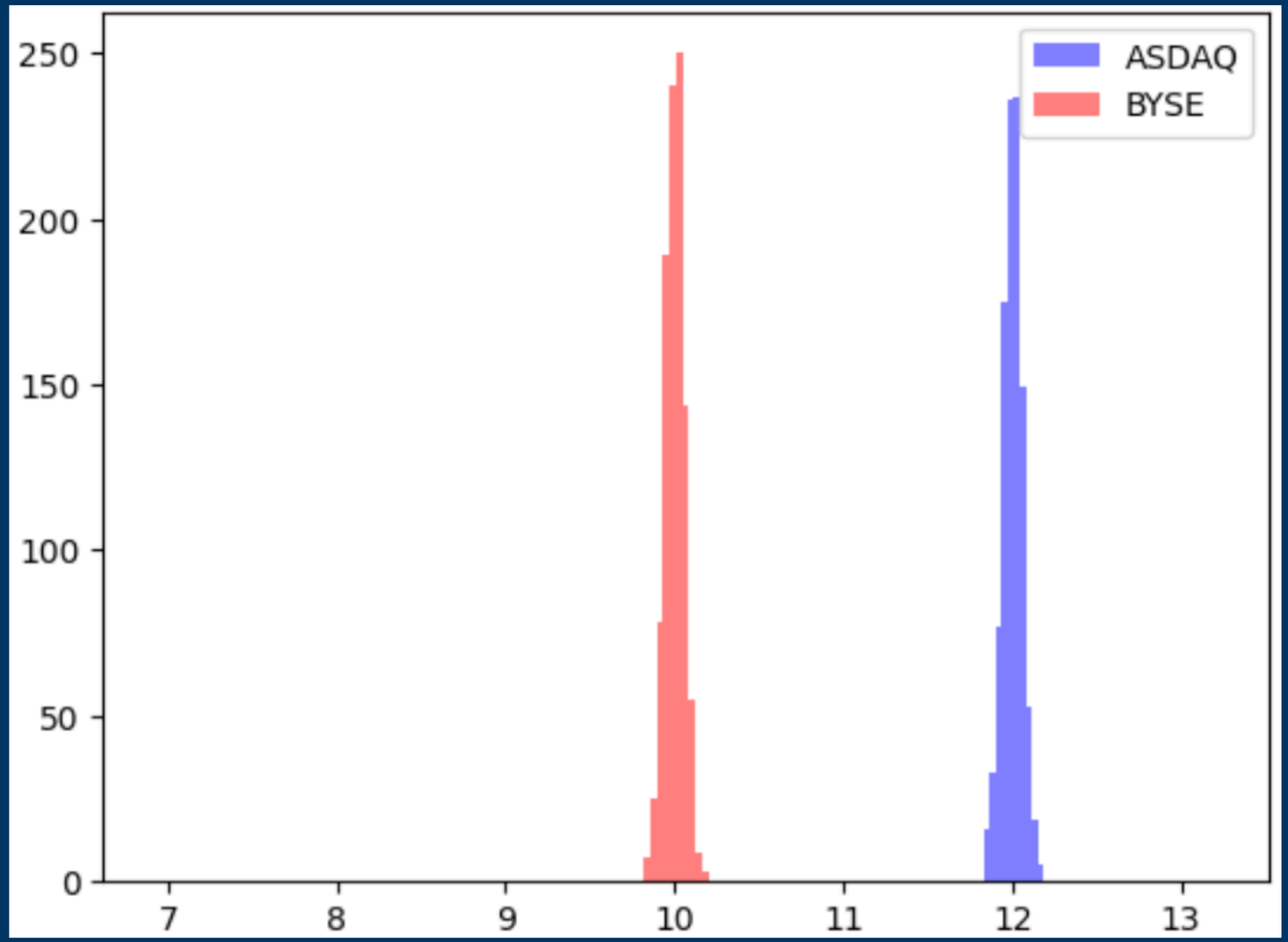
```
1  np.random.seed(17)
2  print(aggregate_measurement_with_se("ASDAQ", 300))
3  print(aggregate_measurement_with_se("BYSE",  300))
```

✓ 0.0s

```
(np.float64(12.000257642551059), np.float64(0.06025475636498122 5))
(np.float64(10.051095649188758), np.float64(0.05714189794415452))
```

A/B - Testing (Stock Exchange Rates)

- **Comparing Two exchanges 300 samples:**



Standard Error

- We can look at these numbers from a single experiment, with no histogram available and claim that BYSE is very likely the better choice by this reasoning:

Standard Error

- We can look at these numbers from a single experiment, with no histogram available and claim that BYSE is very likely the better choice by this reasoning:

  - BYSE's expectation might be higher than the aggregate measurement: Say 10.05 + 0.057 = 10.107

Standard Error

- We can look at these numbers from a single experiment, with no histogram available and claim that BYSE is very likely the better choice by this reasoning:

