Remedial- Session:

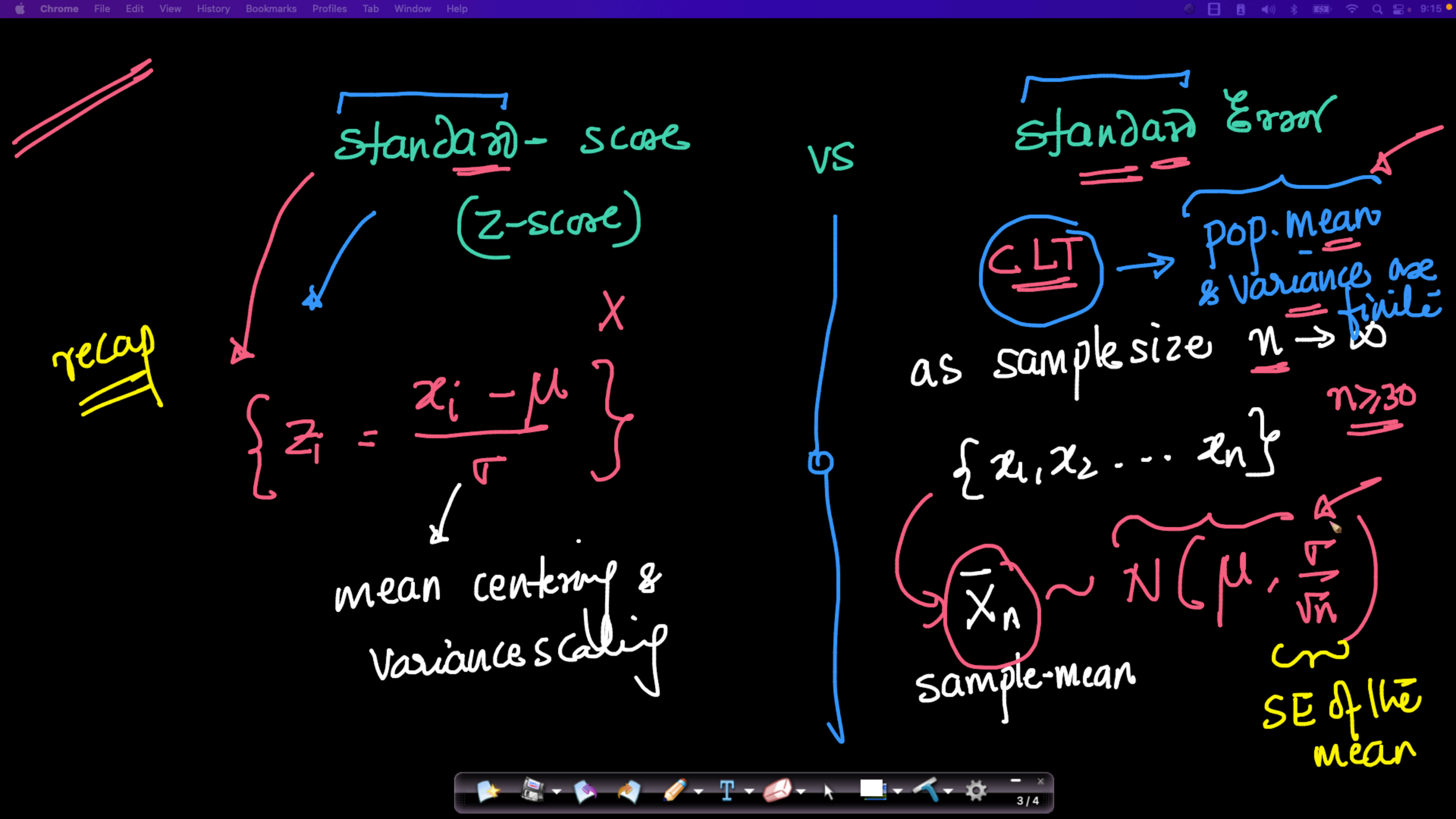
1. Standard-score (z-score) vs standard-error — Practical examples
2. Confidence Intervals → Uber case-study ← (CLT ←, bootstrapping)
3. CLT: sample size & #samples → simulation } ←
4. ✓ Conditional Probability
   ✓ Joint Probability
   ✓ Bayes Theorem
   } → interview Questions + hacks to solve } ✓

Ops:

[Chat → conversation ; Y/N } ✓

✓ { @Questions tab → upvote

Zero it out


Standard- score
(Z-score)
recap
$\{z_i = \frac{x_i - \mu}{\sigma}\}$
mean centering &
Variance scaling
VS
Standard Error
CLT
pop. mean & variance are finite
as sample size $n \to \infty$
$n \geq 30$
$\{x_1, x_2 \cdots x_n\}$
$\bar{X}_n \sim N(\mu, \frac{\sigma}{\sqrt{n}})$
sample-mean
SE of the mean

X ↻ sample

$\mathcal{D} = \{x_1, x_2 \dots x_i \dots x_n\}$

dataset

x: height of babies

$z_i = \dfrac{x_i - \mu}{\sigma}$ ↳ ... ↳ ...

✓ Pop. mean & std-dev

Unbiased

$\dfrac{1}{n-1}\sum_{i=1}^{n}(x_i - \mu)^2$

$\hat{\mu}$ = estimate of pop. mean $\approx$ sample-mean if n is not too small

$\hat{\sigma} \approx$ sample. std-dev if → ✓ $n \geq 30$

(e.g)

Babies:

n - babies

Dataset

hi, wi

cms kgs

std. Scores

i : 1 → n

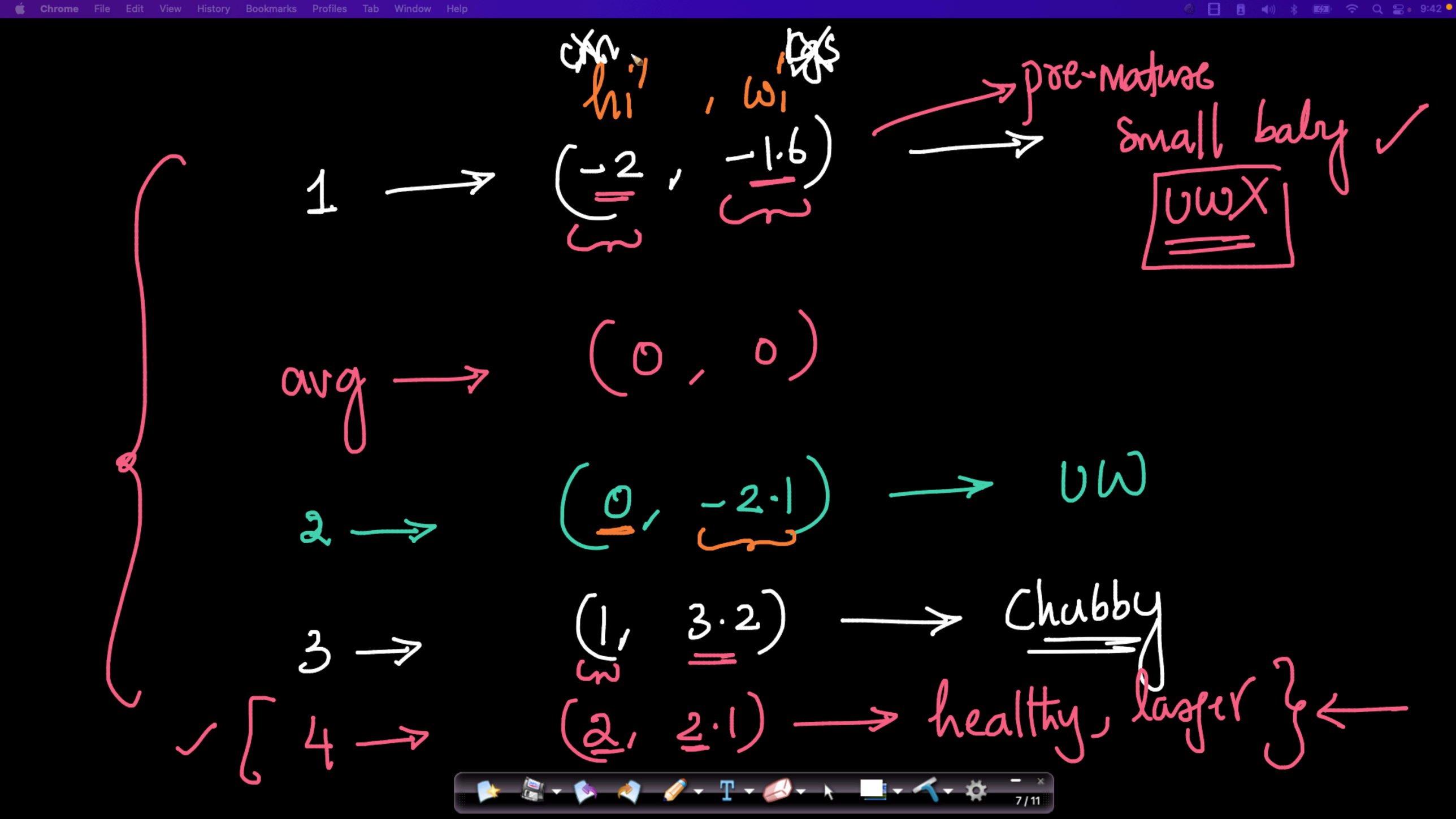
✓ { Task: Underweight; good; overweight/chubby

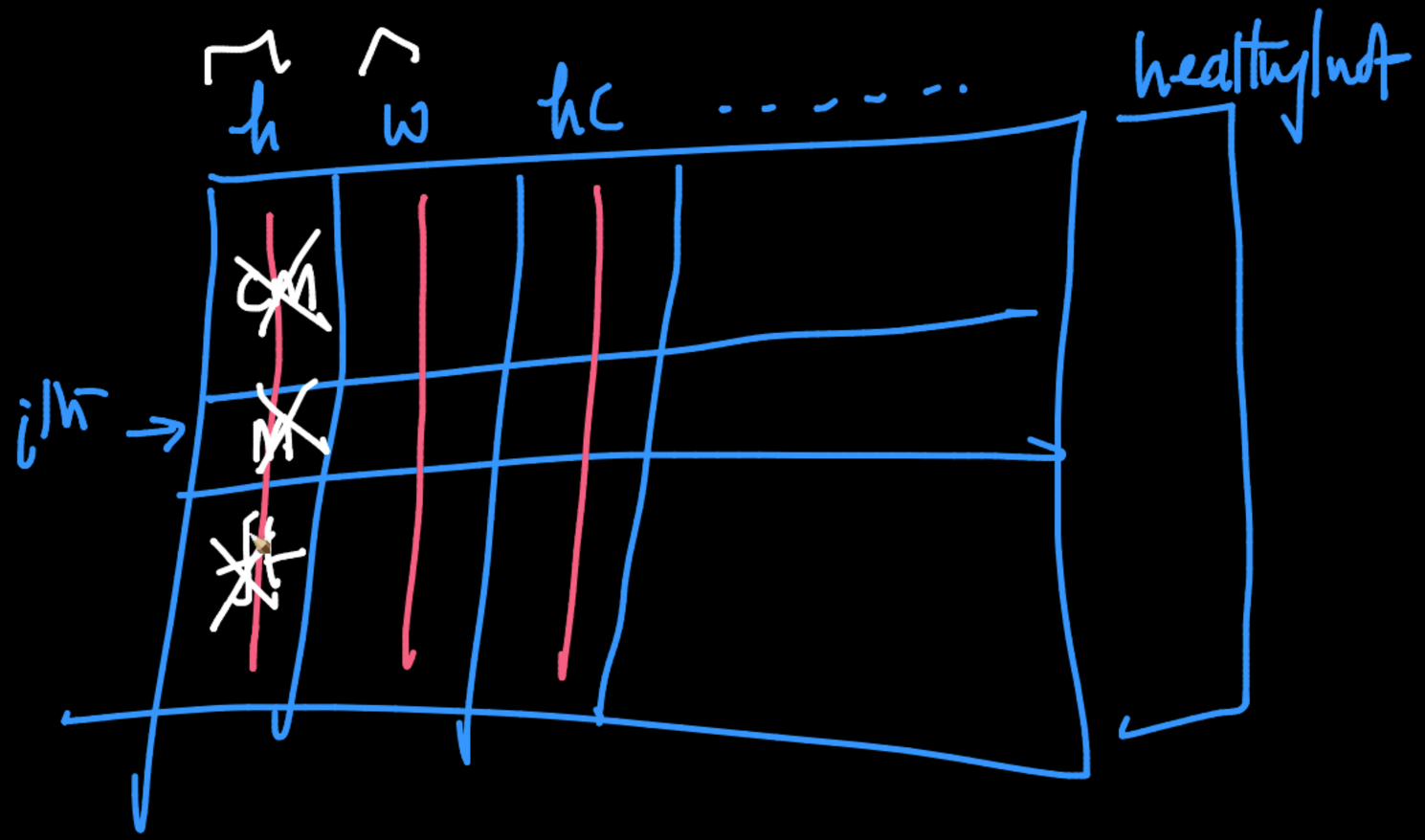
$$h_i' = \frac{h_i - \mu_h}{\sigma_h}$$

$$w_i' = \frac{w_i - \mu_w}{\sigma_w}$$

Standardized

getting rid of the units


h1 , w1
1 → (-2 , -1.6) → pre-mature
small baby ✓
UWX
avg → (0 , 0)
2 → (0 , -2.1) → UW
3 → (1 , 3.2) → Chubby
4 → (2 , 2.1) → healthy, lager


