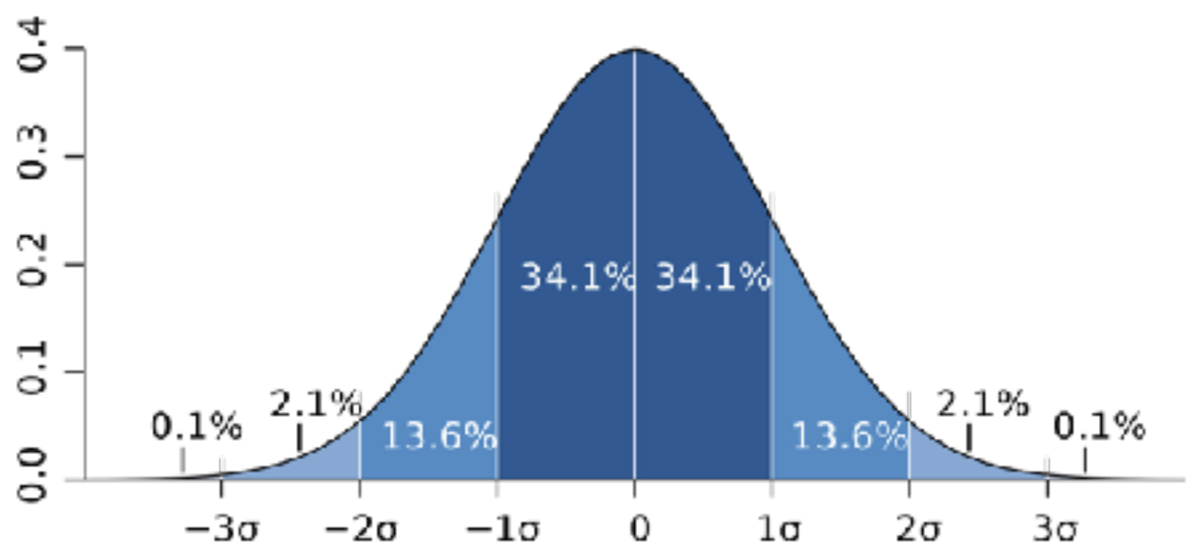
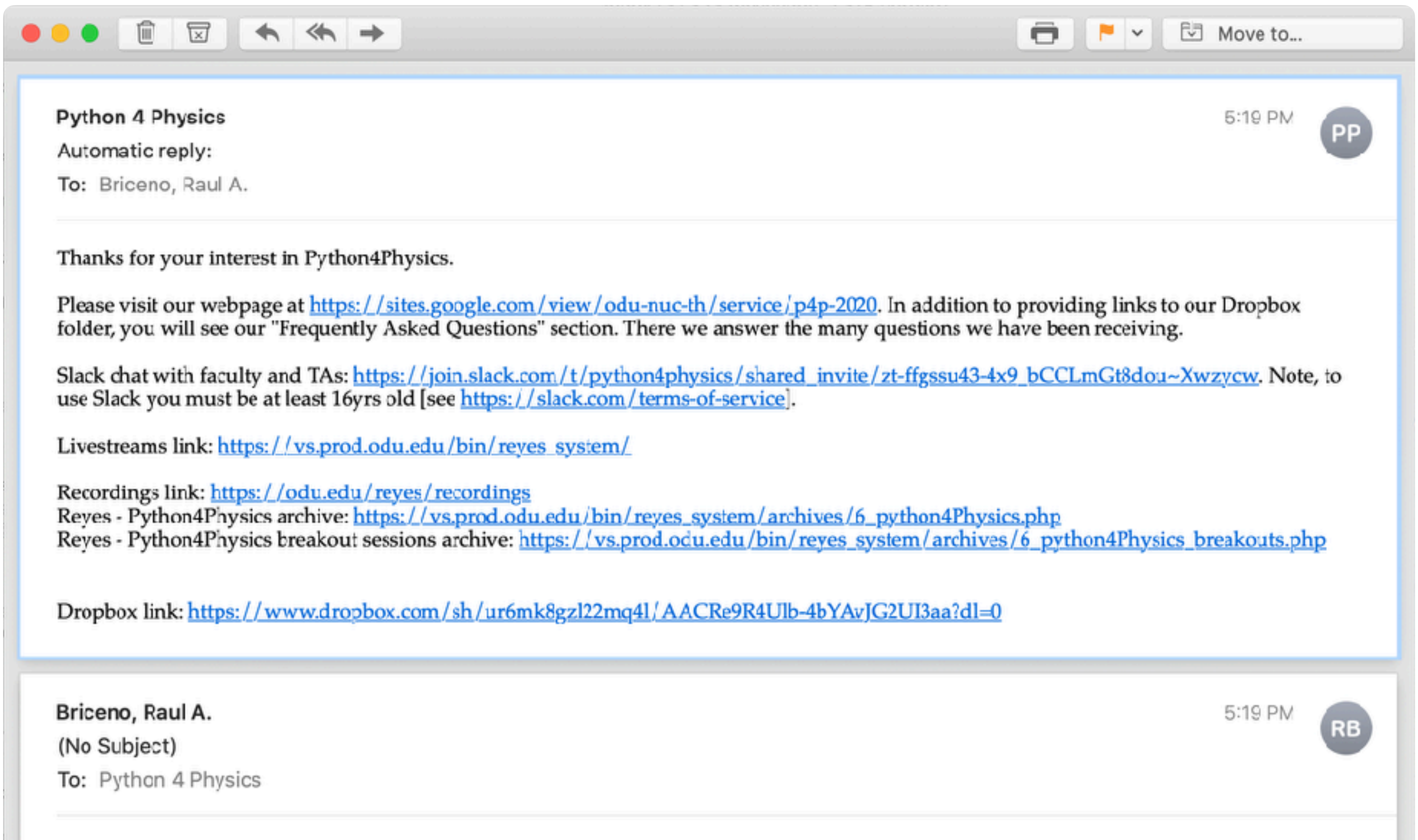


R. BRICEÑO, T. ROGERS

DATA ANALYSIS AND THE NEUTRINO MASS

ADMIN STUFF

Need links, just email python4physics@odu.edu



Move to...

Python 4 Physics 5:19 PM

Automatic reply:

To: Briceno, Raul A.

Thanks for your interest in Python4Physics.

Please visit our webpage at https://sites.google.com/view/odu-nuc-th/service/p4p-2020. In addition to providing links to our Dropbox folder, you will see our "Frequently Asked Questions" section. There we answer the many questions we have been receiving.

Slack chat with faculty and TAs: https://join.slack.com/t/python4physics/shared_invite/zt-ffgssu43-4x9_bCCLmGt8dou~Xwzycw. Note, to use Slack you must be at least 16yrs old [see https://slack.com/terms-of-service].

Livestreams link: https://vs.prod.odu.edu/bin/reyes_system/

Recordings link: https://odu.edu/reyes/recordings
Reyes - Python4Physics archive: https://vs.prod.odu.edu/bin/reyes_system/archives/6_python4Physics.php
Reyes - Python4Physics breakout sessions archive: https://vs.prod.odu.edu/bin/reyes_system/archives/6_python4Physics_breakouts.php

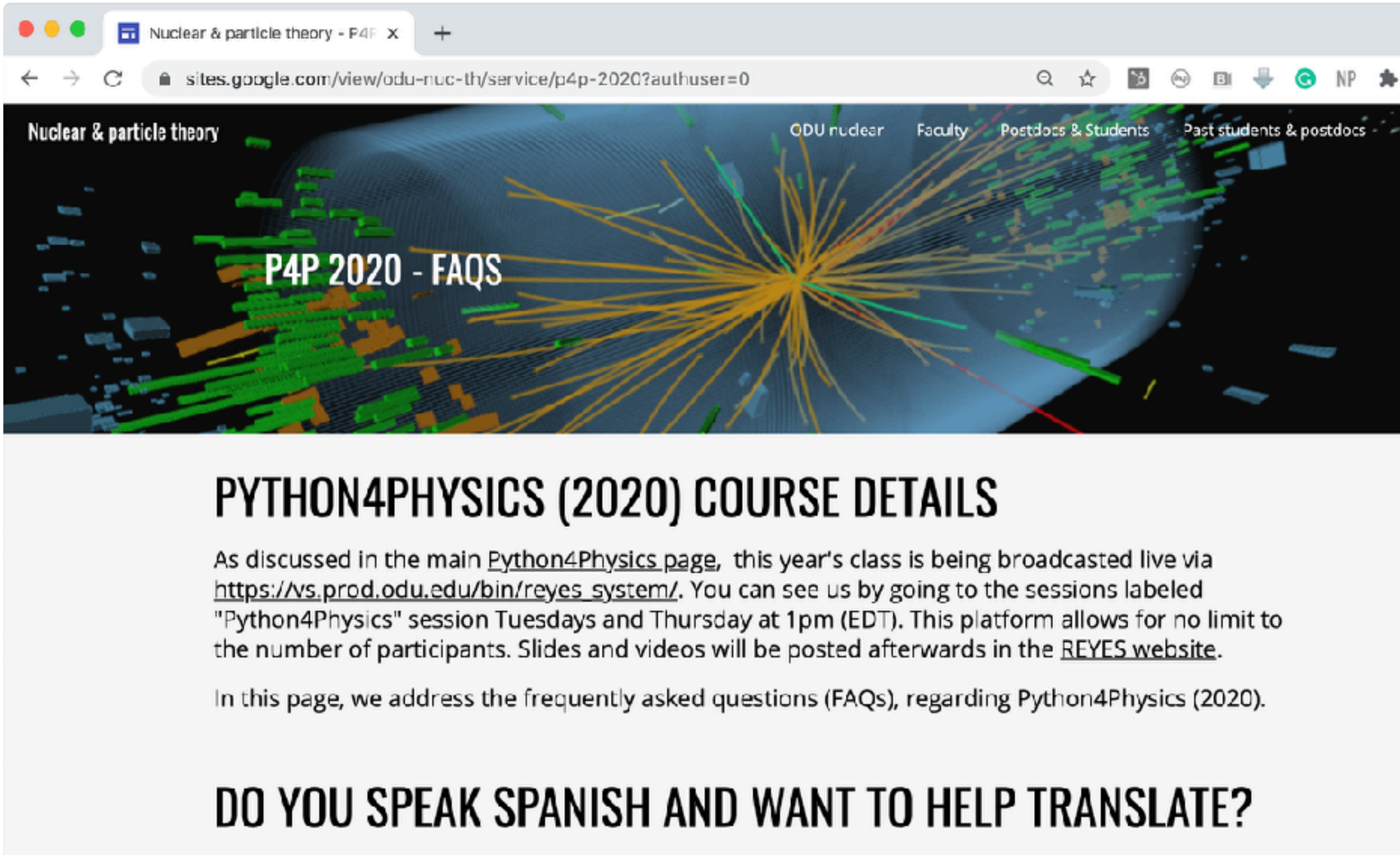
Dropbox link: https://www.dropbox.com/sh/ur6mk8gzl22mq41/AACRe9R4Ulb-4bYAvJG2UI3aa?dl=0

Briceno, Raul A. 5:19 PM

(No Subject)

To: Python 4 Physics

Need links, just email python4physics@odu.edu



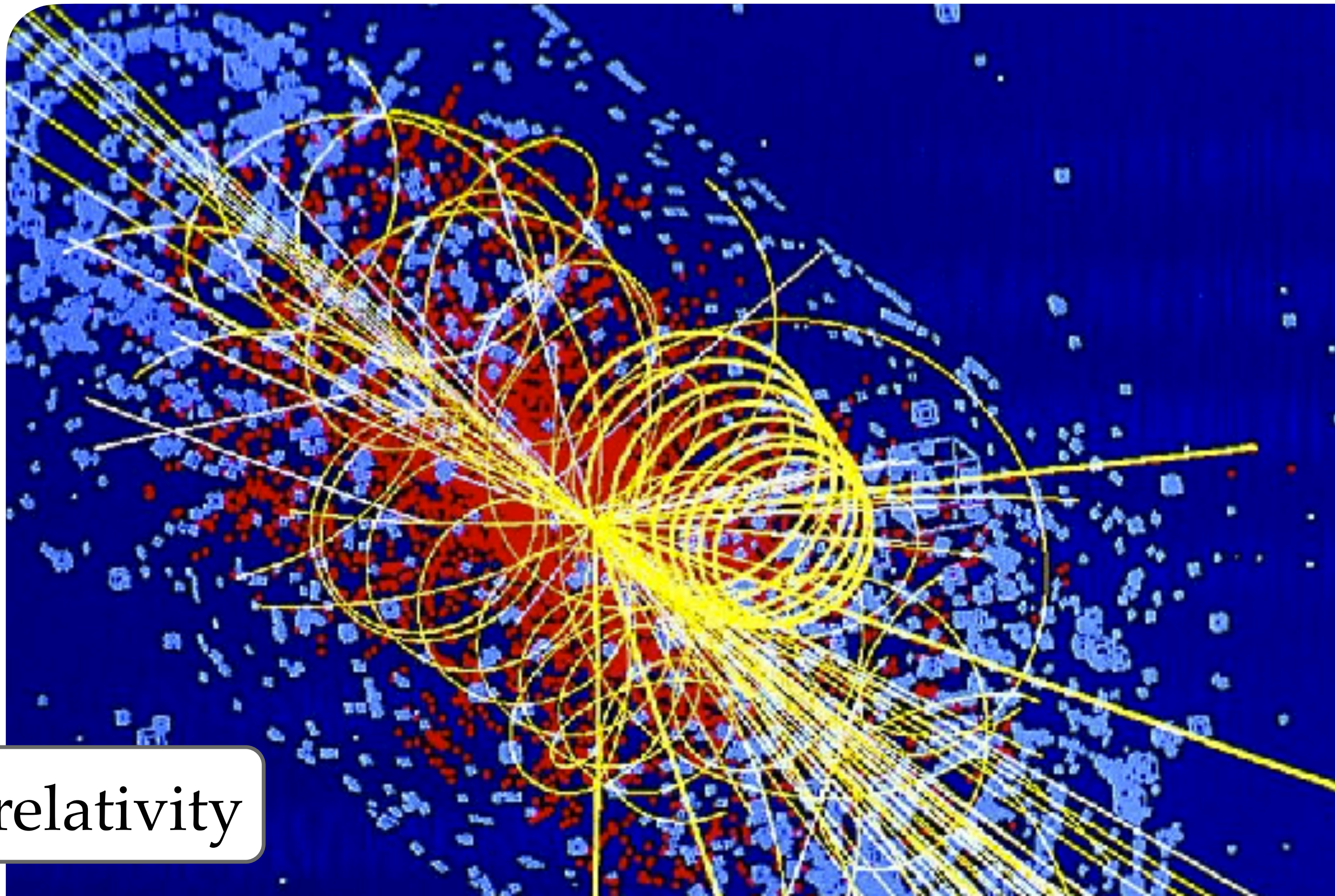
ADMIN STUFF

Need links, just email python4physics@odu.edu

Dropbox for students, email us! Subject: “Dropbox access”

OUTLINE

Intro to data analysis



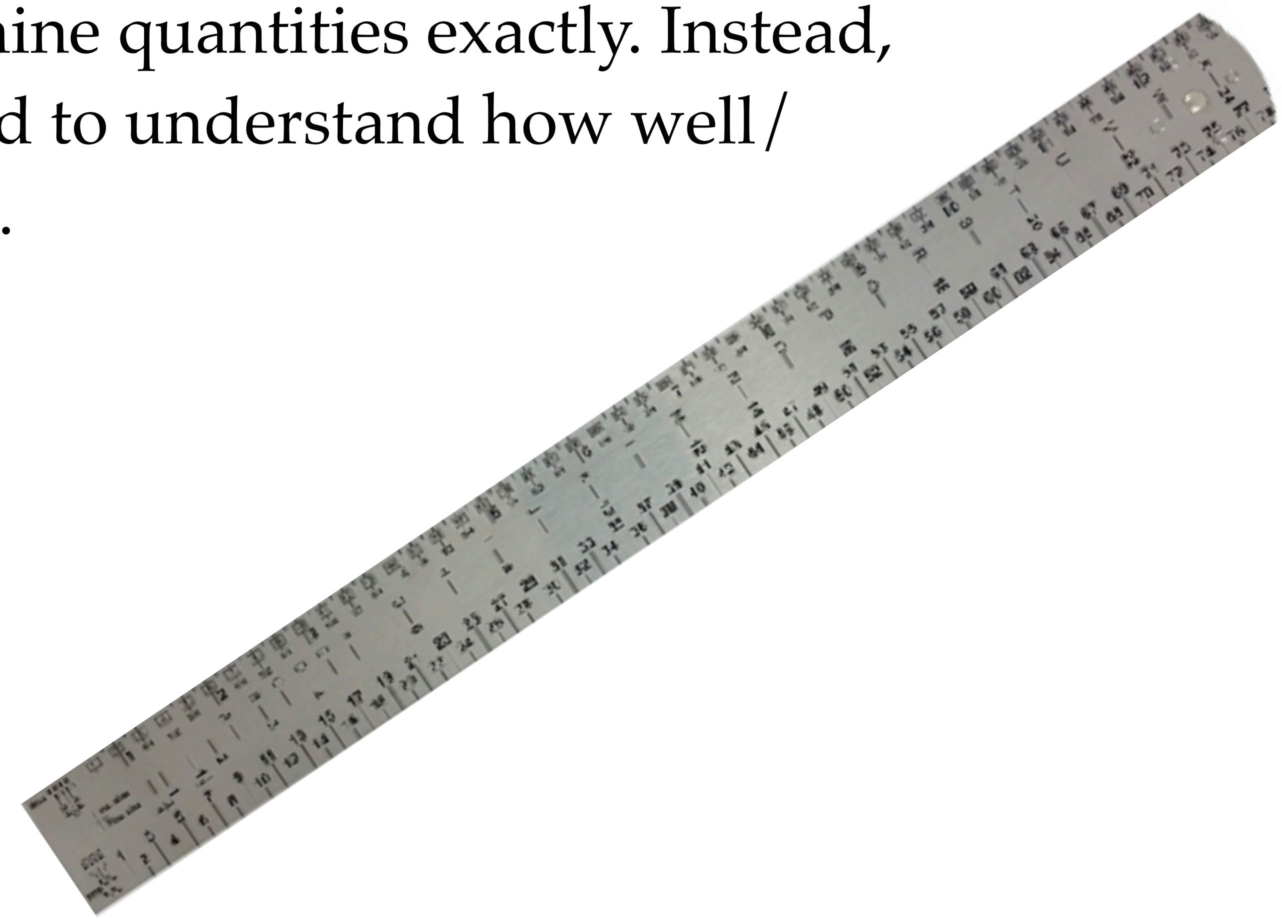
Intro to particle physics and relativity

In order to understand nature, we ***need to observe*** it.

We observe it by ***collecting data.***

Our tools for collecting data are not infinitely precise. In fact, they introduce an error of resolution.

Most times in physics, we cannot determine quantities exactly. Instead, we solve them "approximately". We need to understand how well/ poorly we are actually describing nature.