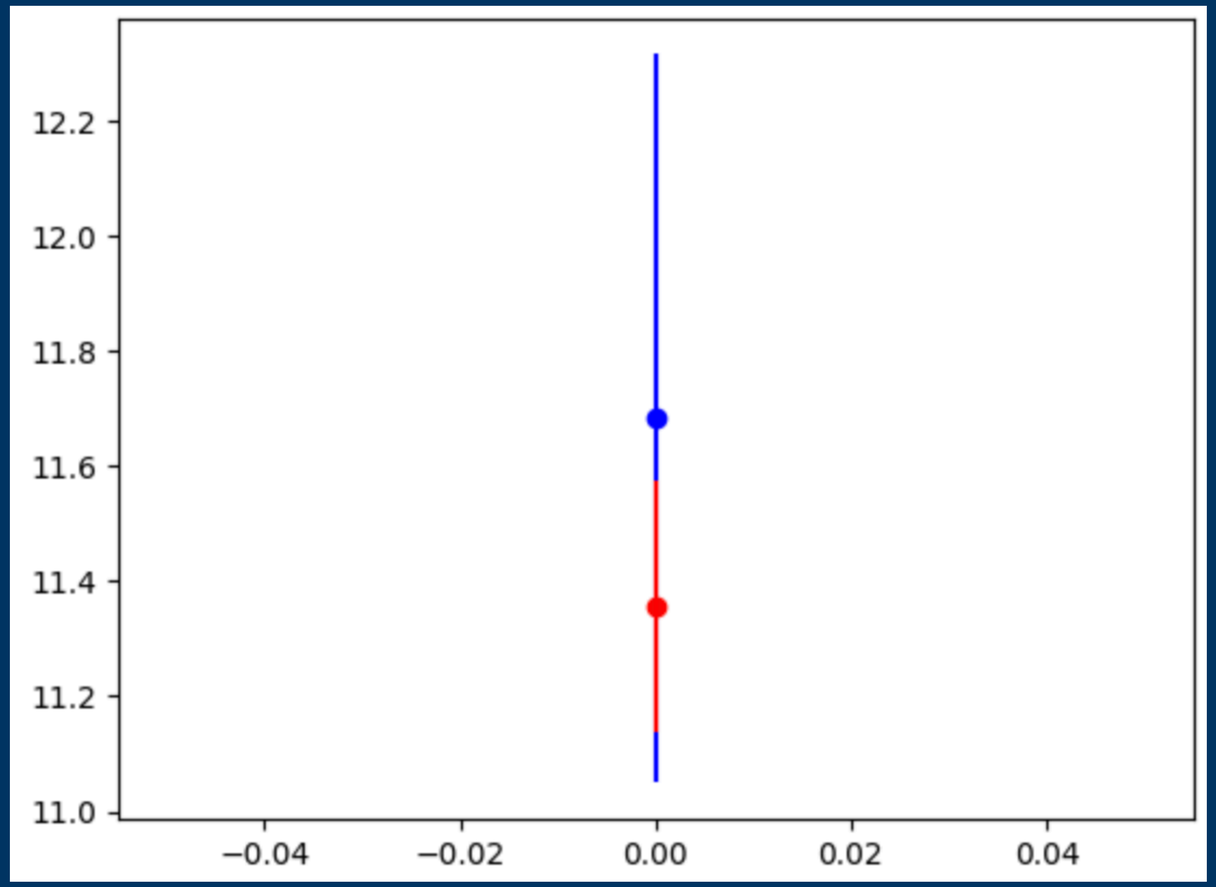
  - BYSE's expectation might be higher than the aggregate measurement: Say 10.05 + 0.057 = 10.107

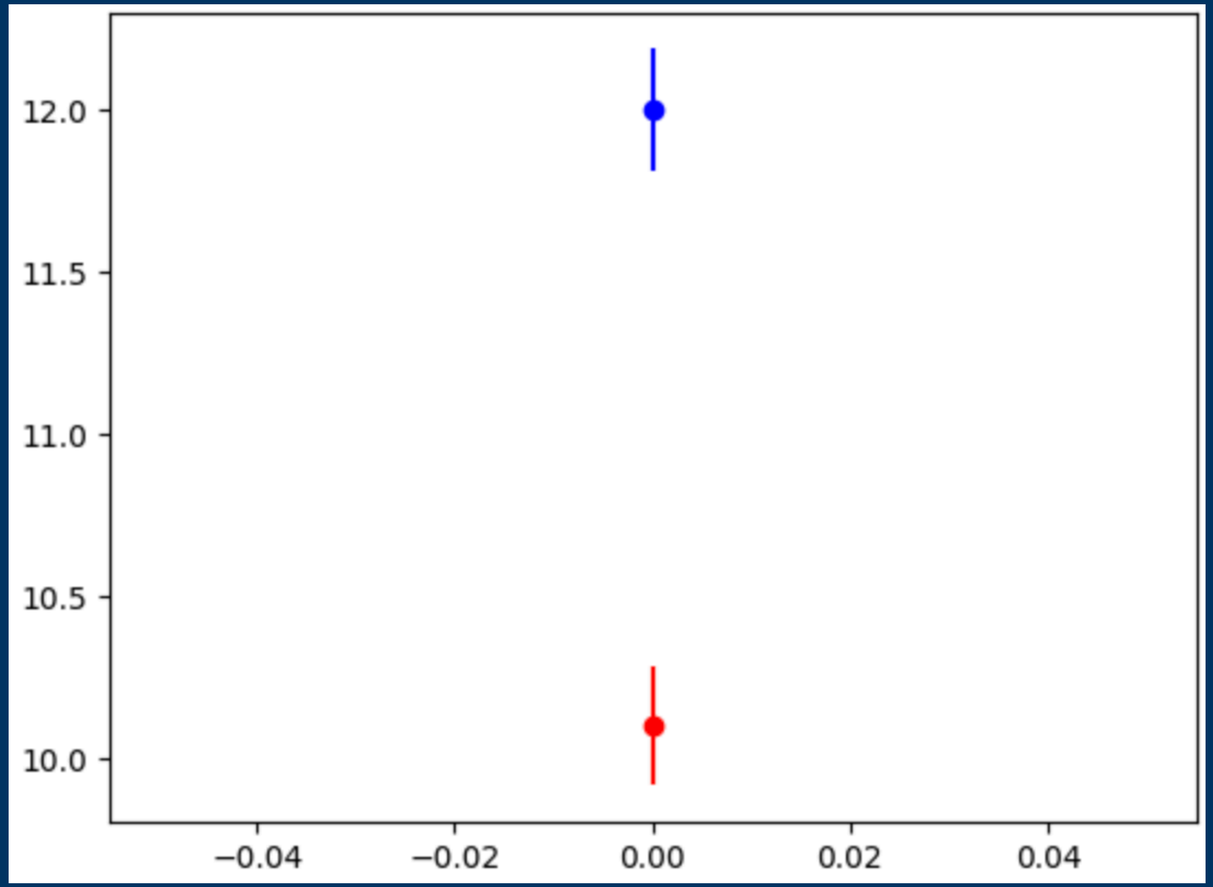
  - Similarly ASDAQ's expectation might be lower than the aggregate by: 12.00 - 0.060 = 11.94