h
w
hc
healthy/not
i/h


dist
0-1KM
2KM,
10KM

✓ [Standardization

↓

widely used

Min-Max scaling

↓

Computer vision

b. e.g

$x_i = \cancel{100}\ 50$

$\mu = 20$

$\sigma = 10$

$z_i = \dfrac{\cancel{100}\ 50 - 20}{10} = \dfrac{30}{10} = 3$ ✓

(1) $x_i = \mu + 2\sigma \Rightarrow z_i = 2$

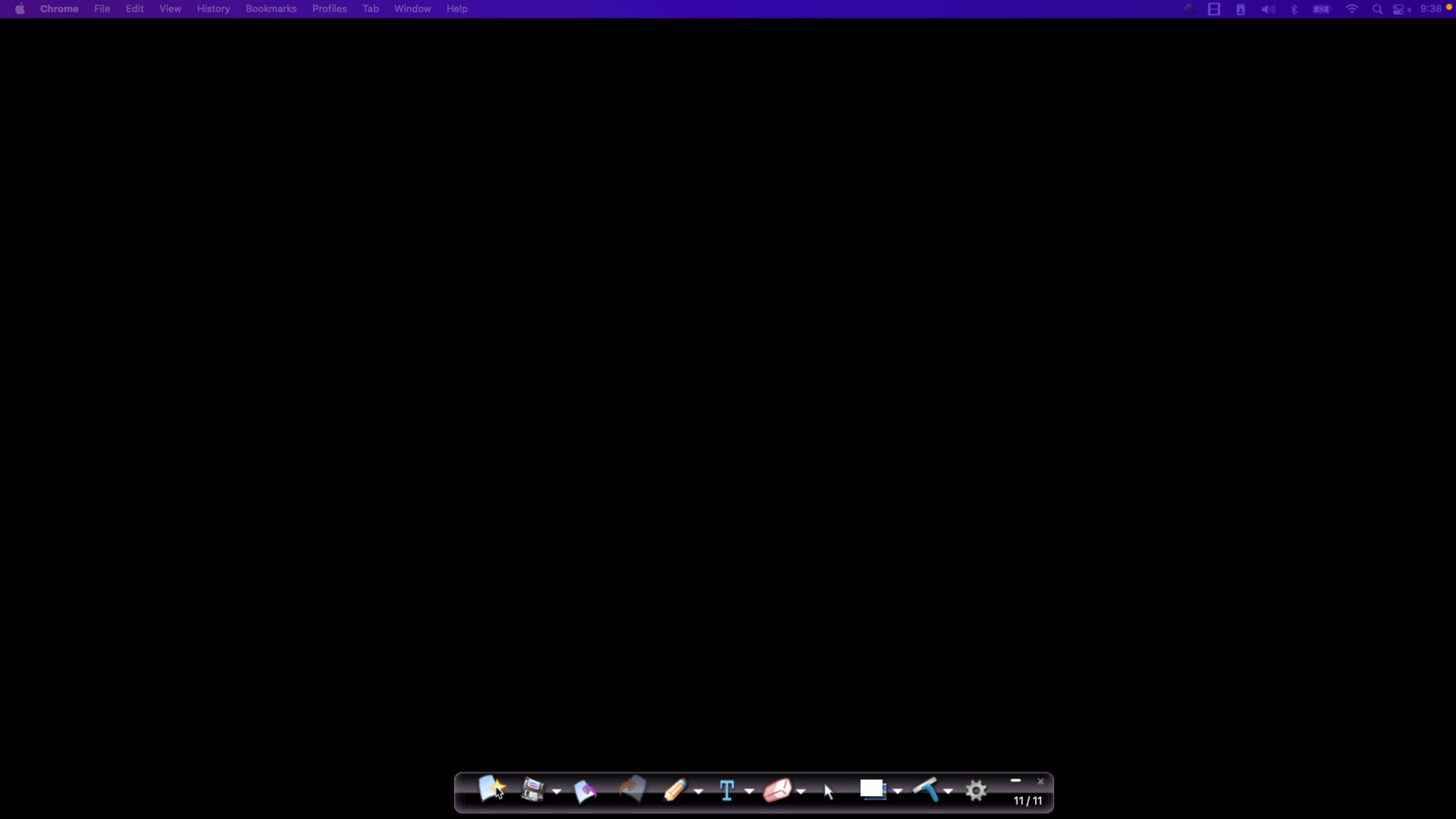
$x_i = \mu \Rightarrow z_i = 0$

$x_i = \mu + \sigma \Rightarrow z_i = 1$

$x_i = \mu - \sigma \Rightarrow z_i = -1$

$z_i = \dfrac{x_i - \mu}{\sigma}$

$(-\infty, \infty)$


Chrome File Edit View History Bookmarks Profiles Tab Window Help
9:38
11 / 11

Standard error:

CLT:

X: finite pop. mean & std-dev

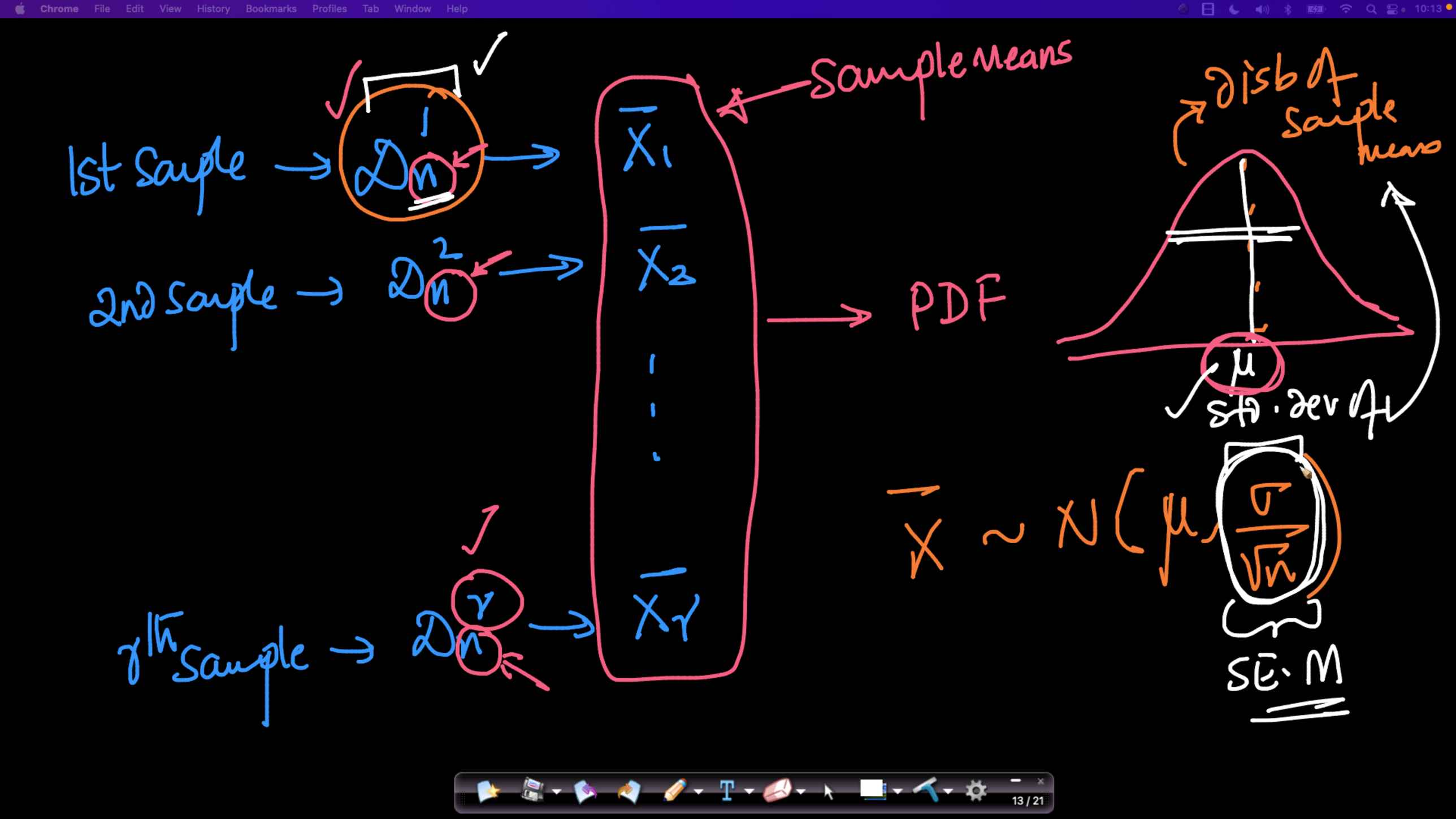
h/ω/hc/ ...

samples of size n

$\mathcal{D}_n = \{x_1, x_2, \ldots x_n\}$ ← amazon delivery time

↓

mean: $\bar{X}_n$



$$\bar{X} \sim N\left(\mu, \frac{\sigma}{\sqrt{n}}\right)$$

$$\bar{X} \sim N\left(\mu, \frac{\sigma}{\sqrt{n}}\right)$$

$\frac{\sigma^2}{n}$; Var

Sample size n ↑

$n\uparrow$ $\sqrt{n}\uparrow$ $\frac{1}{\sqrt{n}}\downarrow$

Std. dev of the dist of Sample Means = SEM

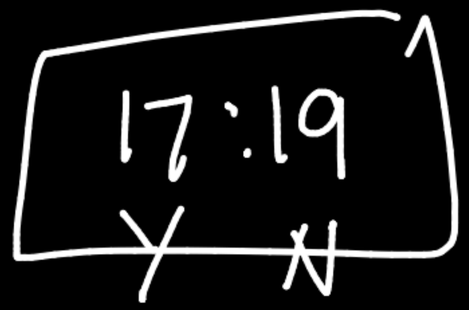
(r & n)

application of SEM $= \frac{\sigma}{\sqrt{n}}$

C.I of the pop. Mean

(little later code & sim)

5-6 Remedial Sessions

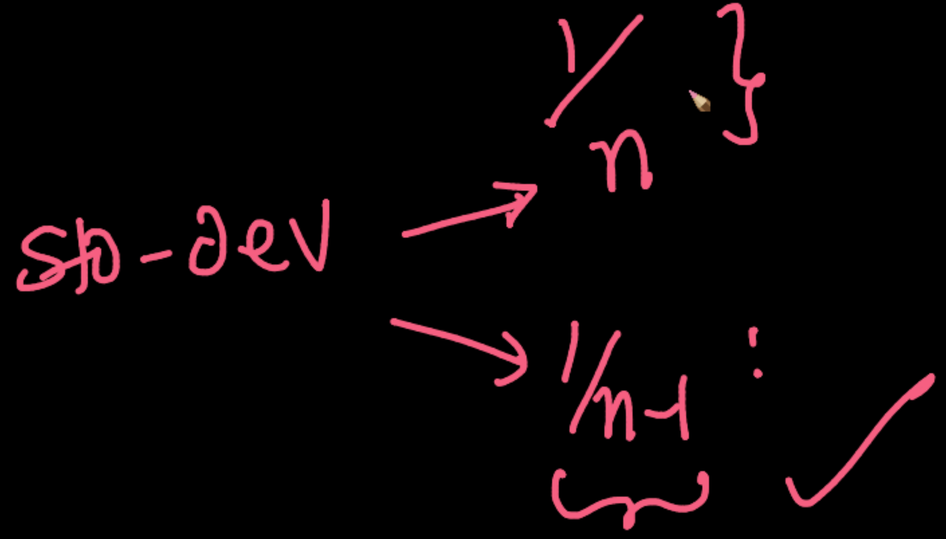


15:12

2 remedial sessions ← stay in this batch

more → other batch & repeat

38 : 1


std-dev
1/n
1/n-1

$$X = \{\overset{0}{H}, \overset{1}{T}\}$$

mean = ?

$$\bar{X} = N\left(\mu, \frac{\sigma}{\sqrt{n}}\right) \quad \text{as} \quad n \to \infty$$

sample-size

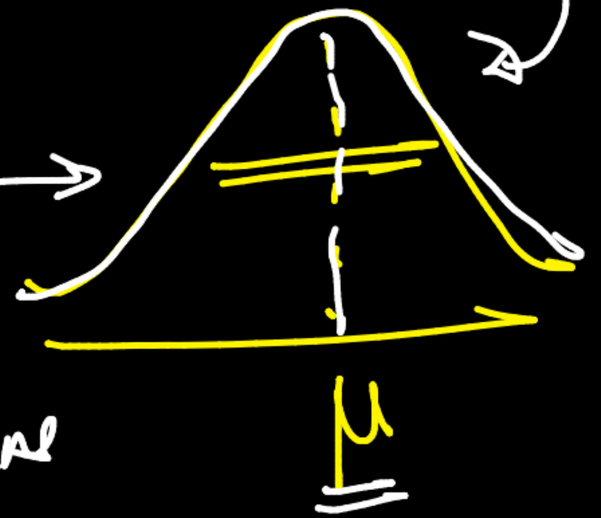
$\bar{X}_1, \bar{X}_2, \bar{X}_3 \cdots \bar{X}_r$ — r-times

