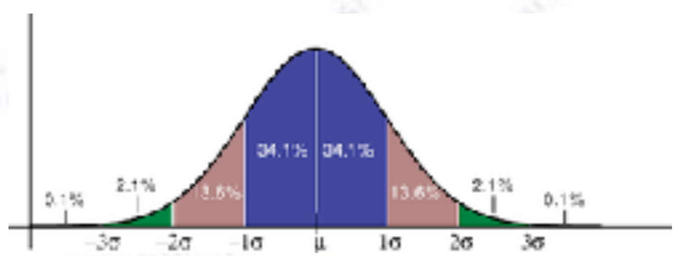
Applied Statistics

Measuring the length of a Table...



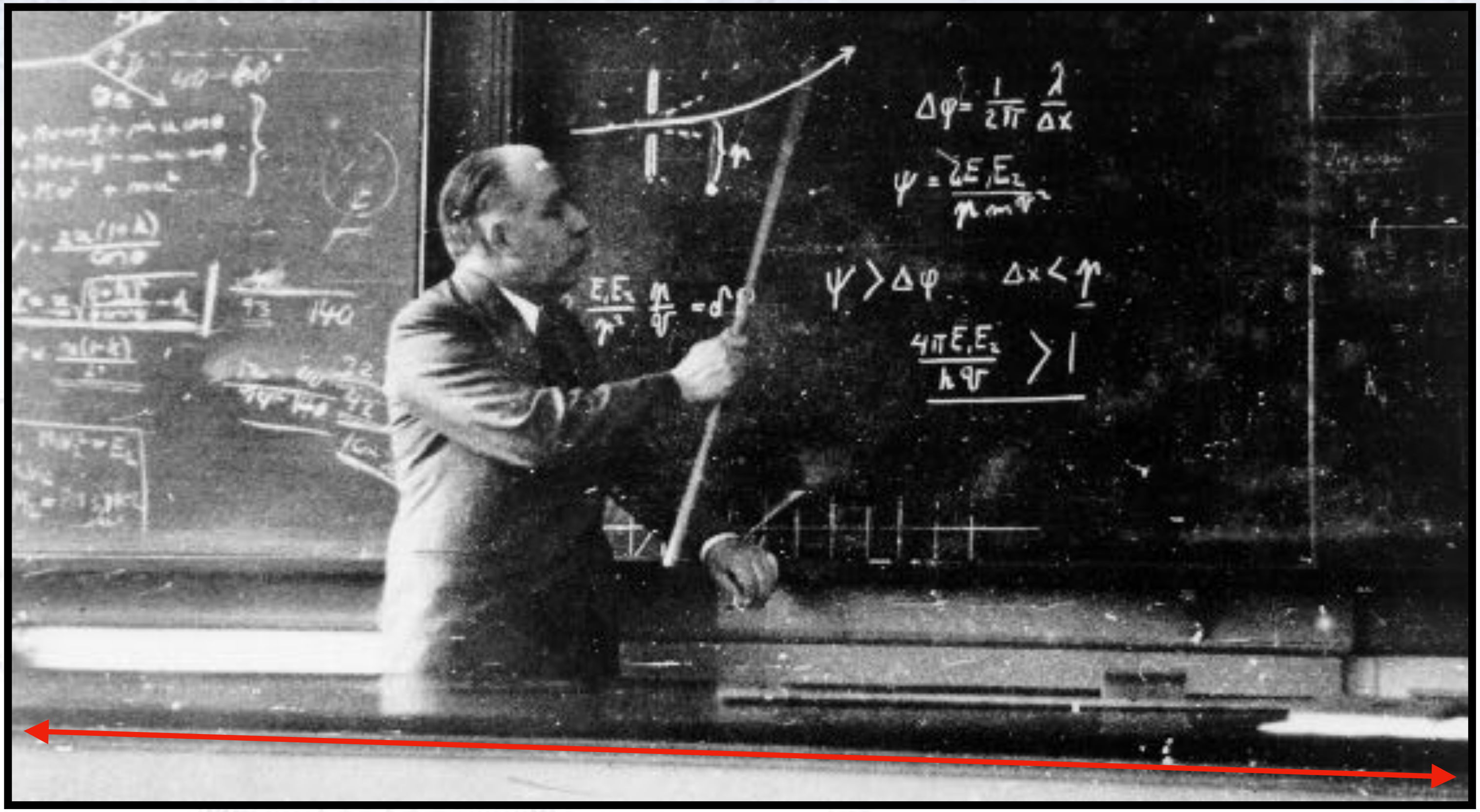



Troels C. Petersen (NBI)



"Statistics is merely a quantisation of common sense"

The table in auditorium A



"Everything is vague to a degree you do not realise till you have tried to make it precise."

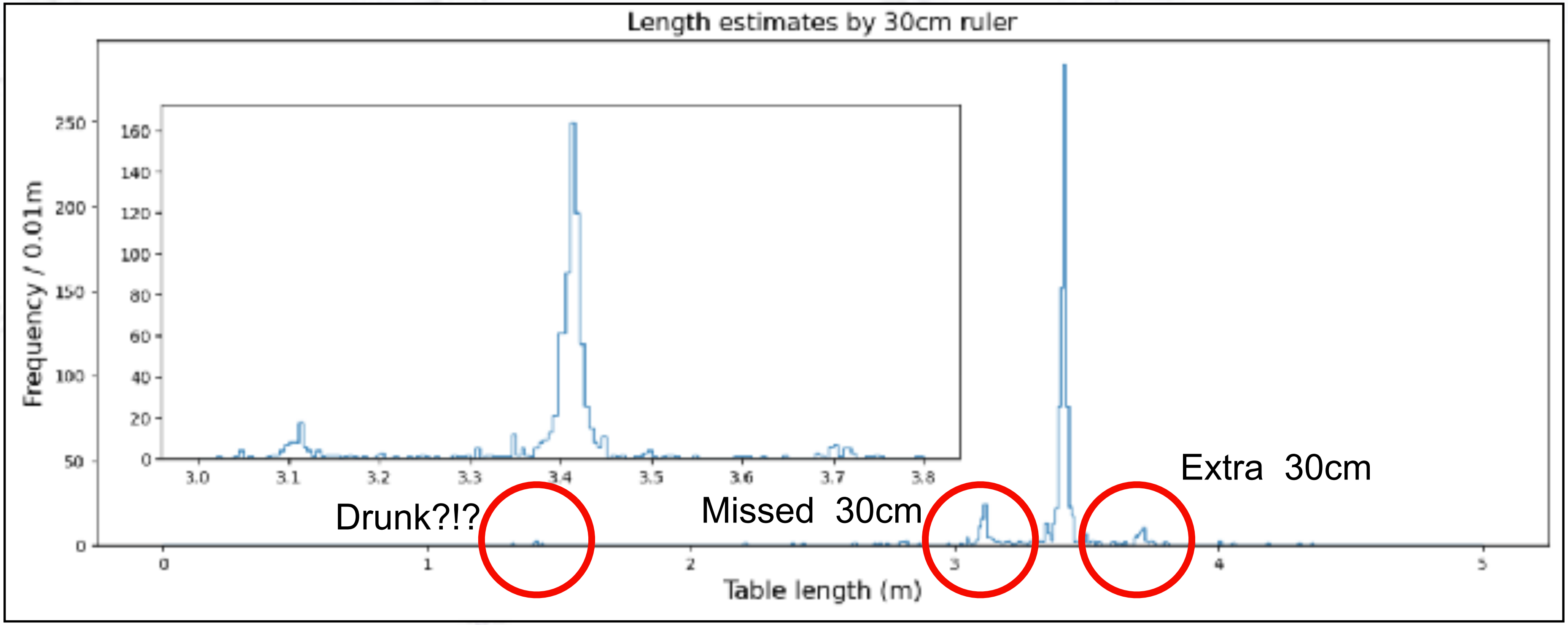
[Bertrand Russell, 1872-1970]

My analysis (2009-2024 data)

The table measurement data

The initial dataset contains (valid measurements):

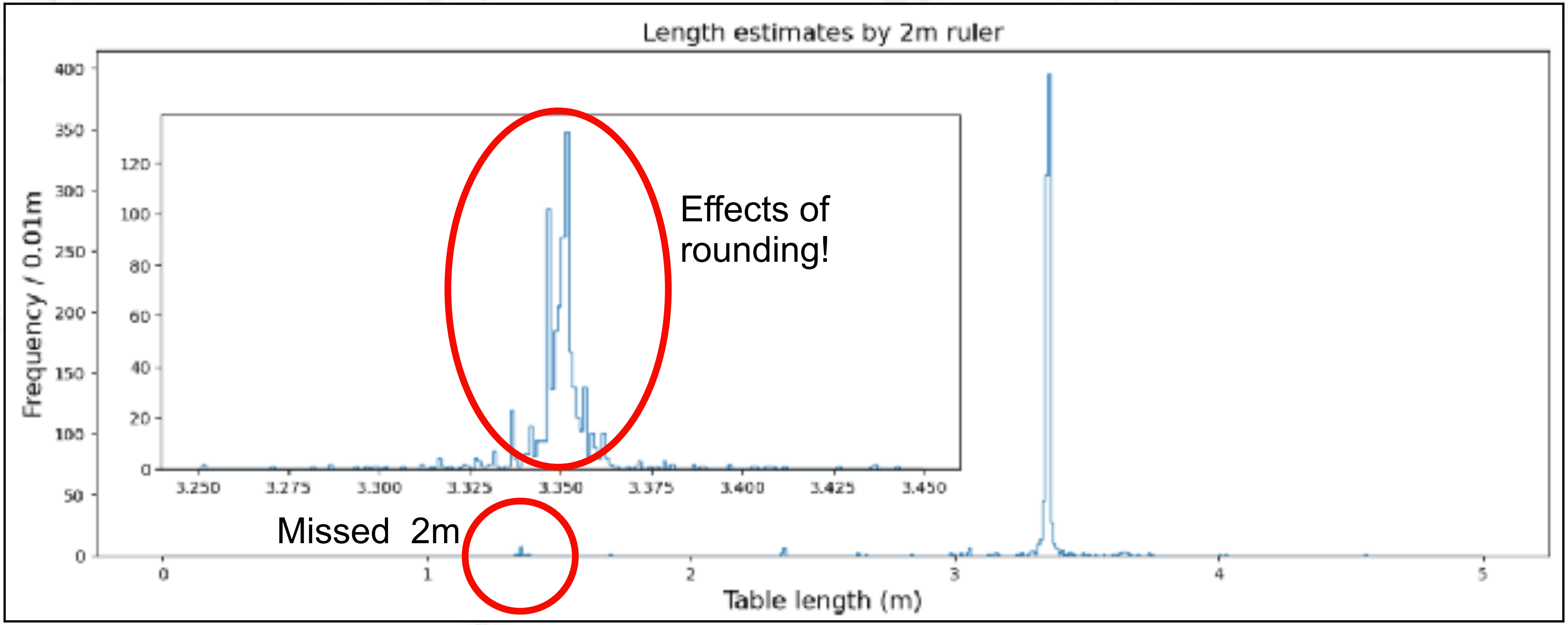
- 30cm measurements: **913** Range: [0.0, 5.0] m
- 2m measurements: **911** Range: [0.0, 5.0] m



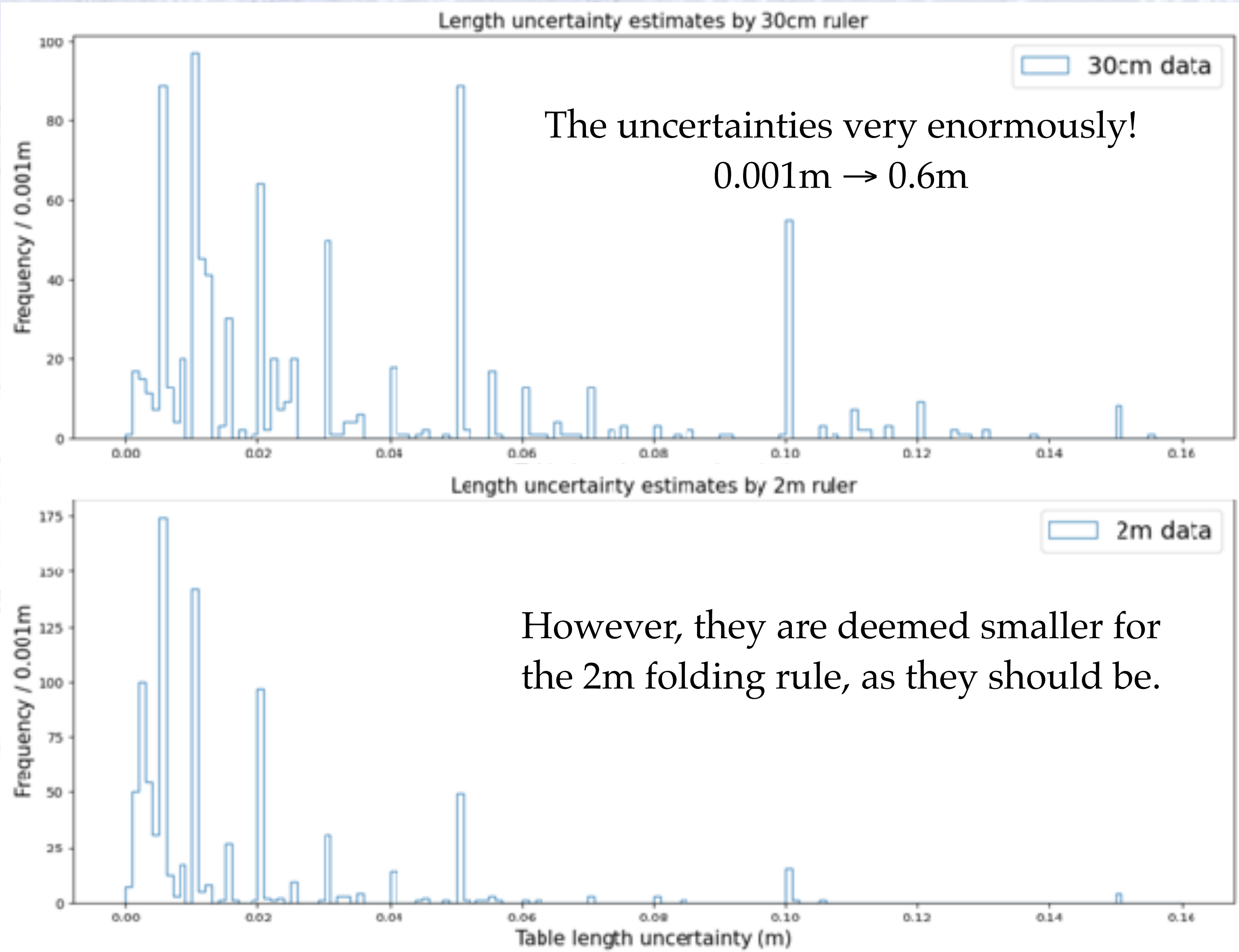
The table measurement data

The initial dataset contains (valid measurements):

- 30cm measurements: **913** Range: [0.0, 5.0] m
- 2m measurements: **911** Range: [0.0, 5.0] m



Uncertainties



The uncertainties very enormously!
0.001m → 0.6m

However, they are deemed smaller for the 2m folding rule, as they should be.