We will learn a bit about:

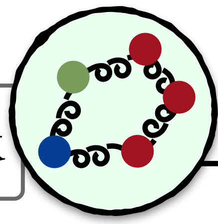
- probability and statistics
- how to understand and describe data
- estimate and calculate statistical and systematic error

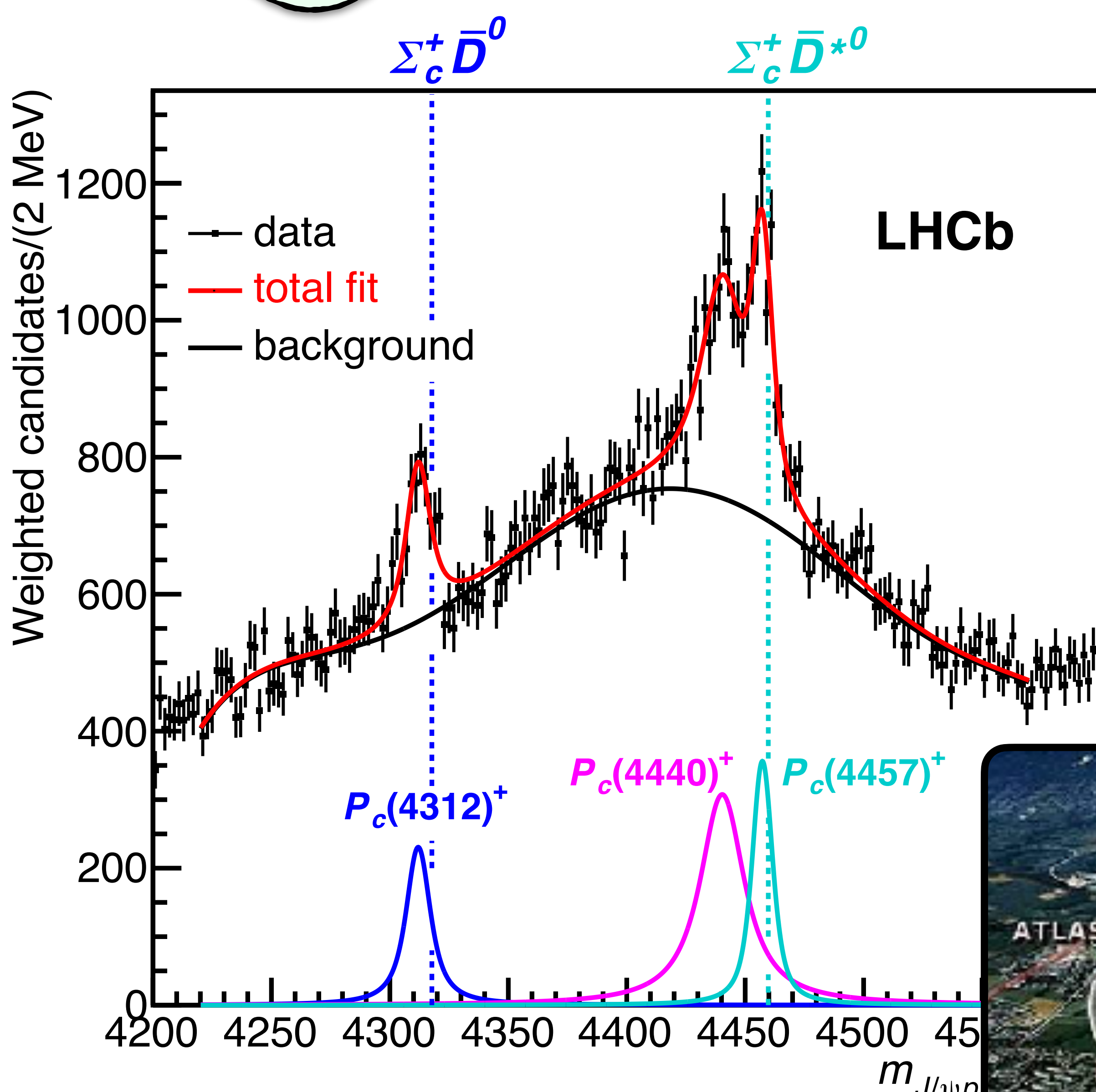
EXPERIMENTS

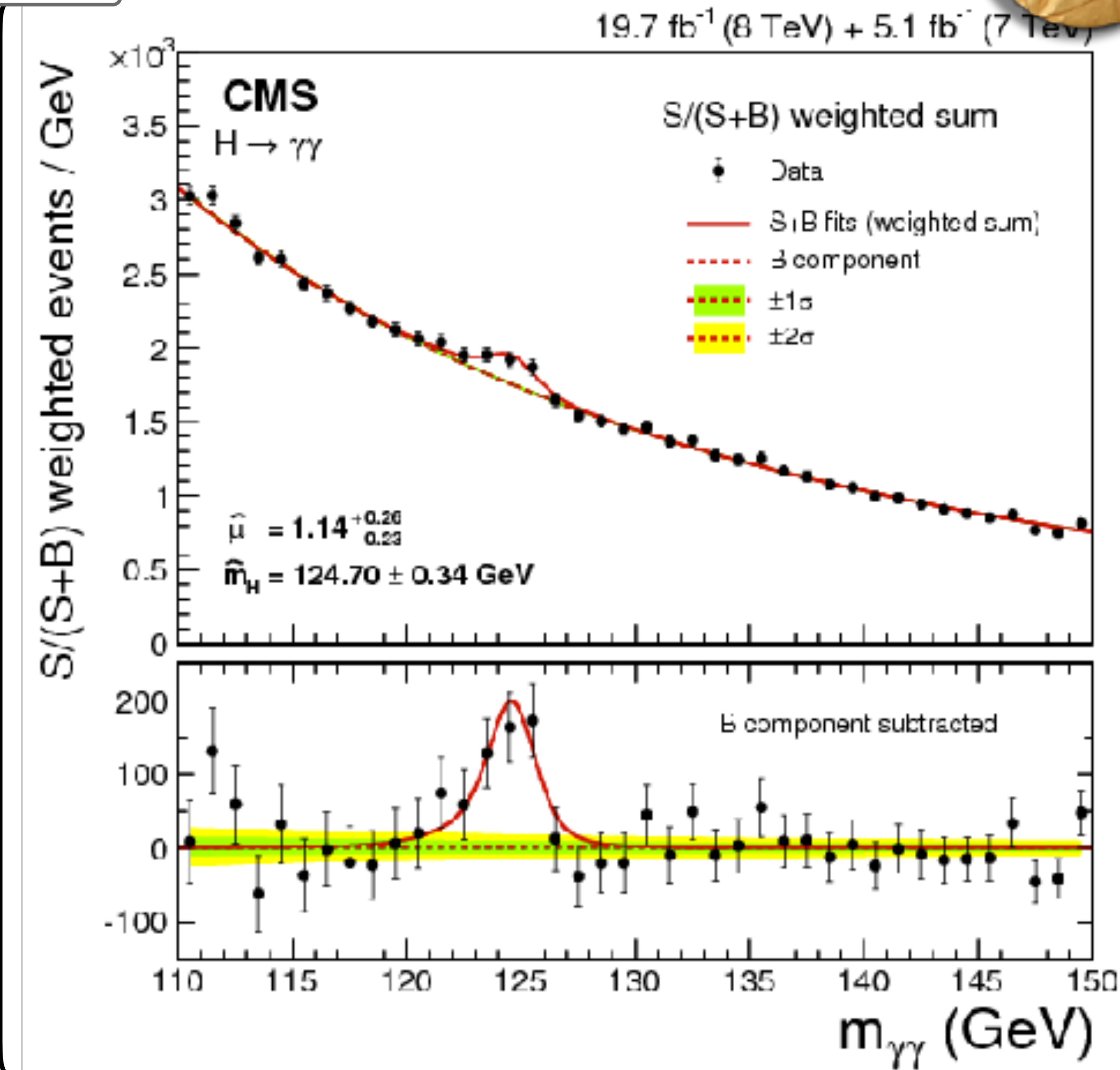
the Higgs

Discoveries require detailed data analysis

pentaquark



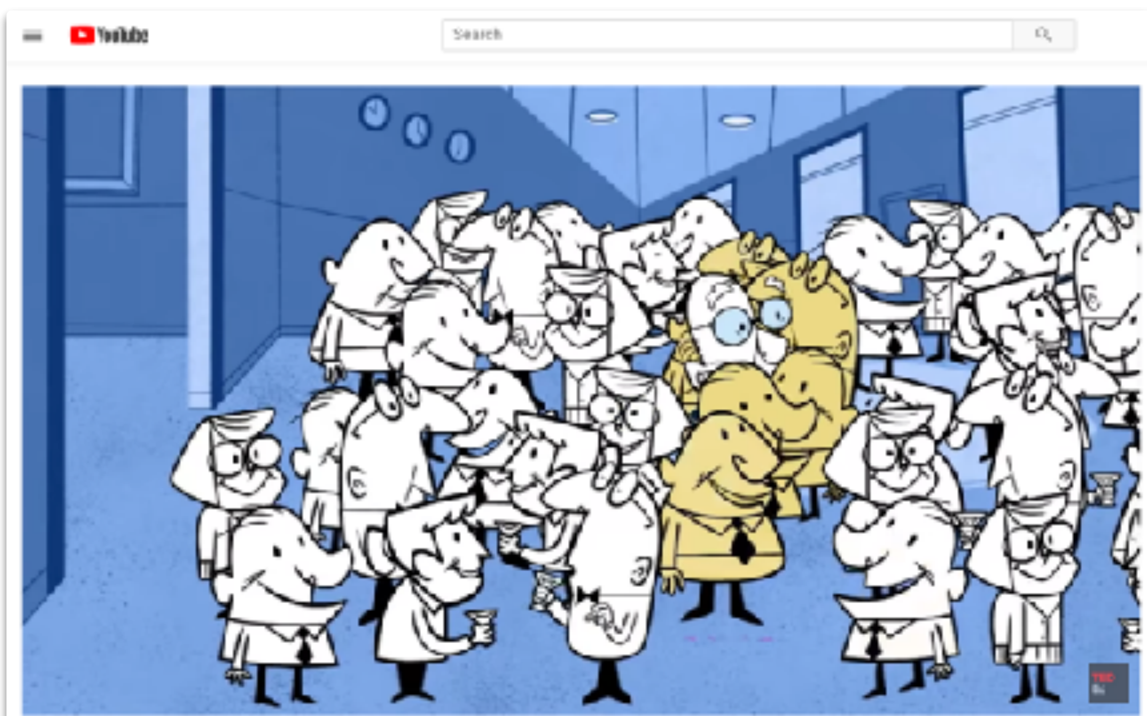


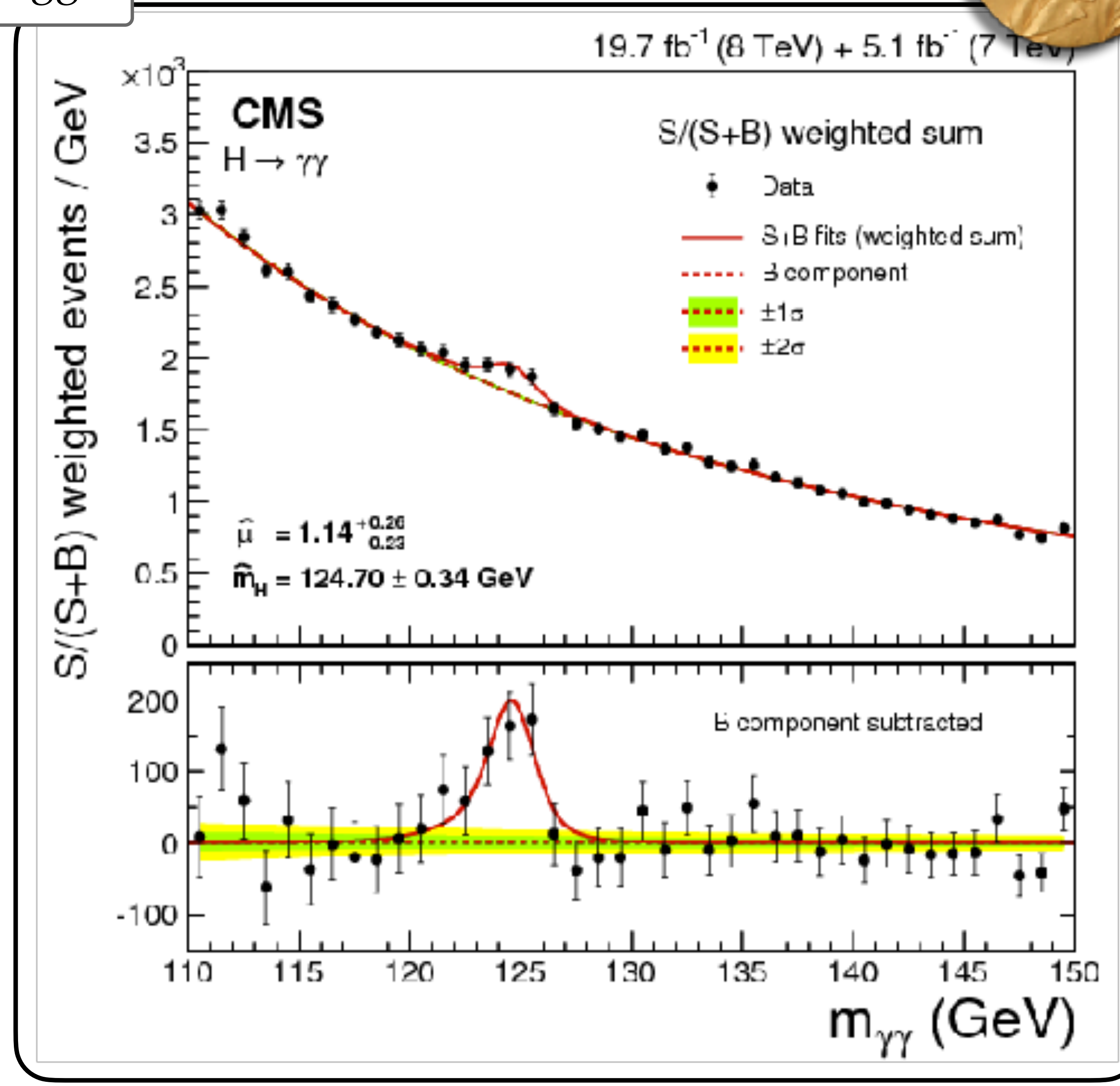


EXPERIMENTS

the Higgs

- the Higgs explained:





https://www.youtube.com/watch?v=joTKd5j3mzk

EXPERIMENTS

the Higgs

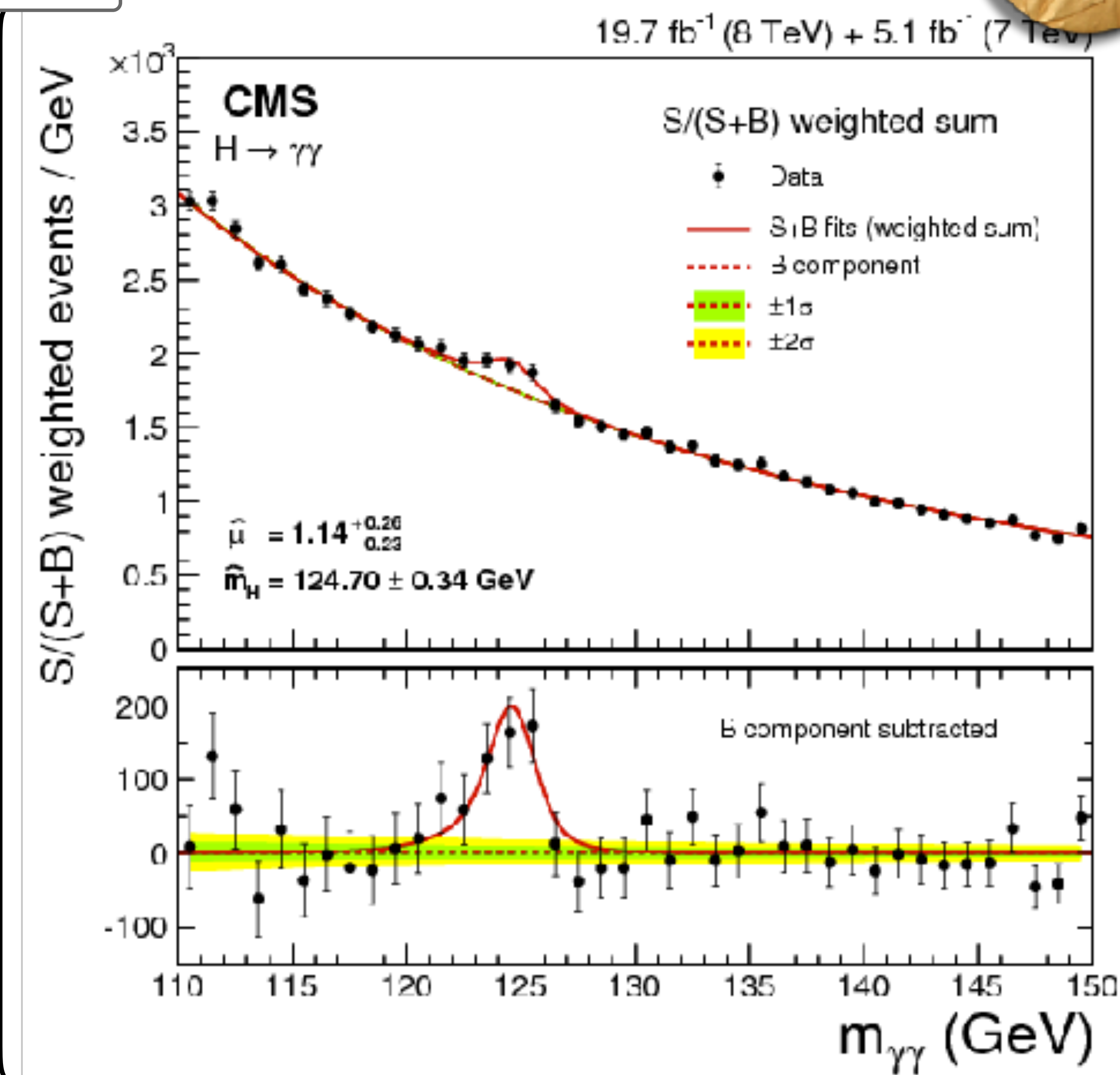
M. T. Hansen, Ph.D.