Standard Error

- We can look at these numbers from a single experiment, with no histogram available and claim that BYSE is very likely the better choice by this reasoning:
  - BYSE's expectation might be higher than the aggregate measurement: Say 10.05 + 0.057 = 10.107
  - Similarly ASDAQ's expectation might be lower than the aggregate by: 12.00 - 0.060 = 11.94
  - Even if both of these are true we still have that BYSE is cheaper.

Standard Error





Reducing Number of Measurements

- It takes time and money to collect measurements. So it's in our interest to reduce the number of measurements while still being able to correctly make inferences about the data collected.

Reducing Number of Measurements

- It takes time and money to collect measurements. So it's in our interest to reduce the number of measurements while still being able to correctly make inferences about the data collected.

- Decreasing the number of measurements increases the SE of the aggregate measure.

Reducing Number of Measurements

- It takes time and money to collect measurements. So it's in our interest to reduce the number of measurements while still being able to correctly make inferences about the data collected.

- Decreasing the number of measurements increases the SE of the aggregate measure.

- To remedy this, we think about the problem backwards. Recall that we are trying to prove the following:

    - The A/B test for making our decision of A over B or B over A is probably not wrong.

Reducing Number of Measurements

- It takes time and money to collect measurements. So it's in our interest to reduce the number of measurements while still being able to correctly make inferences about the data collected.

- Decreasing the number of measurements increases the SE of the aggregate measure.

- To remedy this, we think about the problem backwards. Recall that we are trying to prove the following:

    - The A/B test for making our decision of A over B or B over A is probably not wrong.

Reducing Number of Measurements

- The keyword being: probably!

- To make this a little easier to understand, let's define delta, the difference between the aggregate measurement of BYSE and ASDAQ.

```
1  np.random.seed(17)
2  num_individual_measurements = 10
3  agg_asdaq, se_asdaq = aggregate_measurement_with_se("ASDAQ", num_individual_measurements)
4  agg_byse, se_byse= aggregate_measurement_with_se("BYSE", num_individual_measurements)
5
6  delta = agg_byse - agg_asdaq
7  se_delta = np.sqrt(se_byse**2 + se_asdaq**2)
8  print(delta)
9  print(se_delta)
```

```
-2.2721337833056996
0.5065929285007608
```

Reducing Number of Measurements

- **Notice that if delta were positive, this would imply ASDAQ is cheaper than BYSE.**

Reducing Number of Measurements

- Notice that if delta were positive, this would imply ASDAQ is cheaper than BYSE.

- On the other hand a negative delta suggests that BYSE is cheaper.

Reducing Number of Measurements

- Notice that if delta were positive, this would imply ASDAQ is cheaper than BYSE.
- On the other hand a negative delta suggests that BYSE is cheaper.
- So if our delta is significantly lower than 0 we will start sending trades to BYSE.

Reducing Number of Measurements

- We don't know for certain if this is right, but we're willing to take this bet with at most a 5% chance of being wrong that BYSE is better.