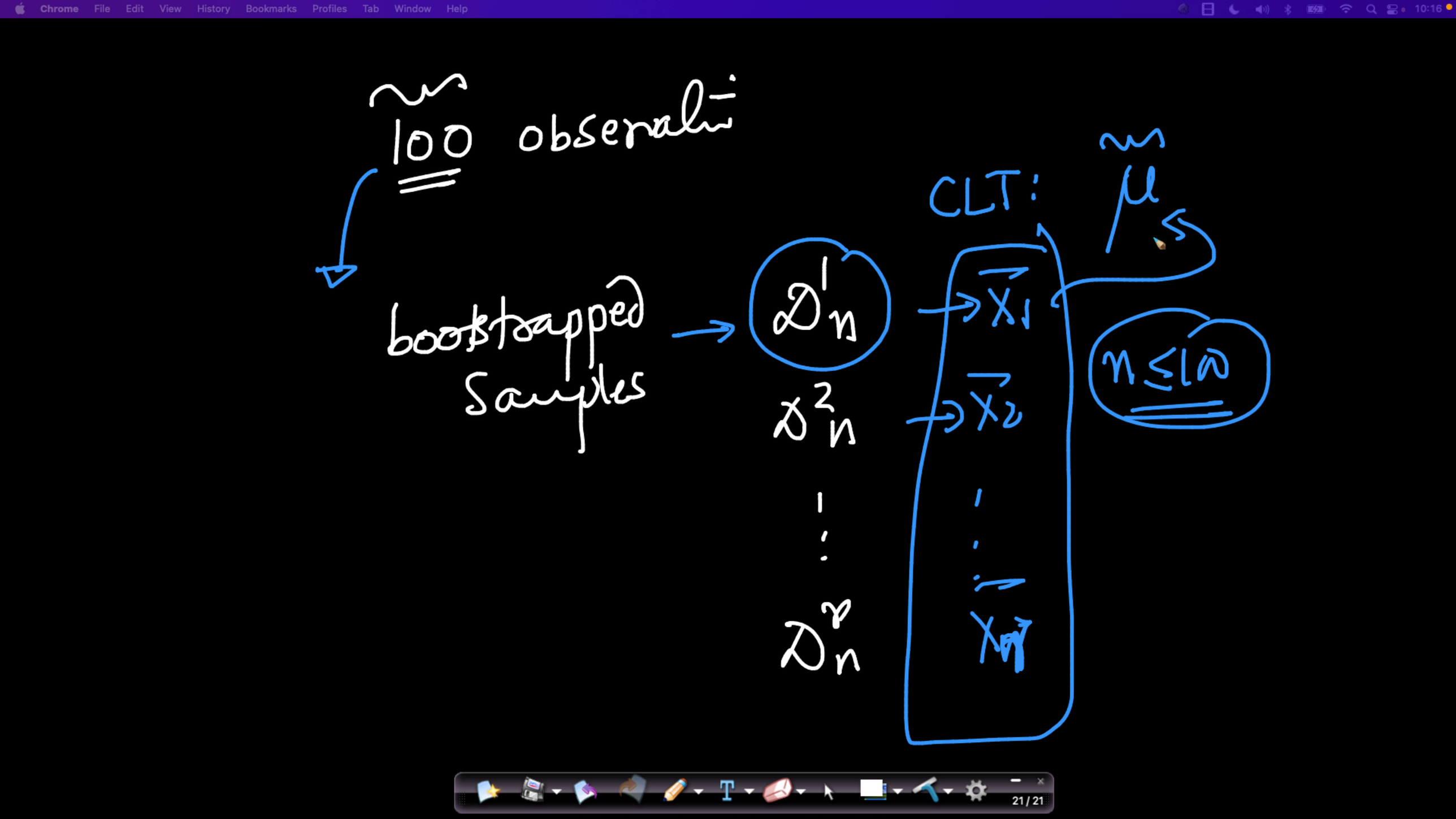
$\sigma = \sqrt{n} \cdot s$

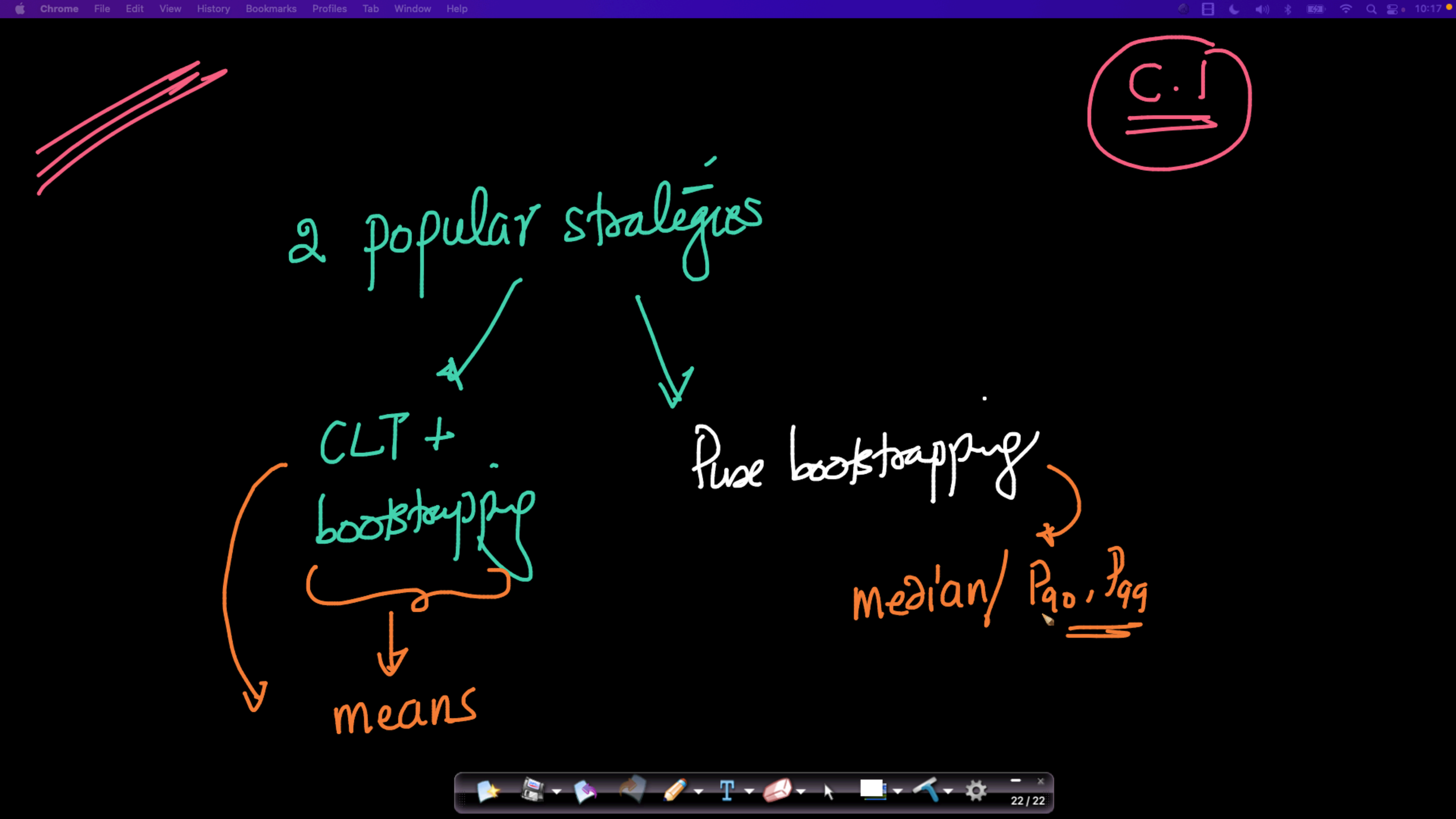
estimate pop. stdev

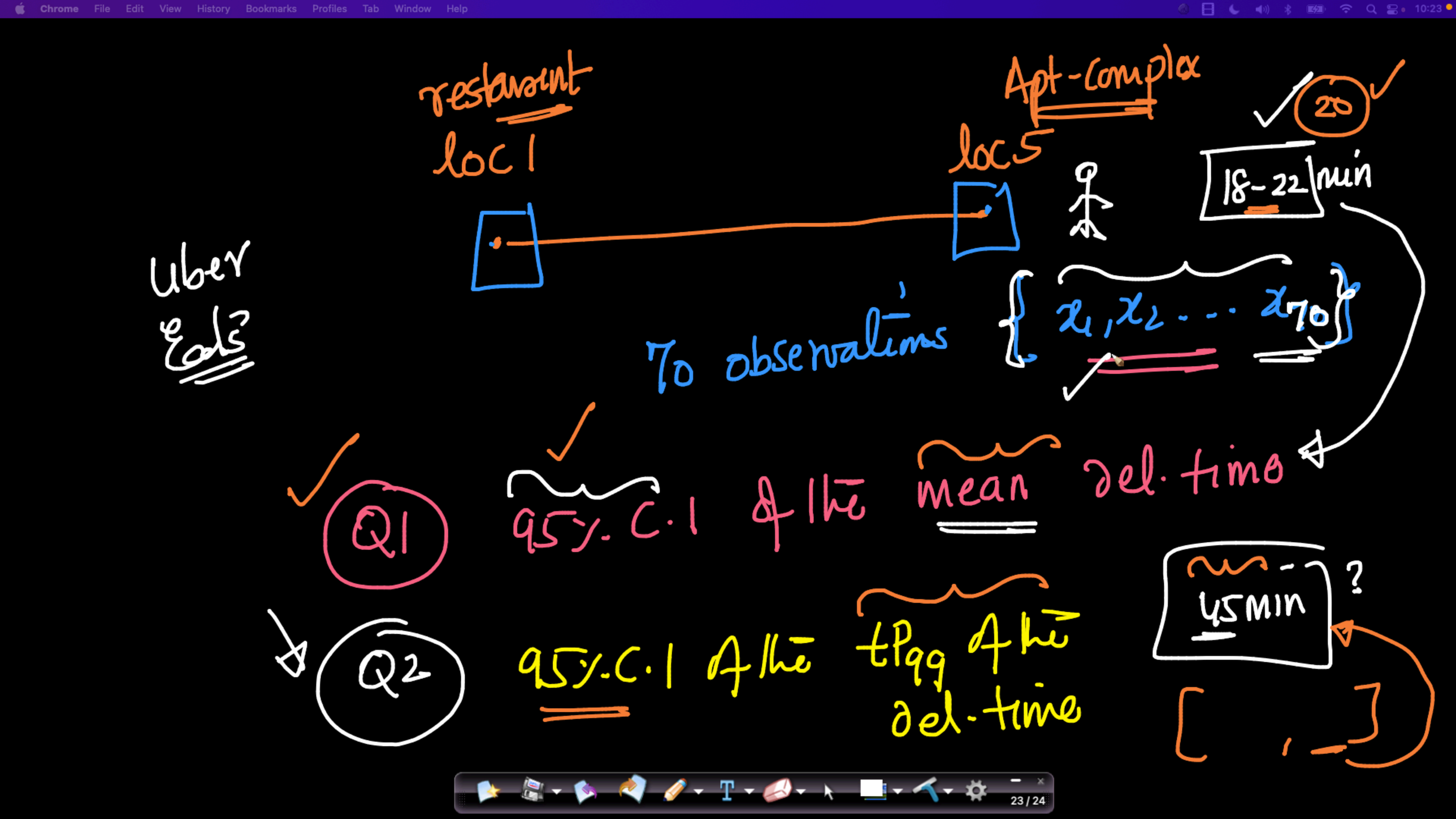
{ std-dev $= s$ of the distb of sample means

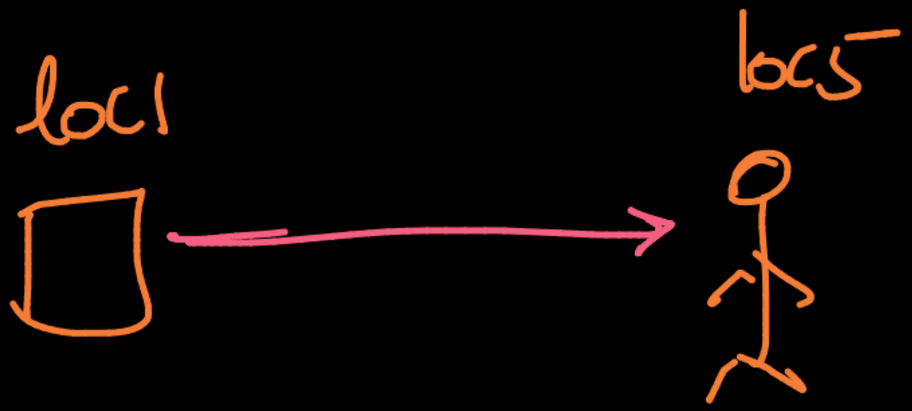
$$\frac{\sigma}{\sqrt{n}} = s$$

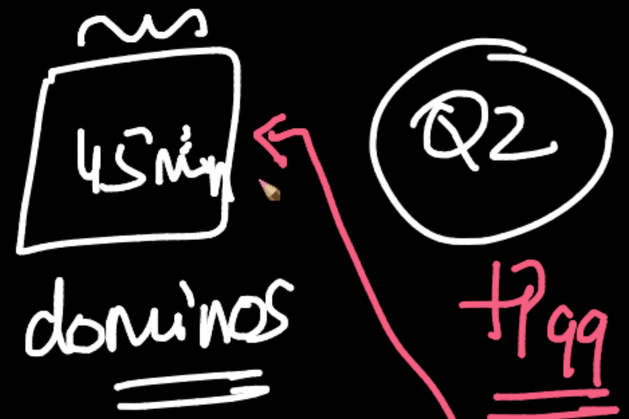



100 observations
bootstrapped samples
CLT:
μ
$n \leq 100$
21 / 21


C.I
2 popular strategies
CLT + bootstrapping
means
Pure bootstrapping
median/ P90, P99


Chrome File Edit View History Bookmarks Profiles Tab Window Help
10:23
restaurant
loc 1
Apt-complex
loc 5
20
18-22 min
Uber Eats
To observations
$x_1, x_2 \ldots x_{70}$
Q1
95% C.I of the mean del. time
Q2
95% C.I of the tP99 of the del. time
45 min
?
[,]
23 / 24





95c · 14
tp99

[41, 45]

+ Code + Text Reconnect

```
[ ]  from scipy import stats
     import seaborn as sns
     import  numpy as np
     import matplotlib.pyplot as plt
```

Uber Data

```
[ ]  id = "1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE"
     print("https://drive.google.com/uc?export=download&id=" + id)
```

```
https://drive.google.com/uc?export=download&id=1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE
```

```
[ ]  !wget "https://drive.google.com/uc?export=download&id=1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE" -O Uber_dataset.zip
```

```
--2022-07-01 13:34:47--  https://drive.google.com/uc?export=download&id=1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE
Resolving drive.google.com (drive.google.com)... 74.125.142.100, 74.125.142.102, 74.125.142.101, ...
Connecting to drive.google.com (drive.google.com)|74.125.142.100|:443... connected.
HTTP request sent, awaiting response... 303 See Other
Location: https://doc-0c-ag-docs.googleusercontent.com/docs/securesc/ha0ro937gcuc7l7deffksulhg5h7mbp1/hg1t9f6r3fq7l0
Warning: wildcards not supported in HTTP.
--2022-07-01 13:34:48--  htt                                                    c/ha0ro937gcuc7l7deffksulhg5h7mbp1
```

+ Code + Text Reconnect

```
import matplotlib.pyplot as plt
```

[ ]

Uber Data

```
id = "1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE"
print("https://drive.google.com/uc?export=download&id=" + id)
```

https://drive.google.com/uc?export=download&id=1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE

```
!wget "https://drive.google.com/uc?export=download&id=1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE" -O Uber_dataset.zip
```

```
--2022-07-01 13:34:47--  https://drive.google.com/uc?export=download&id=1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE
Resolving drive.google.com (drive.google.com)... 74.125.142.100, 74.125.142.102, 74.125.142.101, ...
Connecting to drive.google.com (drive.google.com)|74.125.142.100|:443... connected.
HTTP request sent, awaiting response... 303 See Other
Location: https://doc-0c-ag-docs.googleusercontent.com/docs/securesc/ha0ro937gcuc7l7deffksulhg5h7mbp1/hg1t9f6r3fq7l0
Warning: wildcards not supported in HTTP.
--2022-07-01 13:34:48--  https://doc-0c-ag-docs.googleusercontent.com/docs/securesc/ha0ro937gcuc7l7deffksulhg5h7mbp1
Resolving doc-0c-ag-docs.googleusercontent.com (doc-0c-ag-docs.googleusercontent.com)... 74.125.142.132, 2607:f8b0:4
Connecting to doc-0c-ag-docs.googleusercontent.com (doc-0c-ag-docs.googleusercontent.com)|74.125.142.132|:443... con
HTTP request sent, awaiting response... 200 OK
Length: 18251707 (17M) [application/zip]
Saving to: 'Uber_dataset.zip'
```