Scientific Staff - CERN

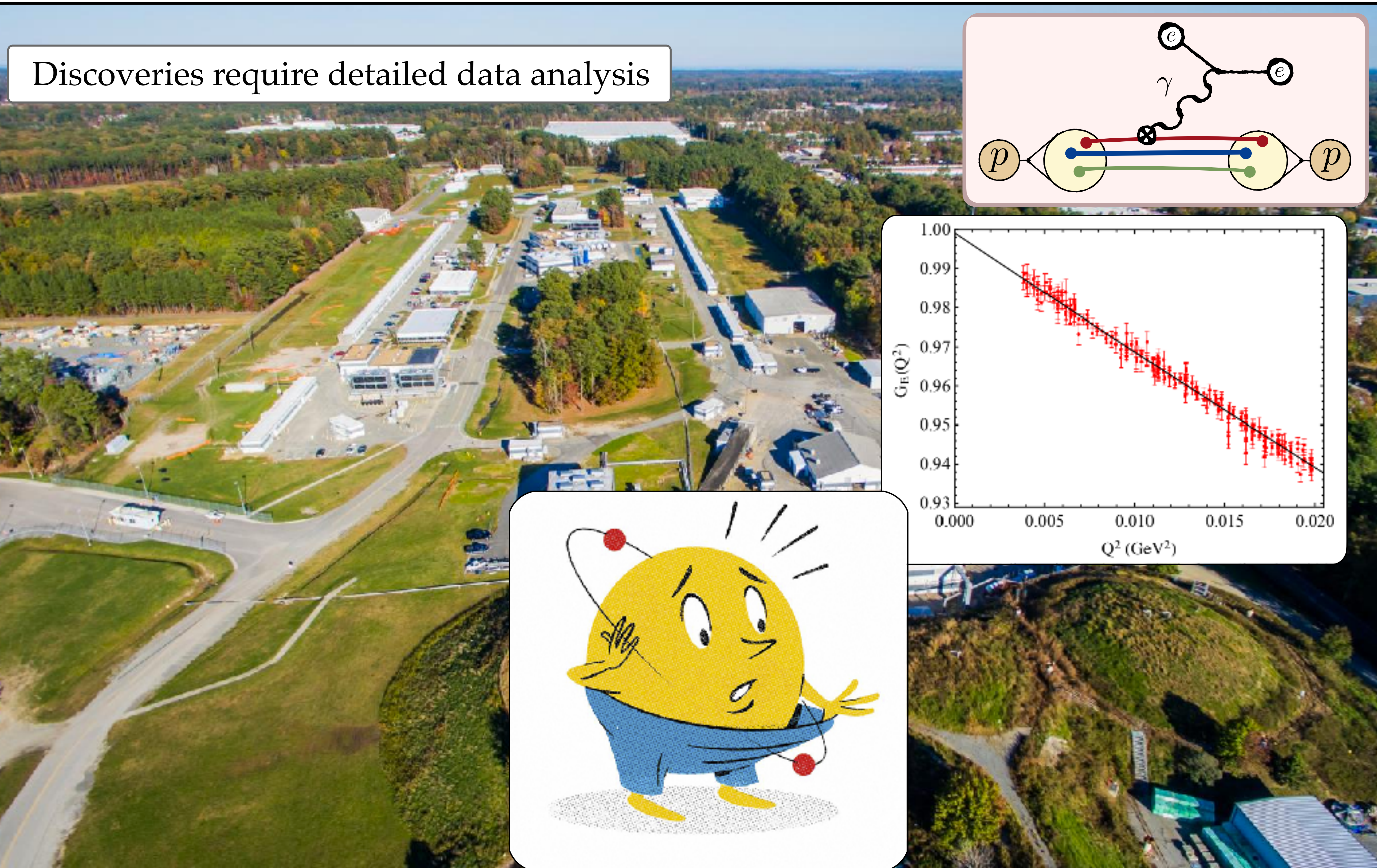
CERN: The past and future

of the world's best microscope

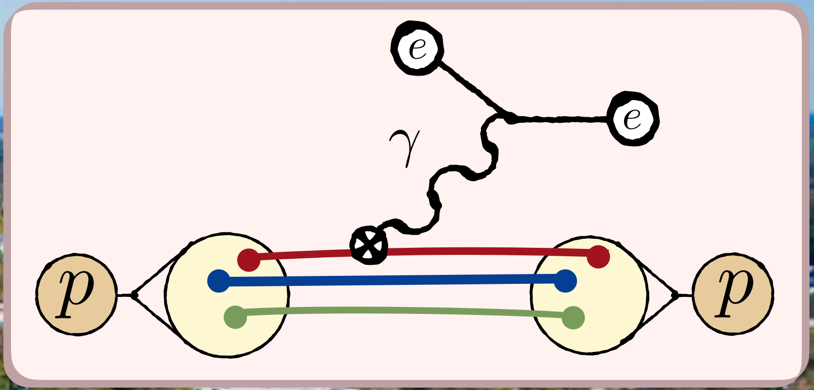
07/20

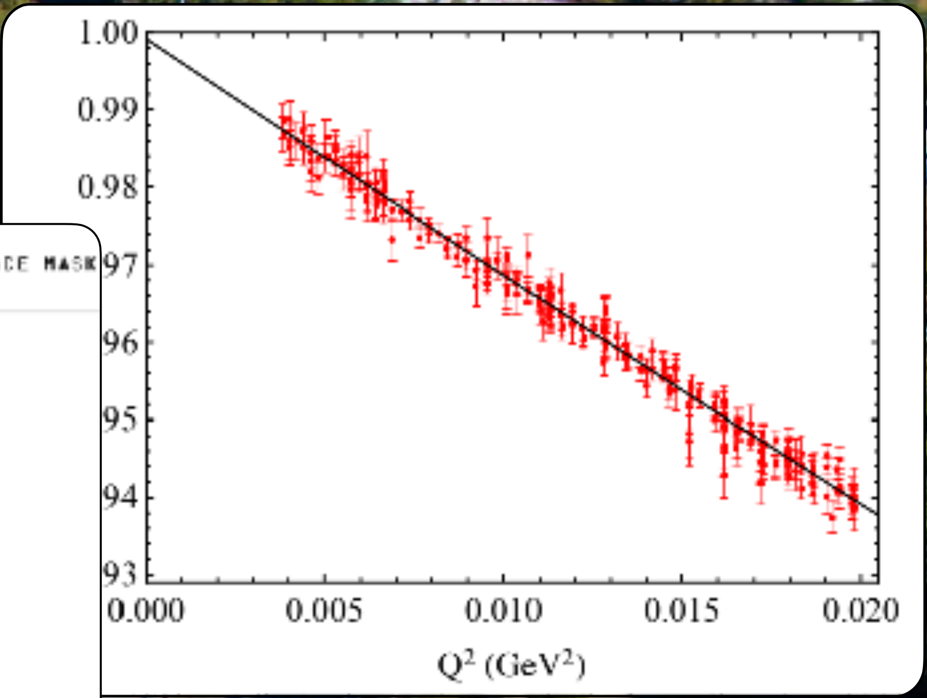


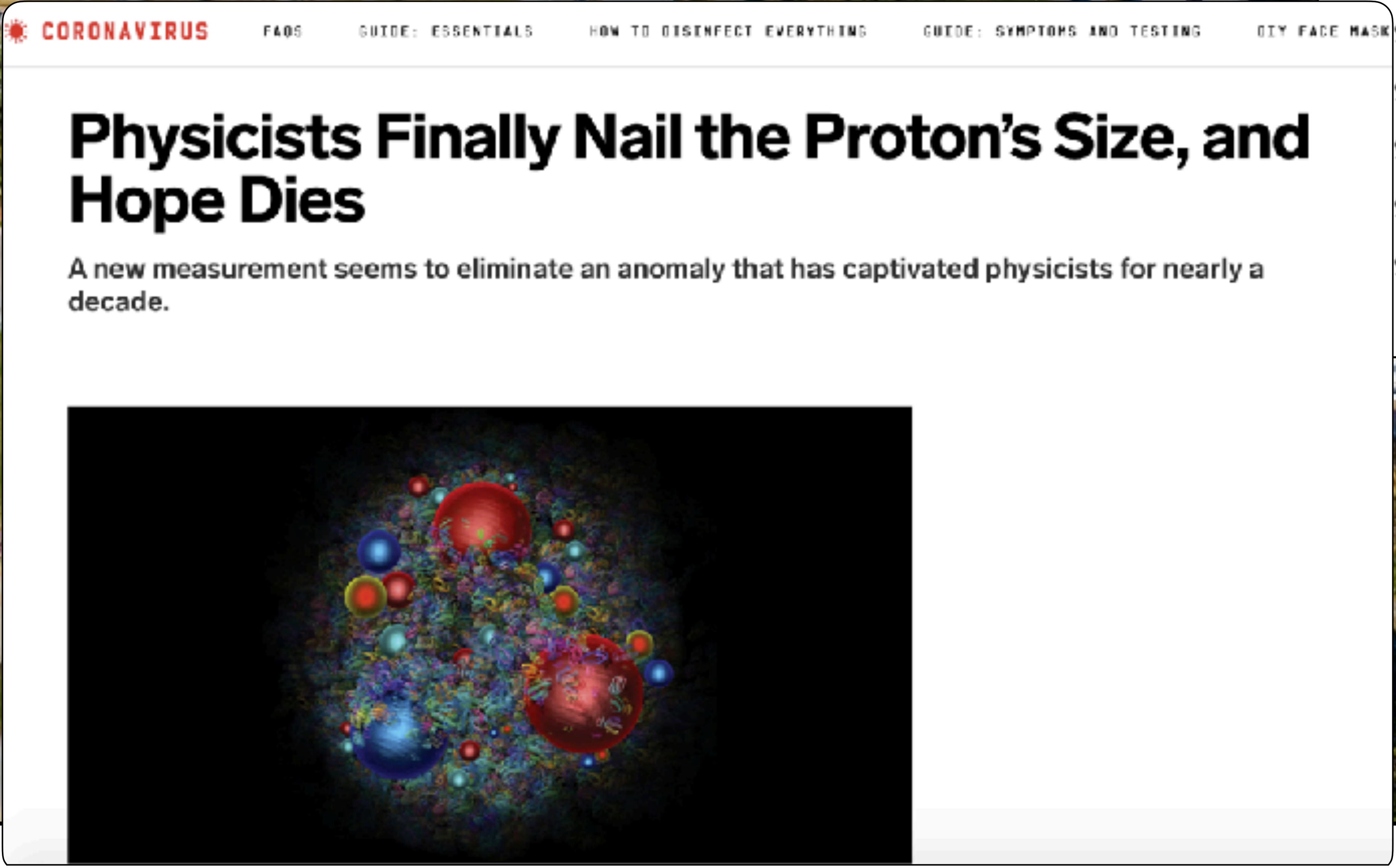
Discoveries require detailed data analysis



Discoveries require detailed data analysis





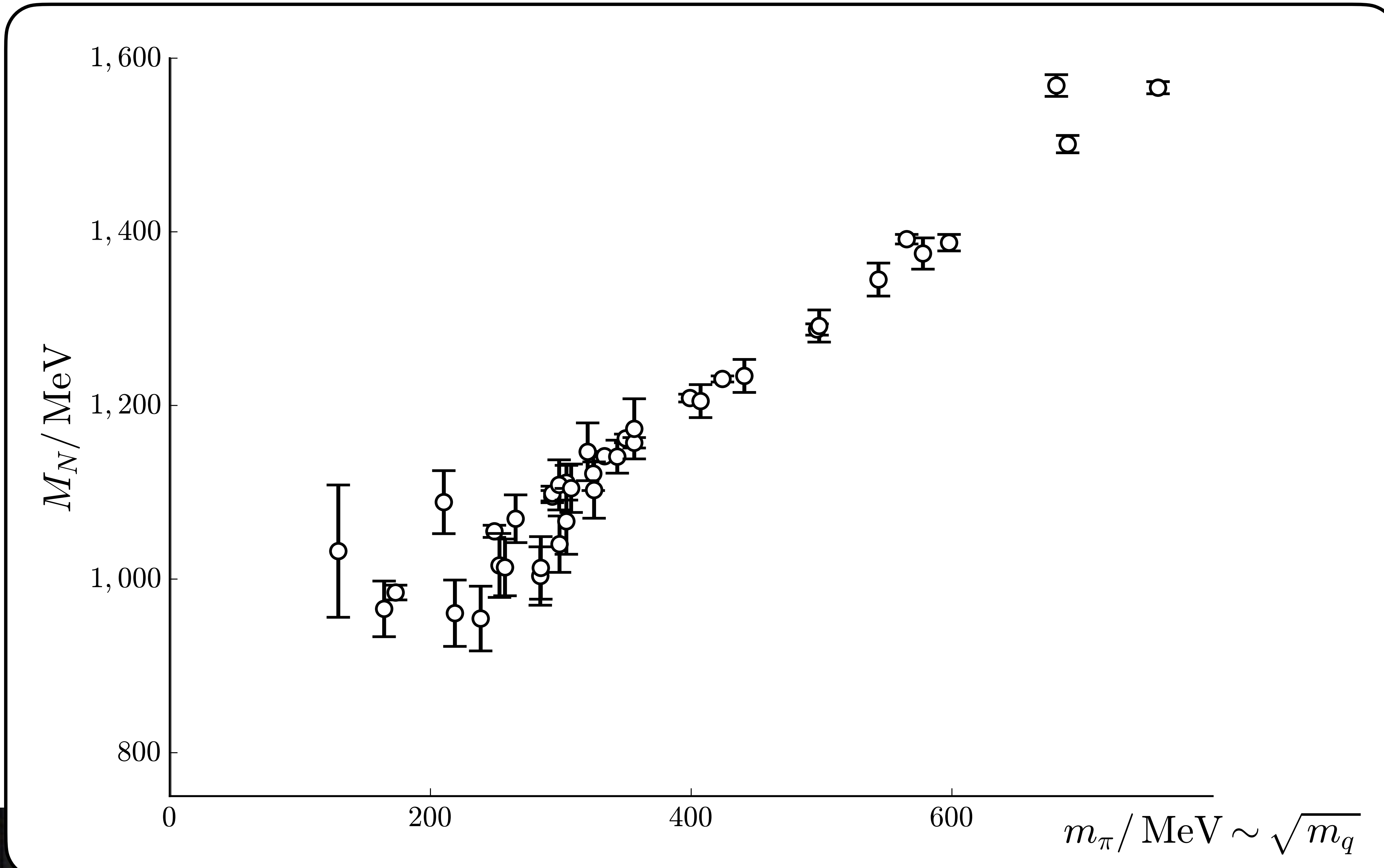

CORONAVIRUS FAQS GUIDE: ESSENTIALS HOW TO DISINFECT EVERYTHING GUIDE: SYMPTOMS AND TESTING DIY FACE MASK

Physicists Finally Nail the Proton's Size, and Hope Dies

A new measurement seems to eliminate an anomaly that has captivated physicists for nearly a decade.

THEORY

Numerical evaluation of the proton mass a function of the parameters of the universe (i.e. the values of the quark masses)



SIMPLE PROBABILITY

What is the probability that if I pick a **random person in the class** that they would be **female**?

SIMPLE PROBABILITY

What is the probability that if I pick a **random person in the class** that they would be **female**?

$$P_W = \boxed{\frac{\text{Number of women}}{\text{total number of people}}} = \frac{\text{Number of women}}{\text{total number of women + men}} \leq 1$$

data! 🤓

What is the probability that if I pick a **random person in the class** that they would be **female**?

$$P_W = \frac{\text{Number of women}}{\text{total number of people}} = \frac{\text{Number of women}}{\text{total number of women + men}} \leq 1$$

What is the probability that if I pick a **random person in the class** that they would be a **high school student**?

What is the probability that if I pick a **random person in the class** that they would be **female**?

$$P_W = \frac{\text{Number of women}}{\text{total number of people}} = \frac{\text{Number of women}}{\text{total number of women + men}} \leq 1$$

What is the probability that if I pick a **random person in the class** that they would be a **high school student**?

$$P_{HS} = \frac{\text{Number of high school student}}{\text{total number of people}} \leq 1$$

What is the probability that if I pick a **random person in the class** that they would be a **female high school student**?

SIMPLE PROBABILITY

What is the probability that if I pick a **random person in the class** that they would be **female**?

$$P_W = \frac{\text{Number of women}}{\text{total number of people}} = \frac{\text{Number of women}}{\text{total number of women + men}} \leq 1$$

What is the probability that if I pick a **random person in the class** that they would be a **high school student**?

$$P_{HS} = \frac{\text{Number of high school student}}{\text{total number of people}} \leq 1$$

What is the probability that if I pick a **random person in the class** that they would be a **female high school student**?

$$P_{WHS} = \frac{\text{Number of female high school student}}{\text{total number of people}} \leq (P_{W,}\ P_{HS}) \leq 1$$

LESS SIMPLE PROBABILITY

What is the probability that if I pick a **random person in the street** that they would be a **female high school student**?

How would we go about got about getting a good estimate of this?

What is the probability that if I pick a **random person in the street** that they would be a **female high school student**?

How would we go about got about getting a good estimate of this?

- we would could go to Hampton roads, start asking every person if they are a female high school student, *tally all the no's (Ns) and all the yes's (Ys). In the limit that we ask every single person in the street this question, we would arrive at our answer.*

$$P_{WHS} = \lim_{\text{limit where we ask everyone}} \left(\frac{Ys}{Ns + Ys} \right)$$

What is the probability that if I pick a **random person in the street** that they would be a **female high school student**?

How would we go about got about getting a good estimate of this?

- we would could go to Hampton roads, start asking every person if they are a female high school student, *tally all the no's (Ns) and all the yes's (Ys). In the limit that we ask every single person in the street this question, we would arrive at our answer.*

$$P_{WHS} = \underset{\substack{\text{limit where} \\ \text{we ask} \\ \text{everyone}}}{}\left(\frac{Ys}{Ns + Ys} \right)$$

for almost every question we ask, we would need to have take almost an infinite amount of data 🤨

this sounds kind of impractical… 🤔

GUESSING THE WEIGHT OF AN OX

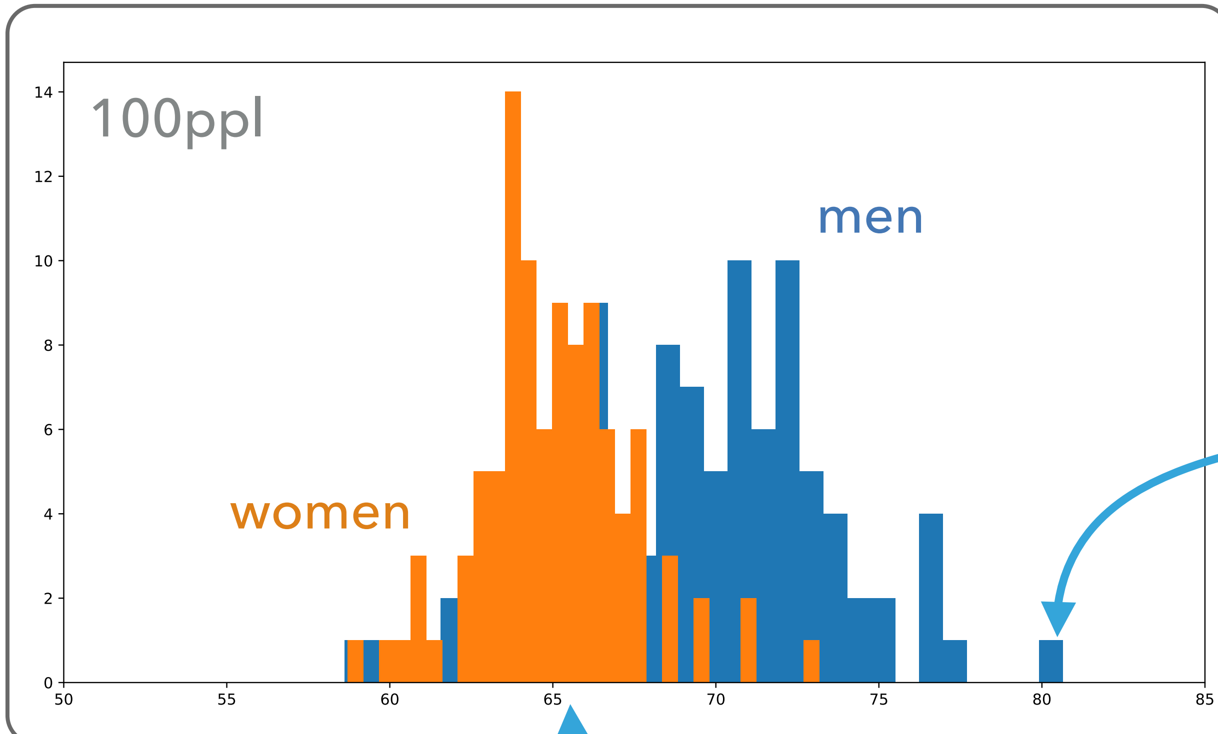
In 1906, visiting a livestock fair, a statistician stumbled upon an intriguing contest. An ox was on display, and the villagers were invited to guess the animal's weight. Nearly 800 participated. The statistician was able to study their individual entries after the event and calculate the mean of the distribution. The statistician reported the mean value as 1,207 pounds. To the statistician's surprise, this was within 0.8% of the weight measured by the judges.

so there has to be a smarter way… 😀

Now let's say that we want to know the height of the average American…
let's go out on the street and start asking folks…

DISTRIBUTIONS AND HISTOGRAMS

Now let's say that we want to know the height of the average American…
let's go out on the street and start asking folks…

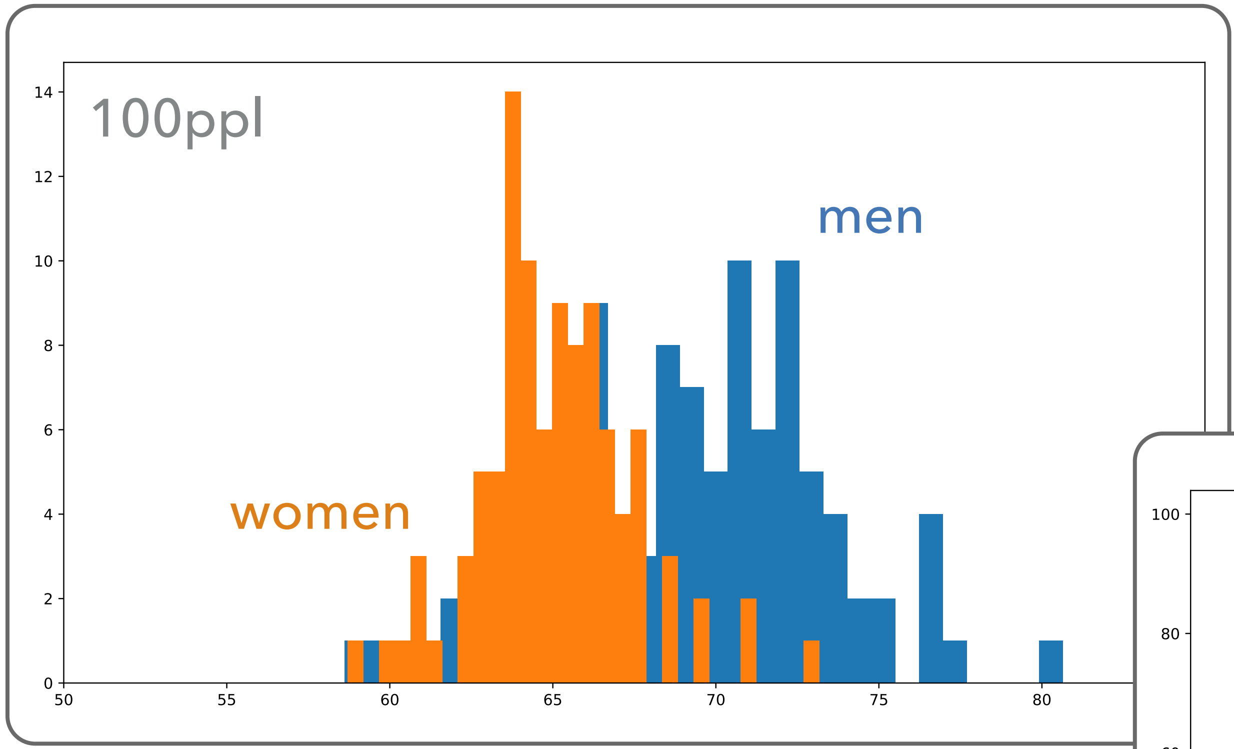


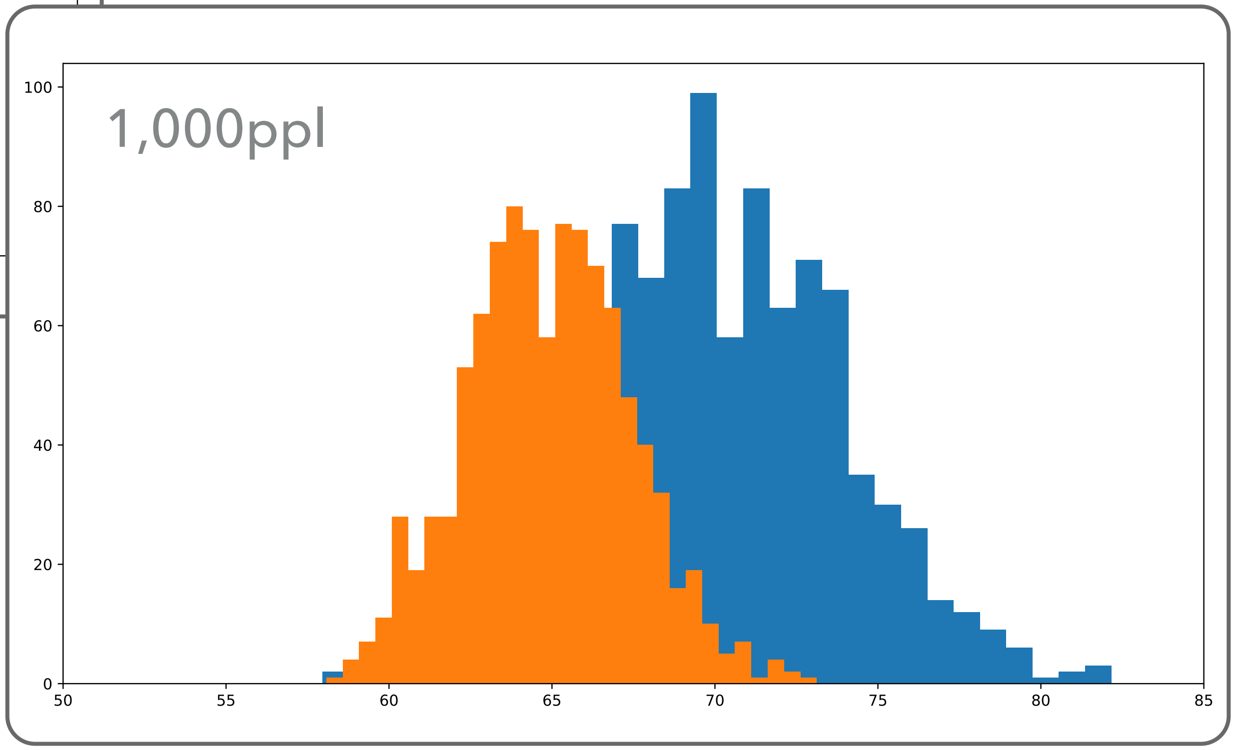
Lebron James (6′8″ tall)

Megan Rapinoe (5′6″ tall)