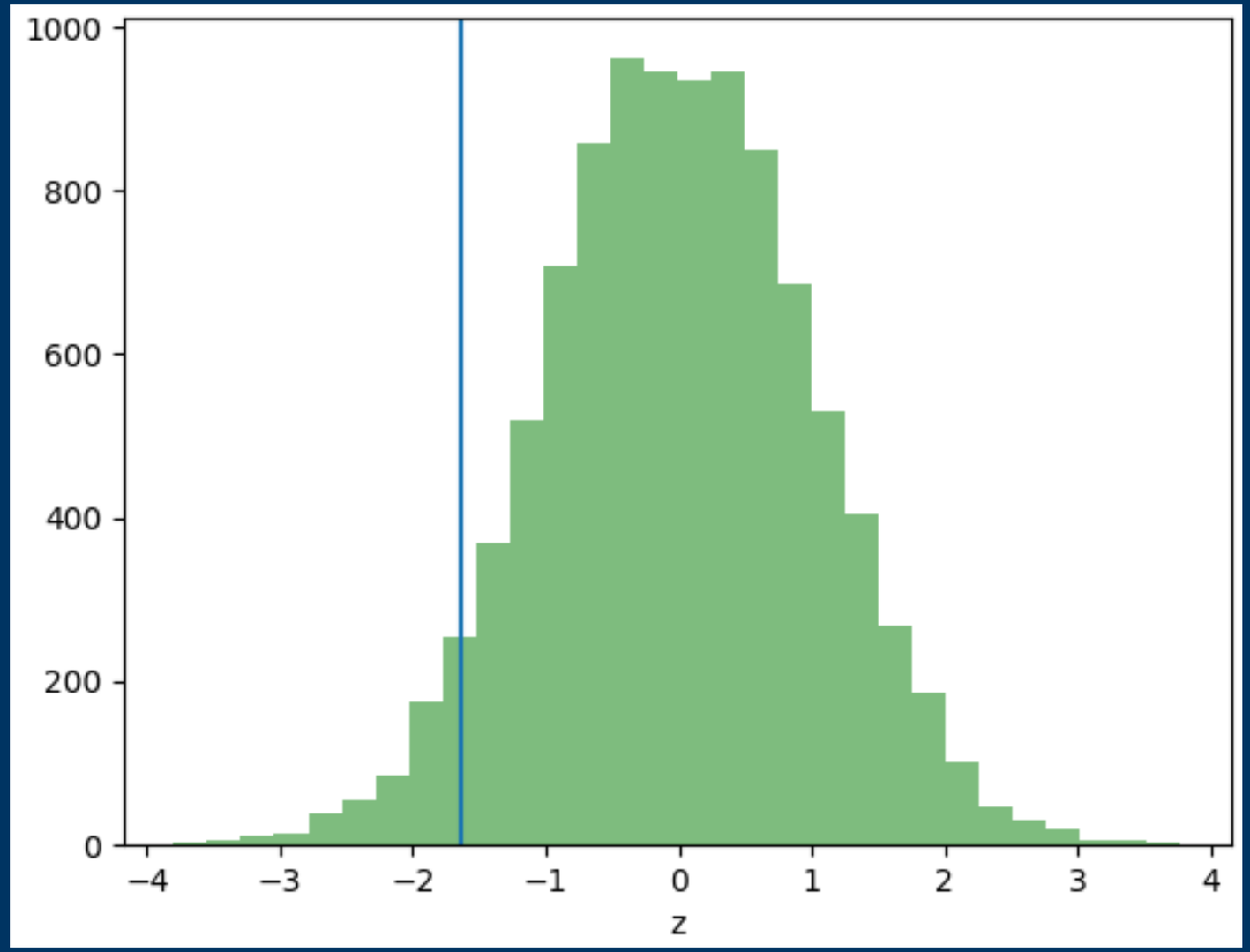
Reducing Number of Measurements

- We don't know for certain if this is right, but we're willing to take this bet with at most a 5% chance of being wrong that BYSE is better.

- For convenience, it's easier to work with z = delta/se_delta, the z-score which standardizes the distribution of the deltas to a gaussian N(0, 1) distribution.

Reducing Number of Measurements

```
1  np.random.seed(17)
2  z = np.random.normal(size=(10000,))
3  plt.hist(z, 30, color = 'green', alpha = 0.5)
4  plt.axvline(-1.64)
5  plt.xlabel("z")
6  plt.show()
```



Reducing Number of Measurements

- The figure above is our standard gaussian distribution.

Reducing Number of Measurements

- The figure above is our standard gaussian distribution.
- There is a 5% chance that our z score (coming from our normalized delta) will fall to the left of the line.

Reducing Number of Measurements

- The figure above is our standard gaussian distribution.
- There is a 5% chance that our z score (coming from our normalized delta) will fall to the left of the line.
- If it does, you should bet that your assumption in step 1. is wrong. We should instead bet that the z is truly less than zero, which means that the expectation of delta is less than zero.

Reducing Number of Measurements

- The figure above is our standard gaussian distribution.
- There is a 5% chance that our z score (coming from our normalized delta) will fall to the left of the line.
- If it does, you should bet that your assumption in step 1. is wrong. We should instead bet that the z is truly less than zero, which means that the expectation of delta is less than zero.
- That implies that BYSE is cheaper than ASDAQ. When z is to the left of the vertical line, we say that the aggregate is statistically significant.

Reducing Number of Measurements

- Remember:
  - If $z < -1.64$, then we act as if BYSE is cheaper than ASDAQ and start trading there.

Reducing Number of Measurements

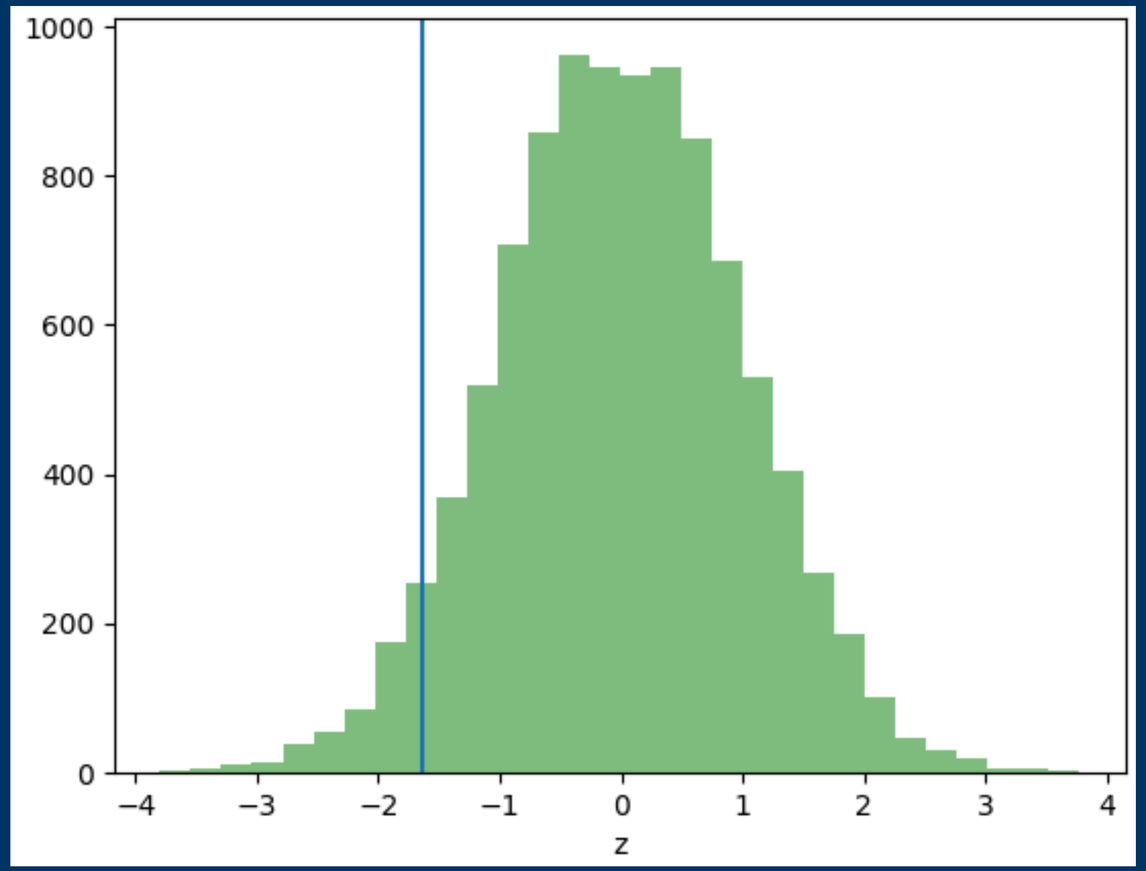
- Remember:
  - If $z < -1.64$, then we act as if BYSE is cheaper than ASDAQ and start trading there.