Raw ("Naive") results

<u>30cm:</u>

Mean = 3.3750 ± 0.0077 m

Std. = 0.23 m (N = 913)

<u>2m:</u>

Mean = 3.3155 ± 0.0086 m

Std. = 0.26 m (N = 911)

From the Std. values alone, it is clear that something is terribly wrong, which is also why the uncertainties on the mean are almost a centimeter!

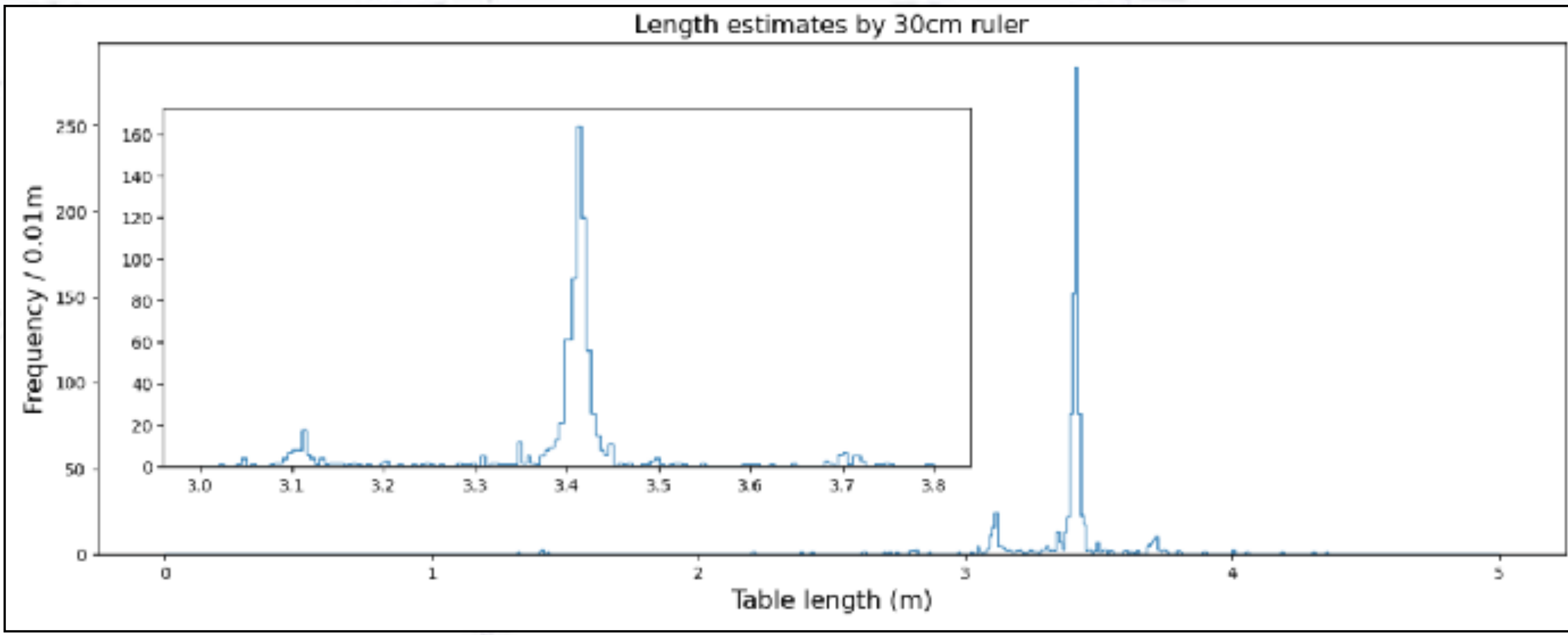
Unweighted analysis

Include offsets?

There are some clear and understandable mis-measurements.

Should one correct and include these? Or reject the values?

Depends on situation, but decide without seeing the final result.

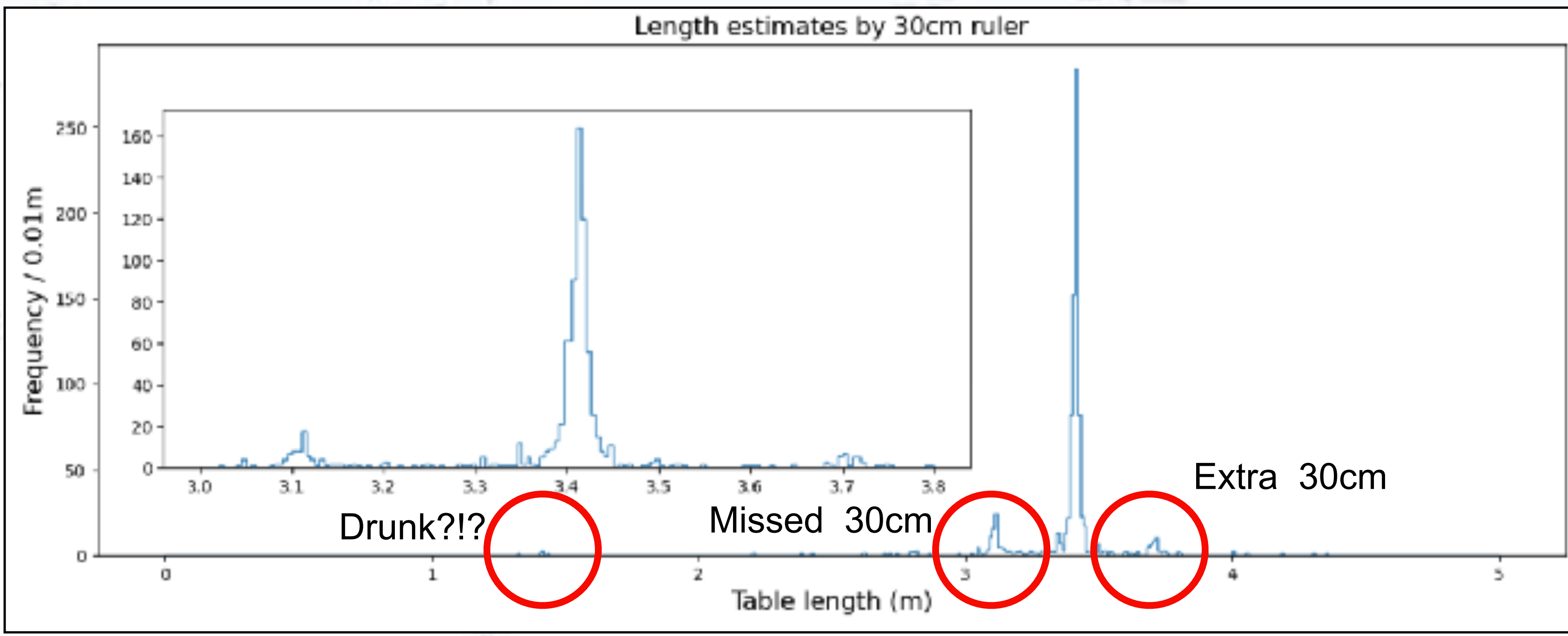


Include offsets?

There are some clear and understandable mis-measurements.

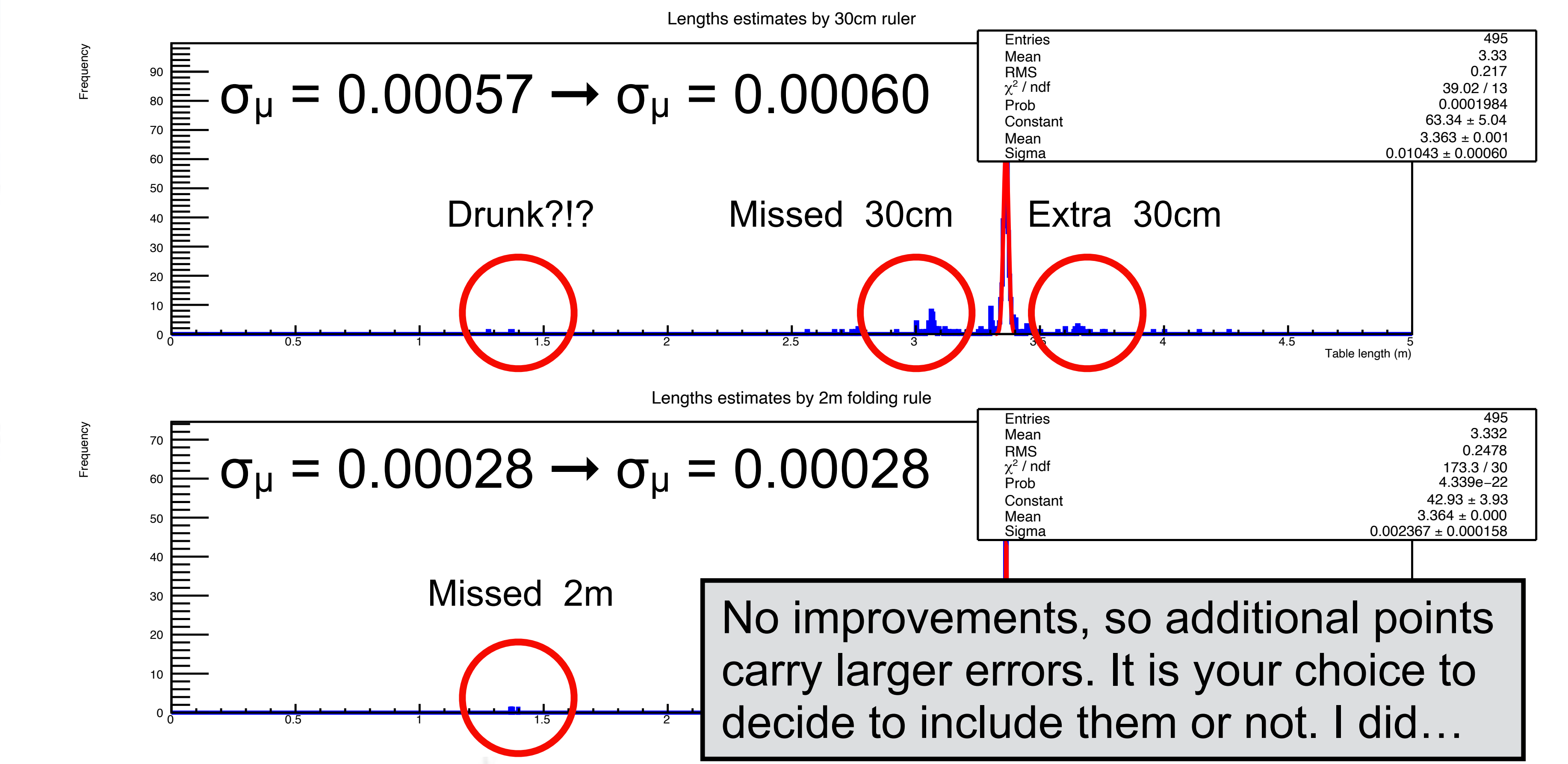
Should one correct and include these? Or reject the values?

Depends on situation, but decide without seeing the final result.