DISTRIBUTIONS AND HISTOGRAMS

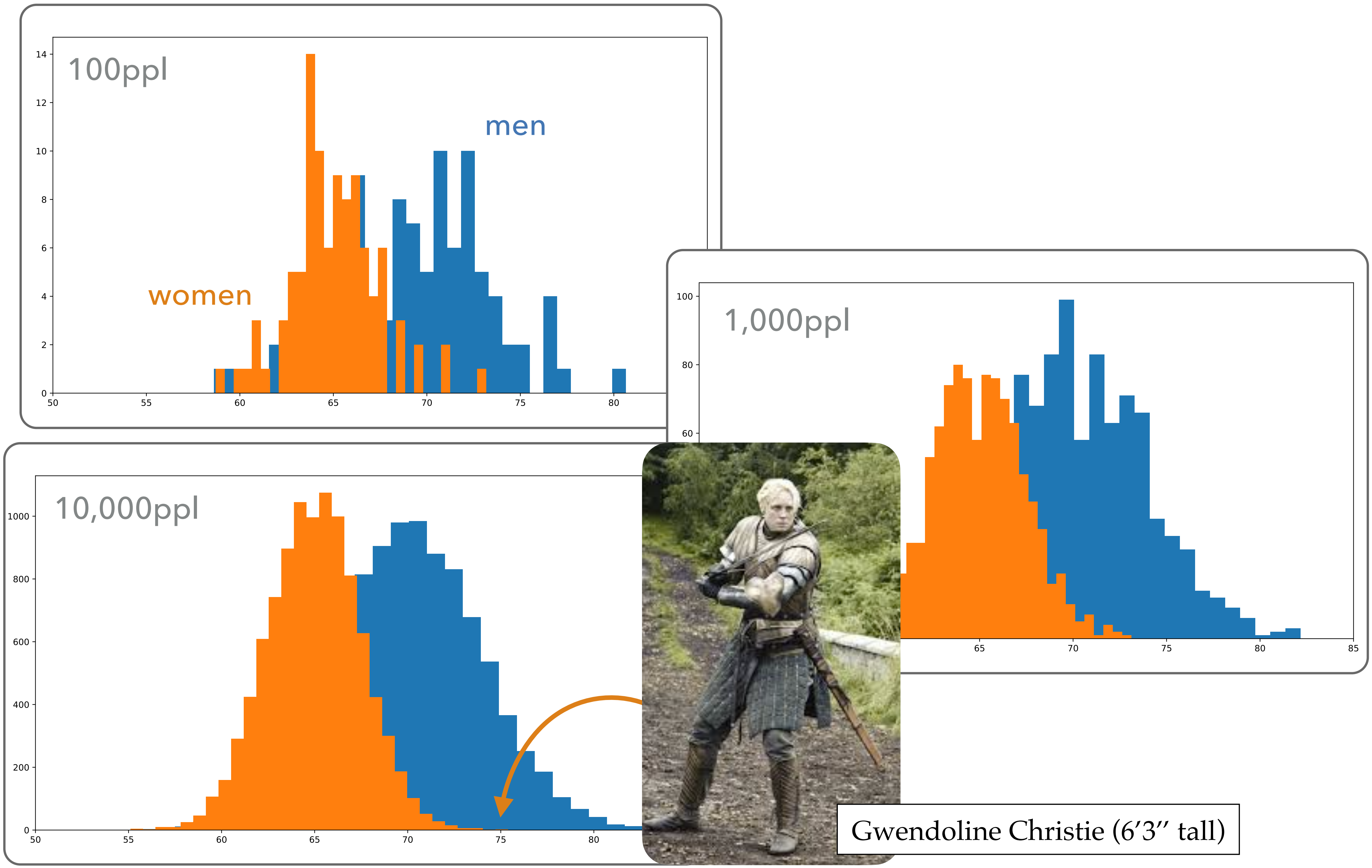
Now let's say that we want to know the height of the average American…
let's go out on the street and start asking folks…





DISTRIBUTIONS AND HISTOGRAMS

Now let's say that we want to know the height of the average American…
let's go out on the street and start asking folks…



Gwendoline Christie (6′3″ tall)

AVERAGE/MEAN

Now, let's call the the distribution of the N data points of heights: $\{x_i\} = \{x_1, x_2, \ldots, x_N\}$

Then the average height is the sum over all heights, divided by the number of measurements made, and give it a special symbol:

$$\boxed{\bar{x} = \frac{\sum_{i=1}^{N} x_i}{N} = \frac{x_1 + x_2 + \cdots + x_N}{N}}$$

Examples:

$$\{x\} = \{55, 60\} \quad \Rightarrow \quad \bar{x} = \frac{55 + 60}{2} = 57.5$$

$$\{x\} = \{55, 68, 60\} \quad \Rightarrow \quad \bar{x} = \frac{55 + 68 + 60}{3} = 61$$

STANDARD DEVIATION OF A SAMPLE

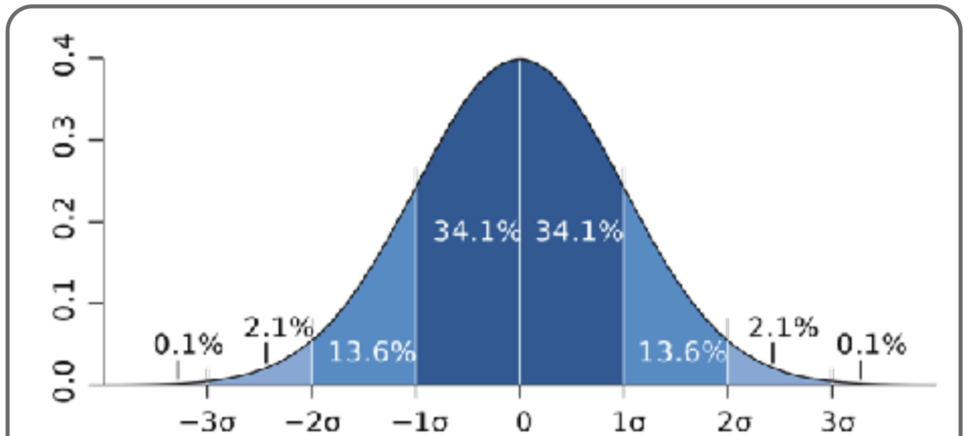
Another measure, which tells you the spread of the distribution is the standard deviation.

$$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(\underline{x_i - \bar{x}})^2}$$

if $N = 1$, then σ is infinite…telling you that one one data point is not enough to tell you the error you have made…

distance away from the mean

All distributions we will consider are so-called normal distributions (https://en.wikipedia.org/wiki/Normal_distribution). These are distributed according to this "probability density" $e^{-\frac{(x-\bar{x})^2}{2\sigma^2}}/\sqrt{2\pi\sigma^2}$



https://en.wikipedia.org/wiki/Standard_deviation

…for these distributions, the standard deviation tells us where 68.27% of the population lies

STANDARD DEVIATION OF SAMPLE VS. POPULATION

A subtlety arise when comparing the standard deviation (std). There are different ways of calculation or estimating the std, here we consider three.

If we know the have sampled the **full population**, then the standard deviation of of the population is defined as

$$\sigma_{\text{population}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2}$$

If instead we have only have a **sample of the population**, then we can only estimate the standard deviation of the distribution. This is what we will largely be using, and it is estimated as

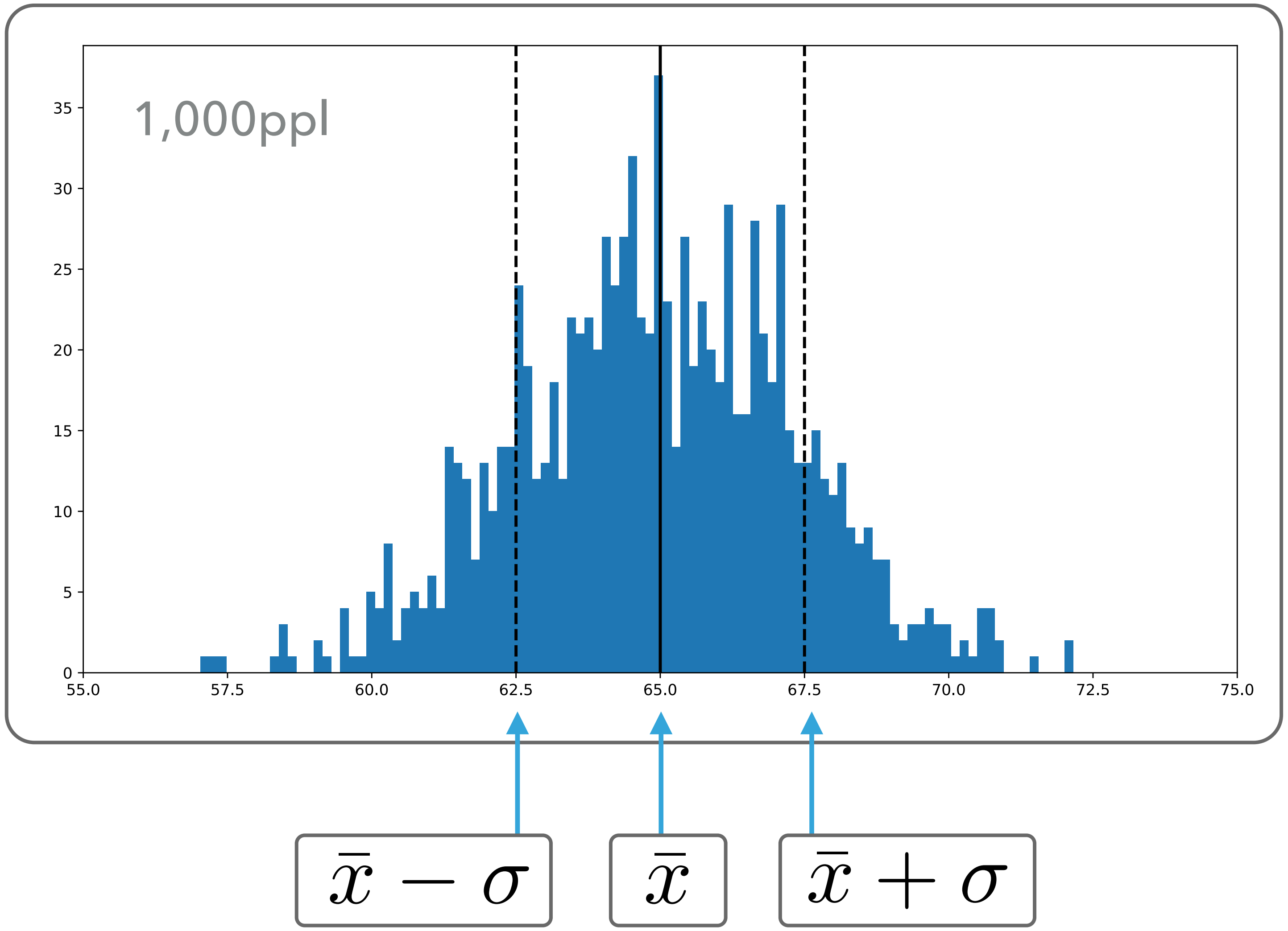
note, that the sample std is always bigger than the population std, because

$$\sigma_{\text{sample}} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{x})^2}$$

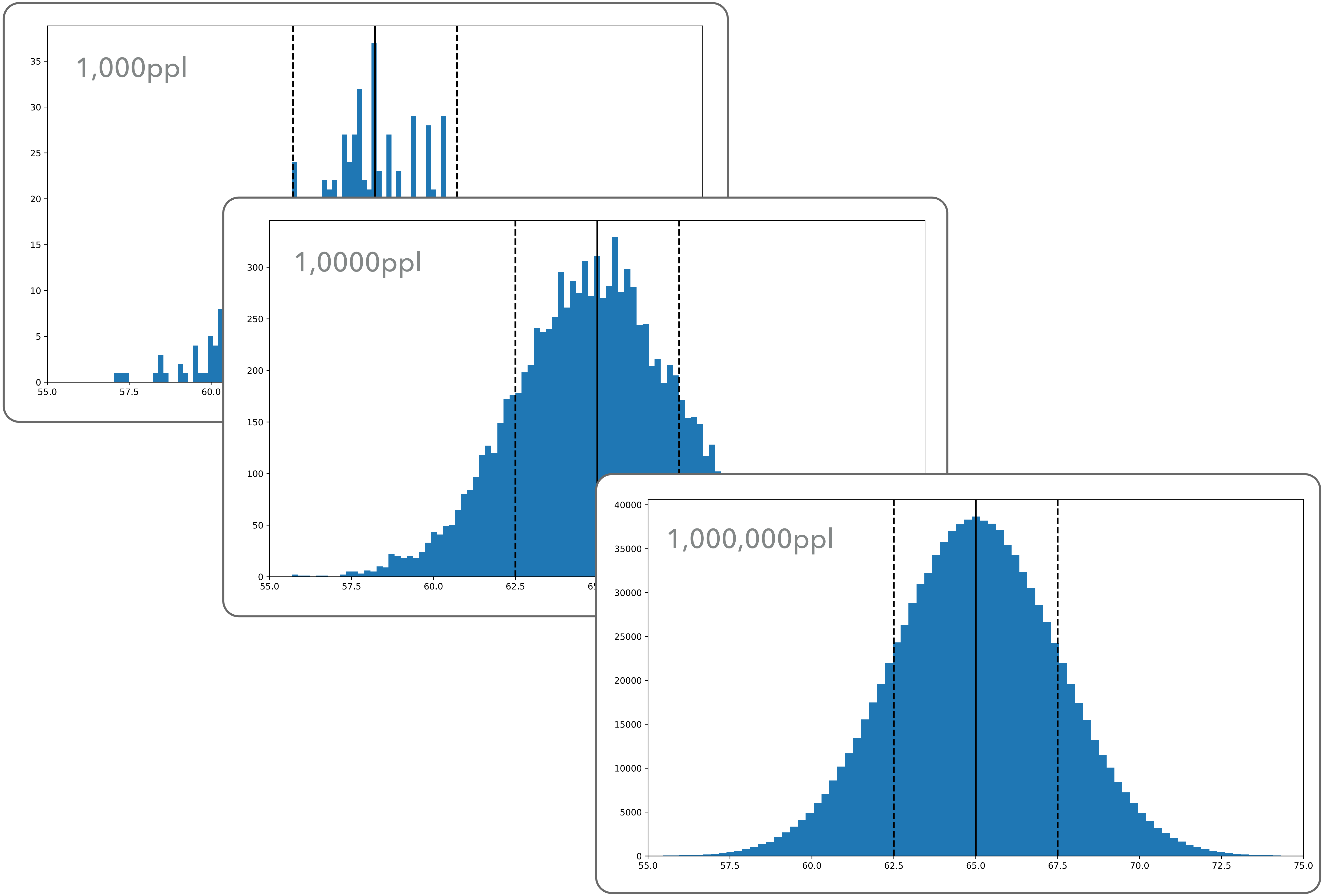
If we then want to estimate the error on the mean, as known as the **standard error**, from the sample distribution, you can estimate this from the std.

$$\sigma_{\text{mean}} = \frac{\sigma_{\text{sample}}}{\sqrt{N}}$$

MEAN AND STANDARD DEVIATION OF WOMEN



MEAN AND STANDARD DEVIATION OF WOMEN



QUESTION

Calculate the mean and standard deviation of: $\{x_i\} = \{78, 50, 69, 95, 69, 19\}$

$$\bar{x} = \frac{\sum_{i=1}^{N} x_i}{N} = \frac{x_1 + x_2 + \cdots + x_N}{N}$$

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} (x_i - \bar{x})^2}$$

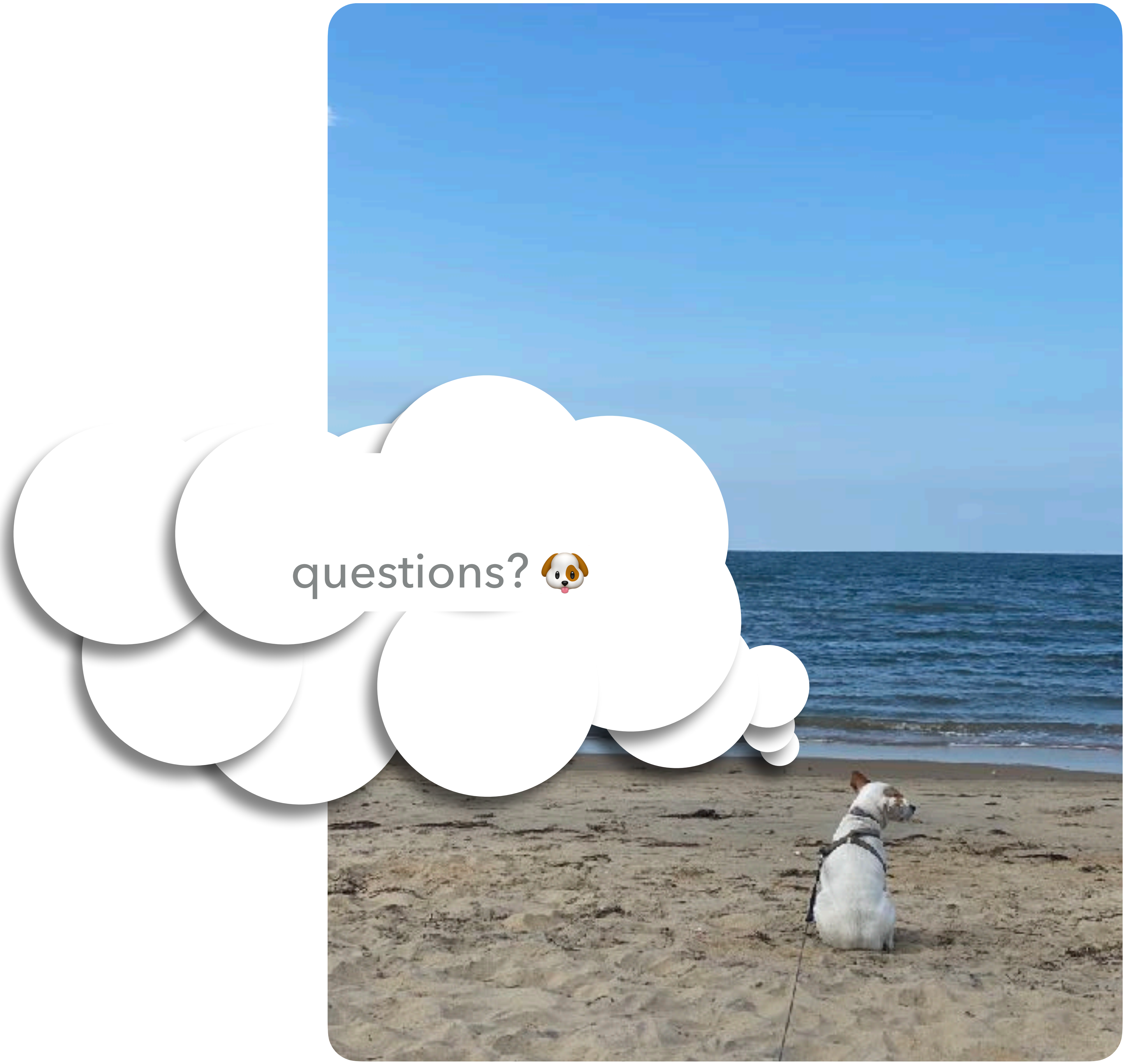
QUESTION

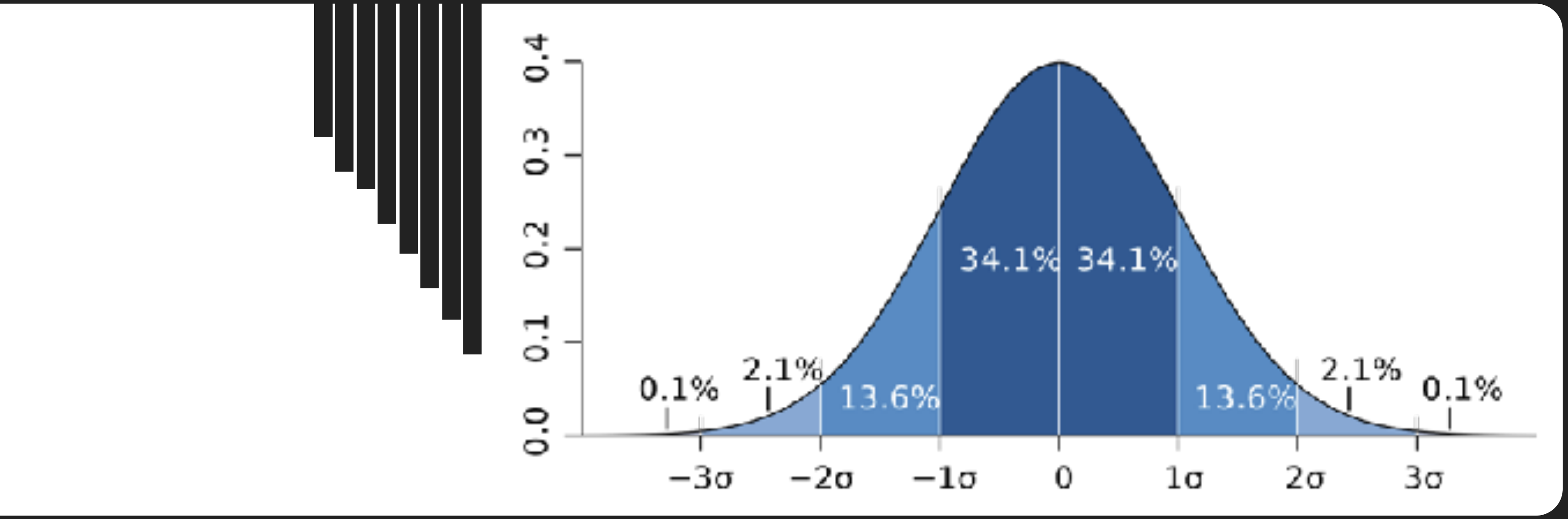
Calculate the mean and standard deviation of: $\{x_i\} = \{78, 50, 69, 95, 69, 19\}$

$$\bar{x} = \frac{78 + 50 + 69 + 95 + 69 + 19}{6} = \frac{380}{6} = 63$$

$$\sigma = \sqrt{\frac{(78-63)^2 + (50-63)^2 + (69-63)^2 + (95-63)^2 + (69-63)^2 + (19-63)^2}{5}} = \sqrt{\frac{3425}{5}} = 26$$

QUESTIONS





R. BRICEÑO, T. ROGERS

DATA ANALYSIS AND THE NEUTRINO MASS

ADMIN STUFF

Need links, just email python4physics@odu.edu

Move to...