  - Otherwise we act as if BYSE is no better than ASDAQ and continue trading on ASDAQ.

False Positives

- When your bet is wrong: when z falls to the left of the line, but the expectation is non-negative, this is called a **false positive**.

- By our design, false positives occur 5% of the time.



Being Practical

- How big of a difference in the execution costs do we actually care about?

Being Practical

- How big of a difference in the execution costs do we actually care about?

- If BYSE is cheaper on average by 1, is it really worth switching over? What about 0.01? Is 0.001 too small?

Being Practical

- How big of a difference in the execution costs do we actually care about?
- If BYSE is cheaper on average by 1, is it really worth switching over? What about 0.01? Is 0.001 too small?
- After all, switching over can introduce us to new risk that our old model was not exposed to.

Being Practical

- How big of a difference in the execution costs do we actually care about?
- If BYSE is cheaper on average by 1, is it really worth switching over? What about 0.01? Is 0.001 too small?
- After all, switching over can introduce us to new risk that our old model was not exposed to.
- So another term to consider is practical significance, denoted prac_sig.

Being Practical

Suppose prac_sig $= 1.0$. Then even if we show a statistically significant measurement, if it turns out that $-0.1 < delta < 0.0$ (i.e. $\text{delta} \geq -\text{prac\_sig}$), we should still act as if $\text{delta} = 0$.

How many measurements is enough?

- **Recall:**
  - **z = delta / se_delta**

How many measurements is enough?

- **Recall:**
  - **z = delta / se_delta**

  - **se_delta = (se_byse^2 + se_asdaq^2)^(1/2)**

How many measurements is enough?

- Recall:
  - z = delta / se_delta

  - se_delta = (se_byse^2 + se_asdaq^2)^(1/2)

- However we also have this formulation:

How many measurements is enough?

- Recall:
  - z = delta / se_delta

  - se_delta = (se_byse^2 + se_asdaq^2)^(1/2)

- However we also have this formulation:
  - se = sd_1 / (N)^(1/2)

How many measurements is enough?

- Recall:
  - z = delta / se_delta

  - se_delta = (se_byse^2 + se_asdaq^2)^(1/2)

- However we also have this formulation:
  - se = sd_1 / (N)^(1/2)

  - se_delta = sd_1_delta / (N)^(1/2)

How many measurements is enough?

- Recall:
  - z = delta / se_delta

  - se_delta = (se_byse^2 + se_asdaq^2)^(1/2)

- However we also have this formulation:
  - se = sd_1 / (N)^(1/2)

  - se_delta = sd_1_delta / (N)^(1/2)

- So z = (N)^(1/2) * delta / sd_1_delta

How many measurements is enough?

- Given z = (N)^(1/2) * delta / sd_1_delta

How many measurements is enough?

- Given z = (N)^(1/2) * delta / sd_1_delta

- If we want a 5% false positive rate, recall this occurs when: z < -1.64

How many measurements is enough?

- Given z = (N)^(1/2) * delta / sd_1_delta
- If we want a 5% false positive rate, recall this occurs when: z < -1.64
- So we can use this inequality and substitute it in the above formula to get:

How many measurements is enough?

- Given z = (N)^(1/2) * delta / sd_1_delta
- If we want a 5% false positive rate, recall this occurs when: z < -1.64
- So we can use this inequality and substitute it in the above formula to get:
- N > (1.64 * sd_1_delta / delta)^2

How many measurements is enough?

- Given z = (N)^(1/2) * delta / sd_1_delta
- If we want a 5% false positive rate, recall this occurs when: z < -1.64
- So we can use this inequality and substitute it in the above formula to get:
- N > (1.64 * sd_1_delta / delta)^2
- Therefore you should choose:

How many measurements is enough?

- Given z = (N)^(1/2) * delta / sd_1_delta
- If we want a 5% false positive rate, recall this occurs when: z < -1.64
- So we can use this inequality and substitute it in the above formula to get:
- N > (1.64 * sd_1_delta / delta)^2
- Therefore you should choose:
  - N = {(1.64 * sd_1_delta / delta)^2}, i.e. the ceiling.

How many measurements is enough?

```python
np.random.seed(17)
num_ind = 100
sd_1_asdaq = np.array([trading_system("ASDAQ") for _ in range(100)]).std() # This is historically logged data we already have
sd_1_byse  = sd_1_asdaq # using the trick we discussed
sd_1_delta = np.sqrt(sd_1_asdaq**2 + sd_1_byse**2)
print(sd_1_delta)

def ab_test_design(sd_1_delta: float, prac_sig: float = 1.0): # we default our practical significance to 1.0
    num_individual_measurements = (1.64 * sd_1_delta / prac_sig) ** 2 # ensure z < -1.64 when delta < -1 * prac_sig
    return np.ceil(num_individual_measurements) # round up to nearest integer

print(ab_test_design(sd_1_delta))
```

```
1.5850244424014406
7.0
```

So if we take 7 individual measurements, we'll have a 5% chance of a false positive of incorrectly acting as if BYSE is better than ASDAQ.