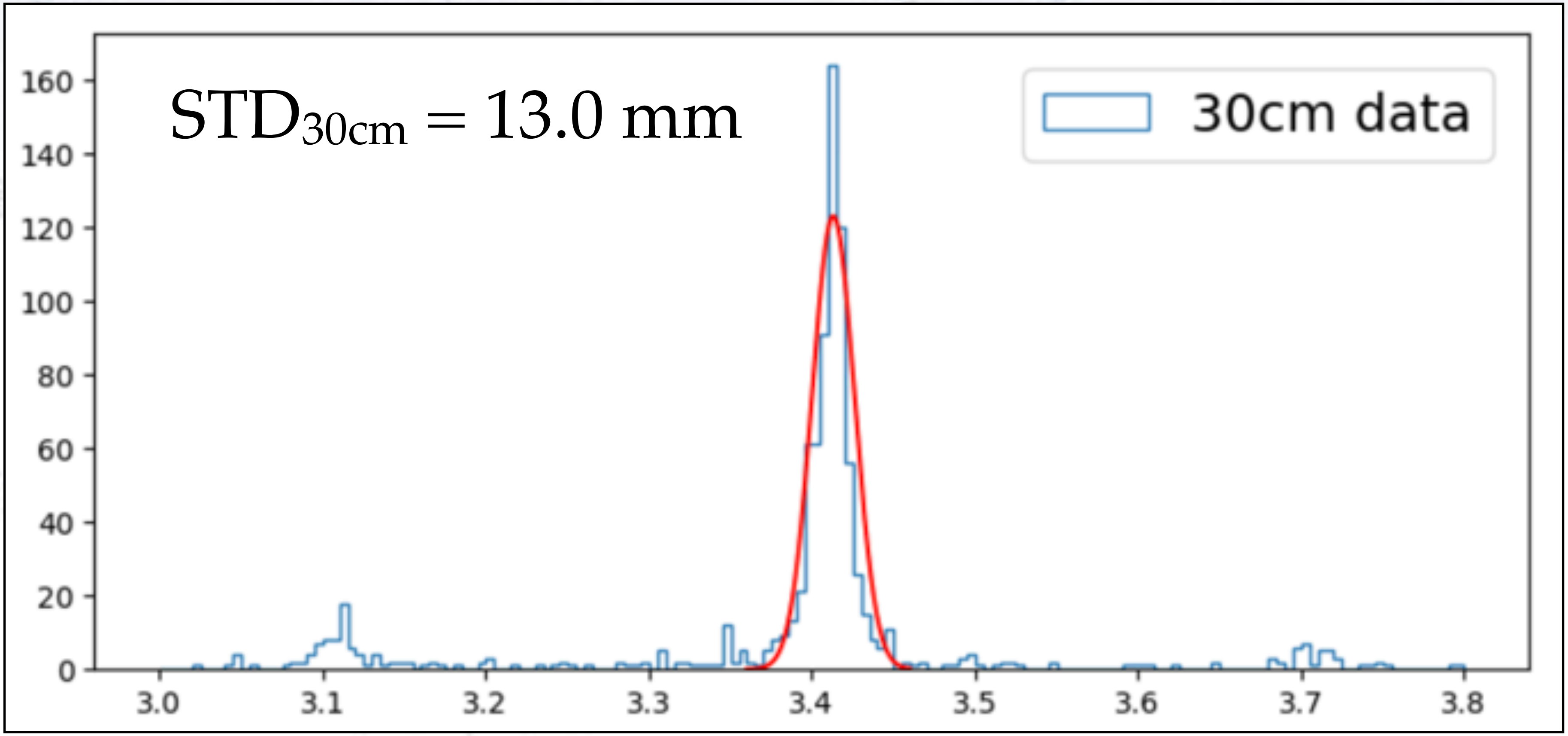
Include offsets?

There are some clear and understandable mis-measurements.
Should one correct and include these?
Depends on resulting improvement, but decide without seeing the final result.



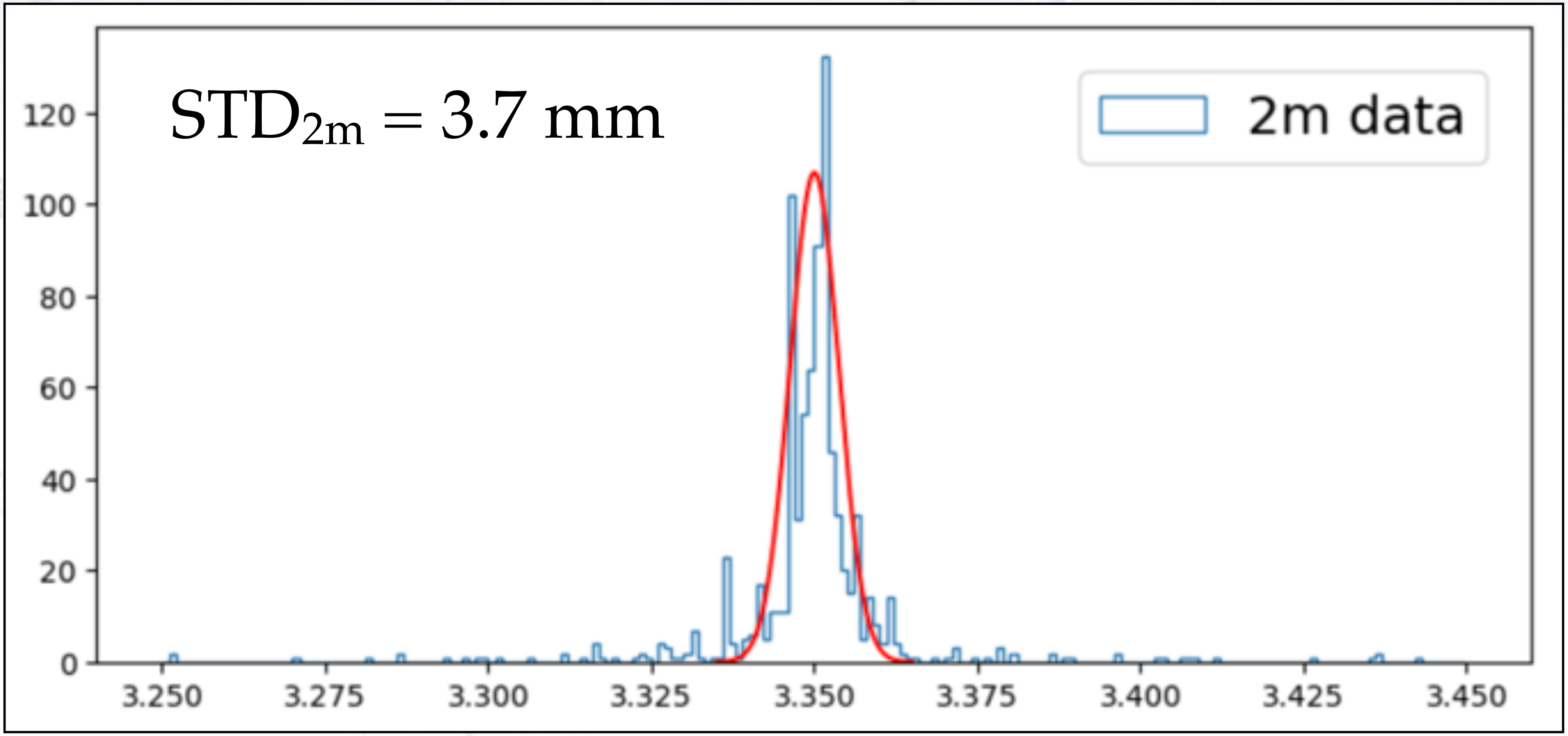
Inspecting the data

The 30cm peak seems somewhat Gaussian (p=2.4%) with binning 0.005m (smaller binning shows discontinuities, i.e. gives peaks).
The 2m peak does not seem Gaussian with any binning (here 0.005), yet "collected".



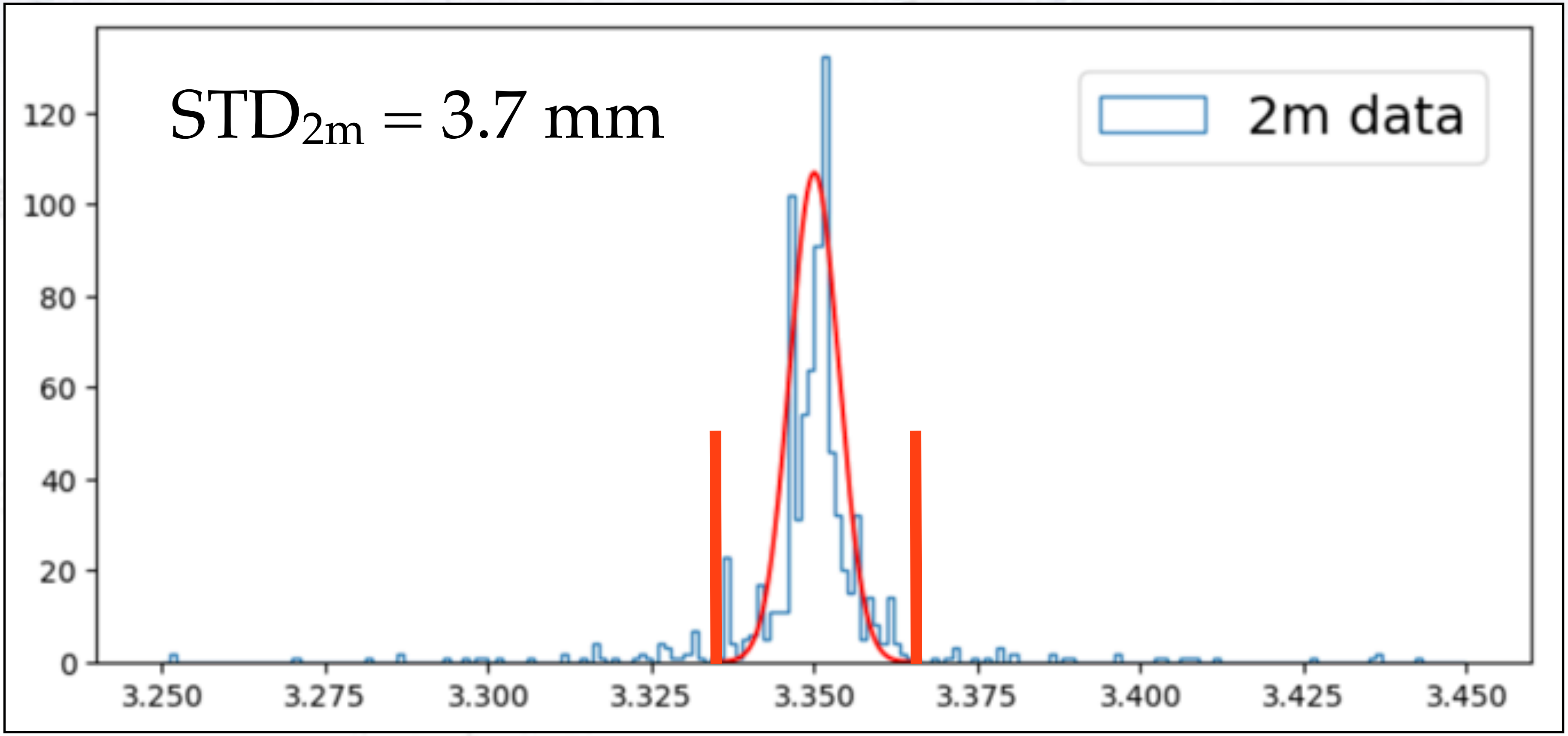
Inspecting the data

The 30cm peak seems somewhat Gaussian (p=2.4%) with binning 0.005m (smaller binning shows discontinuities, i.e. gives peaks).
The 2m peak does not seem Gaussian with any binning (here 0.005), yet "collected".



Inspecting the data

There are clearly some **mis-measurements**, which we would like to **exclude**. Using the fitted width, and accepting that this only includes the best measurements, I could **decide** to include measurements within **4 × STD**:



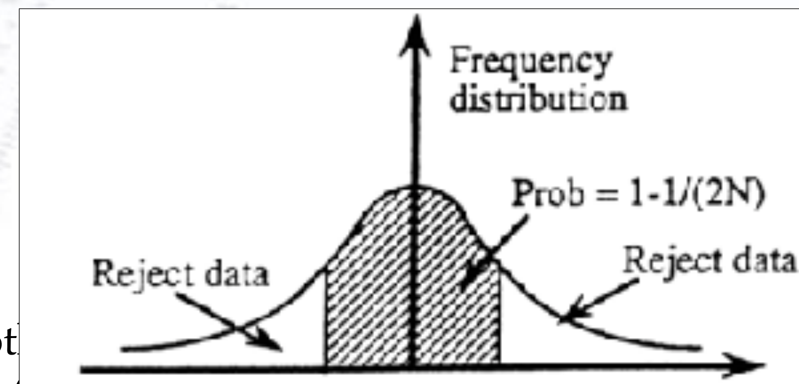
Removing data - General

Some (very "purist") scientists would never allow for the reject of data points!
They would argue, that data reflects reality, and that one should simply model this, including imperfections.

Less "purist" scientists accept exclusion of some data points. However, **one should always be very careful about removing data points**, and only be willing to do so, if very good arguments can be found:

- It is clearly an errornous measurement.
- Measurement is highly improbable.

Preferably, one would like to understand why data points seem faulty.



The procedures for removing points are:

- Without errors: **Chauvenet's Criterion,** though ot
- With errors: Simply reject based on the z-value = $(x - \mu)/\sigma$ of the point

However, **ALWAYS keep a record of your original data**, as it may contain more effects than you originally thought.

Removing data - without errors

Removing improbable data points when no error is given is formalised in **Chauvenet's Criterion,** though many other methods exists (Pierce, Grubbs, etc.)

The overall idea is to assume that the distribution is Gaussian.
One calculates the mean (μ) and standard deviation (σ), and then:

1. Ask what the probability of the farthest point is (given the number of points)
2. Remove point, if it is below some value (e.g. 0.05, preferably decided in advance)
3. If the furthest point was removed, then recalculate μ and σ, and go to 1.

How to calculate the probability of the furthest point with value x (given μ and σ)?