Python 4 Physics 5:19 PM PP

Automatic reply:

To: Briceno, Raul A.

Thanks for your interest in Python4Physics.

Please visit our webpage at https://sites.google.com/view/odu-nuc-th/service/p4p-2020. In addition to providing links to our Dropbox folder, you will see our "Frequently Asked Questions" section. There we answer the many questions we have been receiving.

Slack chat with faculty and TAs: https://join.slack.com/t/python4physics/shared_invite/zt-ffgssu43-4x9_bCCLmGt8dou~Xwzycw. Note, to use Slack you must be at least 16yrs old [see https://slack.com/terms-of-service].

Livestreams link: https://vs.prod.odu.edu/bin/reyes_system/

Recordings link: https://odu.edu/reyes/recordings
Reyes - Python4Physics archive: https://vs.prod.odu.edu/bin/reyes_system/archives/6_python4Physics.php
Reyes - Python4Physics breakout sessions archive: https://vs.prod.odu.edu/bin/reyes_system/archives/6_python4Physics_breakouts.php

Dropbox link: https://www.dropbox.com/sh/ur6mk8gzl22mq41/AACRe9R4Ulb-4bYAvJG2UI3aa?dl=0

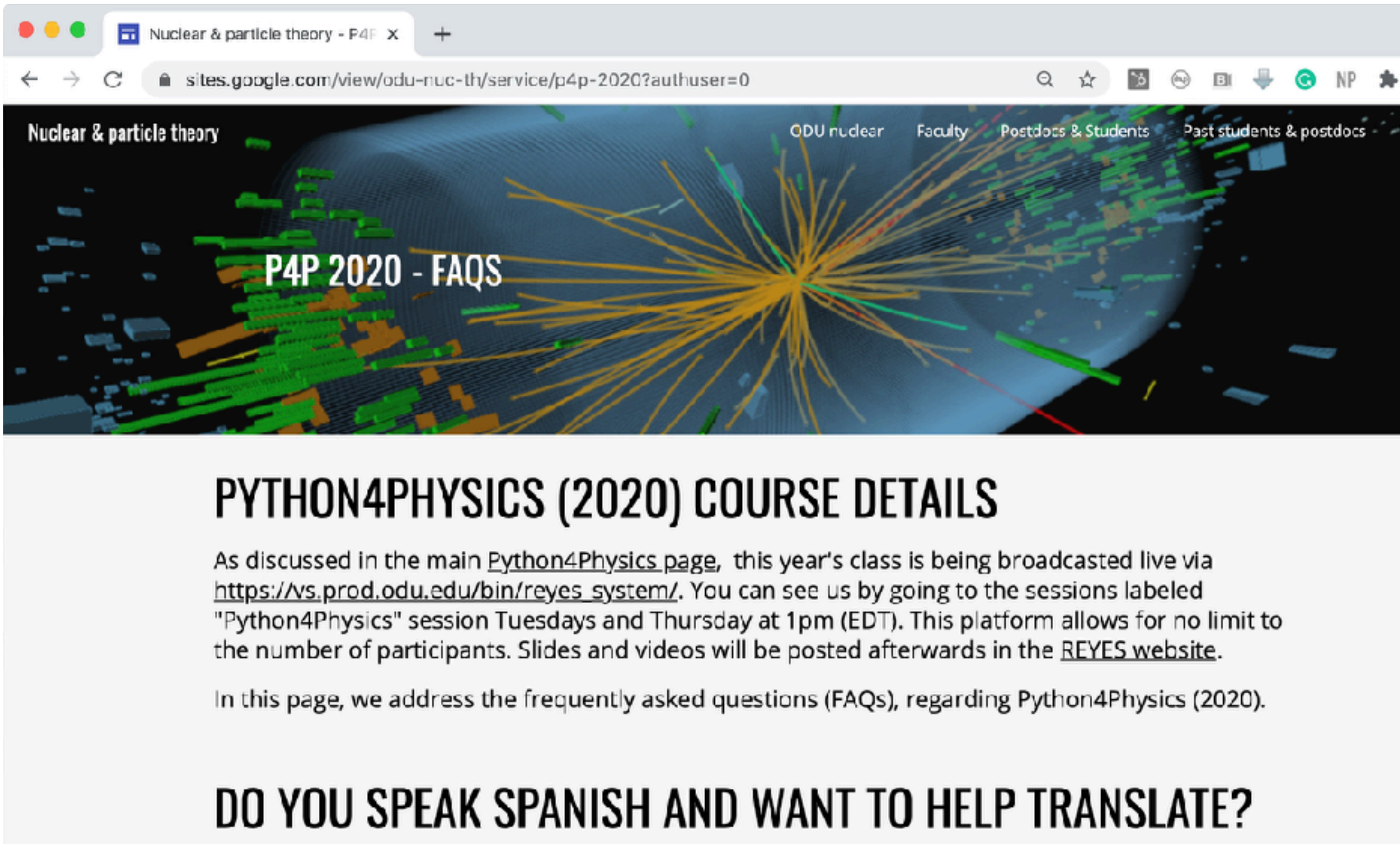
Briceno, Raul A. 5:19 PM RB

(No Subject)

To: Python 4 Physics

ADMIN STUFF

Need links, just email python4physics@odu.edu



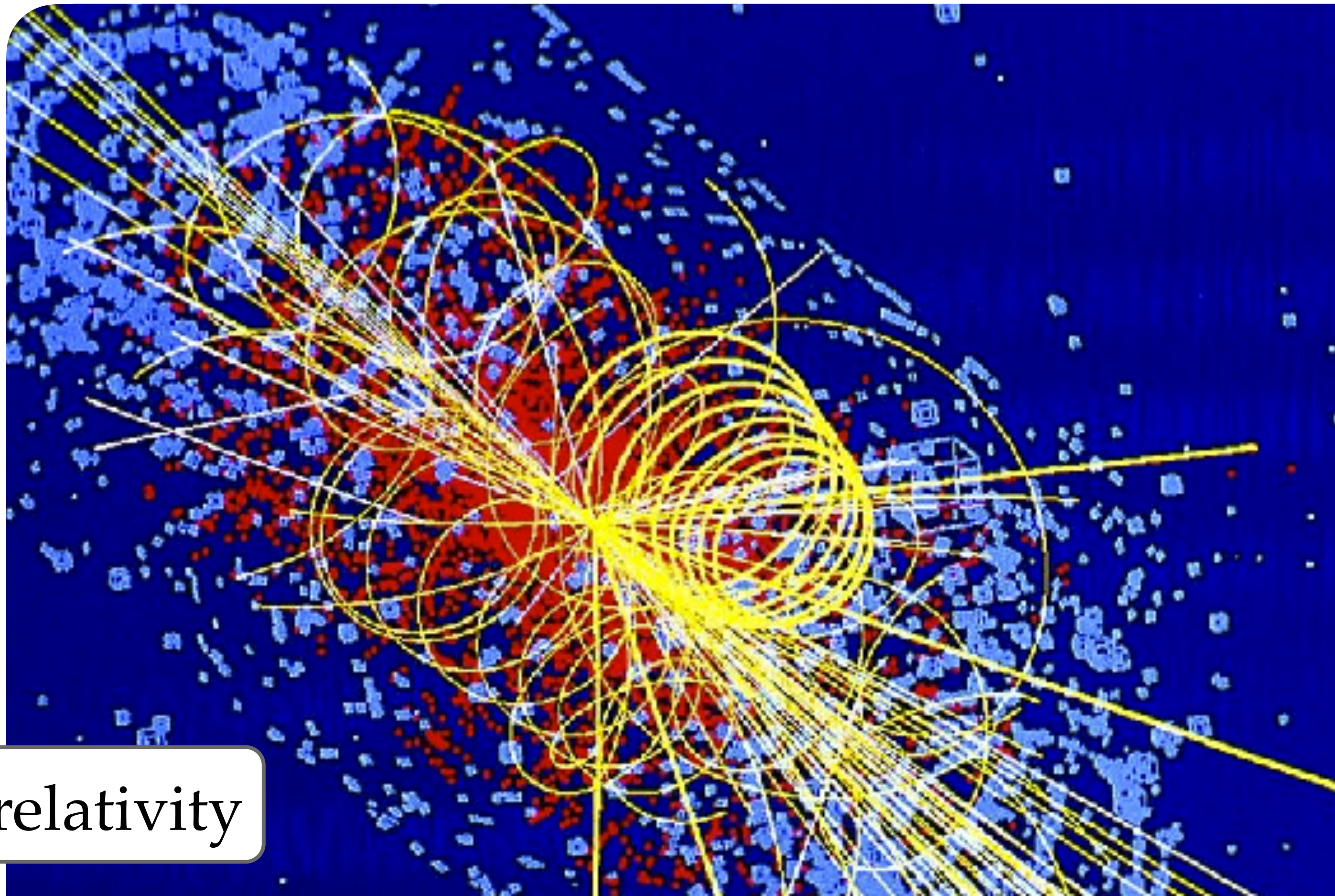
ADMIN STUFF

Need links, just email python4physics@odu.edu

Dropbox for students, email us! Subject: “Dropbox access”

OUTLINE

Intro to data analysis



Intro to particle physics and relativity

AVERAGE/MEAN

Now, let's call the the distribution of the N data points of heights: $\{x_i\} = \{x_1, x_2, \ldots, x_N\}$

Then the average height is the sum over all heights, divided by the number of measurements made, and give it a special symbol:

$$\boxed{\bar{x} = \frac{\sum_{i=1}^{N} x_i}{N} = \frac{x_1 + x_2 + \cdots + x_N}{N}}$$

Examples:

$$\{x\} = \{55, 60\} \quad \Rightarrow \quad \bar{x} = \frac{55 + 60}{2} = 57.5$$

$$\{x\} = \{55, 68, 60\} \quad \Rightarrow \quad \bar{x} = \frac{55 + 68 + 60}{3} = 61$$

STANDARD DEVIATION OF A SAMPLE

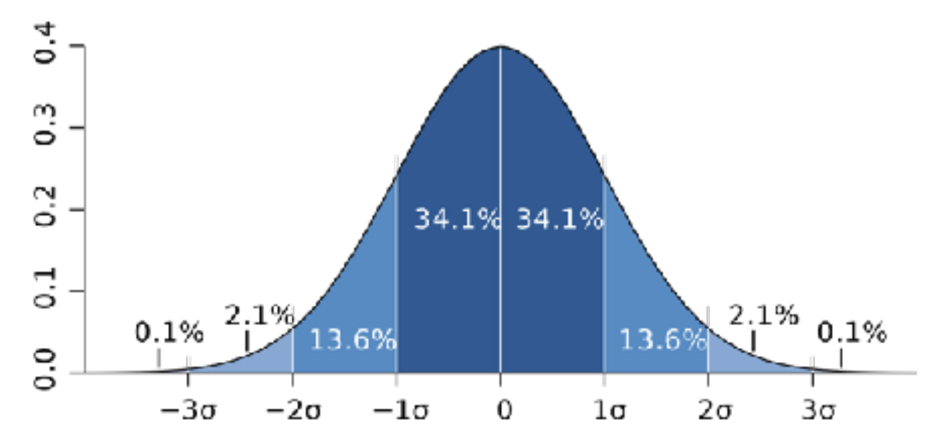
Another measure, which tells you the spread of the distribution is the standard deviation.

$$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\underbrace{(x_i - \bar{x})}_{\text{distance away from the mean}}{}^2}$$

if $N = 1$, then σ is infinite…telling you that one one data point is not enough to tell you the error you have made…

distance away from the mean

All distributions we will consider are so-called normal distributions (https://en.wikipedia.org/wiki/Normal_distribution). These are distributed according to this "probability density" $e^{-\frac{(x-\bar{x})^2}{2\sigma^2}}/\sqrt{2\pi\sigma^2}$



https://en.wikipedia.org/wiki/Standard_deviation

…for these distributions, the standard deviation tells us where 68.27% of the population lies

STANDARD DEVIATION OF SAMPLE VS. POPULATION

A subtlety arise when comparing the standard deviation (std). There are different ways of calculation or estimating the std, here we consider three.

If we know the have sampled the **full population**, then the standard deviation of of the population is defined as

$$\sigma_{\text{population}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2}$$

If instead we have only have a **sample of the population**, then we can only estimate the standard deviation of the distribution. This is what we will largely be using, and it is estimated as

$$\sigma_{\text{sample}} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{x})^2}$$

note, that the sample std is always bigger than the population std, because

If we then want to estimate the error on the mean, as known as the **standard error**, from the sample distribution, you can estimate this from the std.

$$\sigma_{\text{mean}} = \frac{\sigma_{\text{sample}}}{\sqrt{N}}$$

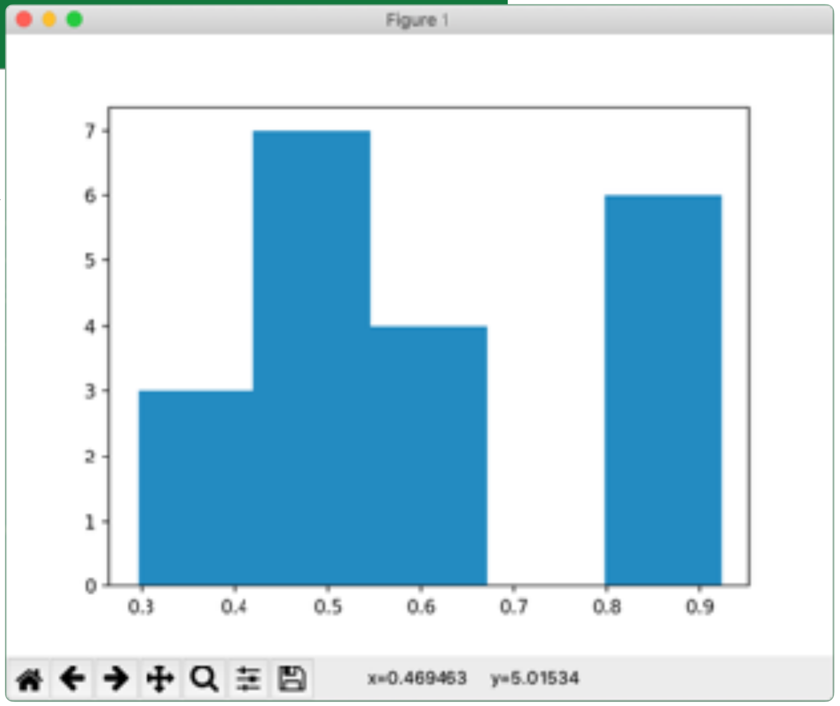
EXPLAIN HISTOGRAM PLOTS

In order to understand histogram plots, it sufficient to consider a small array. Here we generate an array x of 20 random numbers from 0 to 1:

```
In [1]:  x=np.random.random(20)

In [2]: x
Out[2]:
array([0.90872018, 0.50268918, 0.56813294, 0.87005875, 0.4877182 ,
       0.29671119, 0.80359529, 0.33736169, 0.52570809, 0.89248783,
       0.65033269, 0.58342865, 0.52488935, 0.39136942, 0.92260566,
       0.87417851, 0.65245666, 0.44701462, 0.49414769, 0.48217556])
```

We will make a histogram plot, which counts the number of elements of some value. We will break this up into 5 bin. The bins are defined to be evenly separated from the minimum of x to the maximum of x. The minimum here is 0.29671119 and the maximum is 0.92260566. So the bins are marked by a total of 6 points, which are printed by hist() when it is called

```
In [3]: plt.hist(x, bins=5)
Out[3]:
(array([3., 7., 4., 0., 6.]),
 array([0.29671119, 0.42189009, 0.54706898, 0.67224787, 0.79742677,
        0.92260566]),
 <a list of 5 Patch objects>)
```



Finally, what is shown is the number of elements in each bin, which in this case is 3, 7, 4, 0, 6.

Note that 3+7+4+6 =20, as it should!

USEFUL FUNCTIONS

numpy.mean(x) - calculates the mean/average of an array, x

numpy.std(x, ddof=1) - calculates the standard deviation of an array, x

in general numpy.std(x, ddof=n) $= \sqrt{\frac{1}{N-n}\sum_{i=1}^{N}(x_i - \bar{x})^2}$

import matplotlib.pyplot as plt

plt.hist(x, bins=Nbins) - plots a histgram of x separater into Nbins

```
In [1]: import numpy as np
   ...:

In [2]: x=arange(11)

In [3]: x
Out[3]: array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10])

In [4]: np.mean(x)
Out[4]: 5.0

In [5]: np.std(x,ddof=1)
Out[5]: 3.3166247903554
```

PROJECT #3 - DETERMINE Π

You have learn, hopefully, that the area of a circle is $A_c = \pi r^2$, where *r* is its radius. Use random numbers to determine the value of π.

Here are some steps to get you there.

1. Generate random points in the *(x,y)* - plane with $|x|, |y| \leq 1$

2. Next, calculate the distance *(r)*, of any one point from the origin

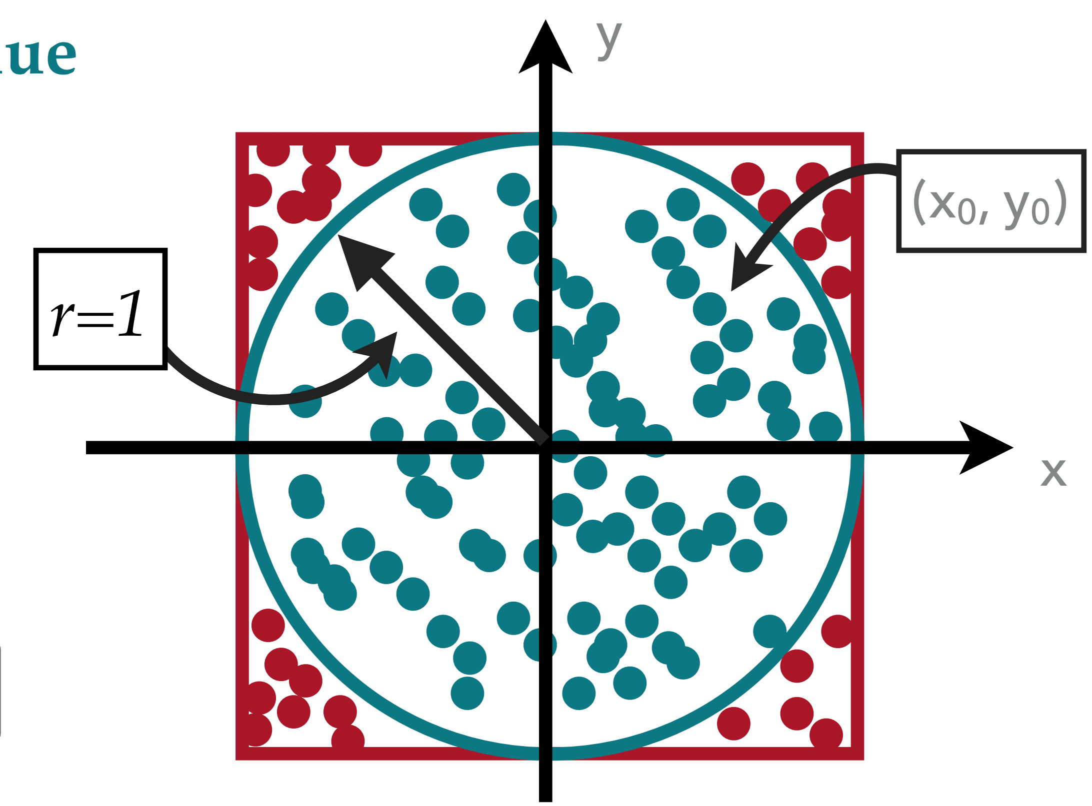
```
x0,y0=np.random.random(Nt),np.random.random(Nt)
x = x0*2.0 - 1.0
y = y0*2.0 - 1.0
r = sqrt( pow(x,2) + pow(y,2) )
```