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:28

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=yrGQlt3n_fgX

+ Code + Text Reconnect

Uber Data

GCP

```
[ ]  id = "1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE"
     print("https://drive.google.com/uc?export=download&id=" + id)
```

https://drive.google.com/uc?export=download&id=1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE

```
[ ]  !wget "https://drive.google.com/uc?export=download&id=1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE" -O Uber_dataset.zip
```

```
--2022-07-01 13:34:47--  https://drive.google.com/uc?export=download&id=1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE
Resolving drive.google.com (drive.google.com)... 74.125.142.100, 74.125.142.102, 74.125.142.101, ...
Connecting to drive.google.com (drive.google.com)|74.125.142.100|:443... connected.
HTTP request sent, awaiting response... 303 See Other
Location: https://doc-0c-ag-docs.googleusercontent.com/docs/securesc/ha0ro937gcuc7l7deffksulhg5h7mbp1/hg1t9f6r3fq7l0
Warning: wildcards not supported in HTTP.
--2022-07-01 13:34:48--  https://doc-0c-ag-docs.googleusercontent.com/docs/securesc/ha0ro937gcuc7l7deffksulhg5h7mbp1
Resolving doc-0c-ag-docs.googleusercontent.com (doc-0c-ag-docs.googleusercontent.com)... 74.125.142.132, 2607:f8b0:4
Connecting to doc-0c-ag-docs.googleusercontent.com (doc-0c-ag-docs.googleusercontent.com)|74.125.142.132|:443... con
HTTP request sent, awaiting response... 200 OK
Length: 18251707 (17M) [application/zip]
Saving to: 'Uber_dataset.zip'

Uber_dataset.zip    100%[===================>]  17.41M  48.6MB/s    in 0.4s
```

27 / 27

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:28

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=yrGQlt3n_fgX

+ Code + Text Reconnect

Uber Data

```
[ ]  id = "1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE"
     print("https://drive.google.com/uc?export=download&id=" + id)
```

```
https://drive.google.com/uc?export=download&id=1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE
```

```
[ ]  !wget "https://drive.google.com/uc?export=download&id=1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE" -O Uber_dataset.zip
```

```
--2022-07-01 13:34:47--  https://drive.google.com/uc?export=download&id=1NokZy4YzavFdTZlWcIUs47WW5M2A4ElE
Resolving drive.google.com (drive.google.com)... 74.125.142.100, 74.125.142.102, 74.125.142.101, ...
Connecting to drive.google.com (drive.google.com)|74.125.142.100|:443... connected.
HTTP request sent, awaiting response... 303 See Other
Location: https://doc-0c-ag-docs.googleusercontent.com/docs/securesc/ha0ro937gcuc7l7deffksulhg5h7mbp1/hg1t9f6r3fq7l0
Warning: wildcards not supported in HTTP.
--2022-07-01 13:34:48--  https://doc-0c-ag-docs.googleusercontent.com/docs/securesc/ha0ro937gcuc7l7deffksulhg5h7mbp1
Resolving doc-0c-ag-docs.googleusercontent.com (doc-0c-ag-docs.googleusercontent.com)... 74.125.142.132, 2607:f8b0:4
Connecting to doc-0c-ag-docs.googleusercontent.com (doc-0c-ag-docs.googleusercontent.com)|74.125.142.132|:443... con
HTTP request sent, awaiting response... 200 OK
Length: 18251707 (17M) [application/zip]
Saving to: 'Uber_dataset.zip'

Uber_dataset.zip    100%[===================>]  17.41M  48.6MB/s    in 0.4s
```

28 / 28

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:29

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=2iCNvRMC1SDa

+ Code + Text Reconnect

```
[ ]
Uber_dataset.zip    100%[===================>]  17.41M  48.6MB/s    in 0.4s

2022-07-01 13:34:49 (48.6 MB/s) - 'Uber_dataset.zip' saved [18251707/18251707]
```

```
!unzip Uber_dataset.zip
```

```
Archive:  Uber_dataset.zip
  inflating: uber_travel_data.csv
  inflating: __MACOSX/._uber_travel_data.csv
```

```
[ ] !ls -lrt
```

```
total 525784
-rw-r--r-- 1 root root 520141836 May 12 14:30 uber_travel_data.csv
drwxr-xr-x 1 root root      4096 Jun 29 13:44 sample_data
-rw-r--r-- 1 root root  18251707 Jul  1 13:34 Uber_dataset.zip
drwxr-xr-x 2 root root      4096 Jul  1 13:34 __MACOSX
```

```
[ ] import pandas as pd

    df = pd.read_csv("./uber_travel_data.csv")
    df.sample(100).head()
```

29 / 29

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:30

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=2iCNvRMC1SDa

+ Code + Text Reconnect

```
!ls -lrt
```

```
total 525784
-rw-r--r-- 1 root root 520141836 May 12 14:30 uber_travel_data.csv
drwxr-xr-x 1 root root      4096 Jun 29 13:44 sample_data
-rw-r--r-- 1 root root  18251707 Jul  1 13:34 Uber_dataset.zip
drwxr-xr-x 2 root root      4096 Jul  1 13:34 __MACOSX
```

```
import pandas as pd

df = pd.read_csv("./uber_travel_data.csv")
df.sample(100).head()
```

	sourceid	source	dstid	destination	travel_time
3699703	234	113, Press Colony, Press Colony, Mayapuri, New...	76	124, SPG Quarters, Sector 4, Pushp Vihar, New ...	2695
2441504	156	Doctor Satpal Sachdeva Marg, Keshav Puram, Tri...	230	N494, Block N, Raghubir Nagar, Tagore Garden E...	958
1824456	119	81, Zulfe Bengal, Dilshad Garden, Delhi	58	Pushta Road, Block A, Rajiv Nagar, Sonia Vihar...	1401
198463	11	Mother Teresa Crescent, Talkatora Garden, Cent...	283	NaN	957
488666	29	Street Number 14, Block C, Sitapuri Part 1, Ja...	60	NaN	3026

30 / 30

```
!ls -lrt
```

```
total 525784
-rw-r--r-- 1 root root 520141836 May 12 14:30 uber_travel_data.csv
drwxr-xr-x 1 root root      4096 Jun 29 13:44 sample_data
-rw-r--r-- 1 root root  18251707 Jul  1 13:34 Uber_dataset.zip
drwxr-xr-x 2 root root      4096 Jul  1 13:34 __MACOSX
```

```
import pandas as pd

df = pd.read_csv("./uber_travel_data.csv")
df.sample(100).head()
```

	sourceid	source	dstid	destination	travel_time
3699703	234	113, Press Colony, Press Colony, Mayapuri, New...	76	124, SPG Quarters, Sector 4, Pushp Vihar, New ...	2695
2441504	156	Doctor Satpal Sachdeva Marg, Keshav Puram, Tri...	230	N494, Block N, Raghubir Nagar, Tagore Garden E...	958
1824456	119	81, Zulfe Bengal, Dilshad Garden, Delhi	58	Pushta Road, Block A, Rajiv Nagar, Sonia Vihar...	1401
198463	11	Mother Teresa Crescent, Talkatora Garden, Cent...	283	NaN	957
488666	29	Street Number 14, Block C, Sitapuri Part 1, Ja...	60	NaN	3026

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:31

Remdesivir for the Treatment c | Remedial Session1.ipynb - Col | New Tab

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=2iCNvRMC1SDa

+ Code + Text Reconnect

1824456	119	81, Zulfe Bengal, Dilshad Garden, Delhi	58	Pushta Road, Block A, Rajiv Nagar, Sonia Vihar...	1401
198463	11	Mother Teresa Crescent, Talkatora Garden, Cent...	283	NaN	957
488666	29	Street Number 14, Block C, Sitapuri Part 1, Ja...	60	NaN	3026

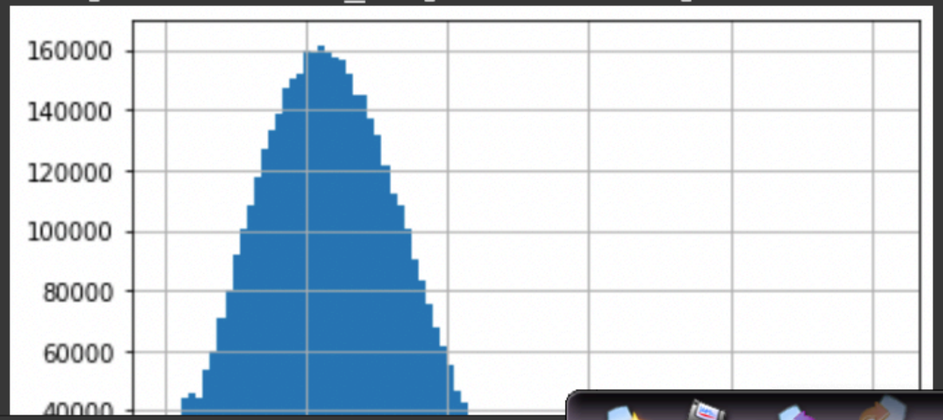
```
df.shape
```

```
(4542026, 5)
```

srlo, destlo : time

```
# histogram of travel_times
df["travel_time"].hist(bins = 100)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fde4ea25dd0>
```



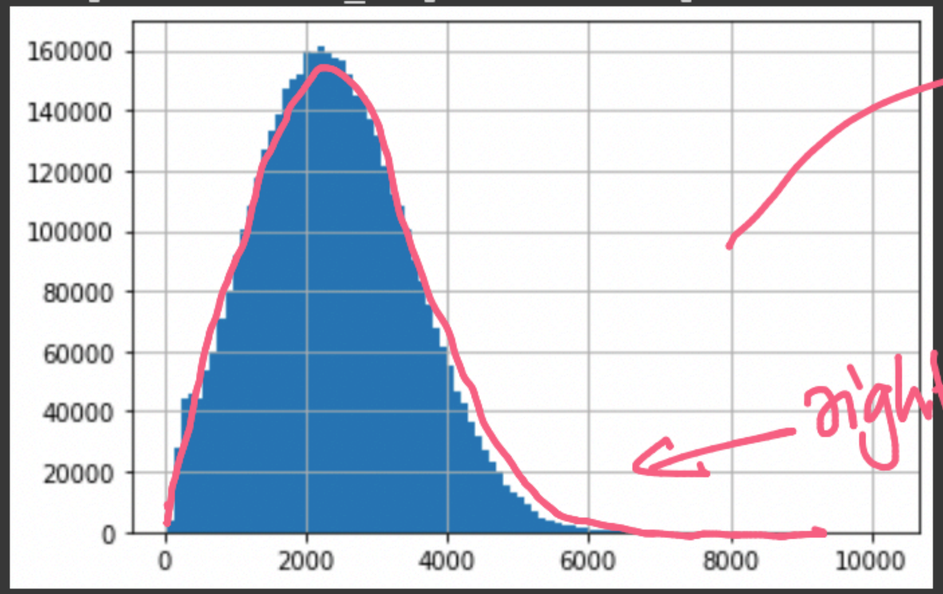
32 / 32

```
[ ] df.shape
```

```
(4542026, 5)
```

```
[ ] # histogram of travel_times
df["travel_time"].hist(bins = 100)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fde4ea25dd0>
```



right skewed

log-normal

```
[ ] df.value_counts(['sourceid', 'dstid']).sort_values()
```

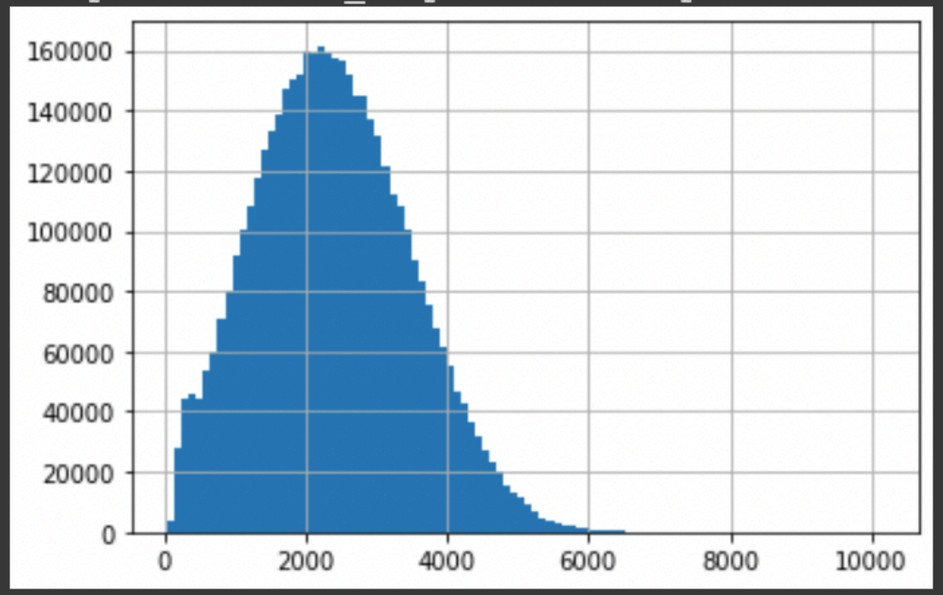
```
sourceid  dstid
```

```
[ ] df.shape
```

```
(4542026, 5)
```