1. Calculate z:

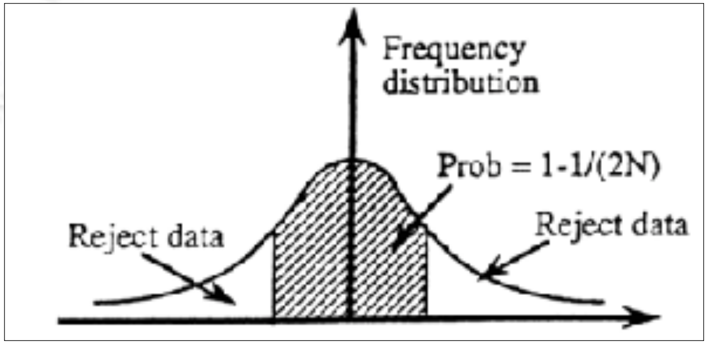
$$z = (x - \mu)/\sigma$$

2. Find the probability of this z, p_{local}:

$$p_{\mathrm{local}} = \int_{-\infty}^{-z} G(z)\,dz + \int_{z}^{\infty} G(z)\,dz$$

3. Take number of point into account, to get p_{global}:

$$p_{\mathrm{global}} = 1 - (1 - p_{\mathrm{local}})^{N_{\mathrm{points}}}$$



Key question:
Is $p_{global} < 0.05$?

Removing data - with errors

Removing improbable data points when each point has an associated uncertainty is much simpler.

The overall idea is that all points should be consistent with a mean value.
One calculates the weighted mean (μ), and then removes all points that are more than z_{cut} sigma away. Done!

No iterative procedure is needed. One can can calculate the value of z_{cut} ahead of applying it as:

$$p_{\mathrm{local}} = 1 - (1 - p_{\mathrm{global}})^{1/N_{\mathrm{points}}}$$

Example:
You have 1000 measurements, all with uncertainties, and decide to discard all points which are less likely than $p_{global} = 0.05$. This yields a cut at $p_{local} = 0.000051$ or 4.05σ. Thus, one would reject all data, which are more than 4.05σ away from the mean.

```
p_local = 1.0 - (1.0 - p_global)**(1.0/Ndata)
Nsigma = np.abs(stats.norm.ppf(p_local/2.0))
```

…a fair hearing?

```
 -------------------   Chauvenet's Criterion (30cm)   -------------------
600: L=1.325  dL=2.050  Nsig= 8.85 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=913 mean=3.3750  std=0.2317  -> Rejected
 61: L=1.405  dL=1.973  Nsig= 8.90 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=912 mean=3.3773  std=0.2216  -> Rejected
 97: L=1.413  dL=1.967  Nsig= 9.28 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=911 mean=3.3795  std=0.2119  -> Rejected
552: L=1.415  dL=1.967  Nsig= 9.75 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=910 mean=3.3816  std=0.2017  -> Rejected
829: L=1.420  dL=1.964  Nsig=10.28 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=909 mean=3.3838  std=0.1910  -> Rejected
189: L=1.434  dL=1.952  Nsig=10.87 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=908 mean=3.3859  std=0.1796  -> Rejected
162: L=2.206  dL=1.182  Nsig= 7.05 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=907 mean=3.3881  std=0.1676  -> Rejected
261: L=2.383  dL=1.007  Nsig= 6.18 p_loc=0.00000000  p_glob=0.00000030 >? pmin=0.100  N=906 mean=3.3894  std=0.1630  -> Rejected
 30: L=2.422  dL=0.969  Nsig= 6.07 p_loc=0.00000000  p_glob=0.00000058 >? pmin=0.100  N=905 mean=3.3905  std=0.1596  -> Rejected
110: L=4.355  dL=0.963  Nsig= 6.16 p_loc=0.00000000  p_glob=0.00000034 >? pmin=0.100  N=904 mean=3.3916  std=0.1564  -> Rejected
826: L=4.310  dL=0.919  Nsig= 6.00 p_loc=0.00000000  p_glob=0.00000090 >? pmin=0.100  N=903 mean=3.3905  std=0.1532  -> Rejected
786: L=4.190  dL=0.800  Nsig= 5.33 p_loc=0.00000005  p_glob=0.00004501 >? pmin=0.100  N=902 mean=3.3895  std=0.1502  -> Rejected
773: L=2.613  dL=0.776  Nsig= 5.25 p_loc=0.00000008  p_glob=0.00006997 >? pmin=0.100  N=901 mean=3.3886  std=0.1479  -> Rejected
119: L=2.625  dL=0.765  Nsig= 5.25 p_loc=0.00000008  p_glob=0.00006887 >? pmin=0.100  N=900 mean=3.3895  std=0.1457  -> Rejected
212: L=2.700  dL=0.691  Nsig= 4.81 p_loc=0.00000075  p_glob=0.00067274 >? pmin=0.100  N=899 mean=3.3903  std=0.1435  -> Rejected
768: L=2.720  dL=0.671  Nsig= 4.74 p_loc=0.00000109  p_glob=0.00097587 >? pmin=0.100  N=898 mean=3.3911  std=0.1418  -> Rejected
599: L=4.054  dL=0.662  Nsig= 4.73 p_loc=0.00000115  p_glob=0.00102772 >? pmin=0.100  N=897 mean=3.3918  std=0.1400  -> Rejected
496: L=2.750  dL=0.641  Nsig= 4.64 p_loc=0.00000178  p_glob=0.00159297 >? pmin=0.100  N=896 mean=3.3911  std=0.1384  -> Rejected
294: L=4.010  dL=0.618  Nsig= 4.52 p_loc=0.00000313  p_glob=0.00279838 >? pmin=0.100  N=895 mean=3.3918  std=0.1368  -> Rejected
179: L=4.009  dL=0.618  Nsig= 4.56 p_loc=0.00000250  p_glob=0.00223068 >? pmin=0.100  N=894 mean=3.3911  std=0.1353  -> Rejected
601: L=4.005  dL=0.614  Nsig= 4.59 p_loc=0.00000220  p_glob=0.00195965 >? pmin=0.100  N=893 mean=3.3904  std=0.1338  -> Rejected
445: L=2.794  dL=0.596  Nsig= 4.51 p_loc=0.00000329  p_glob=0.00292612 >? pmin=0.100  N=892 mean=3.3897  std=0.1323  -> Rejected
350: L=2.800  dL=0.591  Nsig= 4.52 p_loc=0.00000315  p_glob=0.00280203 >? pmin=0.100  N=891 mean=3.3904  std=0.1308  -> Rejected
241: L=2.808  dL=0.583  Nsig= 4.51 p_loc=0.00000325  p_glob=0.00288751 >? pmin=0.100  N=890 mean=3.3911  std=0.1294  -> Rejected
423: L=2.809  dL=0.583  Nsig= 4.56 p_loc=0.00000260  p_glob=0.00230684 >? pmin=0.100  N=889 mean=3.3917  std=0.1280  -> Rejected
 16: L=2.815  dL=0.578  Nsig= 4.57 p_loc=0.00000248  p_glob=0.00220346 >? pmin=0.100  N=888 mean=3.3924  std=0.1265  -> Rejected
654: L=2.818  dL=0.575  Nsig= 4.60 p_loc=0.00000212  p_glob=0.00187759 >? pmin=0.100  N=887 mean=3.3930  std=0.1251  -> Rejected
557: L=2.819  dL=0.575  Nsig= 4.65 p_loc=0.00000166  p_glob=0.00147061 >? pmin=0.100  N=886 mean=3.3937  std=0.1237  -> Rejected
169: L=2.864  dL=0.531  Nsig= 4.34 p_loc=0.00000706  p_glob=0.00622428 >? pmin=0.100  N=885 mean=3.3943  std=0.1222  -> Rejected
205: L=3.891  dL=0.496  Nsig= 4.10 p_loc=0.00002085  p_glob=0.01825865 >? pmin=0.100  N=884 mean=3.3949  std=0.1210  -> Rejected
700: L=2.971  dL=0.424  Nsig= 3.53 p_loc=0.00020441  p_glob=0.16515717 >? pmin=0.100  N=883 mean=3.3944  std=0.1199  -> Accepted

 -------------------   Chauvenet's Criterion (2m)   -------------------
191: L=1.337  dL=1.978  Nsig= 7.64 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=911 mean=3.3154  std=0.2590  -> Rejected
400: L=1.341  dL=1.976  Nsig= 7.88 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=910 mean=3.3176  std=0.2507  -> Rejected
 98: L=1.350  dL=1.970  Nsig= 8.14 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=909 mean=3.3197  std=0.2421  -> Rejected
 61: L=1.351  dL=1.971  Nsig= 8.45 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=908 mean=3.3219  std=0.2332  -> Rejected
199: L=1.351  dL=1.973  Nsig= 8.81 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=907 mean=3.3241  std=0.2240  -> Rejected
894: L=1.351  dL=1.975  Nsig= 9.22 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100  N=906 mean=3.3263  std=0.2143  -> Rejected
```

…a fair hearing?

```
-----------------    Chauvenet's Criterion (30cm)    -----------------
600: L=1.325  dL=2.050  Nsig= 8.85 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=913 mean=3.3750  std=0.2317  -> Rejected
 61: L=1.405  dL=1.973  Nsig= 8.90 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=912 mean=3.3773  std=0.2216  -> Rejected
 97: L=1.413  dL=1.967  Nsig= 9.28 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=911 mean=3.3795  std=0.2119  -> Rejected
552: L=1.415  dL=1.967  Nsig= 9.75 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=910 mean=3.3816  std=0.2017  -> Rejected
829: L=1.420  dL=1.964  Nsig=10.28 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=909 mean=3.3838  std=0.1910  -> Rejected
189: L=1.434  dL=1.952  Nsig=10.87 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=908 mean=3.3859  std=0.1796  -> Rejected
162: L=2.206  dL=1.182  Nsig= 7.05 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=907 mean=3.3881  std=0.1676  -> Rejected
261: L=2.383  dL=1.007  Nsig= 6.18 p_loc=0.00000000  p_glob=0.00000030 >? pmin=0.100   N=906 mean=3.3894  std=0.1630  -> Rejected
 30: L=2.422  dL=0.969  Nsig= 6.07 p_loc=0.00000000  p_glob=0.00000058 >? pmin=0.100   N=905 mean=3.3905  std=0.1596  -> Rejected
110: L=4.355  dL=0.963  Nsig= 6.16 p_loc=0.00000000  p_glob=0.00000034 >? pmin=0.100   N=904 mean=3.3916  std=0.1564  -> Rejected
826: L=4.310  dL=0.919  Nsig= 6.00 p_loc=0.00000000  p_glob=0.00000090 >? pmin=0.100   N=903 mean=3.3905  std=0.1532  -> Rejected
786: L=4.190  dL=0.800  Nsig= 5.33 p_loc=0.00000005  p_glob=0.00004501 >? pmin=0.100   N=902 mean=3.3895  std=0.1502  -> Rejected
773: L=2.613  dL=0.776  Nsig= 5.25 p_loc=0.00000008  p_glob=0.00006997 >? pmin=0.100   N=901 mean=3.3886  std=0.1479  -> Rejected
119: L=2.625  dL=0.765                                                                                td=0.1457  -> Rejected
212: L=2.700  dL=0.691                                                                                td=0.1435  -> Rejected
768: L=2.720  dL=0.671                                                                                td=0.1418  -> Rejected
599: L=4.054  dL=0.662                                                                                td=0.1400  -> Rejected
496: L=2.750  dL=0.641                                                                                td=0.1384  -> Rejected
294: L=4.010  dL=0.618                                                                                td=0.1368  -> Rejected
179: L=4.009  dL=0.618                                                                                td=0.1353  -> Rejected
601: L=4.005  dL=0.614                                                                                td=0.1338  -> Rejected
445: L=2.794  dL=0.596                                                                                td=0.1323  -> Rejected
350: L=2.800  dL=0.591                                                                                td=0.1308  -> Rejected
241: L=2.808  dL=0.583                                                                                td=0.1294  -> Rejected
423: L=2.809  dL=0.583                                                                                td=0.1280  -> Rejected
 16: L=2.815  dL=0.578  Nsig= 4.57 p_loc=0.00000248  p_glob=0.00220346 >? pmin=0.100   N=888 mean=3.3924  std=0.1265  -> Rejected
654: L=2.818  dL=0.575  Nsig= 4.60 p_loc=0.00000212  p_glob=0.00187759 >? pmin=0.100   N=887 mean=3.3930  std=0.1251  -> Rejected
557: L=2.819  dL=0.575  Nsig= 4.65 p_loc=0.00000166  p_glob=0.00147061 >? pmin=0.100   N=886 mean=3.3937  std=0.1237  -> Rejected
169: L=2.864  dL=0.531  Nsig= 4.34 p_loc=0.00000706  p_glob=0.00622428 >? pmin=0.100   N=885 mean=3.3943  std=0.1222  -> Rejected
205: L=3.891  dL=0.496  Nsig= 4.10 p_loc=0.00002085  p_glob=0.01825865 >? pmin=0.100   N=884 mean=3.3949  std=0.1210  -> Rejected
700: L=2.971  dL=0.424  Nsig= 3.53 p_loc=0.00020441  p_glob=0.16515717 >? pmin=0.100   N=883 mean=3.3944  std=0.1199  -> Accepted

-----------------    Chauvenet's Criterion (2m)    -----------------
191: L=1.337  dL=1.978  Nsig= 7.64 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=911 mean=3.3154  std=0.2590  -> Rejected
400: L=1.341  dL=1.976  Nsig= 7.88 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=910 mean=3.3176  std=0.2507  -> Rejected
 98: L=1.350  dL=1.970  Nsig= 8.14 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=909 mean=3.3197  std=0.2421  -> Rejected
 61: L=1.351  dL=1.971  Nsig= 8.45 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=908 mean=3.3219  std=0.2332  -> Rejected
199: L=1.351  dL=1.973  Nsig= 8.81 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=907 mean=3.3241  std=0.2240  -> Rejected
894: L=1.351  dL=1.975  Nsig= 9.22 p_loc=0.00000000  p_glob=0.00000000 >? pmin=0.100   N=906 mean=3.3263  std=0.2143  -> Rejected
```

I rejected:
30 data points from the 30cm sample (*),
132 data points from the 2m sample.
And I inspected each and every one!