3. If r>1 then make it **red**, otherwise make them **blue**

4. Then, as we take more and more point, the number of blue point (N_B) divided by the total number of points, will be equal to the ratio of area of the circle and the square shown:

$$\frac{N_B}{N_B + N_R} = \frac{A_c}{A_S} = \frac{\pi r^2}{(2r)^2}$$

solve for π 😎



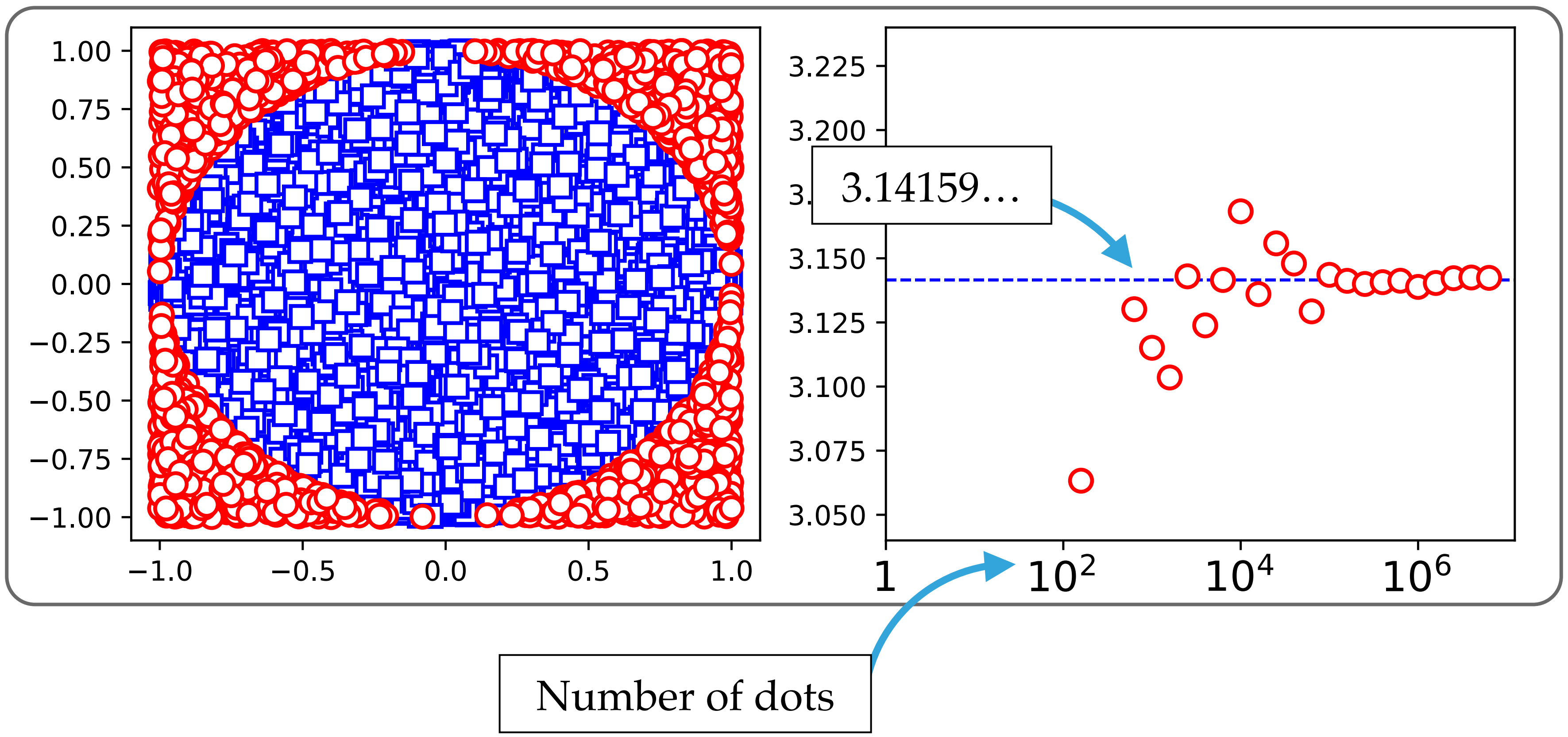
WHAT TO EXPECT

Some of other useful excerpts from my code:

```
"in condition"
condition0 = r <1
xin = np.extract(condition0, x)
yin = np.extract(condition0, y)

"out condition"
condition1 = r >=1
xout = np.extract(condition1, x)
yout = np.extract(condition1, y)
```

```
plt.subplot(221)
plt.errorbar(xin,yin,markersize=8,fmt='s',color='b',mfc='white',mec='b', elinewidth=2, capsize=4, mew=1.4)
plt.errorbar(xout,yout,markersize=8,fmt='o',color='r',mfc='white',mec='r', elinewidth=2, capsize=4, mew=1.4)
```

After some more work…



PROJECT #3 - EXTRA CHALLENGES

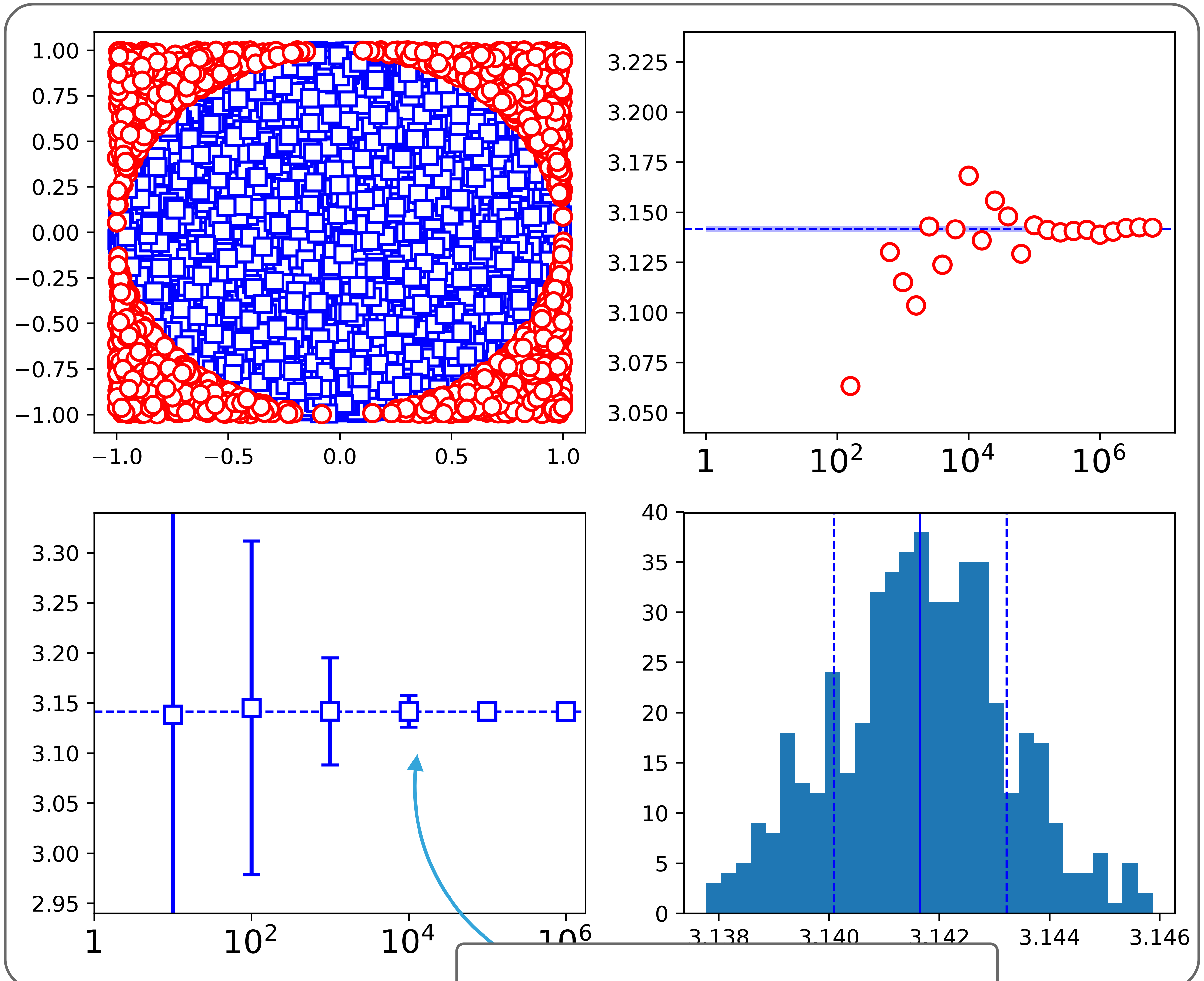
1) Estimate the error made as a function of the number of points used

2) Plot the histograms for the largest value consider into 30 bins. Show that the distribution is center about the average value and that the one standard deviation includes most of the points of the distribution.

3) Write your own mean and standard deviation functions, check that you agree with the one in numpy.

4) Extra-extra challenge: write code that will give you π to a desired level of accuracy. Plot how long your code takes as a function of the desired accuracy.

WHAT TO EXPECT



notice the error is decreasing with increasing number of points 😎

USEFUL FUNCTIONS

numpy.mean(x) - calculates the mean/average of an array, x

numpy.std(x, ddof=1) - calculates the standard deviation of an array, x

in general numpy.std(x, ddof=n) $= \sqrt{\frac{1}{N-n}\sum_{i=1}^{N}(x_i - \bar{x})^2}$

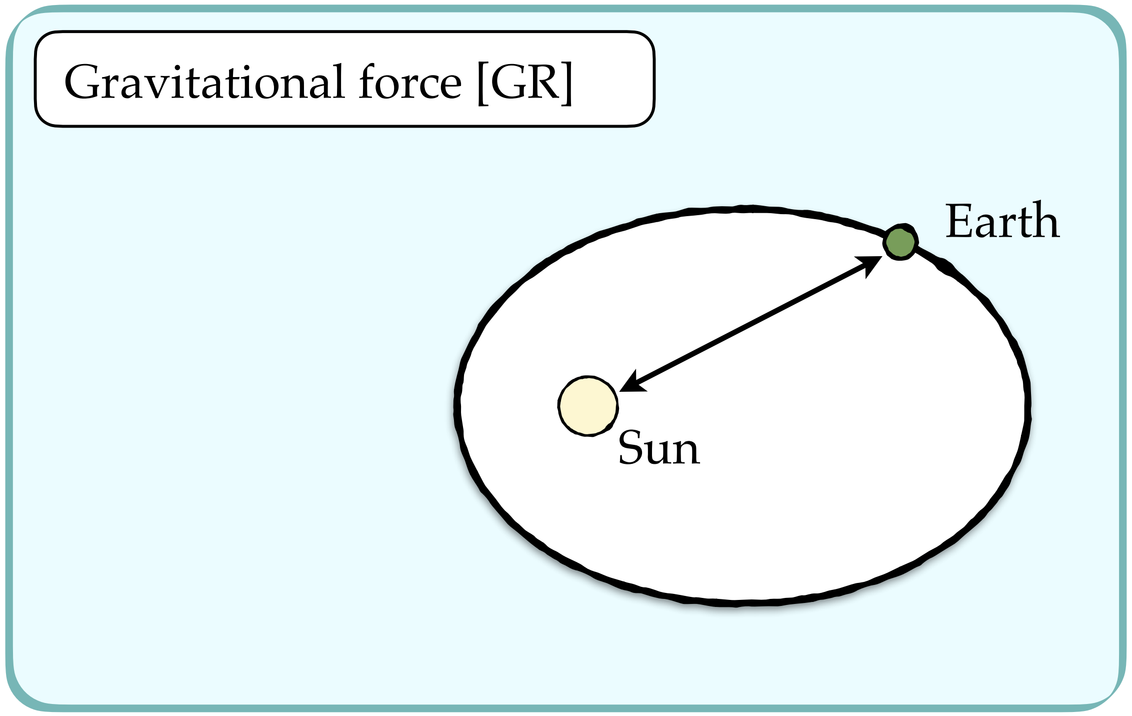
import matplotlib.pyplot as plt

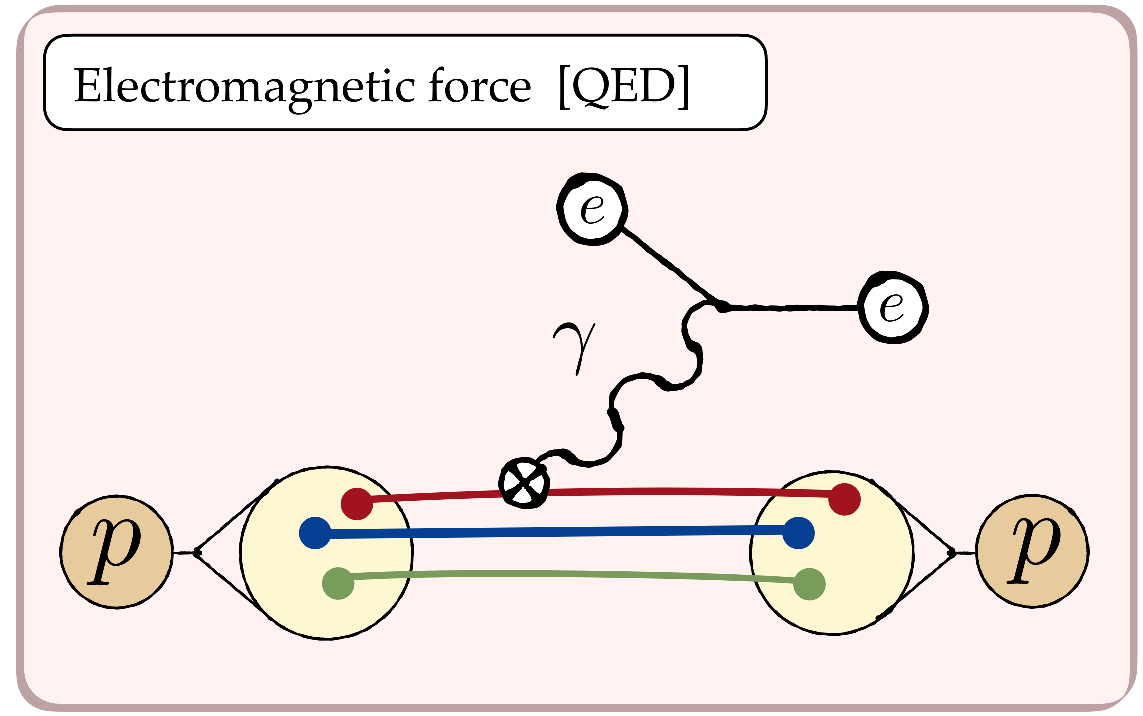
plt.hist(x, bins=Nbins) - plots a histgram of x separater into Nbins

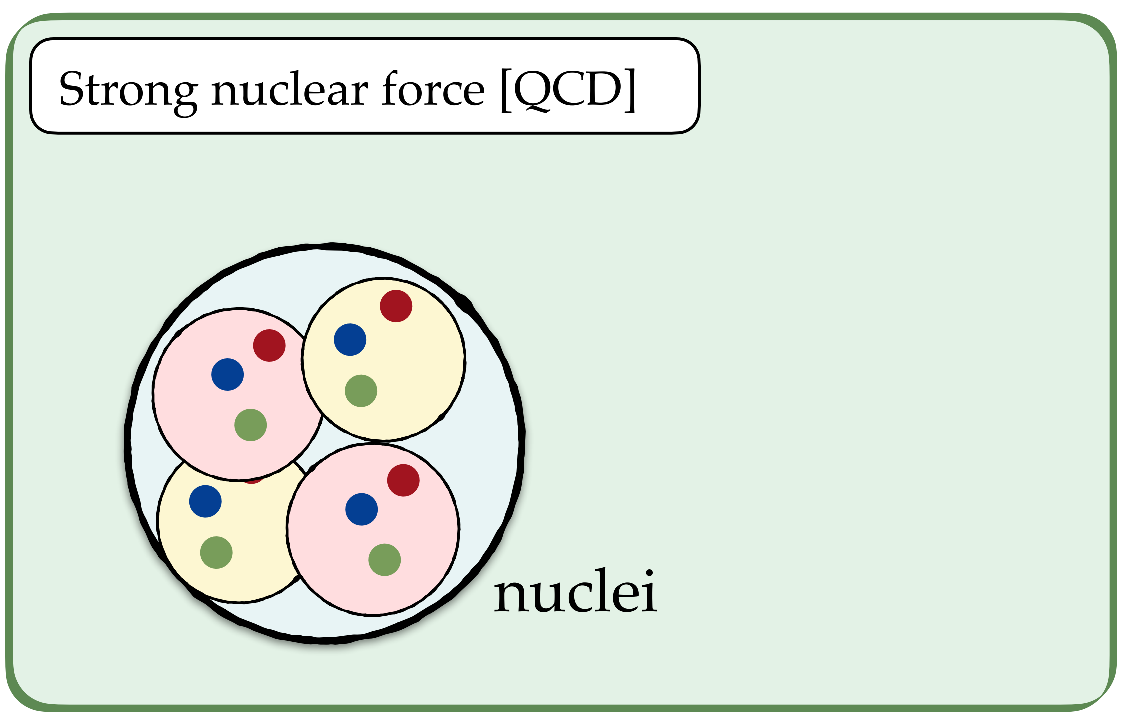
```
In [1]: import numpy as np
   ...:

In [2]: x=arange(11)

In [3]: x
Out[3]: array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10])

In [4]: np.mean(x)
Out[4]: 5.0

In [5]: np.std(x,ddof=1)
Out[5]: 3.3166247903554
```

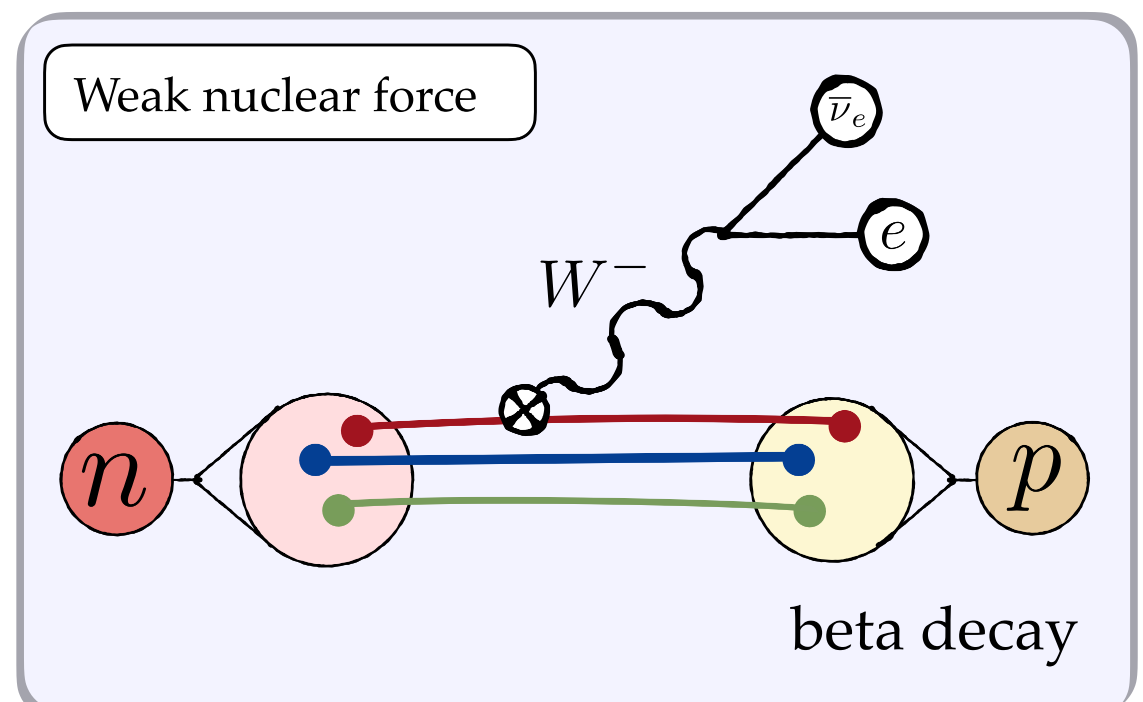

questions? 🐶

THE FORCES OF NATURE

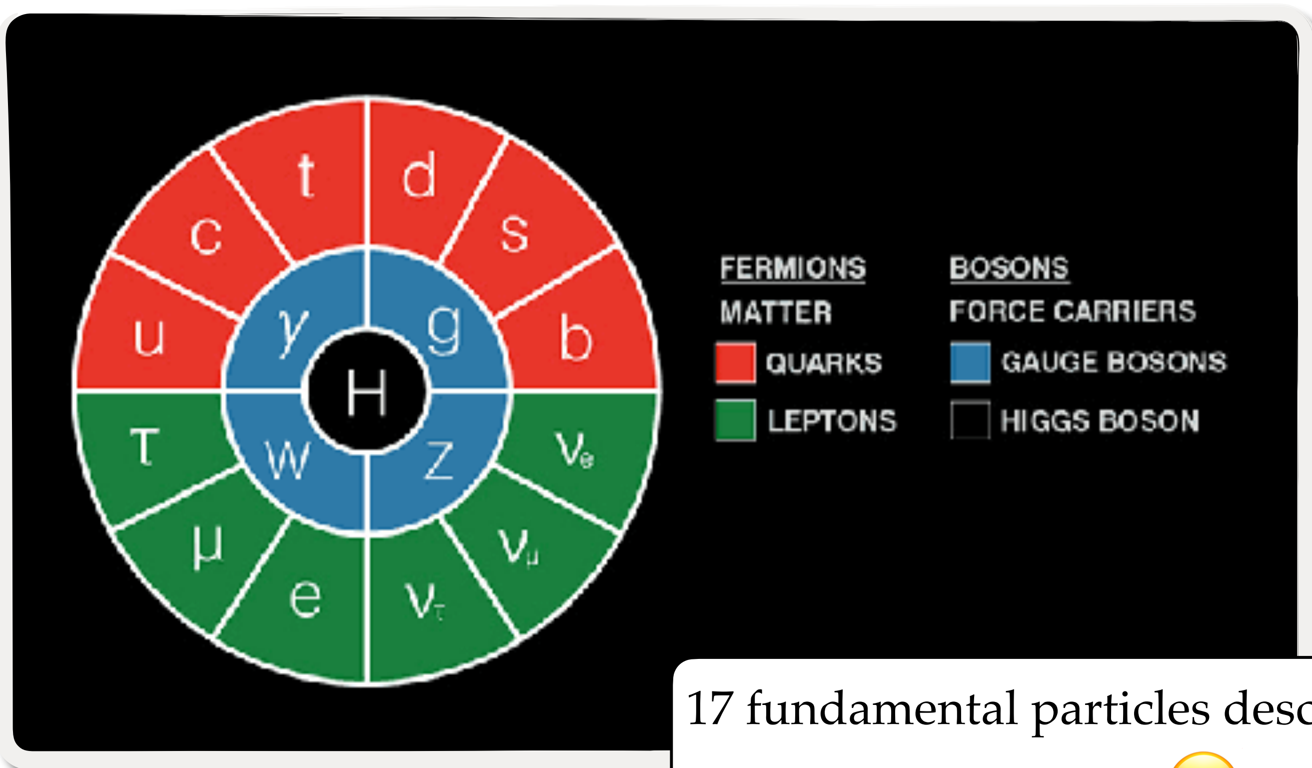






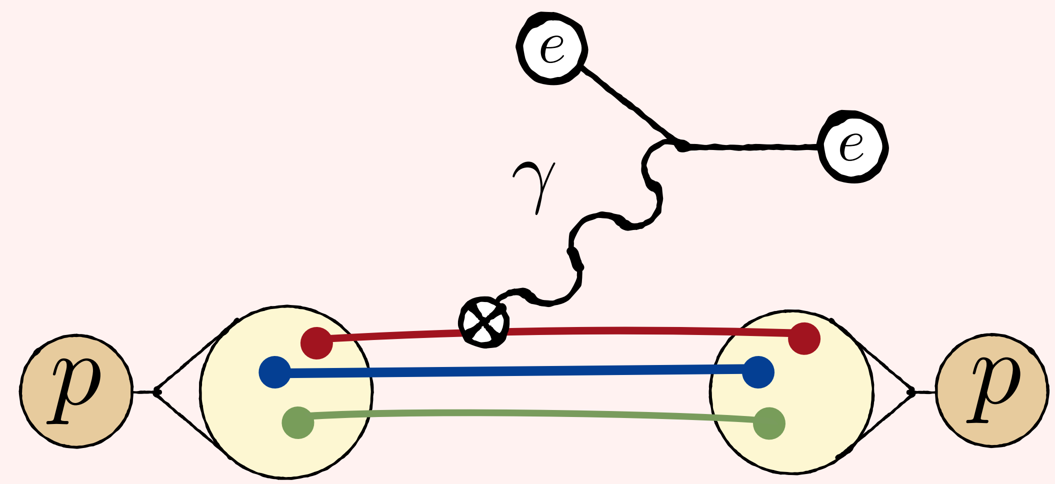


THE STANDARD MODEL

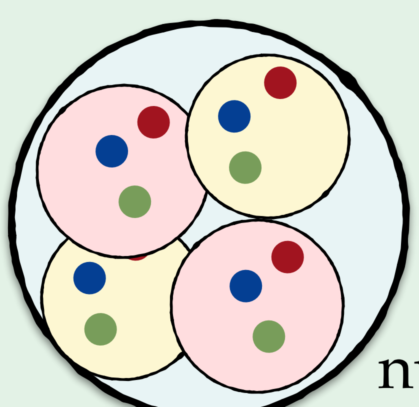


17 fundamental particles describe essentially everything 🤓

Electromagnetic force [QED]