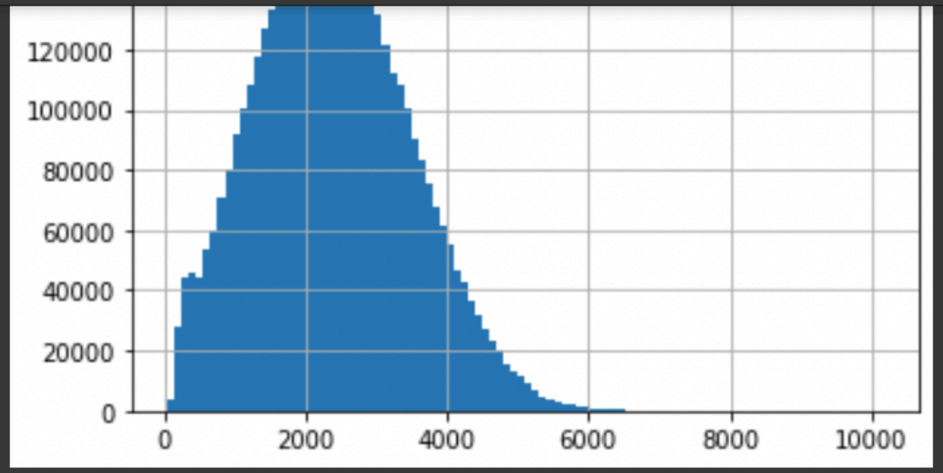
```
[ ] # histogram of travel_times
    df["travel_time"].hist(bins = 100)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fde4ea25dd0>
```



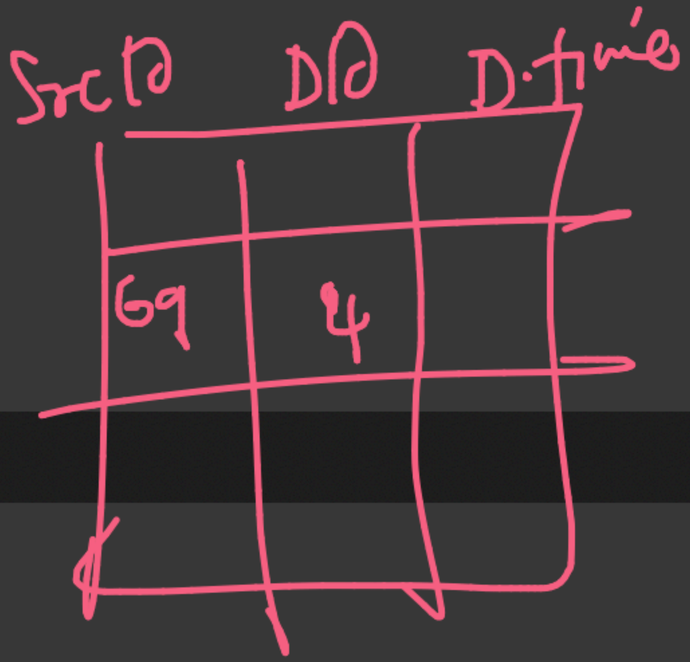

```
[ ] df.value_counts(['sourceid', 'dstid']).sort_values()
```

[ ]

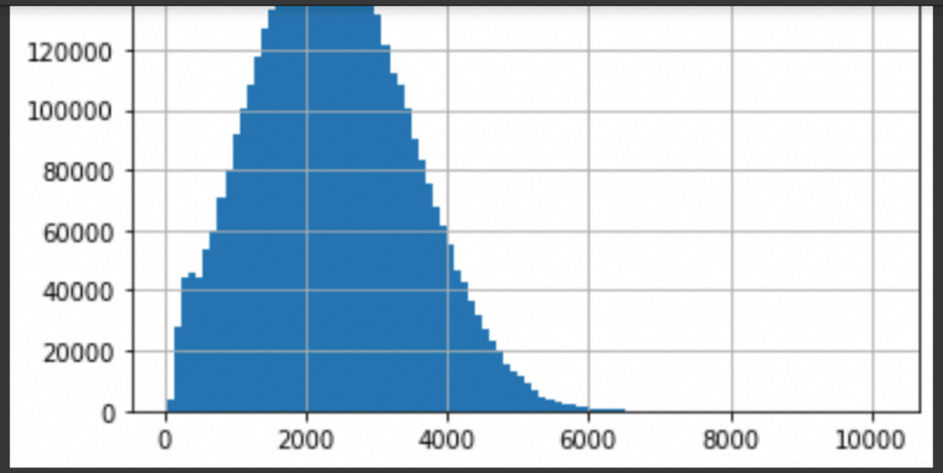


[ ] `df.value_counts(['sourceid', 'dstid']).sort_values()`

```
sourceid  dstid
69        4        50
167       107      50
          101      50
264       14       50
167       100      50
                   ..
83        88       79
244       32       79
202       201      79
          135      79
45        170      79
Length: 70429, dtype: int64
```



[ ]



[ ]
```
df.value_counts(['sourceid', 'dstid']).sort_values()
```

```
sourceid  dstid
69        4        50
167       107      50
          101      50
264       14       50
167       100      50
                   ..
83        88       79
244       32       79
202       201      79
          135      79
45        170      79
Length: 70429, dtype: int64
```

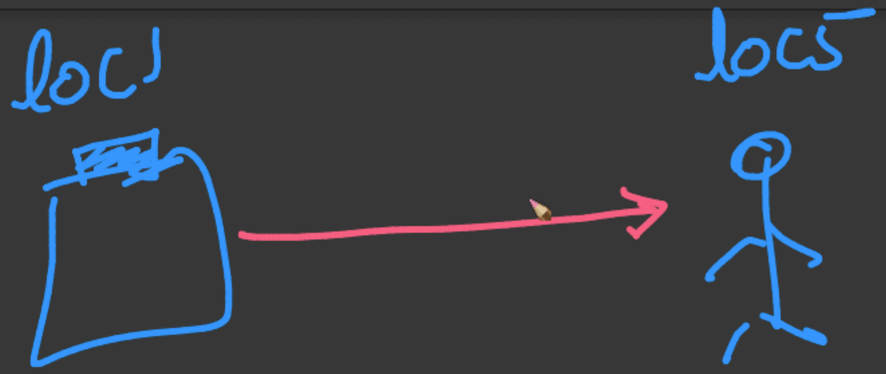
Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:36

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=2iCNvRMC1SDa

+ Code + Text Reconnect

```
69          4          50
167         107        50
            101        50
264         14         50
167         100        50
                       ..
83          88         79
244         32         79
202         201        79
            135        79
45          170        79
Length: 70429, dtype: int64
```




```
data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)]["travel_time"]
data.shape
```

```
(75,)
```

```
data.hist(bins=30)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fde4e9b9f50>
```

37 / 37

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:38

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=rhun6Z8uLhrm

+ Code + Text Reconnect

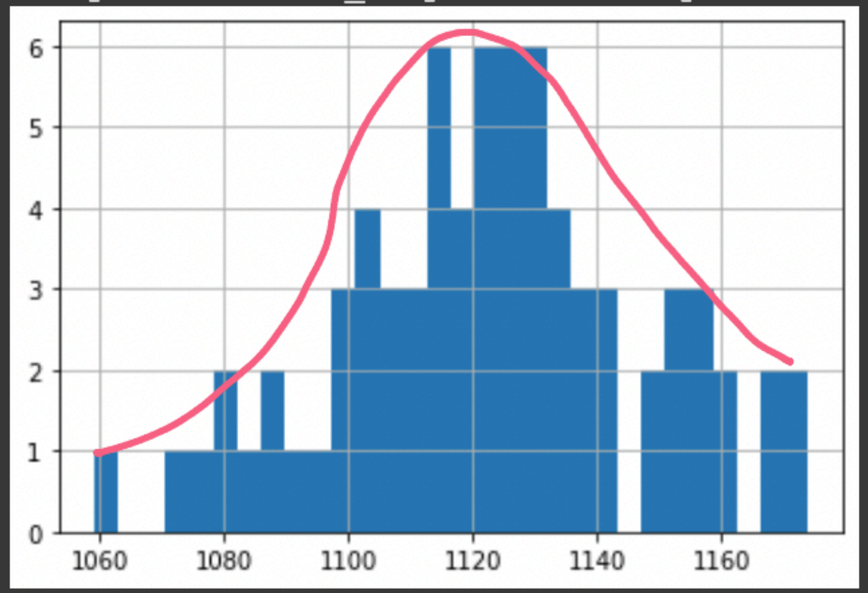
```
[ ]  data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
     data.shape
```

(75,)

(75,)

```
data.hist(bins=30)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fde4e9b9f50>
```



CLT for C.I on mean of trave

38 / 38

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:38

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=rhun6Z8uLhrm

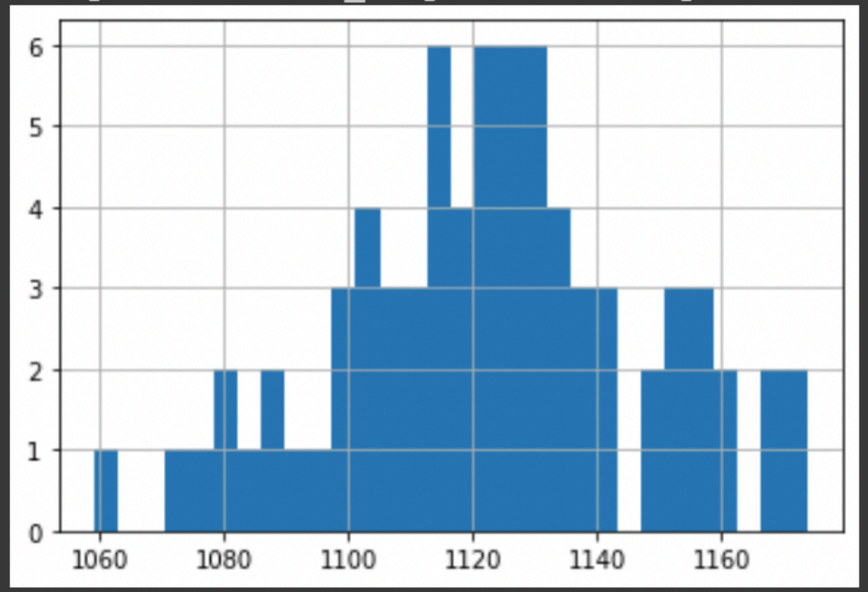
+ Code + Text Reconnect

```
[ ]  data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
     data.shape
```

(75,)

```
data.hist(bins=30)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fde4e9b9f50>
```



CLT for C.I on mean of trave

39 / 39

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:39

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=rhun6Z8uLhrm

+ Code + Text Reconnect

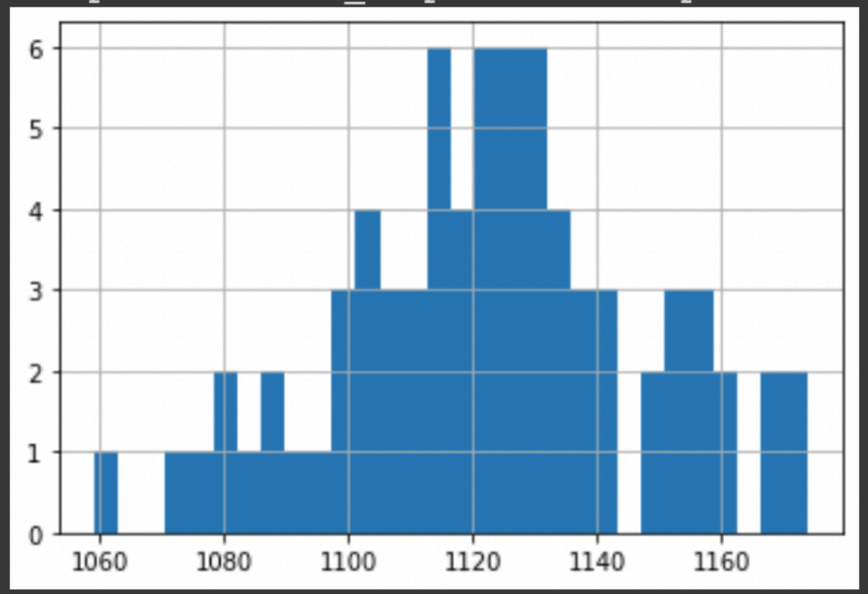
```
Length: 70429, dtype: int64
```

```
[ ]  data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
     data.shape
```

(75,) → bootstrapping with replacement

```
data.hist(bins=30)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fde4e9b9f50>
```



asy. CI of Mean DT from 1 to 5 → CLT

40 / 40

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:40

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

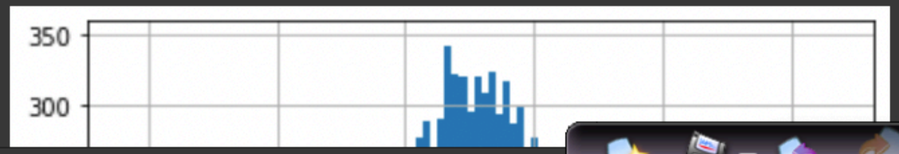
colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=-XRxzwtX8h5s

+ Code + Text Reconnect

CLT for C.I on mean of travel_time

```
# Let's create r=10000 bootstrap samples, and let each bootstrap sample be of size=50
# bs_means is a list of 'r' bootstrap sample means
r = 10000
data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
size = 50
bs_means = np.empty(r)

for i in range(r):
    bs_sample = np.random.choice(data, size=size)
    bs_means[i] = np.mean(bs_sample)
```

```
import matplotlib.pyplot as plt
plt.figure()
plt.hist(bs_means, bins=100)
plt.grid()
plt.show()
```



41 / 41

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:40

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

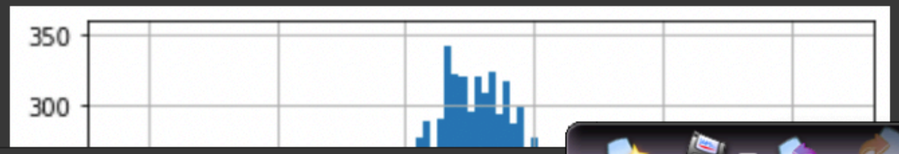
colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=-XRxzwtX8h5s

+ Code + Text Reconnect

CLT for C.I on mean of travel_time

```
# Let's create r=10000 bootstrap samples, and let each bootstrap sample be of size=50
# bs_means is a list of 'r' bootstrap sample means
r = 10000
data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
size = 50
bs_means = np.empty(r)

for i in range(r):
    bs_sample = np.random.choice(data, size=size)
    bs_means[i] = np.mean(bs_sample)
```

```
import matplotlib.pyplot as plt
plt.figure()
plt.hist(bs_means, bins=100)
plt.grid()
plt.show()
```