Unweighted results

30cm:

Mean = 3.39438 ± 0.00404 m

Std. = 0.120 m (N = 882)

2m:

Mean = 3.34942 ± 0.00020 m

Std. = 0.0055 m (N = 779)

Without corrections and Chavenet's (p=0.10)

While the 2m result starts looking realistic (Std. of 5.5mm) and precise (1/5th of a millimeter), the 30cm result is still terrible.

This is because the two ±30cm peaks are not removed by Chavenet's Criterion.

30cm: Reject or correct?

The poor 30cm result is due to the (many) mis-measurements of ± 30cm, which are not rejected by the Chauvenet's Criterion with p_global = 0.10.

At least two solutions exist:

1. Decide to **reject** all measurements more than 15cm away from the mean and run Chauvenet's Criterion again.
2. Decide to **correct** measurements 15-45cm away from the mean, and run Chauvenet's Criterion again.

Rejecting the events removes a total of 172 (± 15cm) + 64 (CC) measurements.
Correcting the events first removes a total of 91 measurements.

The uncertain on the mean resulting from each strategy is:

1. Rejection: 0.49 mm
2. Correction: 0.62 mm

So, it is worthwhile to focus on the good measurements, if this can be argued.

Unweighted results

30cm:

Mean = 3.41190 ± 0.00049 m

Std. = 0.0126 m (N = 677)

2m:

Mean = 3.34942 ± 0.00020 m

Std. = 0.0055 m (N = 779)

Without corrections and Chavenet's (p=0.10)

Now the results are precise, and the 2m result is about a factor 2.5 more so, as would also be expected from the initial Std. observed for the peaks.

The improvement over the naive 30cm / 2m results are factores of 19 / 43

Cross Check

Now we have gotten two precision results. How to cross check if there is any realism in the values and uncertainties?

We compare the 30cm and 2m results.

So far, the results have been blinded, and so the difference is very large:
L30cm - L2m (fully blinded) = 0.06248 ± 0.00052 m (119.4σ → prob=0.0000)

Subtracting the *difference in blinding value* yields the real difference:
L30cm - L2m (partially blinded) = **-0.00102 ± 0.00052** m (**2.0σ** → prob=0.0508)

This means that the two results are reasonably within each other, and the results and their uncertainty are (more) trustworthy.

Weighted analysis

Checking for valid errors

In order to do a weighted analyst, the measurements of course have to have valid uncertainties.

You may wonder why there are negative uncertainties!

The reason is, that this is a (good?) way of putting measurements without uncertainties, without putting NaNs into the table.

```
The 30cm entry L =  3.413 +- -1.000 was not considered valid!
The 30cm entry L =  3.412 +- -1.000 was not considered valid!
The 30cm entry L =  3.388 +- -1.000 was not considered valid!
The 30cm entry L =  3.416 +- -1.000 was not considered valid!
The 30cm entry L =  3.421 +- -1.000 was not considered valid!
The 30cm entry L =  3.415 +- -1.000 was not considered valid!
The 30cm entry L =  3.443 +- -1.000 was not considered valid!
The 30cm entry L =  3.422 +- -1.000 was not considered valid!
The 30cm entry L =  3.410 +- -1.000 was not considered valid!
The 30cm entry L =  3.416 +- -1.000 was not considered valid!
The 30cm entry L =  3.416 +- -1.000 was not considered valid!
The 30cm entry L =  3.405 +- -1.000 was not considered valid!
The 30cm entry L =  3.417 +- -1.000 was not considered valid!
The 30cm entry L =  3.414 +- -1.000 was not considered valid!
The 30cm entry L =  3.430 +- -1.000 was not considered valid!
The 30cm entry L =  3.720 +- -1.000 was not considered valid!
The number of accepted / rejected 30cm points is 897 / 16

The 2m entry L =  3.350 +- -1.000 was not considered valid!
The 2m entry L =  3.361 +- -1.000 was not considered valid!
The 2m entry L =  3.351 +-  0.000 was not considered valid!
The 2m entry L =  3.350 +- -1.000 was not considered valid!
The 2m entry L =  1.361 +-  0.000 was not considered valid!
The 2m entry L =  3.342 +- -1.000 was not considered valid!
The 2m entry L =  3.350 +- -1.000 was not considered valid!
The 2m entry L = -1.014 +- -1.000 was not considered valid!
The 2m entry L =  3.355 +- -1.000 was not considered valid!
The 2m entry L =  3.353 +- -1.000 was not considered valid!
The 2m entry L =  3.349 +- -1.000 was not considered valid!
The 2m entry L =  3.356 +- -1.000 was not considered valid!
The 2m entry L =  3.348 +- -1.000 was not considered valid!
The 2m entry L =  3.350 +- -1.000 was not considered valid!
The 2m entry L = -1.014 +- -0.010 was not considered valid!
The 2m entry L =  3.336 +-  0.000 was not considered valid!
The 2m entry L =  3.341 +- -0.010 was not considered valid!
The 2m entry L =  3.346 +- -0.010 was not considered valid!
The 2m entry L =  3.351 +- -0.010 was not considered valid!
The 2m entry L =  3.426 +- -0.010 was not considered valid!
The 2m entry L =  3.626 +- -0.010 was not considered valid!
The number of accepted / rejected 2m points is 892 / 21
```

“Naive” weighted results

30cm:

Mean = 3.38764 ± 0.00017 m

RMS = undefined! (N = 896)

2m:

Mean = 3.31486 ± 0.00009 m

RMS = undefined! (N = 891)

Now the results are really precise, and the 2m result is about a factor 2.5 more so, as would also be expected from the initial Std. observed for the peaks.

The improvement over the naive 30cm / 2m results are factores of 19 / 43

“Naive” weighted results

30cm:

Mean = 3.38764 ± 0.00017 m

Chi2 = 1743866.2, Ndof = 896, Prob = 0.0!

2m:

Mean = 3.31486 ± 0.00009 m

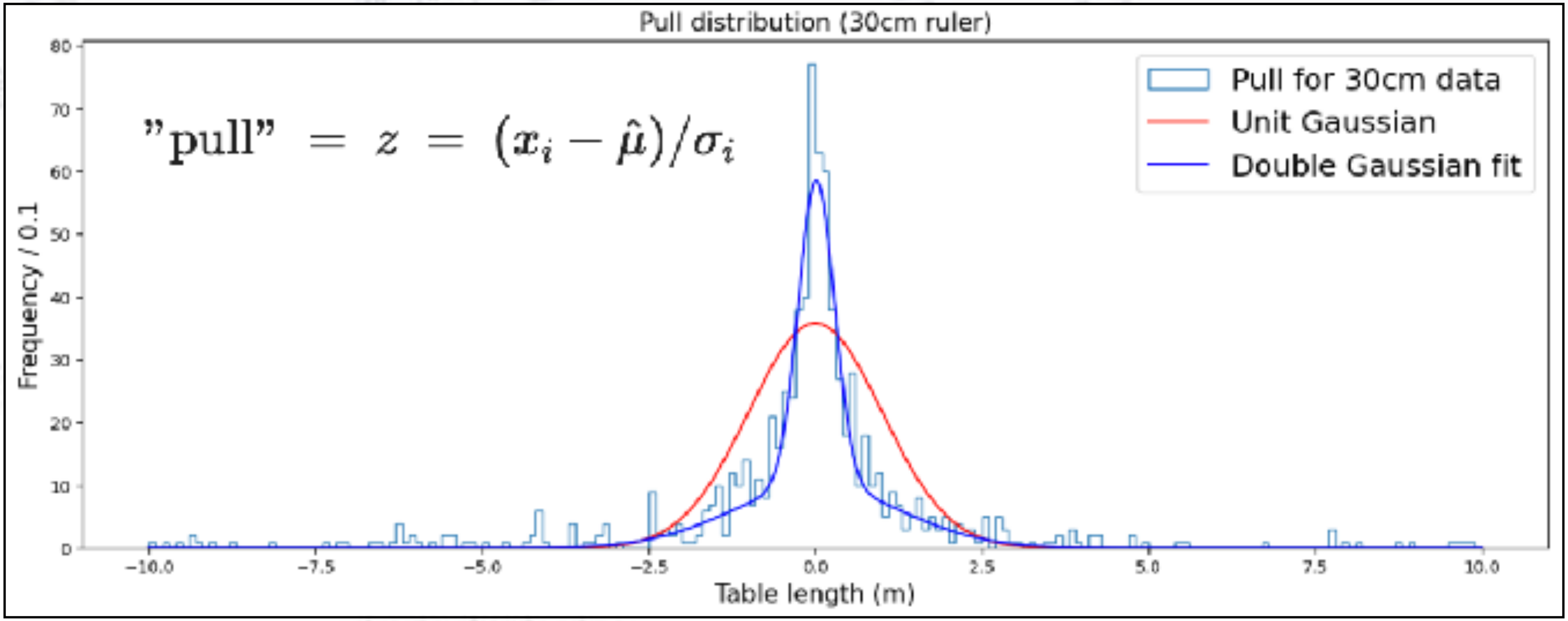
Chi2 = 7595341.9, Ndof = 891, Prob = 0.0!

While the values of the results may look "alluring" and the uncertainties amazingly small, the ChiSquare reveals that this is not the case. **The measurements disagree enormously when uncertainties are taken into account**. Clearly, the naive approach is way off.

The pull distribution

Considering the quoted uncertainties, we first need to evaluate their quality. The plot to consider is a **PULL** plot, i.e. the distribution of z-values.

The pulls should be unit Gaussian. However, it is far from. In fact, **most pull values are small, which is caused by an overestimation of the uncertainty**. We are too conservative and don't trust, that we can do things fairly accurately.

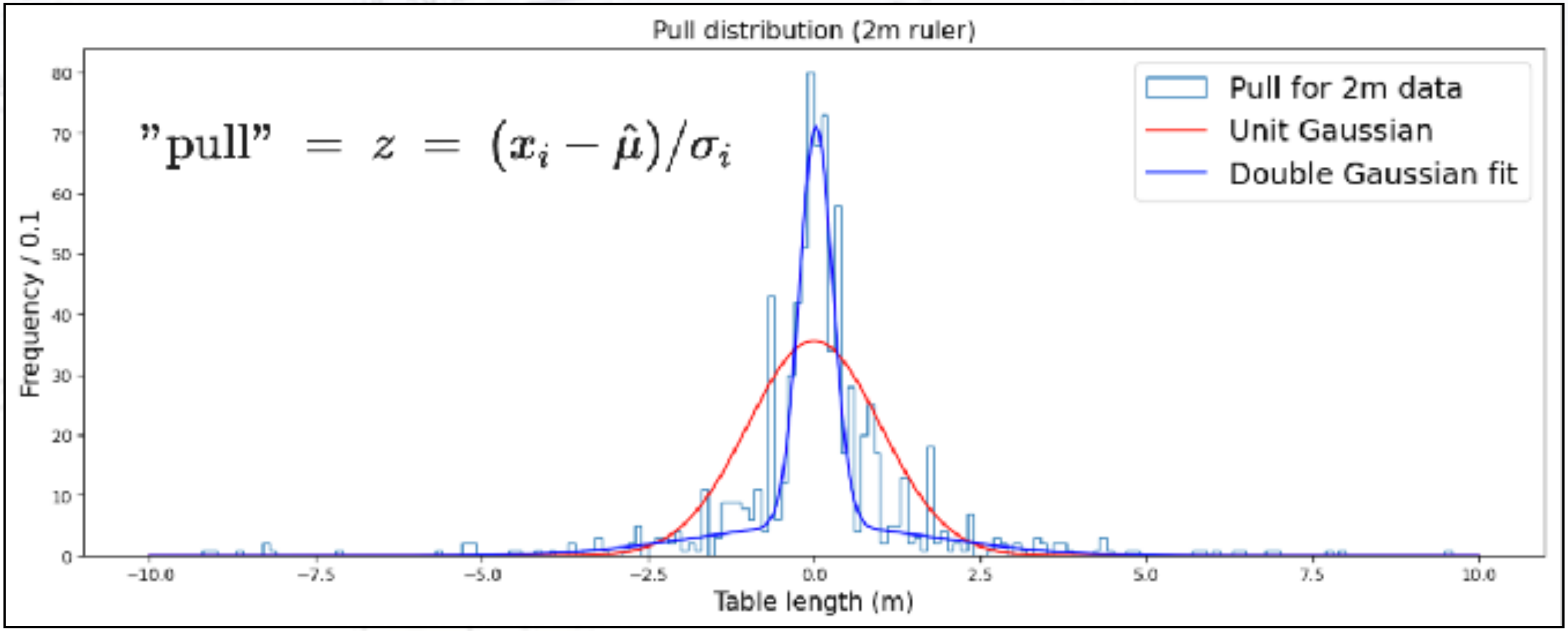


In the case at hand, we take the mean to be the unweighted best result.

The pull distribution

Considering the quoted uncertainties, we first need to evaluate their quality.
The plot to consider is a **PULL** plot, i.e. the distribution of z-values.