The more data the better?

False Discovery Rates

False Discovery Rates (FDR)

Okay first thing's first, let's de-brief on hypothesis testing:

Hypothesis Testing

We have 72 lukemia patients

1. 47 with ALL (acute Lymphobastic leukemia)
2. 25 with AML (acute myeloid leukemia, a worse prognosis)

Each disease had a genetic activity measured for a panel of 7,128 genes.

Let's download the data and construct a histogram comparing the genetic activities in the two groups for gene 136

False Discovery Rates

False Discovery Rates (FDR)

Okay first thing's first, let's de-brief on hypothesis testing:

Hypothesis Testing

We have 72 lukemia patients

1. 47 with ALL (acute Lymphobastic leukemia)
2. 25 with AML (acute myeloid leukemia, a worse prognosis)

Each disease had a genetic activity measured for a panel of 7,128 genes.

Let's download the data and construct a histogram comparing the genetic activities in the two groups for gene 136

```
1  data = pd.read_csv("https://hastie.su.domains/CASI_files/DATA/leukemia_big.csv")
```

✓ 0.6s

False Discovery Rates

	ALL	ALL.1	ALL.2	ALL.3	ALL.4	ALL.5	ALL.6	ALL.7	ALL.8	ALL.9	...
0	-1.533622	-0.867610	-0.433172	-1.671903	-1.187689	-1.127234	-1.045409	-0.106917	-1.198796	-1.190899	...
1	-1.235673	-1.275501	-1.184492	-1.596424	-1.335256	-1.113730	-0.800880	-0.745177	-0.849312	-1.190899	...
2	-0.333983	0.375927	-0.459196	-1.422571	-0.797493	-1.362768	-0.671954	-1.175674	0.320813	0.646610	...
3	0.488702	0.444011	0.436264	0.193353	0.235632	-0.360312	0.184941	0.425653	0.333983	0.235270	...
4	-1.300893	-1.229660	-1.325882	-1.818329	-1.311206	-1.513975	-1.651624	-1.339555	-0.593132	0.133302	...
...	...	...	...	...	...	...	...	...	...	...	...
7123	1.295992	-0.218494	1.132893	1.113077	0.719203	1.490610	0.483163	1.433292	0.737309	0.633018	...
7124	0.733853	0.378380	0.475669	0.148928	0.419502	1.000031	0.258833	-0.498831	-0.657700	-0.373663	...
7125	-0.301622	-0.663166	-0.530138	-0.625945	-0.487514	-0.172972	-0.052590	-0.512817	-1.005845	-1.245923	...
7126	0.133657	-0.663166	1.566946	0.871972	0.358999	0.080430	0.029891	1.553879	-0.144841	0.129578	...
7127	-0.825596	-0.611045	-0.805978	-1.037246	-0.742858	-0.670192	-0.758939	-0.959684	-1.044802	-1.204950	...

7128 rows × 72 columns

False Discovery Rates

```
index = 135
ALL = []
AML = []
for key in data.iloc[index].keys():
    if "ALL" in key and key != "ALL":
        ALL.append(data.iloc[index][key])
    else:
        if key != "AML":
            AML.append(data.iloc[index][key])
ALL = np.array(ALL)
ALL_mean = ALL.mean()
AML = np.array(AML)
AML_mean = AML.mean()
```