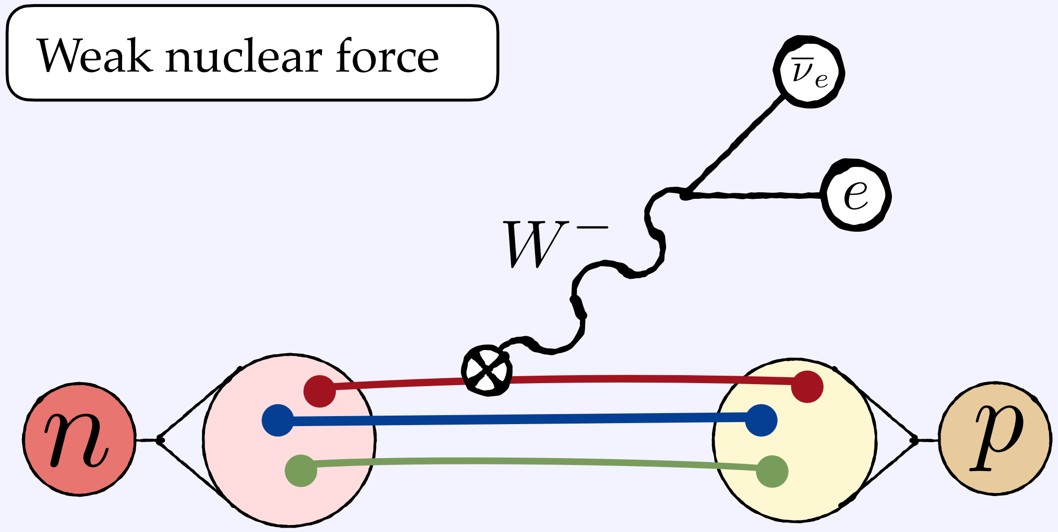


Strong nuclear force [QCD]

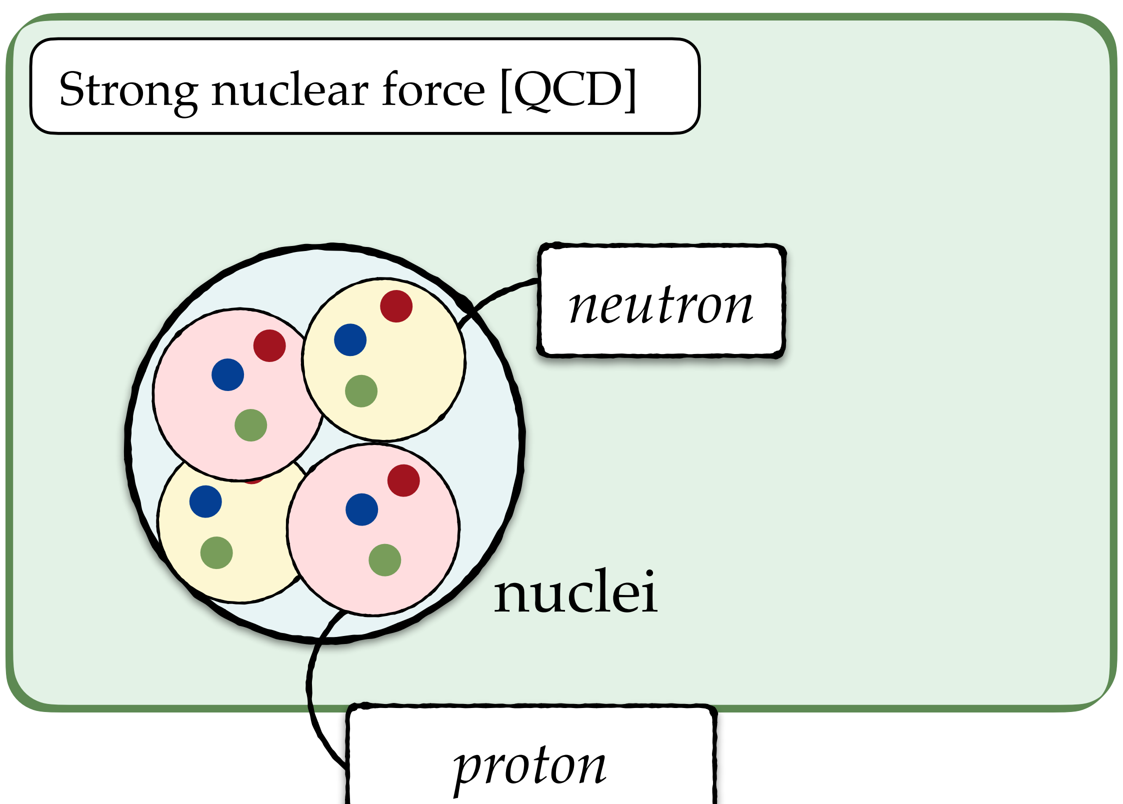


nuclei

Weak nuclear force



beta decay



atomic nuclei are composed of **protons** and **neutrons**

example: ^{12}C, the basis of life, is made of 6 protons and 6 neutrons

these interact and are bound together by the strong nuclear force

the proton and neutron are made of quarks and gluons, which also interact via the strong nuclear force

nuclear physics = physics of the strong nuclear force

NEUTRONS ARE UNSTABLE

Orginos

Monahan

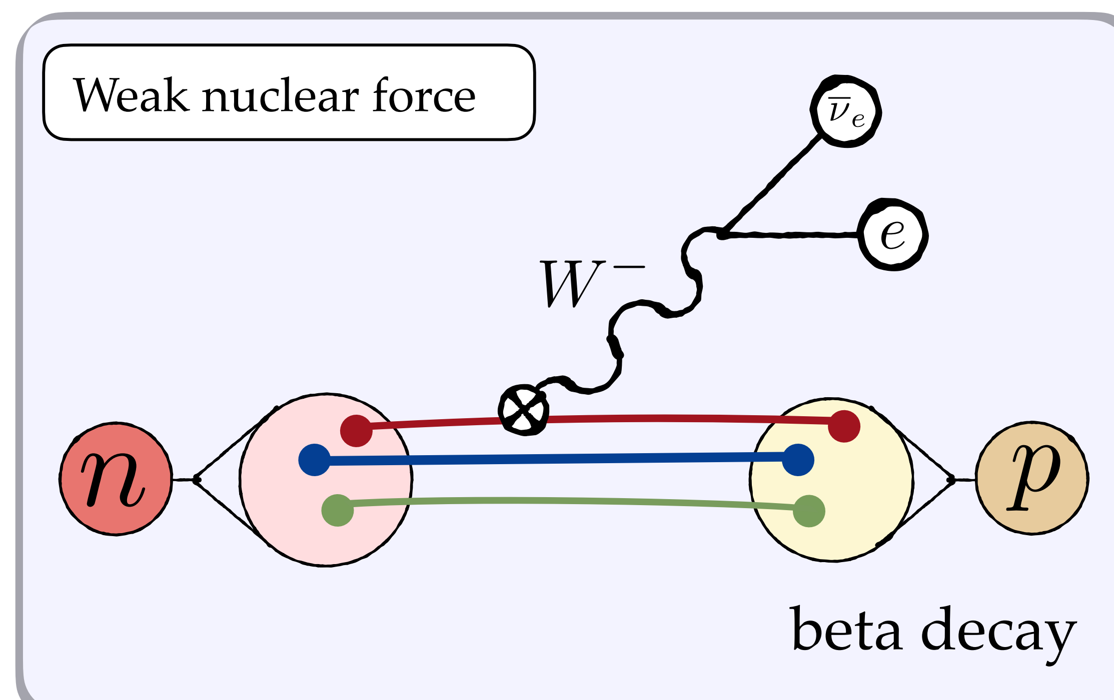
nature
International journal of science

Letter | Published: 30 May 2018

A per-cent-level determination of the nucleon axial coupling from quantum chromodynamics

C. C. Chang, A. N. Nicholson, E. Rinaldi, E. Berkowitz, N. Garron, D. A. Brantley, H. Monge-Camacho, C. J. Monahan, C. Bouchard, M. A. Clark, B. Joó, T. Kurth, K. Orginos, P. Vranas & A. Walker-Loud

Nature **558**, 91–94 (2018) | Download Citation

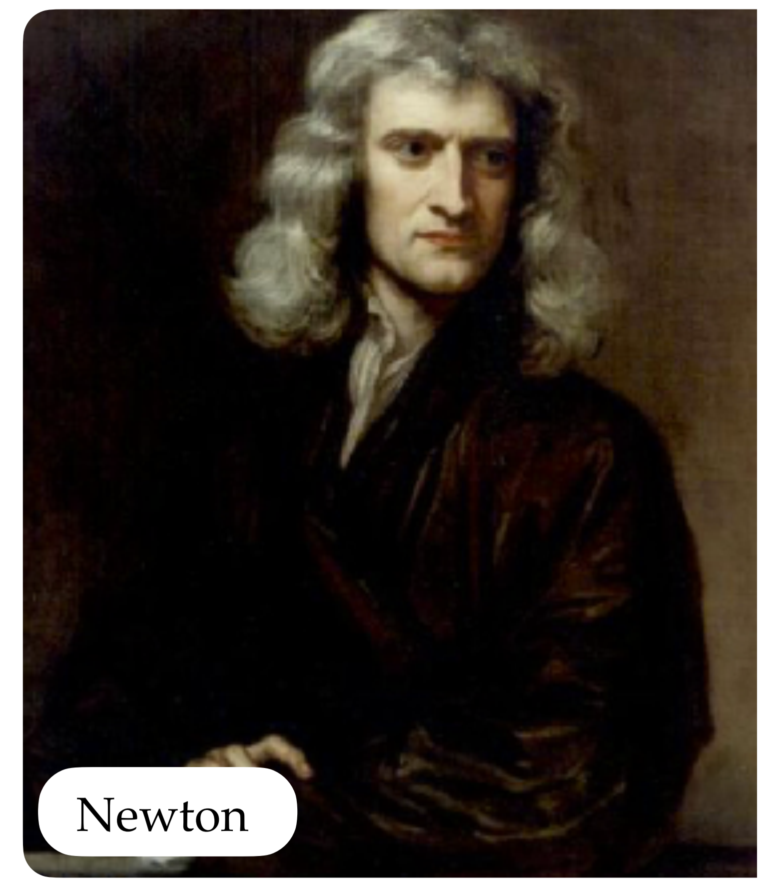


MOMENTUM

If a particle is moving at some speed v and it has mass m, it has **momentum**

$$v = \frac{\Delta x}{\Delta t}$$

In Newtonian mechanics: $p = mv$



Newton

Einstein taught us that this was a low-energy approximation of

$$\boxed{p = \frac{mv}{\sqrt{1 - \frac{v^2}{c^2}}} \longrightarrow \frac{mv}{\sqrt{1 - v^2}}}$$

c is the speed of light, and we will use where it's equal to 1 🤓

If the particle is moving, then it also carries energy.

Einstein taught us that the energy of a particle is related to its momentum via

$$\boxed{E = \sqrt{m^2 + p^2}}$$

energy

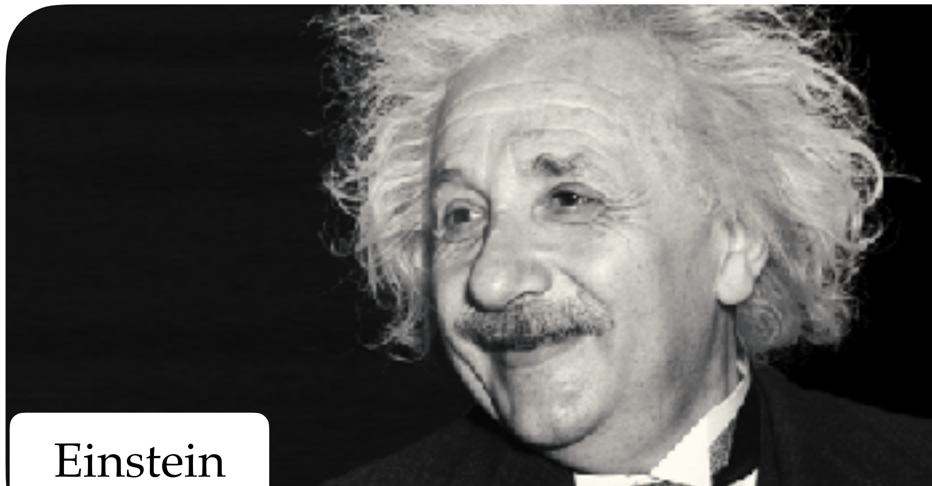
mass

momentum

You are probably more familiar with: $E = mc^2$

this holds when the momentum (p) is equal to 0.

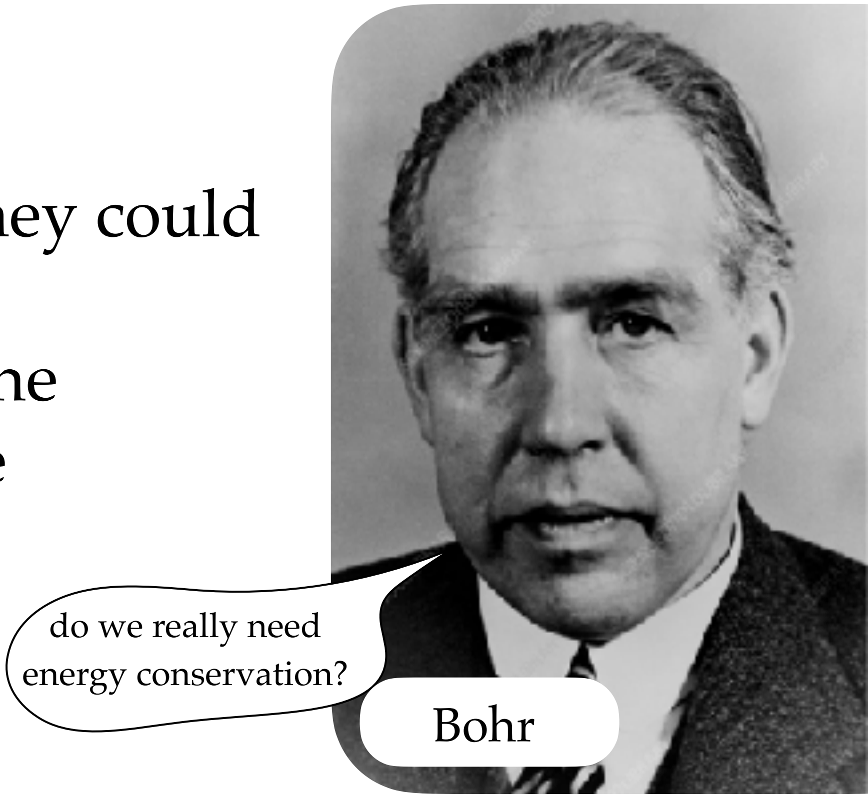
also, remember we are setting c=1 🤓



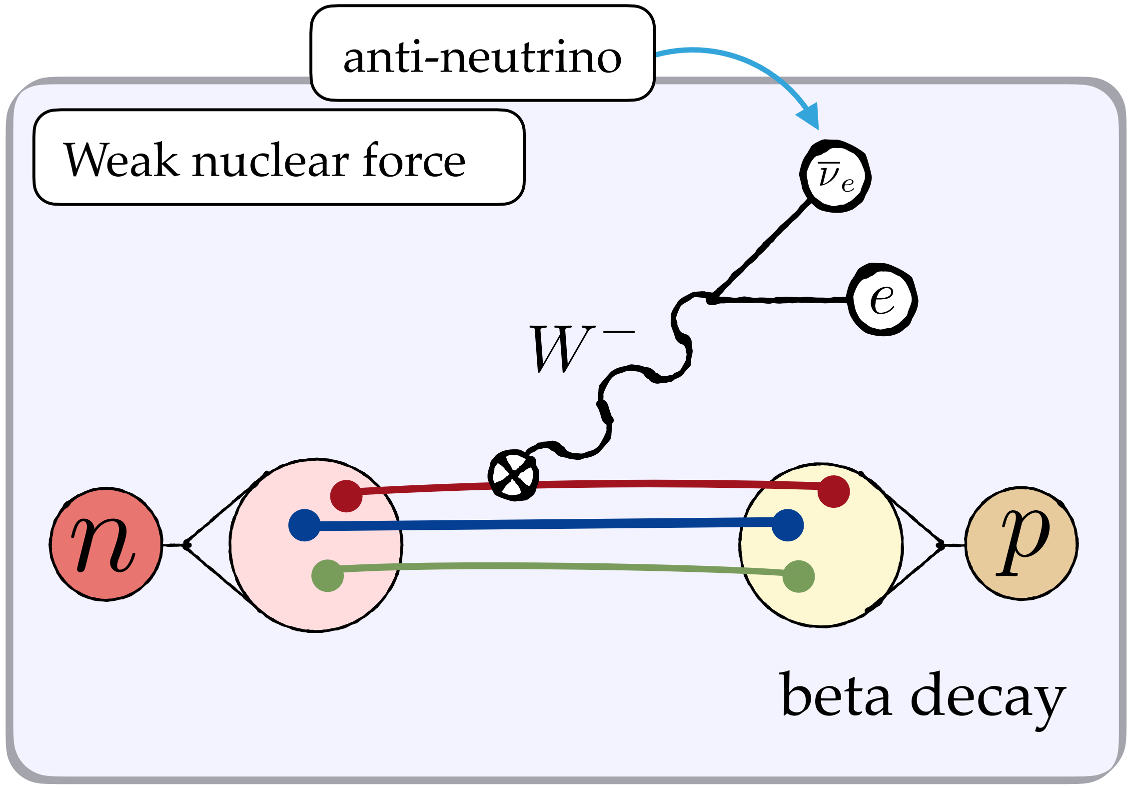
Einstein

So far, we see that both energy and momentum is conserved in a reaction in nature.

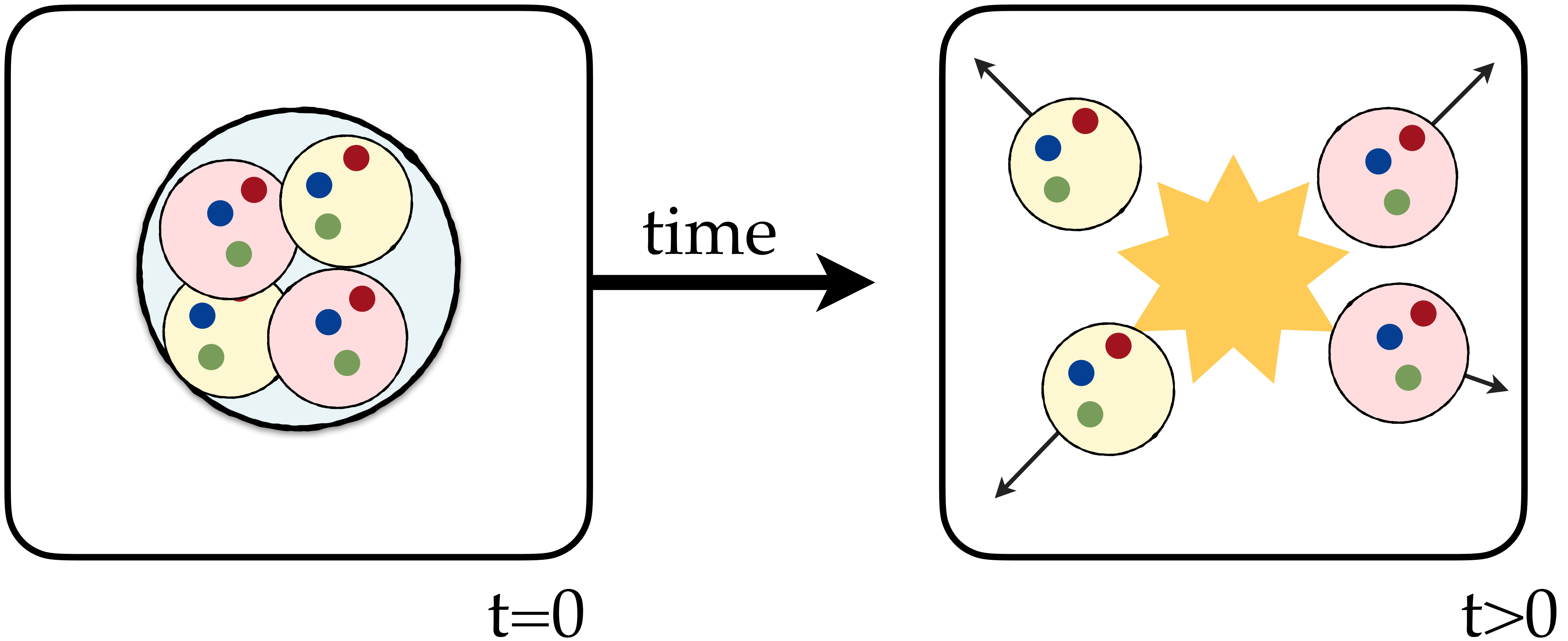
In fact, when physicists realized that neutrons decay, and they could not account for all the energy lost.
Before Wolfgang Pauli correctly proposed the existence of the **neutrinos**, many proposed that energy might not always be conserved! 😬



Neutrinos have no charge, very tiny masses, and barely interact with all other matter.