The pulls should be unit Gaussian. However, it is far from. In fact, **most pull values are small, which is caused by an overestimation of the uncertainty**. We are too conservative and don't trust, that we can do things fairly accurately.



In the case at hand, we take the mean to be the unweighted best result.

Where to select?

Assuming unit Gaussian distributions (not the case, but still) we calculate at what level it is reasonable to discard individual measurement based on their z-value, i.e. how many sigmas they are away from the mean.

This only depends on the number of measurements and $p_{global} = 0.05$, and the result is 4.0σ for both 30cm and 2m data.

```
p_local = 1.0 - (1.0 - p_global)**(1.0/Ndata)
Nsigma = np.abs(stats.norm.ppf(p_local/2.0))
```

Given that the assumption is not really fulfilled, the real level should be set below this value, as many low z-values will make the ChiSquare unnaturally low. I chose 80% and 90% of the 4.0σ, i.e. 3.2σ and 3.6σ

Once the selection level is fitting, we then discard unlikely events (i.e. beyond a certain number of sigmas) and then proceed to calculate the weighted mean (with error, Chi2, and ProbChi2 of course!).

Excluded data due to bad pull

```
Warning! Large pull: L = 3.750 +- 0.013 m      z =   25.98
Warning! Large pull: L = 3.697 +- 0.020 m      z =   14.24
Warning! Large pull: L = 3.323 +- 0.009 m      z =   -9.91
Warning! Large pull: L = 2.815 +- 0.091 m      z =   -6.56
Warning! Large pull: L = 3.250 +- 0.002 m      z =  -81.11
Warning! Large pull: L = 3.395 +- 0.005 m      z =   -3.45
Warning! Large pull: L = 2.422 +- 0.005 m      z = -198.05
Warning! Large pull: L = 3.753 +- 0.010 m      z =   34.08
Warning! Large pull: L = 3.425 +- 0.003 m      z =    4.26
Warning! Large pull: L = 3.085 +- 0.070 m      z =   -4.67
Warning! Large pull: L = 3.115 +- 0.023 m      z =  -12.92
Warning! Large pull: L = 3.595 +- 0.005 m      z =   36.55
Warning! Large pull: L = 3.701 +- 0.030 m      z =    9.63
Warning! Large pull: L = 1.405 +
Warning! Large pull: L = 3.356 +
Warning! Large pull: L = 3.089 +
Warning! Large pull: L = 3.112 +
Warning! Large pull: L = 3.417 +
Warning! Large pull: L = 3.405 +
Warning! Large pull: L = 3.409 +
Warning! Large pull: L = 3.743 +
Warning! Large pull: L = 3.721 +- 0.005 m      z =   61.75
Warning! Large pull: L = 3.134 +- 0.050 m      z =   -5.56
Warning! Large pull: L = 1.413 +- 0.013 m      z = -153.79
Warning! Large pull: L = 3.142 +- 0.005 m      z =  -54.05
Warning! Large pull: L = 3.107 +- 0.005 m      z =  -61.05
Warning! Large pull: L = 4.355 +- 0.260 m      z =    3.63
Warning! Large pull: L = 3.795 +- 0.100 m      z =    3.83
Warning! Large pull: L = 3.091 +- 0.015 m      z =  -21.42
Warning! Large pull: L = 2.625 +- 0.090 m      z =   -8.75
Warning! Large pull: L = 3.106 +- 0.073 m      z =   -4.19
Warning! Large pull: L = 3.132 +- 0.022 m      z =  -12.74
Warning! Large pull: L = 3.168 +- 0.045 m      z =   -5.43
Warning! Large pull: L = 2.206 +- 0.002 m      z = -603.11
Warning! Large pull: L = 3.094 +- 0.002 m      z = -159.11
Warning! Large pull: L = 3.085 +- 0.005 m      z =  -65.45
Warning! Large pull: L = 3.320 +- 0.005 m      z =  -18.45
Warning! Large pull: L = 3.106 +- 0.050 m      z =   -6.12
Warning! Large pull: L = 3.110 +- 0.010 m      z =  -30.22
Warning! Large pull: L = 3.717 +- 0.061 m      z =    5.00
```

```
Warning! Large pull: L = 3.696 +- 0.050 m      z =    6.93
Warning! Large pull: L = 3.371 +- 0.005 m      z =    4.35
Warning! Large pull: L = 2.350 +- 0.015 m      z =  -66.67
Warning! Large pull: L = 3.746 +- 0.020 m      z =   19.84
Warning! Large pull: L = 3.051 +- 0.002 m      z = -149.13
Warning! Large pull: L = 2.350 +- 0.001 m      z = -999.25
Warning! Large pull: L = 1.351 +- 0.020 m      z =  -99.91
Warning! Large pull: L = 3.122 +- 0.025 m      z =   -9.09
Warning! Large pull: L = 3.631 +- 0.006 m      z =   46.96
Warning! Large pull: L = 3.535 +- 0.005 m      z =   37.15
Warning! Large pull: L = 1.350 +- 0.010 m      z = -199.93
Warning! Large pull: L = 4.550 +- 0.023 m      z =   52.21
Warning! Large pull: L = 3.322 +- 0.001 m      z =  -27.25
= 3.166 +- 0.020 m      z =   -9.16
= 3.344 +- 0.001 m      z =   -5.25
= 3.653 +- 0.025 m      z =   12.15
= 3.386 +- 0.010 m      z =    3.67
= 3.660 +- 0.084 m      z =    3.70
= 3.353 +- 0.001 m      z =    3.75
= 3.324 +- 0.001 m      z =  -25.25
= 2.351 +- 0.015 m      z =  -66.55
Warning! Large pull: L = 1.337 +- 0.003 m      z = -670.75
Warning! Large pull: L = 3.641 +- 0.014 m      z =   20.04
Warning! Large pull: L = 1.351 +- 0.005 m      z = -399.65
Warning! Large pull: L = 3.357 +- 0.002 m      z =    3.87
Warning! Large pull: L = 3.299 +- 0.002 m      z =  -25.13
Warning! Large pull: L = 3.286 +- 0.005 m      z =  -12.65
Warning! Large pull: L = 2.636 +- 0.050 m      z =  -14.27
Warning! Large pull: L = 3.361 +- 0.002 m      z =    5.87
Warning! Large pull: L = 2.631 +- 0.010 m      z =  -71.83
Warning! Large pull: L = 2.665 +- 0.150 m      z =   -4.56
Warning! Large pull: L = 3.358 +- 0.002 m      z =    4.37
Warning! Large pull: L = 3.341 +- 0.001 m      z =   -8.25
Warning! Large pull: L = 3.344 +- 0.001 m      z =   -5.25
Warning! Large pull: L = 2.341 +- 0.025 m      z =  -40.33
Warning! Large pull: L = 3.618 +- 0.013 m      z =   20.67
Warning! Large pull: L = 3.341 +- 0.002 m      z =   -4.13
Warning! Large pull: L = 3.378 +- 0.002 m      z =   14.37
Warning! Large pull: L = 3.335 +- 0.002 m      z =   -7.13
Warning! Large pull: L = 3.354 +- 0.001 m      z =    4.75
```

I rejected:
176 data points from the 30cm sample,
102 data points from the 2m sample.
And I inspected each and every one!

Excluded data due to bad pull

```
Warning! Large pull: L = 3.750 +- 0.013 m     z =   25.98
Warning! Large pull: L = 3
Warning! Large pull: L = 3
Warning! Large pull: L = 2
Warning! Large pull: L = 3
Warning! Large pull: L = 3
Warning! Large pull: L = 2
Warning! Large pull: L = 3
Warning! Large pull: L = 3
Warning! Large pull: L = 3
Warning! Large pull: L = 3.115 +- 0.023 m     z =  -12.92
Warning! Large pull: L = 3.595 +- 0.005 m     z =   36.55
Warning! Large pull: L = 3.701 +- 0.030 m     z =    9.63
Warning! Large pull: L = 1.405 +-
Warning! Large pull: L = 3.356 +-
Warning! Large pull: L = 3.089 +-
Warning! Large pull: L = 3.112 +-
Warning! Large pull: L = 3.417 +-
Warning! Large pull: L = 3.405 +-
Warning! Large pull: L = 3.409 +-
Warning! Large pull: L = 3.743 +-
Warning! Large pull: L = 3.721 +- 0.005 m     z =   61.75
Warning! Large pull: L = 3.134 +- 0.050 m     z =   -5.56
Warning! Large pull: L = 1.413 +- 0.013 m     z = -153.79
Warning! Large pull: L = 3.142 +- 0.005 m     z =  -54.05
Warning! Large pull: L = 3.107 +- 0.005 m     z =  -61.05
Warning! Large pull: L = 4.355 +- 0.260 m     z =    3.63
Warning! Large pull: L = 3.795 +- 0.100 m     z =    3.83
Warning! Large pull: L = 3.091 +- 0.015 m     z =  -21.42
Warning! Large pull: L = 2.625 +- 0.090 m     z =   -8.75
Warning! Large pull: L = 3.106 +- 0.073 m     z =   -4.19
Warning! Large pull: L = 3.132 +- 0.022 m     z =  -12.74
Warning! Large pull: L = 3.168 +- 0.045 m     z =   -5.43
Warning! Large pull: L = 2.206 +- 0.002 m     z = -603.11
Warning! Large pull: L = 3.094 +- 0.002 m     z = -159.11
Warning! Large pull: L = 3.085 +- 0.005 m     z =  -65.45
Warning! Large pull: L = 3.320 +- 0.005 m     z =  -18.45
Warning! Large pull: L = 3.106 +- 0.050 m     z =   -6.12
Warning! Large pull: L = 3.110 +- 0.010 m     z =  -30.22
Warning! Large pull: L = 3.717 +- 0.061 m     z =    5.00
```

Largest pull is about 1000, the result of a fine measurement with tiny uncertainty…
…but 1 meter wrong!

I rejected:
176 data points from the 30cm sample,
102 data points from the 2m sample.
And I inspected each and every one!

```
Warning! Large pull: L = 3.696 +- 0.050 m     z =    6.93
Warning! Large pull: L = 3.371 +- 0.005 m     z =    4.35
Warning! Large pull: L = 2.350 +- 0.015 m     z =  -66.62
Warning! Large pull: L = 3.746 +- 0.020 m     z =   19.84
Warning! Large pull: L = 3.051 +- 0.002 m     z = -149.13
Warning! Large pull: L = 2.350 +- 0.001 m     z = -999.25
Warning! Large pull: L = 1.351 +- 0.020 m     z =  -99.91
Warning! Large pull: L = 3.122 +- 0.025 m     z =   -9.09
Warning! Large pull: L = 3.631 +- 0.006 m     z =   46.96
Warning! Large pull: L = 3.535 +- 0.005 m     z =   37.15
Warning! Large pull: L = 1.350 +- 0.010 m     z = -199.93
Warning! Large pull: L = 4.550 +- 0.023 m     z =   52.21
Warning! Large pull: L = 3.322 +- 0.001 m     z =  -27.25
                         = 3.166 +- 0.020 m     z =   -9.16
                         = 3.344 +- 0.001 m     z =   -5.25
                         = 3.653 +- 0.025 m     z =   12.15
                         = 3.386 +- 0.010 m     z =    3.67
                         = 3.660 +- 0.004 m     z =    3.70
                         = 3.353 +- 0.001 m     z =    3.75
                         = 3.324 +- 0.001 m     z =  -25.25
                         = 2.351 +- 0.015 m     z =  -66.55
Warning! Large pull: L = 1.337 +- 0.003 m     z = -670.75
Warning! Large pull: L = 3.641 +- 0.014 m     z =   20.04
Warning! Large pull: L = 1.351 +- 0.005 m     z = -399.65
Warning! Large pull: L = 3.357 +- 0.002 m     z =    3.87
Warning! Large pull: L = 3.299 +- 0.002 m     z =  -25.13
Warning! Large pull: L = 3.286 +- 0.005 m     z =  -12.65
Warning! Large pull: L = 2.636 +- 0.050 m     z =  -14.27
Warning! Large pull: L = 3.361 +- 0.002 m     z =    5.87
Warning! Large pull: L = 2.631 +- 0.010 m     z =  -71.83
Warning! Large pull: L = 2.665 +- 0.150 m     z =   -4.56
Warning! Large pull: L = 3.358 +- 0.002 m     z =    4.37
Warning! Large pull: L = 3.341 +- 0.001 m     z =   -8.25
Warning! Large pull: L = 3.344 +- 0.001 m     z =   -5.25
Warning! Large pull: L = 2.341 +- 0.025 m     z =  -40.33
Warning! Large pull: L = 3.618 +- 0.013 m     z =   20.67
Warning! Large pull: L = 3.341 +- 0.002 m     z =   -4.13
Warning! Large pull: L = 3.378 +- 0.002 m     z =   14.37
Warning! Large pull: L = 3.335 +- 0.002 m     z =   -7.13
Warning! Large pull: L = 3.354 +- 0.001 m     z =    4.75
```

Weighted results

<u>30cm:</u>

Mean = 3.41300 ± 0.00027 m

RMS = undefined!   (N = 677)

<u>2m:</u>

Mean = 3.34985 ± 0.00011 m

RMS = undefined! (N = 779)

Now the results are really precise, and the 2m result is about a factor 2 more so.

Improvement over the unweighted 30cm / 2m results are factores of 1.8 / 1.8

So the uncertainties carry information about the measurement quality.