False Discovery Rates

```
1  print(ALL_mean)
2  print(AML_mean)
```

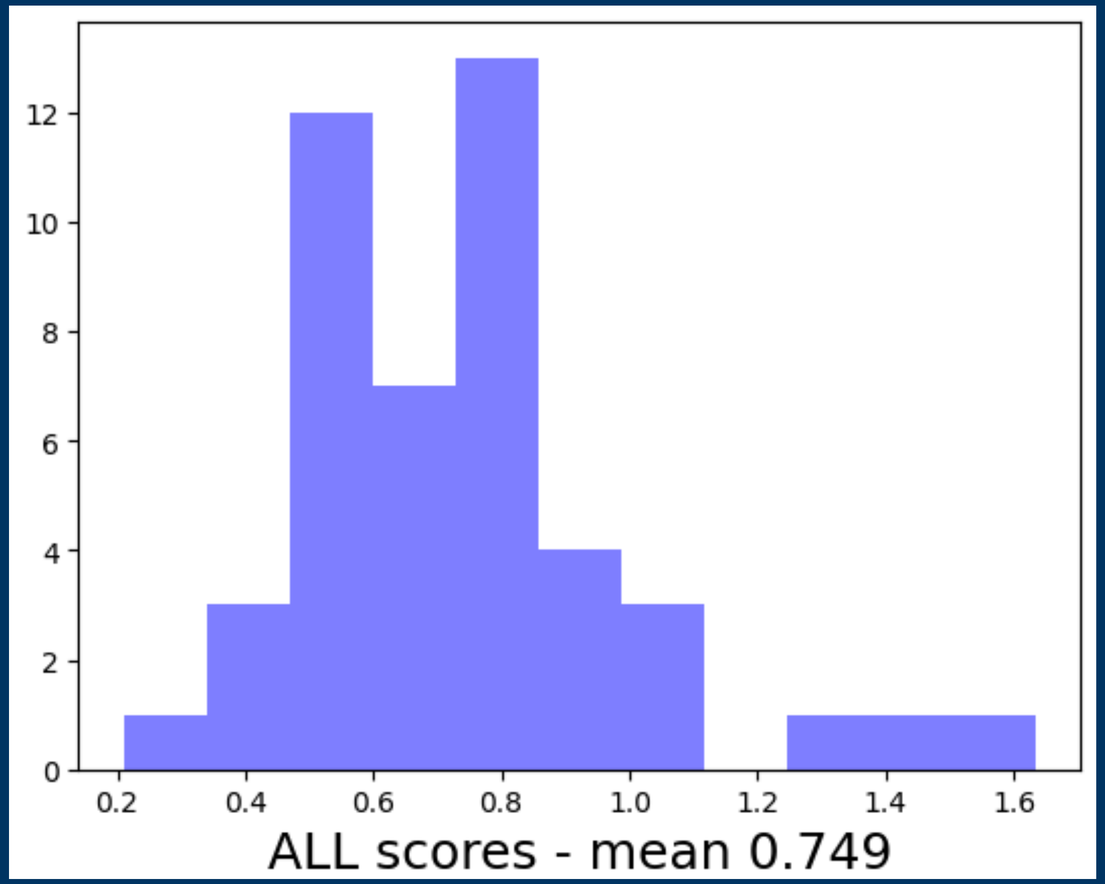
0.0s

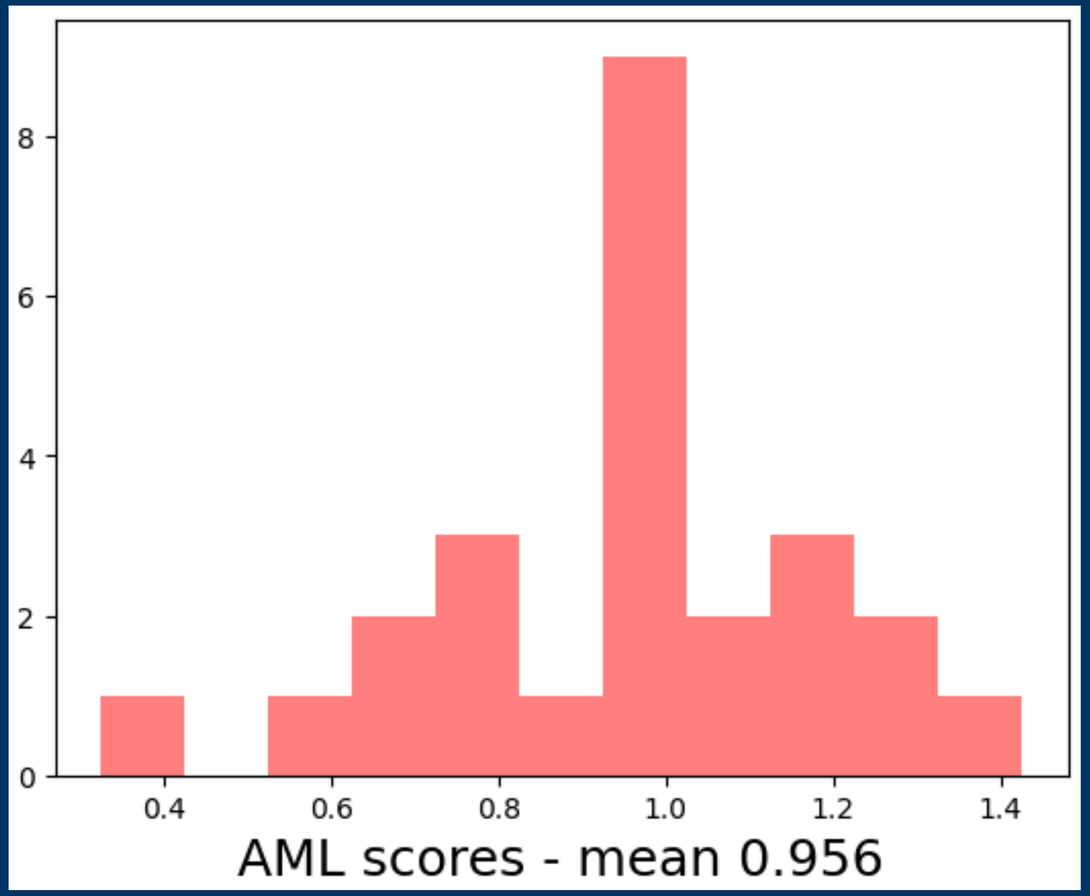
```
0.7488660087608695
0.9560340174399999
```

```
1  plt.hist(ALL, color = "blue", alpha=0.5, bins=11)
2  plt.xlabel(f"ALL scores - mean {ALL_mean:.3f}", size=18)
3  plt.show()
4  plt.hist(AML, color = "red", alpha=0.5, bins=11)
5  plt.xlabel(f"AML scores - mean {AML_mean:.3f}", size=18)
6  plt.show()
```

0.0s

False Discovery Rates





It appears that the AML group shows greater activity for gene 136 on average compared to the ALL group. Let's compute the t and p values.

False Discovery Rates

```
1  t_val, p_val = stats.ttest_ind(AML, ALL)
```

✓ 0.0s

```
1  print(t_val, p_val)
```

✓ 0.0s

We see a p value far below 0.05 which is suggestive that perhaps there is a difference betwen ALL and AML on gene 136. Could it be that gene 136 is an indicator of the presence of AML?

Is there anything wrong with our analysis?

False Discovery Rates

Too Many candidates:

Firstly there are 7128 genes. Don't you think by chance we'd just naturally get some t values exceeding 3 when performing a two-sample t-test?