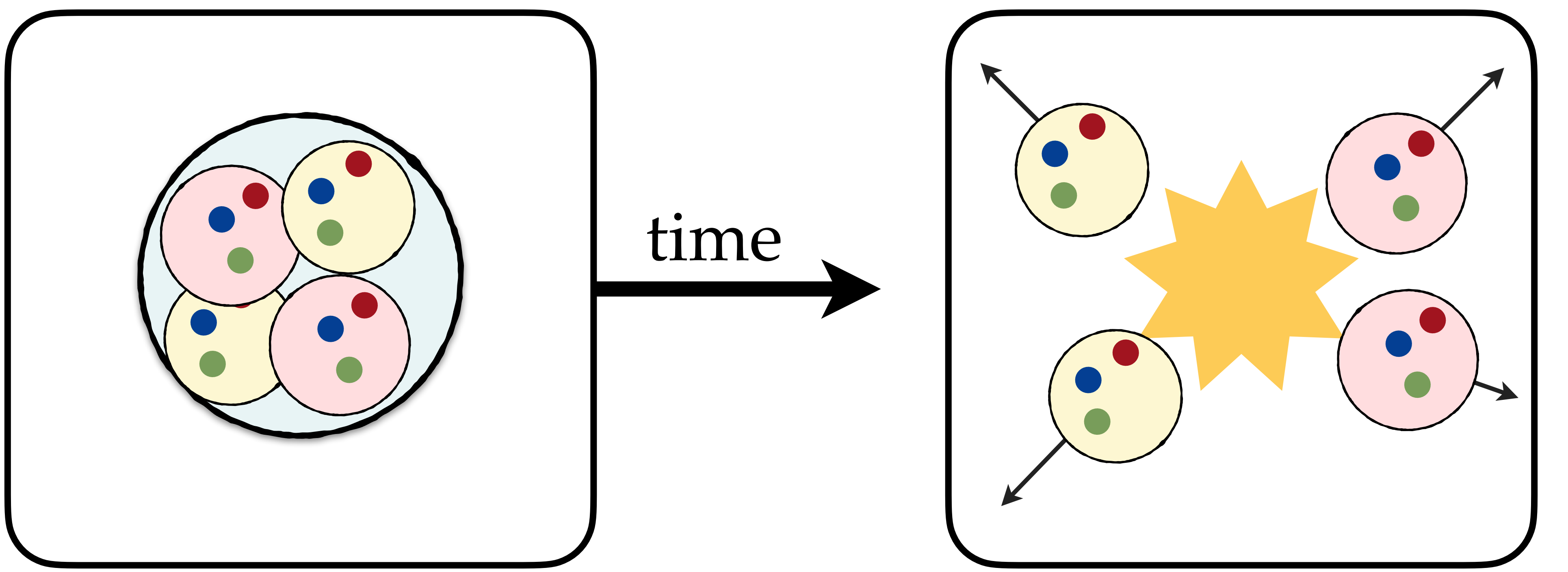
If a systems of particles with energy E_i and momentum P_i goes through a reaction, and it has a final total energy E_f and momentum P_f



then energy and momentum conservation states:

$$\boxed{\begin{aligned} E_i &= E_f \\ P_i &= P_f \end{aligned}}$$

If a systems of particles with energy E_i and momentum P_i goes through a reaction, and it has a final total energy E_f and momentum P_f



t=0 t>0

then energy and momentum conservation states:

$$\boxed{\begin{aligned} E_i &= E_f \\ P_i &= P_f \end{aligned}}$$

If you find any evidence against this, you would certainly get a Nobel prize 🤓

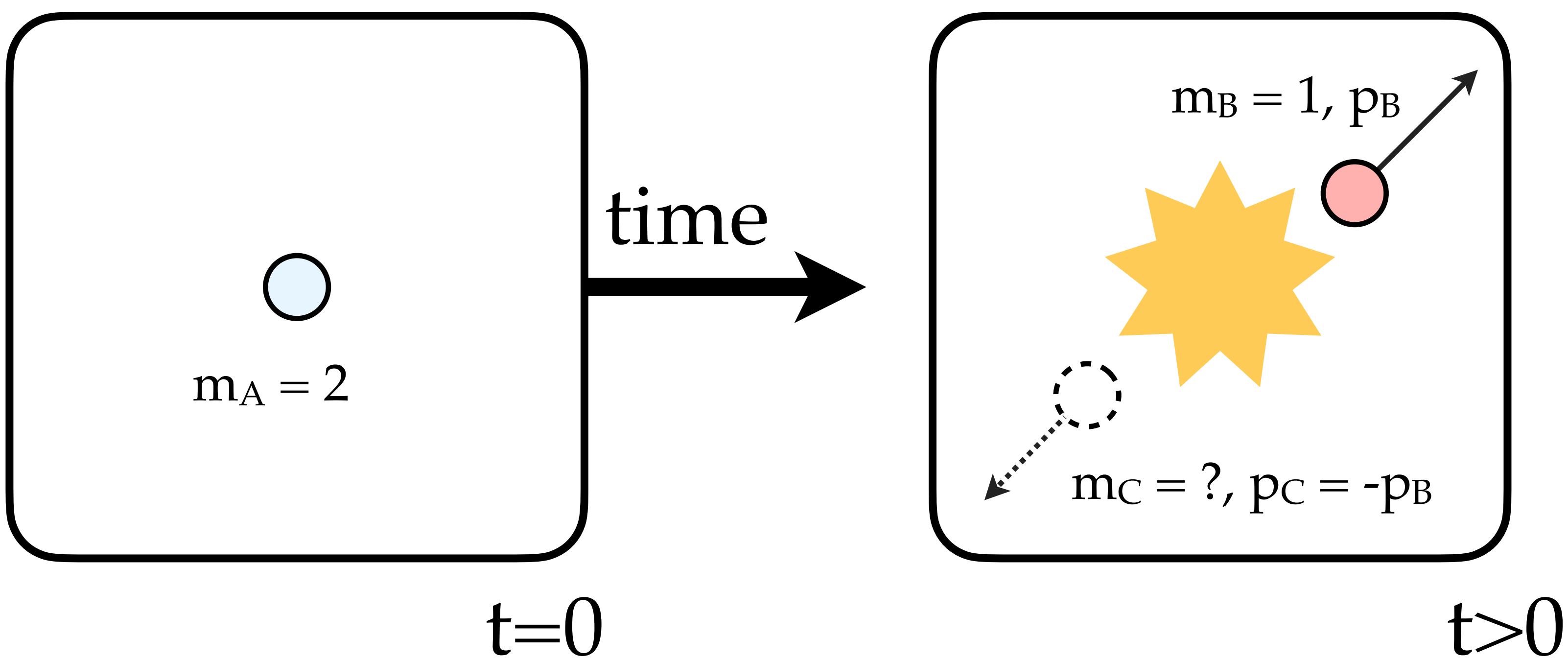
PROJECT #4 - PARTICLE DECAYS TO 2 PARTICLES

Consider a particle with mass $m_A = 2$.

When it is at rest ($P_A = 0$), it decays to two other particles, one particle with mass $m_B = 1$, the other particle you do not know what mass it has, but we will label it as m_C.

Momentum conservation: $p_C = -p_B$

Energy conservation:
$$\begin{aligned} m_A &= \sqrt{m_B^2 + p_B^2} + \sqrt{m_C^2 + p_C^2} \\ &= \sqrt{m_B^2 + p_B^2} + \sqrt{m_C^2 + (-p_B)^2} \\ &= \sqrt{m_B^2 + p_B^2} + \sqrt{m_C^2 + p_B^2} \end{aligned}$$



With a little bit of algebra, we can solve for the missing mass, m_C, as a function of the momentum measured:

$$\begin{aligned}
\sqrt{m_C^2 + p_B^2} &= m_A - \sqrt{m_B^2 + p_B^2} \\
\Rightarrow m_C^2 + p_B^2 &= \left(m_A - \sqrt{m_B^2 + p_B^2}\right)^2 \\
\Rightarrow m_C^2 &= \left(m_A - \sqrt{m_B^2 + p_B^2}\right)^2 - p_B^2 \\
&= m_A^2 - 2m_A\sqrt{m_B^2 + p_B^2} + m_B^2 \\
\Rightarrow m_C &= \sqrt{m_A^2 - 2m_A\sqrt{m_B^2 + p_B^2} + m_B^2}
\end{aligned}$$

Determine the mean value and standard deviation of the m_C, given the measure values of p_B.

These are in **Dropbox:/P4P_2020/project# 3/pBs_2body_decay.txt**

PROJECT #4 - PARTICLE DECAYS TO 2 PARTICLES

Open **Dropbox:/P4P_2020/project# 3/pBs_2body_decay.txt**

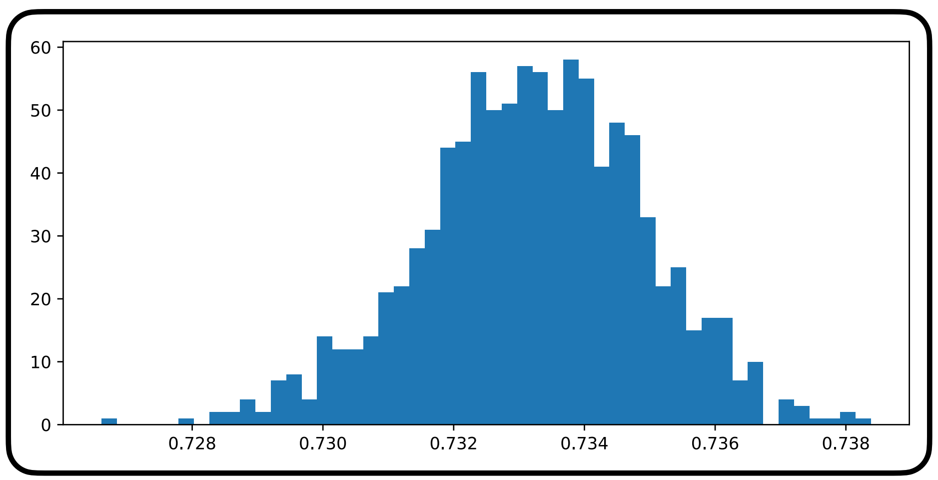
```
7.338821980534157019e-01
7.341499939912857764e-01
7.321368844320390590e-01
7.327569761651976377e-01
7.331937868928702917e-01
7.348416728855912128e-01
7.335555555424789764e-01
7.328387254467044754e-01
7.321660883826330846e-01
7.345873579284167842e-01
7.351338392574563452e-01
7.357170128743750759e-01
7.340901234594668923e-01
7.326525649529516127e-01
7.317402999142356146e-01
7.319256451883046077e-01
7.324645934796907287e-01
7.371238937888494602e-01
7.313561078590937736e-01
7.347029727679132982e-01
7.360628356914329373e-01
7.310879408601412832e-01
7.323464890388832194e-01
7.330003420112729540e-01
7.336390311806597619e-01
7.299171911865390827e-01
7.333730337370883978e-01
7.321668025523996315e-01
7.327239217872473231e-01
7.330904601820998900e-01
7.311308635636277753e-01
7.333209736048642569e-01
7.336773696634950603e-01
```

1,000 measurements of the momentum.

to load them into look at **~/P4P_2020/project# 3/momenta_hist.py**

```python
import matplotlib.pyplot as plt
import numpy as np

filename = 'pBs_2body_decay.txt'
q0s = np.loadtxt(filename)
print("q0s = ", q0s)
print("type(q0s) = ", type(q0s))
print("shape(q0s) = ", np.shape(q0s))
Nbins=50
plt.hist(q0s, bins=Nbins)
```

you should see an output of this form




```
 0.73167441 0.73436523 0.73417144 0.73412524 0.73284964 0.73119216
 0.73183324 0.73326519 0.73296209 0.73322869 0.73241116 0.73231934
 0.73162328 0.73581308 0.7331647  0.73555658 0.73106951 0.73321146
 0.7329967  0.73146933 0.73467377 0.73017257 0.73585629 0.73369258
 0.73541336 0.73149528 0.73480574 0.73142763 0.728334   0.73614393
 0.73259801 0.7340329  0.73409469 0.73340496 0.733278   0.7318795
 0.73008713 0.7312059  0.73583695 0.73435047 0.73231069 0.73357277
 0.73492108 0.73331342 0.73273318 0.73157123 0.73521194 0.73060941
 0.73305422 0.73392535 0.73132553 0.73600348 0.731705   0.73297316
 0.73450881 0.73256617 0.73386239 0.73178461 0.73559862 0.73165861
 0.73478557 0.7336575  0.73489538 0.73161492 0.73356875 0.73340258
 0.73263223 0.73082317 0.73474089 0.73539007 0.73371127 0.73352628
 0.73009778 0.73402765 0.7339779  0.73472165 0.73140423 0.73485026
 0.73640759 0.73589802 0.73183026 0.73438454 0.73089128 0.73288993
 0.73518338 0.73543439 0.73810418 0.7324432 ]
type(q0s) =  <class 'numpy.ndarray'>
shape(q0s) =  (1000,)
```

PROJECT #4 - WHAT TO EXPECT

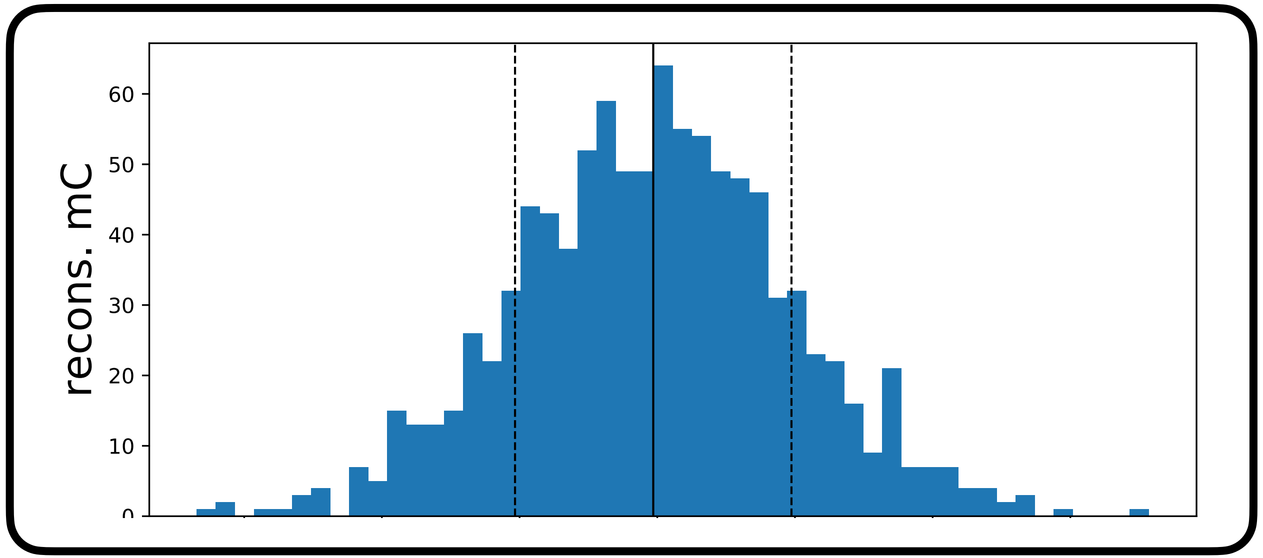
Use the p_B data and the equation below, to determine the mean value and standard deviation of m_C

$$m_C = \sqrt{m_A^2 - 2m_A\sqrt{m_B^2 + p_B^2} + m_B^2}$$

tips: define a function *missing_mass()* that returns the value of mC, given the other variable. You can then make a distribution of the mass, evaluate the mean and standard deviation just as before. Here is a glimpse into my code:

```
mCfs = missing_mass(mA, mB, q0s )
print("mCfs = ",np.mean(mCfs),np.std(mCfs,ddof=1))
plt.hist(mCfs, bins=Nbins)
plt.axvline(x=np.mean(mCfs),color='k',linewidth=1)
plt.axvline(x=np.mean(mCfs)+np.std(mCfs,ddof=1),color='k',linewidth=1,linestyle='dashed')
plt.axvline(x=np.mean(mCfs)-np.std(mCfs,ddof=1),color='k',linewidth=1,linestyle='dashed')
```

output



USEFUL FUNCTIONS

numpy.mean(x) - calculates the mean/average of an array, x

numpy.std(x, ddof=1) - calculates the standard deviation of an array, x

in general numpy.std(x, ddof=n)

import matplotlib.pyplot as plt

plt.hist(x, bins=Nbins) - plots a histgram of x separater into Nbins

SUMMARY - EQUATIONS USED

Statistics:

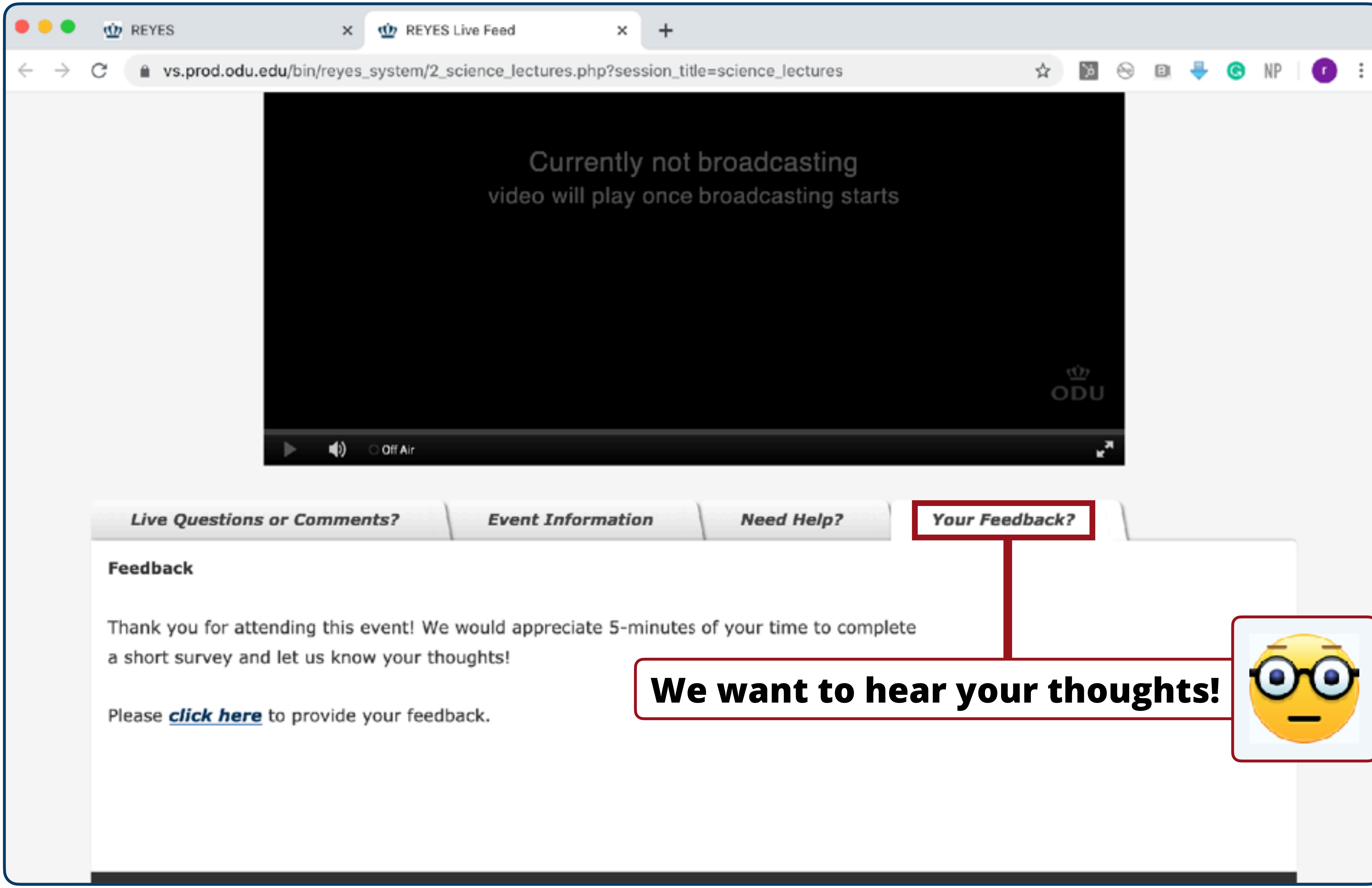
Mean: $\bar{x} = \frac{\sum_{i=1}^{N} x_i}{N}$ Standard deviation: $\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{x})^2}$

Physics:

Energy and momentum conservation: $E_i = E_f, \qquad P_i = P_f$

Relativistic energy of a single particle: $E = \sqrt{m^2 + p^2}$

VIDEO STREAMS // VS.PROD.ODU.EDU/BIN/REYES_SYSTEM/



VIDEO STREAMS // VS.PROD.ODU.EDU/BIN/REYES_SYSTEM/