Weighted results

30cm:

Mean = 3.41300 ± 0.00027 m

Chi2 = 642.8, Ndof = 720, Prob = 0.98

2m:

Mean = 3.34985 ± 0.00011 m

Chi2 = 715.9, Ndof = 789, Prob = 0.97

Now the results are really precise, and the 2m result is about a factor 2 more so.

Improvement over the unweighted 30cm / 2m results are factores of 1.8 / 1.8

So the uncertainties carry information about the measurement quality.

Cross Checks

Once again, we compare the 30cm and 2m weighted results.

Subtracting the *difference in blinding value* yields the real difference:
L30cm - L2m (partially blinded) = **-0.00034 ± 0.00029** m (1**.2σ** → prob=0.24)

This means that the two results are reasonably within each other, and the results and their uncertainty are (more) trustworthy.

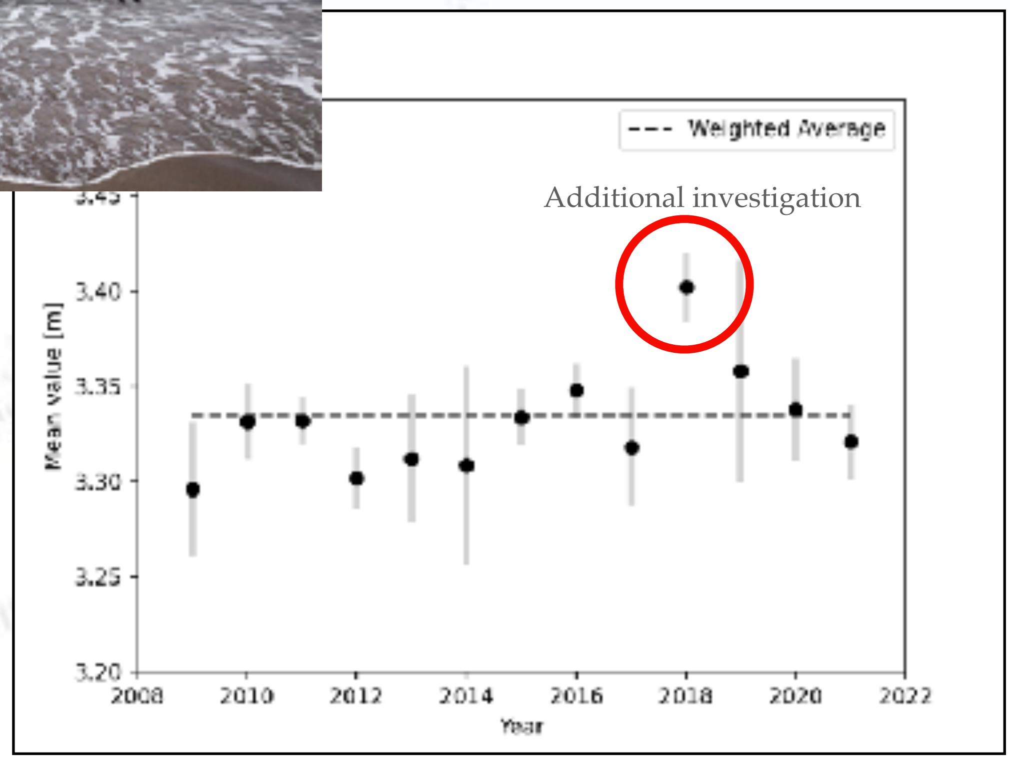
We can also check the unweighted against the weighted results. Here, there is not even a partial unblinding, as they have the same offsets.

30cm: Unweighted-Weighted = -0.00110 ± 0.00056 m (2.0σ → prob = 0.048)
2m: Unweighted-Weighted = -0.00042 ± 0.00023 m (1.9σ → prob = 0.062)

Thus, now the four results (30cm, 2m) x (unweighted, weighted) seem to be in agreement.

A problem?

Things may look very good, yet it remains to investigate the data further.
A question is the homogeneity of the data. And here we find problems!



Somehow, the 2018 data seems different (read: biased) compared to the other years.

One could also ask, if the order in the data file mattered.

Fitting analysis

Fitting for a result

A completely different approach is to fit the RAW data, hence describing **all** data points instead of excluding some.

This approach is philosophically more clean, but certainly not easy!

Challenges:

- Measurements has many different resolutions.
- There are several peaks in the data (30cm case).
- Some measurements are clearly rounded.

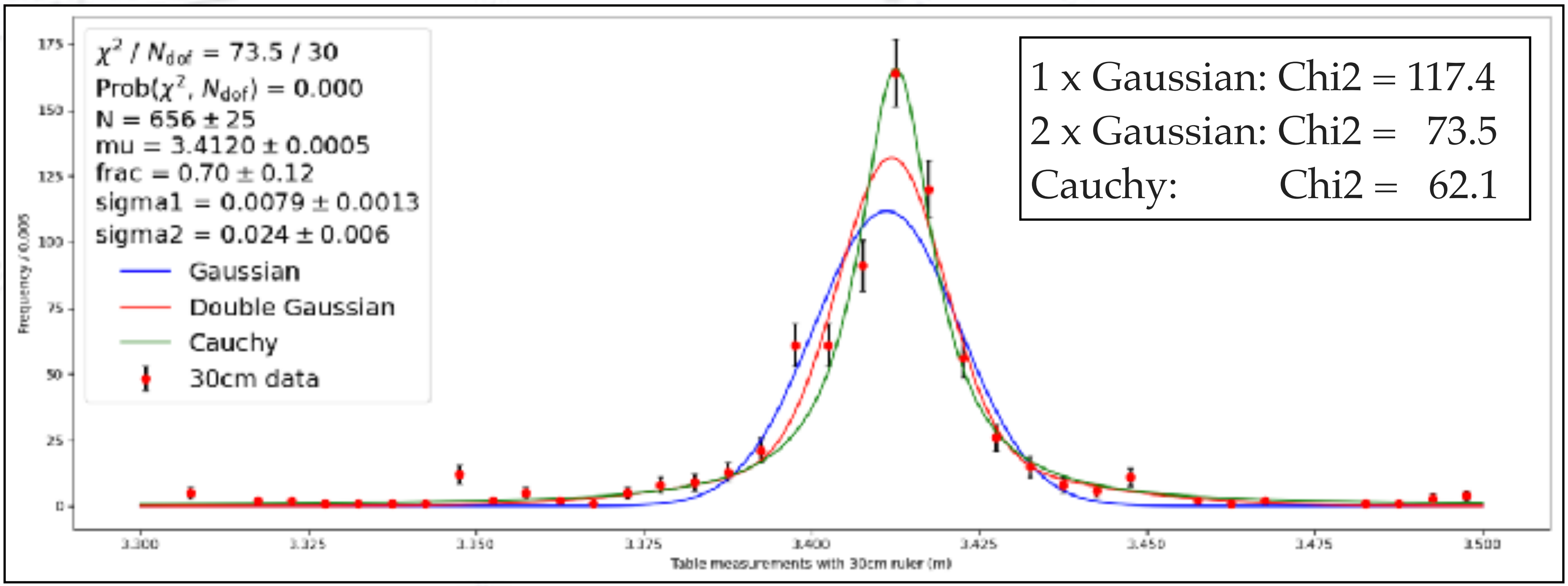
While all of these can be accommodated, it is still a challenge, at the following “fitting around” took me several hours!

Fitting for a result

First step is to establish what PDF the measurements follow.

I have tried the following three:

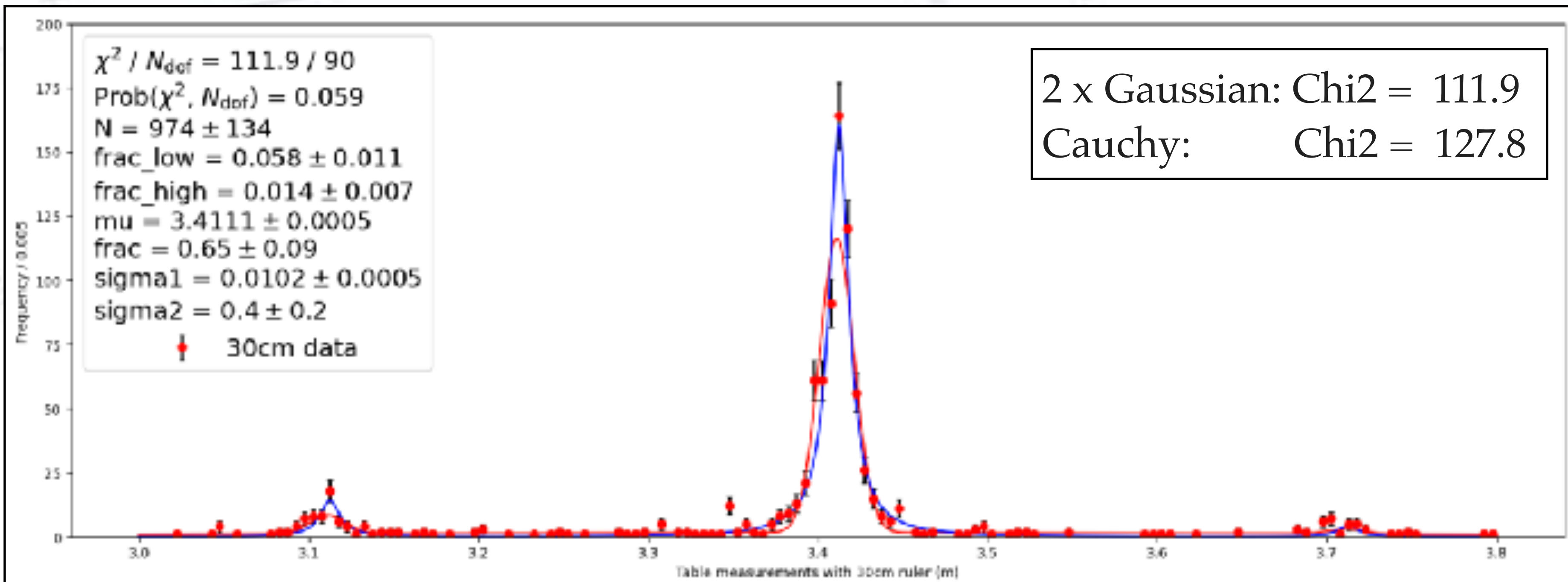
- Single Gaussian: Simplest and mandatory first step.
- Double Gaussian: To accommodate different resolutions.
- Cauchy: Alternative to Gaussian with long tails as expected.



Fitting for a result

The fits converge and gives OK values.However, both models have a problem modelling the far outliers. The second Gaussian starts being used for this, thus not matching the peak.

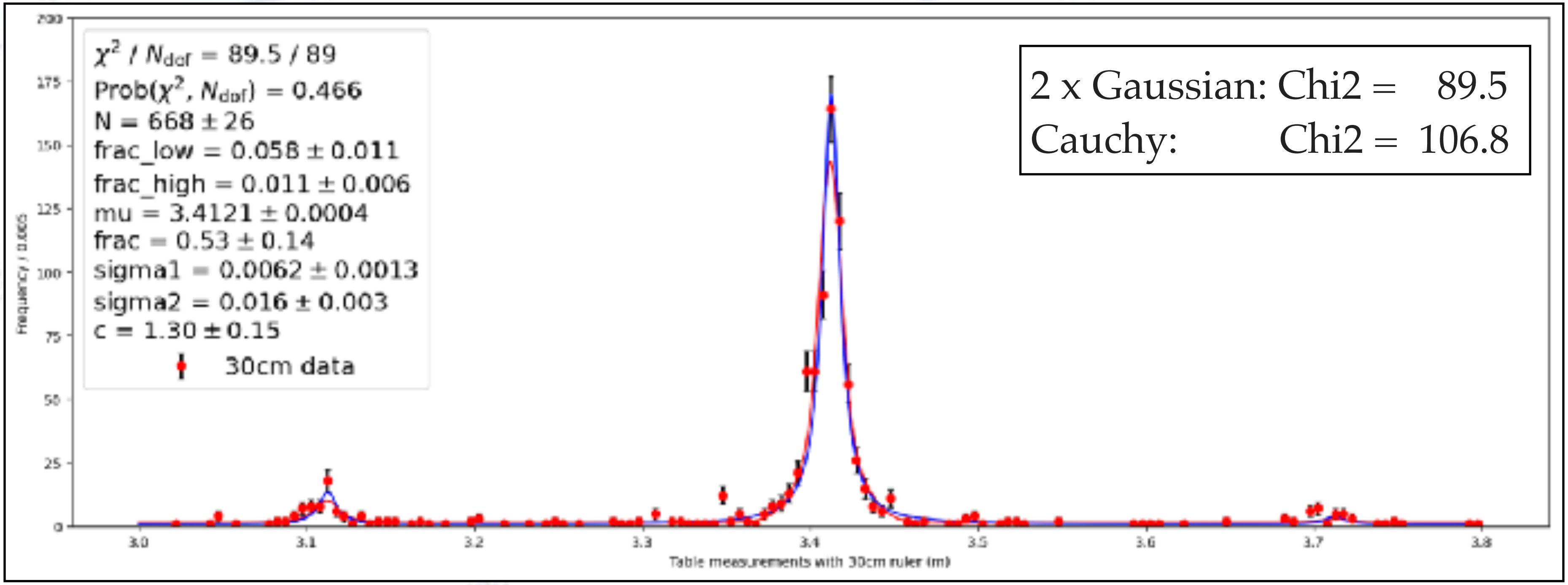
A better model, which avoids this problem should have a separate PDF for the far outliers.



Fitting for a result

The fits converge and gives OK values.However, both models have a problem modelling the far outliers. The second Gaussian starts being used for this, thus not matching the peak.