42 / 42

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:41

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=-XRxzwtX8h5s

+ Code + Text Reconnect

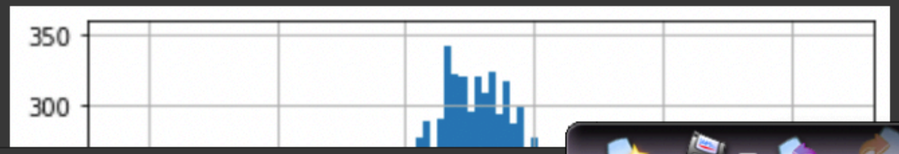
CLT for C.I on mean of travel_time

75 obs

```
# Let's create r=10000 bootstrap samples, and let each bootstrap sample be of size=50
# bs_means is a list of 'r' bootstrap sample means
r = 10000
data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
size = 50
bs_means = np.empty(r)

for i in range(r):
    bs_sample = np.random.choice(data, size=size)
    bs_means[i] = np.mean(bs_sample)
```

n=

```
import matplotlib.pyplot as plt
plt.figure()
plt.hist(bs_means, bins=100)
plt.grid()
plt.show()
```



43 / 43

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:41

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=-XRxzwtX8h5s

+ Code + Text Reconnect

CLT for C.I on mean of travel_time

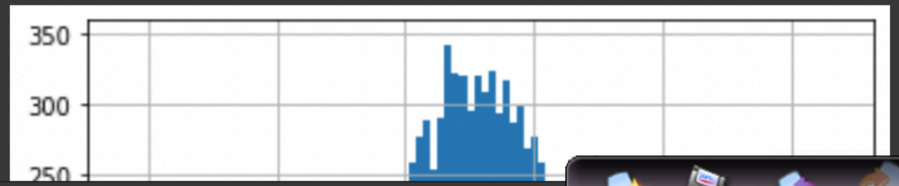
```
# Let's create r=10000 bootstrap samples, and let each bootstrap sample be of size=50
# bs_means is a list of 'r' bootstrap sample means
r = 10000
data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
size = 50
bs_means = np.empty(r)

for i in range(r):
    bs_sample = np.random.choice(data, size=size)
    bs_means[i] = np.mean(bs_sample)
```

r

n

75 obs

```
[ ] import matplotlib.pyplot as plt
    plt.figure()
    plt.hist(bs_means, bins=100)
    plt.grid()
    plt.show()
```



44 / 44

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:43

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

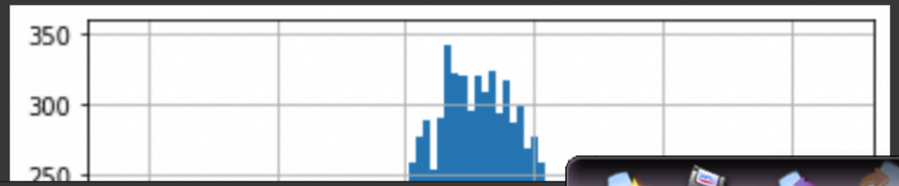
colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=-XRxzwtX8h5s

+ Code + Text Reconnect

CLT for C.I on mean of travel_time

```
# Let's create r=10000 bootstrap samples, and let each bootstrap sample be of size=50
# bs_means is a list of 'r' bootstrap sample means
r = 10000
data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
size = 50
bs_means = np.empty(r)

for i in range(r):
    bs_sample = np.random.choice(data, size=size)
    bs_means[i] = np.mean(bs_sample)
```

```
[ ] import matplotlib.pyplot as plt
    plt.figure()
    plt.hist(bs_means, bins=100)
    plt.grid()
    plt.show()
```



45 / 45

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:43

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

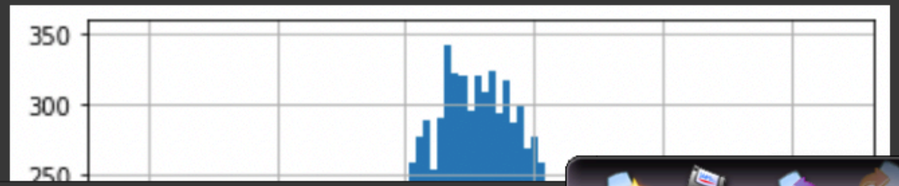
colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=-XRxzwtX8h5s

+ Code + Text Reconnect

CLT for C.I on mean of travel_time

```
# Let's create r=10000 bootstrap samples, and let each bootstrap sample be of size=50
# bs_means is a list of 'r' bootstrap sample means
r = 10000
data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
size = 50
bs_means = np.empty(r)

for i in range(r):
    bs_sample = np.random.choice(data, size=size)
    bs_means[i] = np.mean(bs_sample)
```

```
[ ] import matplotlib.pyplot as plt
    plt.figure()
    plt.hist(bs_means, bins=100)
    plt.grid()
    plt.show()
```

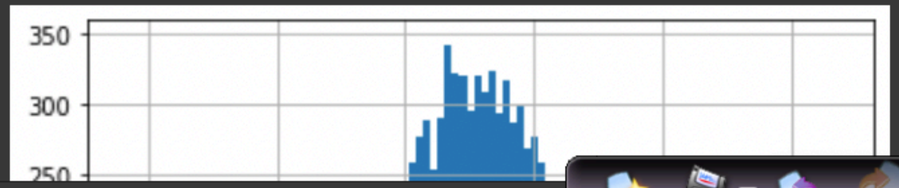


46 / 46

CLT for C.I on mean of travel_time

```
# Let's create r=10000 bootstrap samples, and let each bootstrap sample be of size=50
# bs_means is a list of 'r' bootstrap sample means
r = 10000
data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
size = 50
bs_means = np.empty(r)

for i in range(r):
    bs_sample = np.random.choice(data, size=size)
    bs_means[i] = np.mean(bs_sample)
```

```
[ ] import matplotlib.pyplot as plt
    plt.figure()
    plt.hist(bs_means, bins=100)
    plt.grid()
    plt.show()
```



Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:44

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

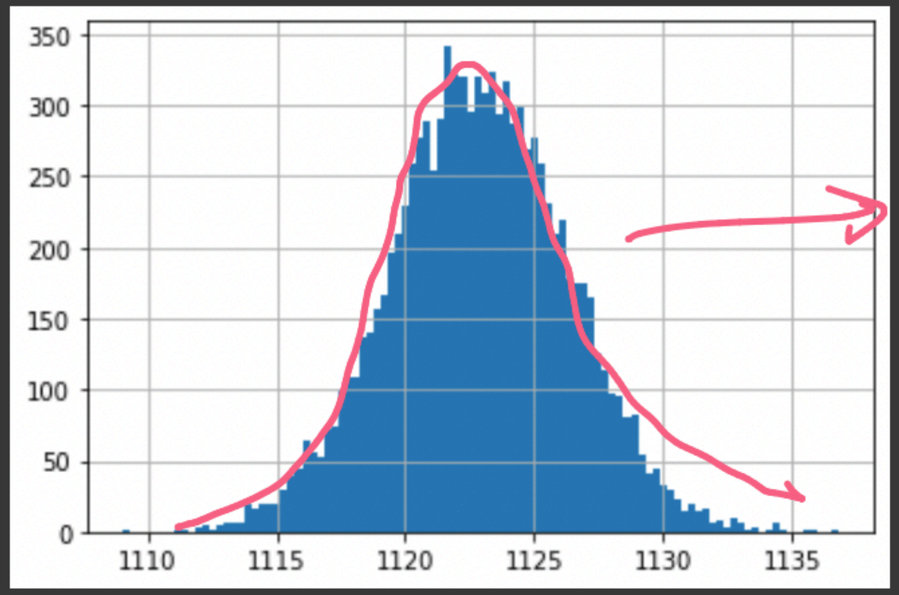
colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=-XRxzwtX8h5s

+ Code + Text Reconnect

```
    bs_means[i] = np.mean(bs_sample)
```

```
[ ] import matplotlib.pyplot as plt
    plt.figure()
    plt.hist(bs_means, bins=100)
    plt.grid()
    plt.show()
```

$\bar{X}_1, \bar{X}_2 \cdots \bar{X}_r$

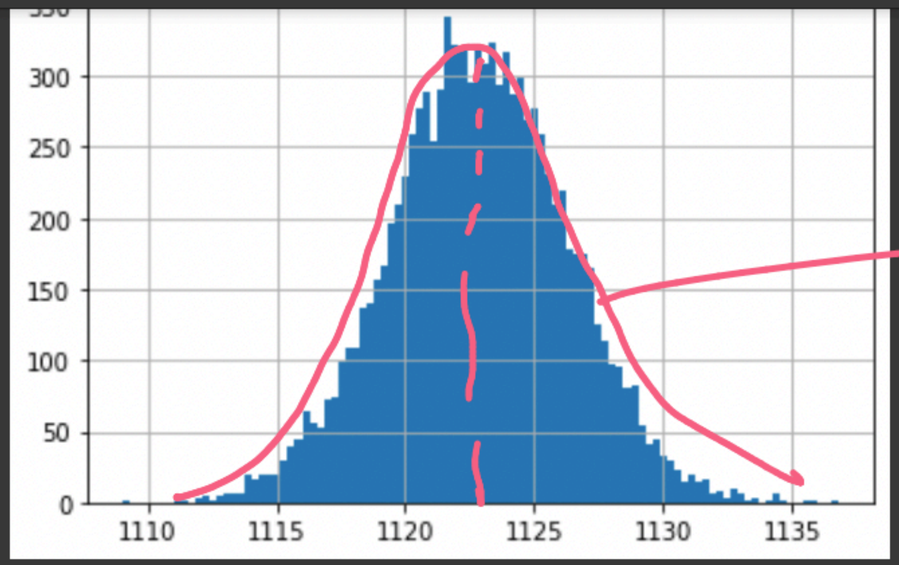


Gaussian → QQ plot, KS test (later)

```
[ ] # QQ-plot with normal distribution
```

48 / 48

+ Code + Text Reconnect



distbn of bs-means

$\bar{X}_1, \bar{X}_2 \cdots \bar{X}_n$

$N\left(\mu, \frac{\sigma}{\sqrt{n}}\right)$ as $n \to \infty$

```
[ ] # QQ-plot with normal distribution
```

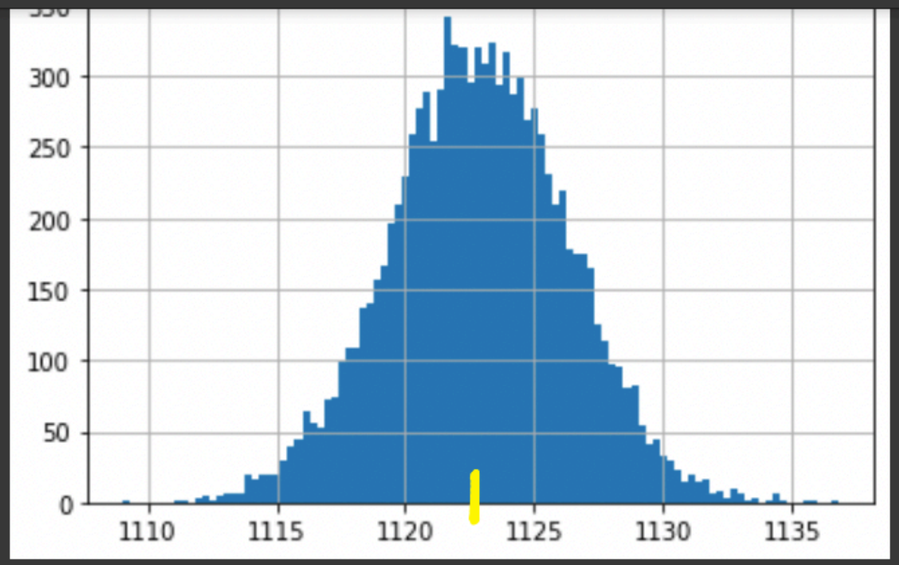
```
[ ] # compute C.I on the mean given that bs_means follows Gaussian distribution: CLT
    print(np.mean(bs_means))
    print(np.std(bs_means))
```

```
1122.85326
3.4374772628193493
```

SEM

```
[ ] print(np.mean(bs_means)-2*np
```

+ Code + Text Reconnect



95% C.I on Mean. DT

[M - 1.96 S , M + 1.96 S]

```
# QQ-plot with normal distribution
```

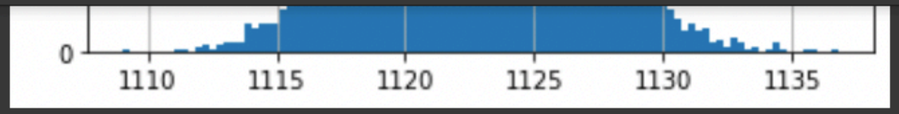
```
# compute C.I on the mean given that bs_means follows Gaussian distribution: CLT
print(np.mean(bs_means))
print(np.std(bs_means))
```

M

S

```
1122.85326
3.4374772628193493
```

```
print(np.mean(bs_means)-2*np
```




```
# QQ-plot with normal distribution
```

```
# compute C.I on the mean given that bs_means follows Gaussian distribution: CLT
print(np.mean(bs_means))
print(np.std(bs_means))
```

```
1122.85326
3.4374772628193493
```

```
print(np.mean(bs_means)-2*np.std(bs_means))
print(np.mean(bs_means)+2*np.std(bs_means))
```

```
1115.9783054743614
1129.7282145256388
```

```
# could we just use the 2.5th percentile and 97.5th percentile value
print(np.percentile(bs_means,2.5))
print(np.percentile(bs_means,97.5))
```

loc1 loc5

[,]

51 / 51

Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:49

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=-XRxzwtX8h5s

+ Code + Text Reconnect

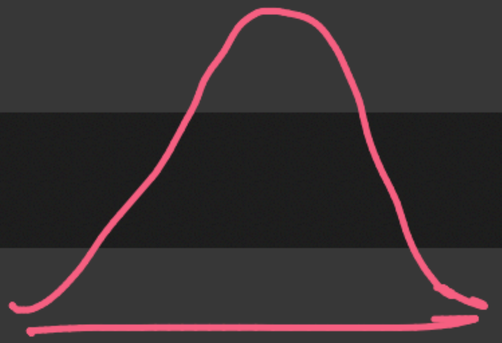
```
[ ] print(np.mean(bs_means))
    print(np.std(bs_means))
```

```
1122.85326
3.4374772628193493
```

CLT ✓

```
[ ] print(np.mean(bs_means)-2*np.std(bs_means))
    print(np.mean(bs_means)+2*np.std(bs_means))
```

```
1115.9783054743614
1129.7282145256388
```



```
[ ] # could we just use the 2.5th percentile and 97.5th percentile value
    print(np.percentile(bs_means,2.5))
    print(np.percentile(bs_means,97.5))

    # what if r is say 100 and not 10,000?
```

```
1116.02
1129.52
```