To see the chances of this, let's empirically compute the distribution of t-values:

```
1  cols_all = [col for col in data.columns if col.startswith("ALL")]
2  cols_aml = [col for col in data.columns if col.startswith("AML")]
```

✓ 0.0s

```
1  t_values = []
2  for _, row in data.iterrows():
3      values_all = row[cols_all].values
4      values_aml = row[cols_aml].values
5      t, _ = stats.ttest_ind(values_aml, values_all)
6      t_values.append(t)
```

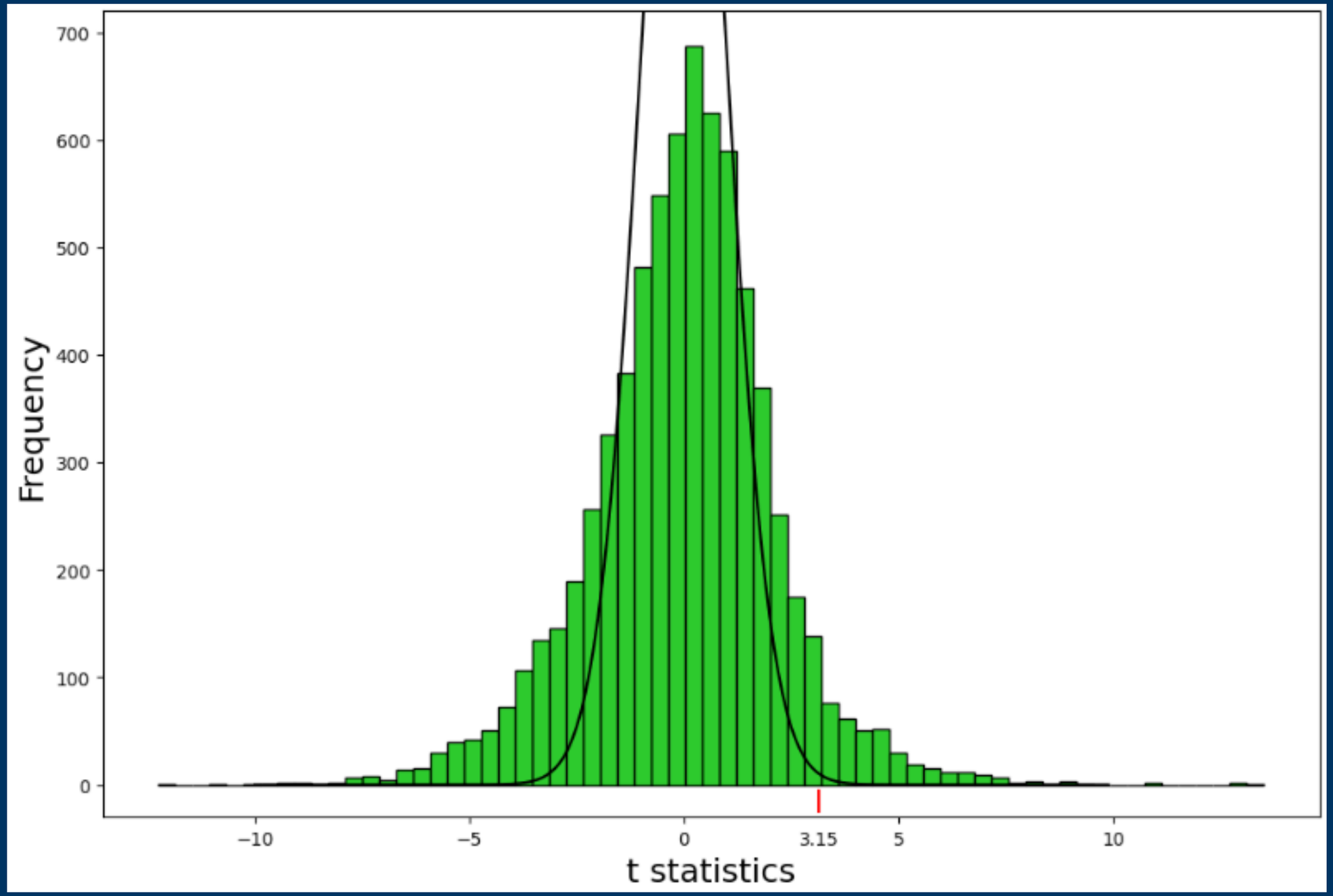
False Discovery Rates

```
1  dist_x = np.arange(-10, 10, 0.1)
2  dist_t = stats.t.pdf(dist_x, df=70)
```

✓ 0.0s

```
1  fig, ax = plt.subplots(figsize=(12, 8))
2
3  hist_info = ax.hist(t_values, bins=65, edgecolor="k", facecolor="limegreen")
4  bin_y, bin_x = hist_info[0], hist_info[1]
5  hist_area = ((bin_x[1:] - bin_x[:-1]) * bin_y).sum()
6
7  ax.plot(dist_x, dist_t * hist_area, c="k")
8
9  ax.set_xticks([t_val], minor=True)
10 ax.set_xticklabels([np.round(t_val, 2)], minor=True)
11 ax.tick_params(axis='both', which='minor', length=4, color="white")
12
13 ax.plot([t_val, t_val], [-5, -25], c="r")
14
15 ax.set_xlabel("t statistics", size=18)
16 ax.set_ylabel("Frequency", size=18)
17 ax.set_ylim(-30, 720)
```

False Discovery Rates



False Discovery Rates

This plot reveals two complications!

1. There are 400 other genes that have a t value exceeding 3.
2. The theoretical t-distribution in black is far more limiting than what we are seeing empirically.

This is a compilication of having many many candidates that we can look at, thus increasing our chances of a `False Discovery`. Later we will learn that we would actually need a t-value of about 6.16 to actually show if there is any statistical power in gene indicating AML.

Thank You

(Up next: Multi Armed Bandits (MAB))

Contact Me: andrewlizarraga@g.ucla.edu