95% C.I on 99th percentile value for travel_time via bootsrapping

52 / 52

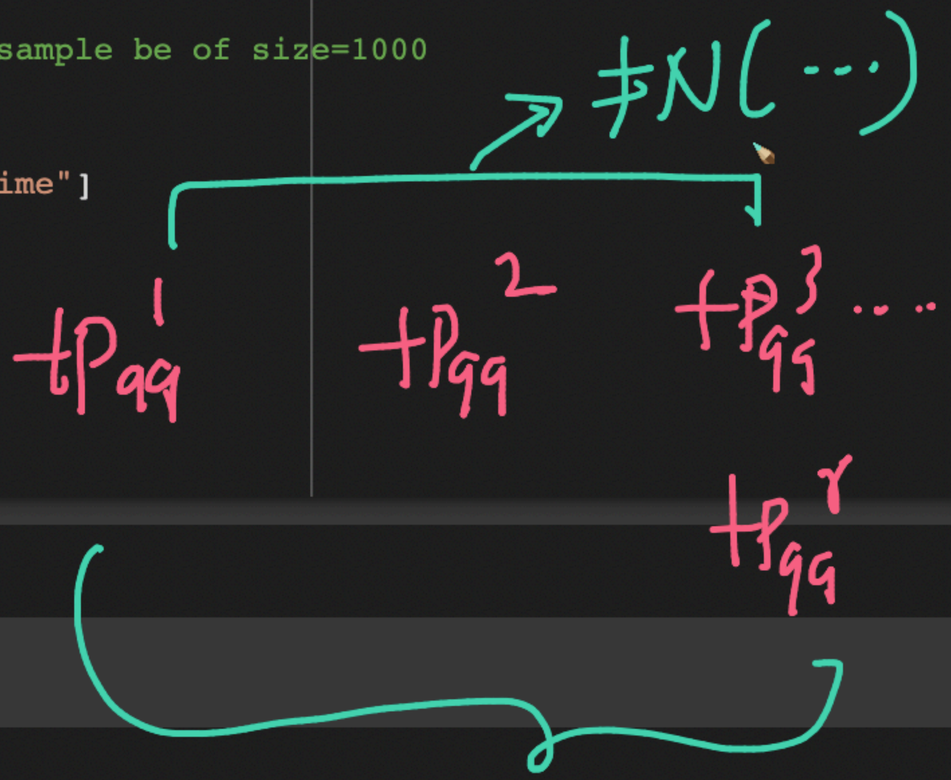
Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:55

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab | numpy.std — NumPy v1.23 Ma

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=tmxw002-9n-A

+ Code + Text Reconnect

```
# What if we want a C.I on the 99th precentile?
#Let's create r=10000 bootstrap samples, and let each bootstrap sample be of size=1000
# bs_99p is a list of 'r' bootstrap sample's 99th percentiles
r = 10000
data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
size = 75
bs_99p = np.empty(r)

for i in range(r):
    bs_sample = np.random.choice(data, size=size)
    bs_99p[i] = np.percentile(bs_sample,99)
```




```
[ ] len(bs_99p)
```

```
10000
```

```
[ ] bs_99p
```

```
array([1167., 1167., 1174., ..., 1174., 1174., 1174.])
```

```
[ ] #bs_99p may or maynot be normally distributed.
    print(np.percentile(bs_99p,2.5))
    print(np.percentile(bs_99p,9
```

53 / 53

```
[ ]     bs_99p[i] = np.percentile(bs_sample,99)
```

```
[ ] len(bs_99p)
```

```
10000
```

```
[ ] bs_99p
```

```
array([1167., 1167., 1174., ..., 1174., 1174., 1174.])
```

```
#bs_99p may or maynot be normally distributed.
print(np.percentile(bs_99p,2.5))
print(np.percentile(bs_99p,97.5))
```

```
1162.56
1174.0
```

95% C.I for the t_{99} rel. time from loc 1 to loc 5

```
[ ] # Point estimate of the 99th percenitle of the 75 observed samples
print(np.percentile(data,99))
```

```
1174.0
```

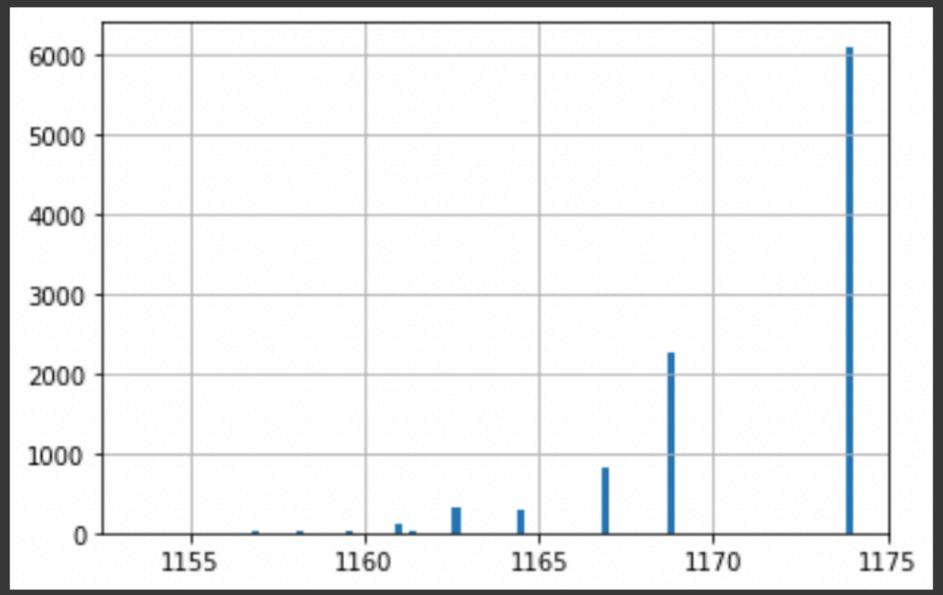
Chrome File Edit View History Bookmarks Profiles Tab Window Help

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab | numpy.std — NumPy v1.23 Ma

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=zyYEvchf-C6q

+ Code + Text Reconnect

```
[ ]  # plot the pdf of bs_99p
     import matplotlib.pyplot as plt
     plt.figure()
     plt.hist(bs_99p, bins=100)
     plt.grid()
     plt.show()
```



CLT as 'n' and 'r' changes

55 / 55

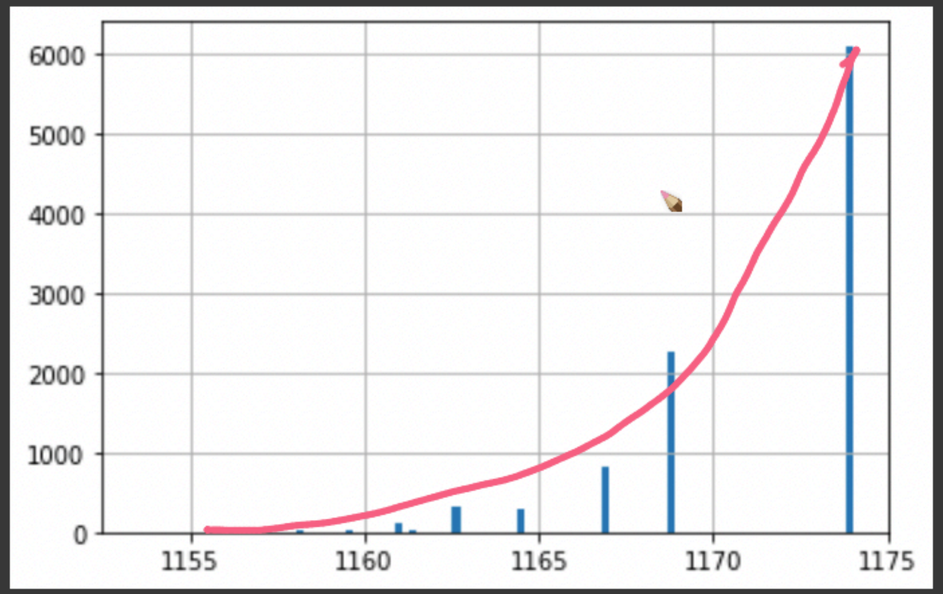
Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:57

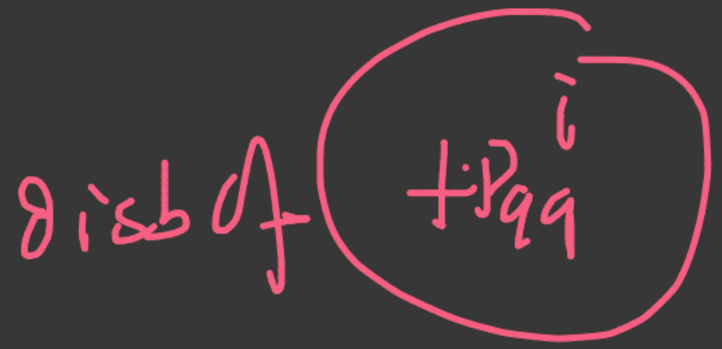
Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab | numpy.std — NumPy v1.23 Ma

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=JIlQOyNe6VnF

+ Code + Text Reconnect

```
plt.show()
```





CLT as 'n' and 'r' changes

```
[ ] data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
    data.shape
```

```
(75,)
```

56 / 56

```
1116.02
1129.52
```

95% C.I on 99th percentile value for travel_time via bootsrapping

asy. C.I of the median del.time

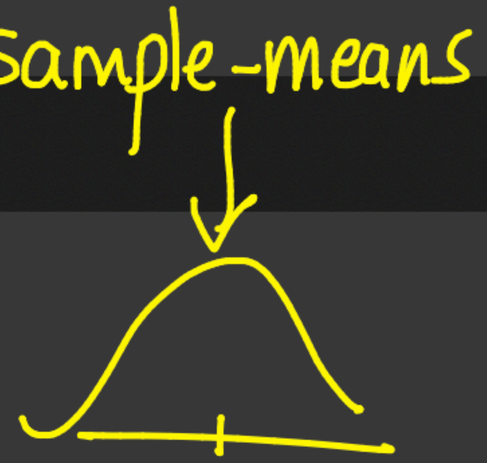
```
# What if we want a C.I on the 99th precentile?
#Let's create r=10000 bootstrap samples, and let each bootstrap sample be of size=1000
# bs_99p is a list of 'r' bootstrap sample's 99th percentiles
r = 10000
data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
size = 75
bs_99p = np.empty(r)

for i in range(r):
    bs_sample = np.random.choice(data, size=size)
    bs_99p[i] = np.percentile(bs_sample, 99)
```

```
len(bs_99p)
```

```
10000
```

```
bs_99p
```

CLT as 'n' and 'r' changes




```
[ ] data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
    data.shape
```

```
(75,)
```

Change "r"

```
[ ] # n=30, r=10000
    # Let's create r=10000 bootstrap samples, and let each bootstrap sample be of size=30
    # bs_means is a list of 'r' bootstrap sample means
    r = 10000
    data = df[(df["sourceid"] == 1) & (df["dstid"] == 5)] ["travel_time"]
    size = 30
    bs_means = np.empty(r)

    for i in range(r):
        bs_sample = np.random.choice(data, size=size)
        bs_means[i] = np.mean(bs_sample)
```

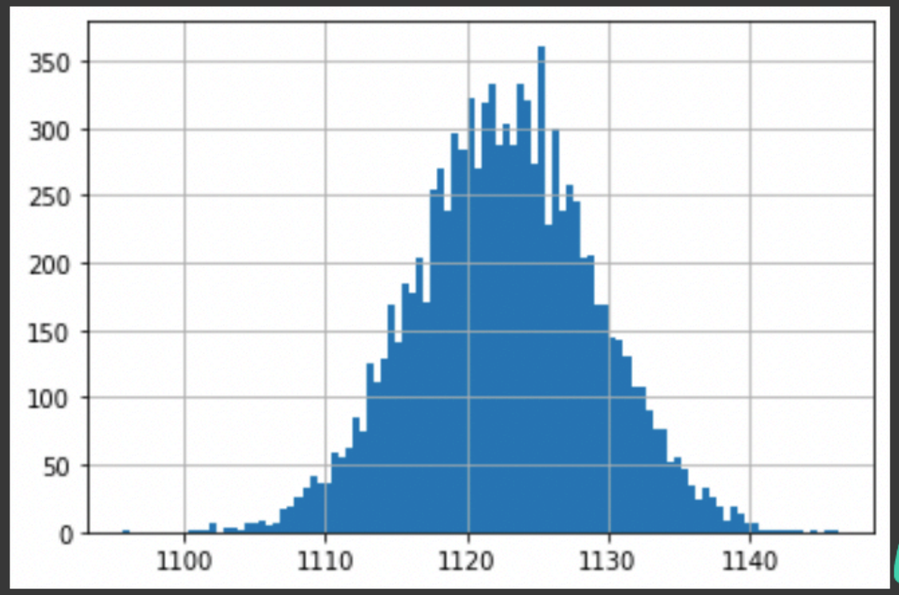
Chrome File Edit View History Bookmarks Profiles Tab Window Help 11:08

Remdesivir for the Treatment | Remedial Session1.ipynb - Col | New Tab | numpy.std — NumPy v1.23 Ma

colab.research.google.com/drive/14XXagrZxnpzYGe_dHUZxE0xqRCngh9F1#scrollTo=D5_0dcfW9rHv

+ Code + Text Reconnect

```
[ ] print(np.mean(bs_means), np.std(bs_means))
```



1122.7105866666666 6.337015344627504

#rep
r

Sample-size
n

SEM
std-dev

mean

r	n	mean	SEM (std-dev)
10K	30	1122.73	4.42
100	30	1122.39	3.98
1000	30	1122.58	4.34

r	n	mean	SEM (std-dev)
10K	30	1122.73	4.42
10K	60	1122.82	3.21
10K	154	1122.71	6.33

$\frac{\sigma}{\sqrt{n}}$

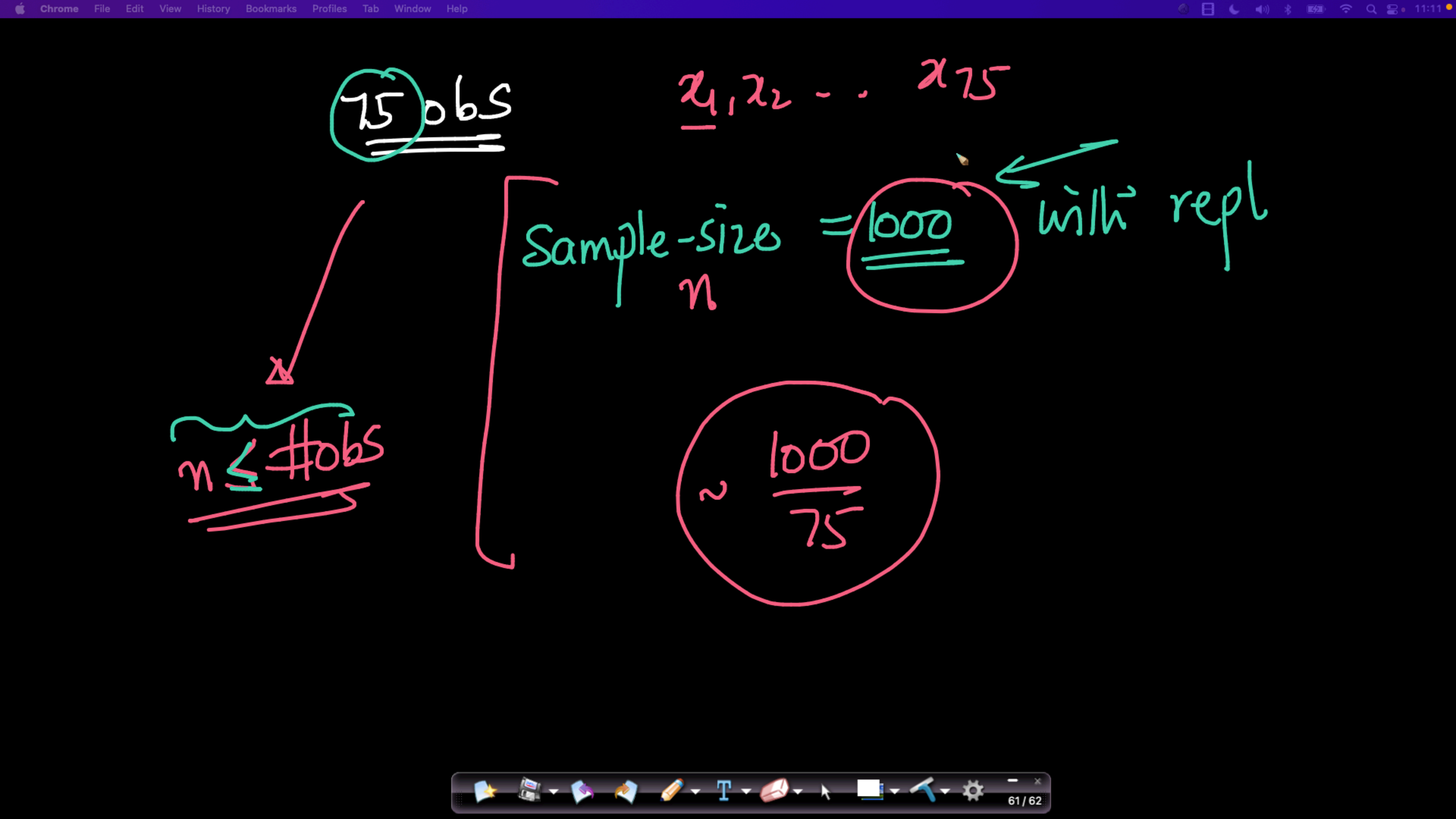
59 / 60

CLT

sample-size

as $n \to \infty$

$$\bar{x} \sim N\left(\mu, \frac{\sigma}{\sqrt{n}}\right)$$


75 obs
$x_1, x_2 \ldots x_{75}$
Sample-size
n
$= 1000$
with repl
$n \leq$ #obs
$\sim \frac{1000}{75}$

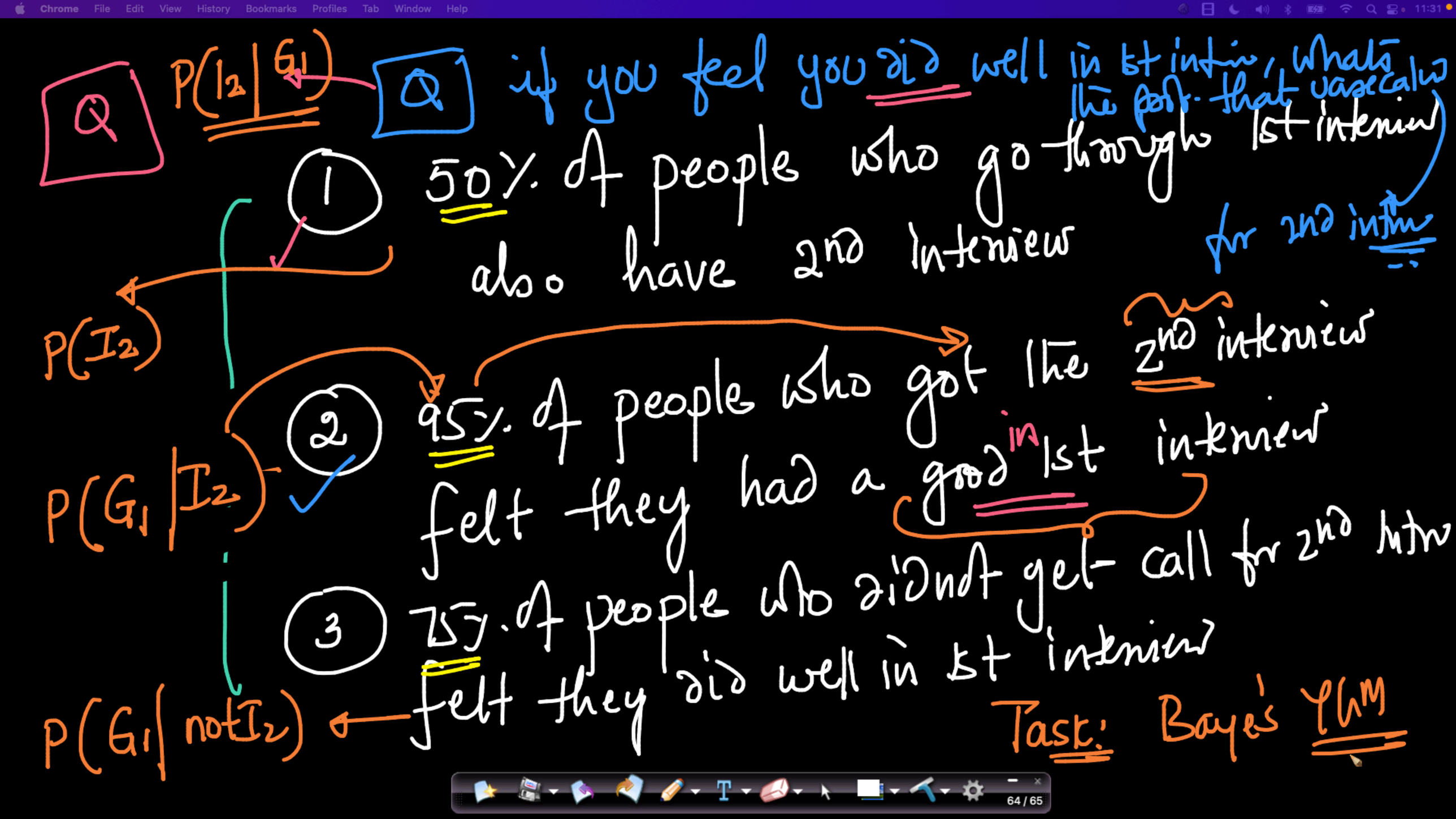
n = sample size $\longrightarrow$ as large as possible

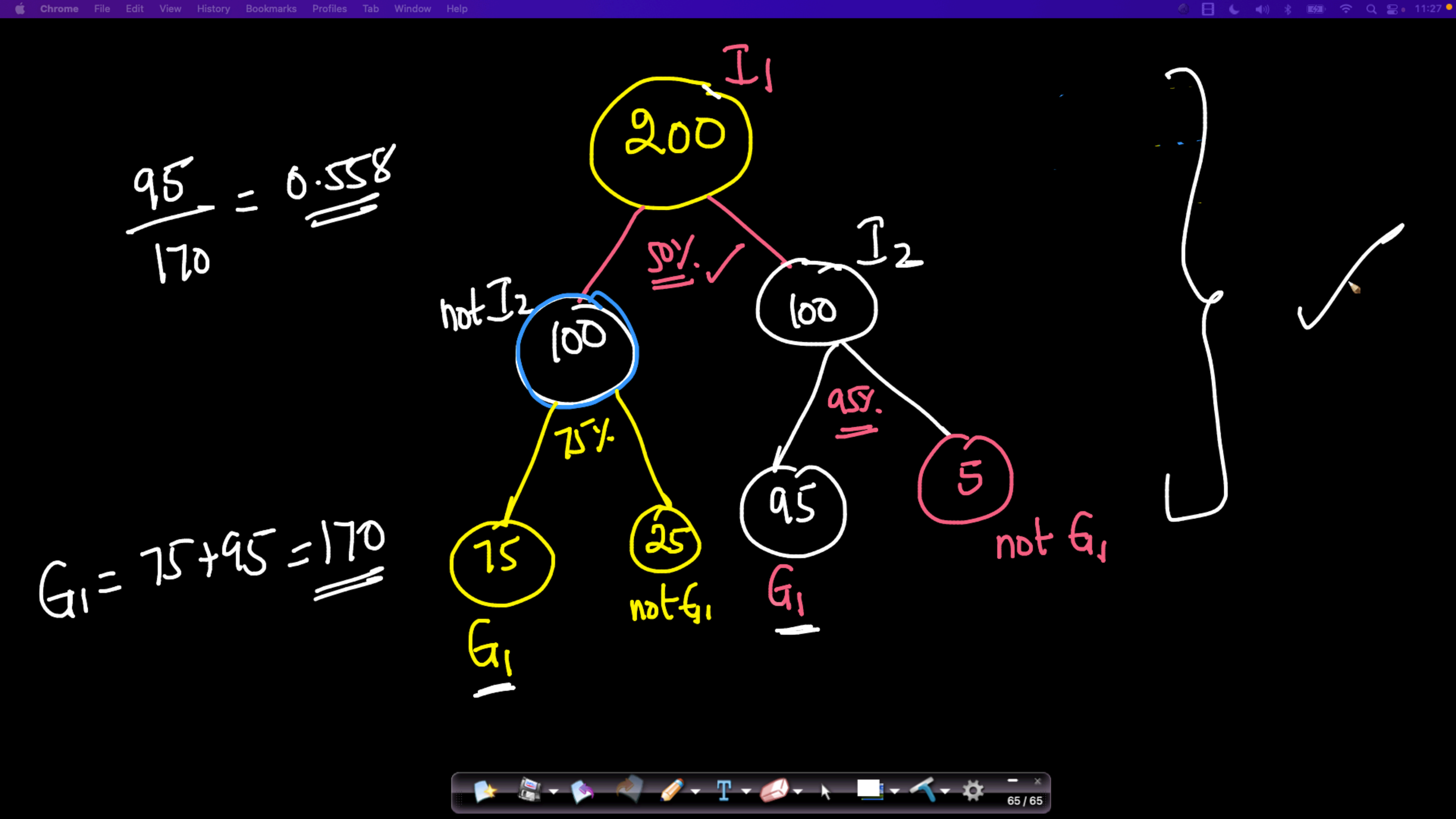
bootstapping $r \Rightarrow$ comp. time

$n = 75$ without replacement

$\sigma_n^1 = \sigma_n^2 = \sigma_3^2 = \cdots$

$\{ \bar{X}_1 = \bar{X}_2 = \bar{X}_3 \cdots$


Q
P(I2 | G1)
Q
if you feel you did well in 1st intw, what's the prob. that you'll call for 1st interview
1
50% of people who go through 1st interview also have 2nd interview
for 2nd intw
P(I2)
2
95% of people who got the 2nd interview felt they had a good in 1st interview
P(G1 | I2)
3
25% of people who didn't get call for 2nd intw felt they did well in 1st interview
P(G1 | not I2)
Task: Bayes Thm
64 / 65


$\frac{95}{170} = 0.558$
I_1
200
50%
not I_2
100
I_2
100
75%
95%
75
25
95
5
$G_1 = 75 + 95 = 170$
G_1
not G_1
not G_1
G_1

ORIGINAL ARTICLE

Remdesivir for the Treatment of Covid-19 — Final Report

John H. Beigel, M.D., Kay M. Tomashek, M.D., M.P.H., Lori E. Dodd, Ph.D., Aneesh K. Mehta, M.D., Barry S. Zingman, M.D., Andre C. Kalil, M.D., M.P.H., Elizabeth Hohmann, M.D., Helen Y. Chu, M.D., M.P.H., Annie Luetkemeyer, M.D., Susan Kline, M.D., M.P.H., Diego Lopez de Castilla, M.D., M.P.H., Robert W. Finberg, M.D., et al., for the ACTT-1 Study Group Members*

November 5, 2020
N Engl J Med 2020; 383:1813-1826
DOI: 10.1056/NEJMoa2007764
Chinese Translation 中文翻译

Abstract

BACKGROUND

Although several therapeutic agents have been evaluated for the treatment of coronavirus disease 2019 (Covid-19), no antiviral agents have yet been shown to be efficacious.

METHODS

We conducted a double-blind, randomized, placebo-controlled trial of intravenous remdesivir in adults who were hospitalized with Covid-19 and had evidence of lower respiratory tract infection. Patients were randomly assigned to receive either remdesivir (200 mg loading dose on day 1, followed by 100 mg daily for up to 9 additional days) or placebo for up to 10 days. The primary outcome was the time to recovery, defined by either discharge from the hospital or hospitalization for infection-control purposes only.

RESULTS

A total of 1062 patients underwent ran

Related Articles

EDITORIAL NOV 5, 2020
Remdesivir — An Important First Step
R. Dolin and M.S. Hirsch

ORIGINAL ARTICLE NOV 5, 2020
Remdesivir for 5 or 10 Days in Patients with Severe Covid-19
J.D. Goldman and Others

CORRESPONDENCE SEP 3, 2020
Remdesivir for the Treatment of Covid-19 — Preliminary Report