A better model, which avoids this problem should have a separate PDF for the far outliers. Adding a constant improves the fits, especially the double Gaussian.



Fitting results

Summarising all the fitting results (below), it is clear that the quality of the fit slowly improves.

```
Chi2 - Single Gaussian:              Prob(chi2 = 117.4, Ndof = 94) = 0.052    Mu = 3.411227 +- 0.000455
Chi2 - Double Gaussian:              Prob(chi2 =  73.5, Ndof = 92) = 0.922    Mu = 3.411950 +- 0.000547
Chi2 - 3 x Double Gaussian:          Prob(chi2 = 111.9, Ndof = 90) = 0.059    Mu = 3.411138 +- 0.000452
Chi2 - 3 x Double Gaussian + c:      Prob(chi2 =  89.5, Ndof = 89) = 0.466    Mu = 3.412124 +- 0.000448
ULLH - 3 x Double Gaussian + c:      Likelihood value = -6327.9               Mu = 3.413071 +- 0.000321
ULLH - 3 x Cauchy + c:               Likelihood value = -6241.9               Mu = 3.413009 +- 0.000318
```

The double Gaussian tripple peak fit has a good ChiSquare, but the statistics is often low, and hence a likelihood fit is used.

Even with a good PDF, this was not easy to get running, and amendments were needed. However, the result is significantly more precise, and in the end we reach an uncertainty of 0.32mm.

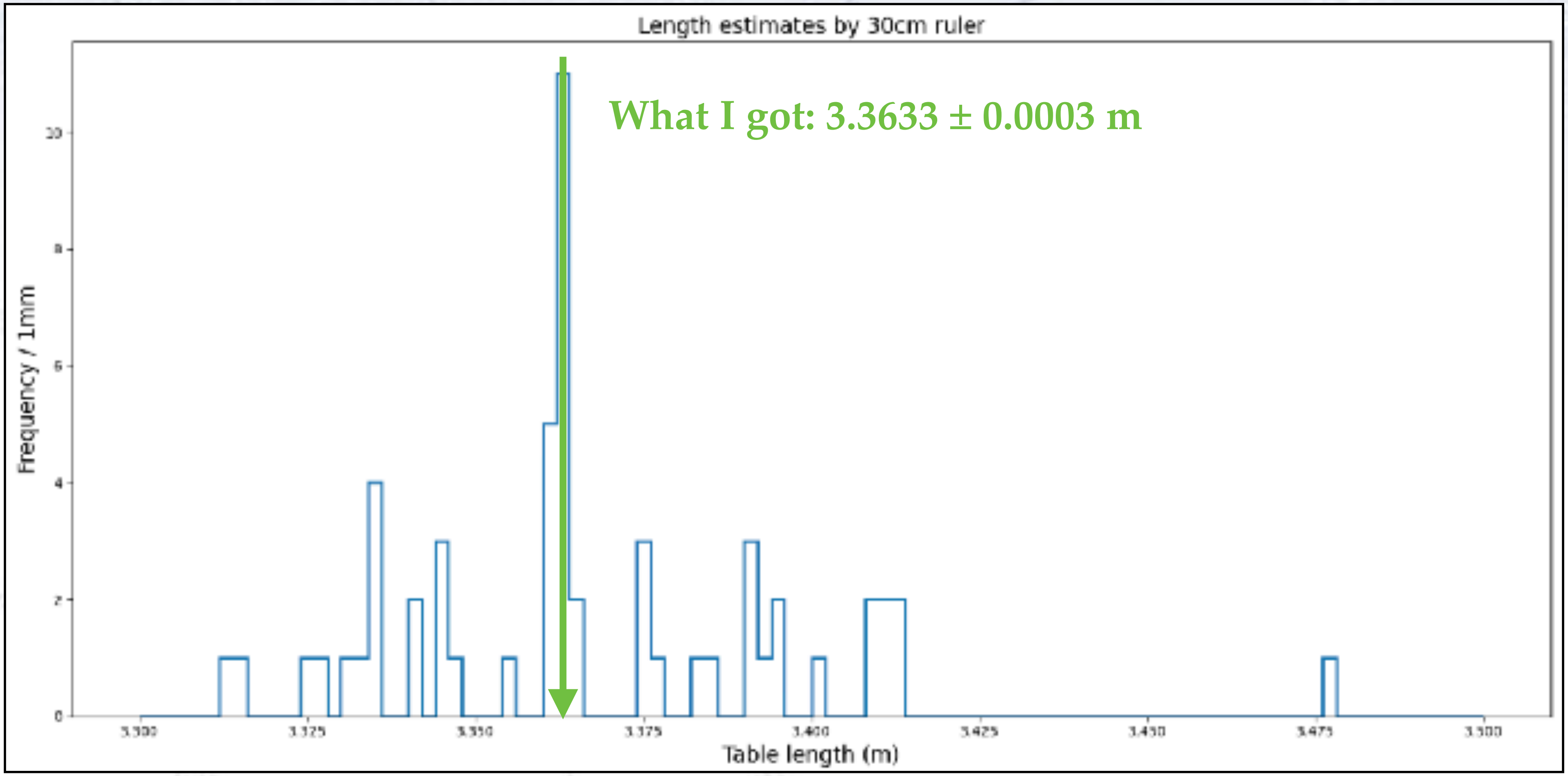
Both value and uncertainty are remarkably comparable to the weighted mean result: Weighted mean = 3.41300 +- 0.00027m Fit: 3.413071 +- 0.00032m

The fitting method starts being a significant systematic uncertainty!

Student analyses comparison

Your measurement value

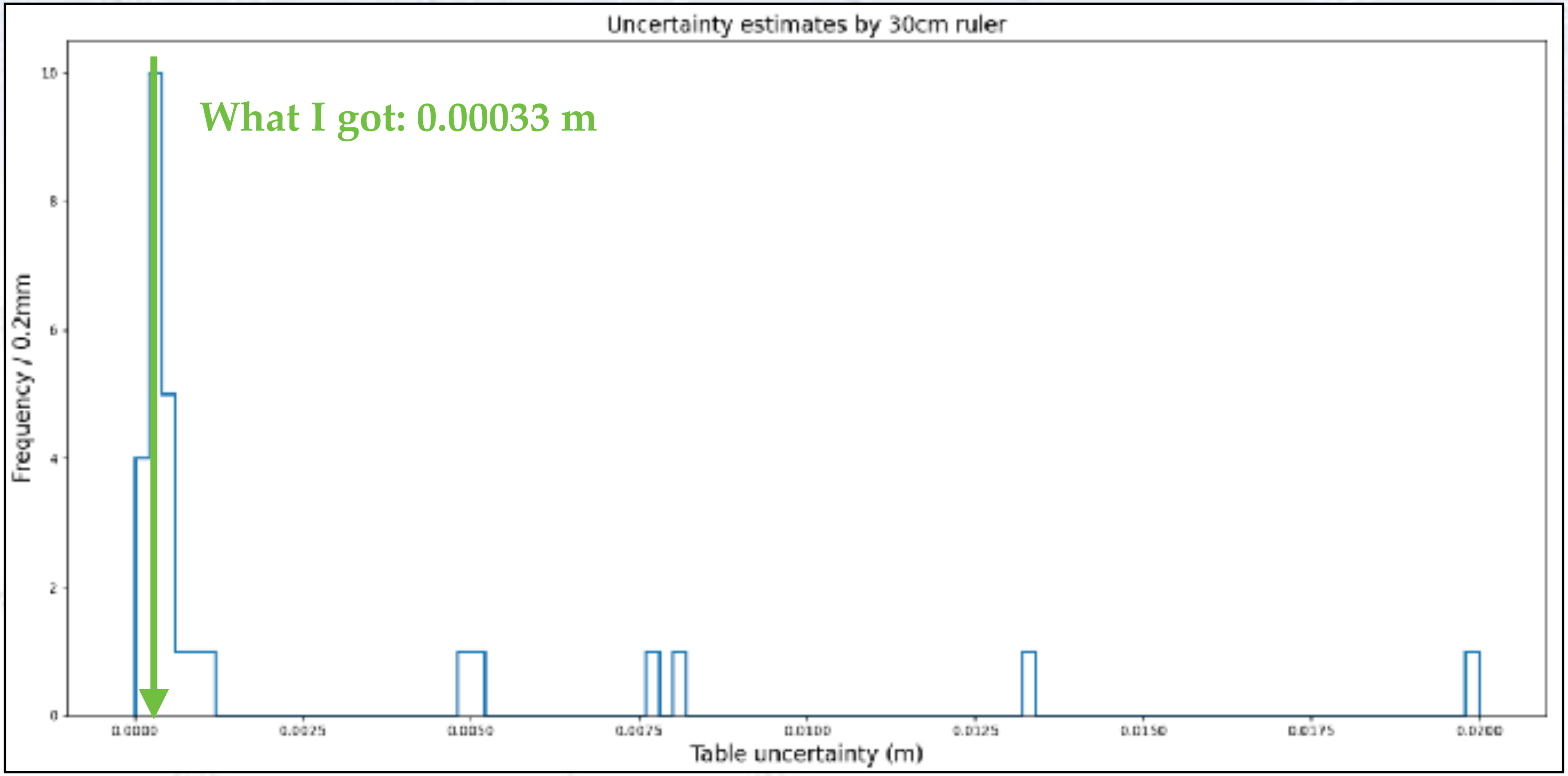
The uncertainties varied quite a bit…. from 3.3 to beyond 3.4.



Estimating uncertainties is (still) hard.

Your measurement uncertainty

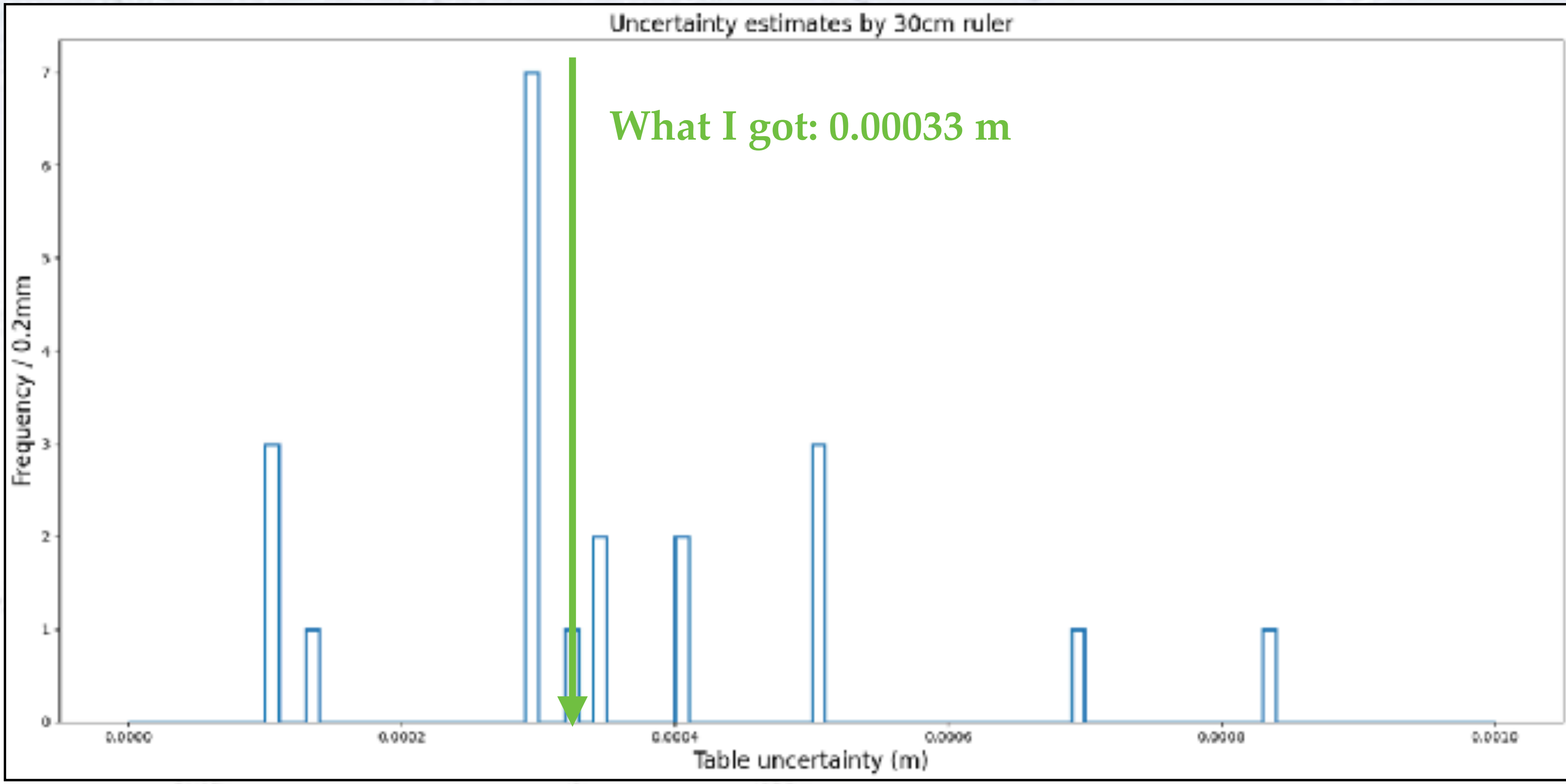
The measurement uncertainties varied even more wildly!!!



The lowest was 0.0001, while the highest was 0.02 (two orders of magnitude).

Your measurement uncertainty

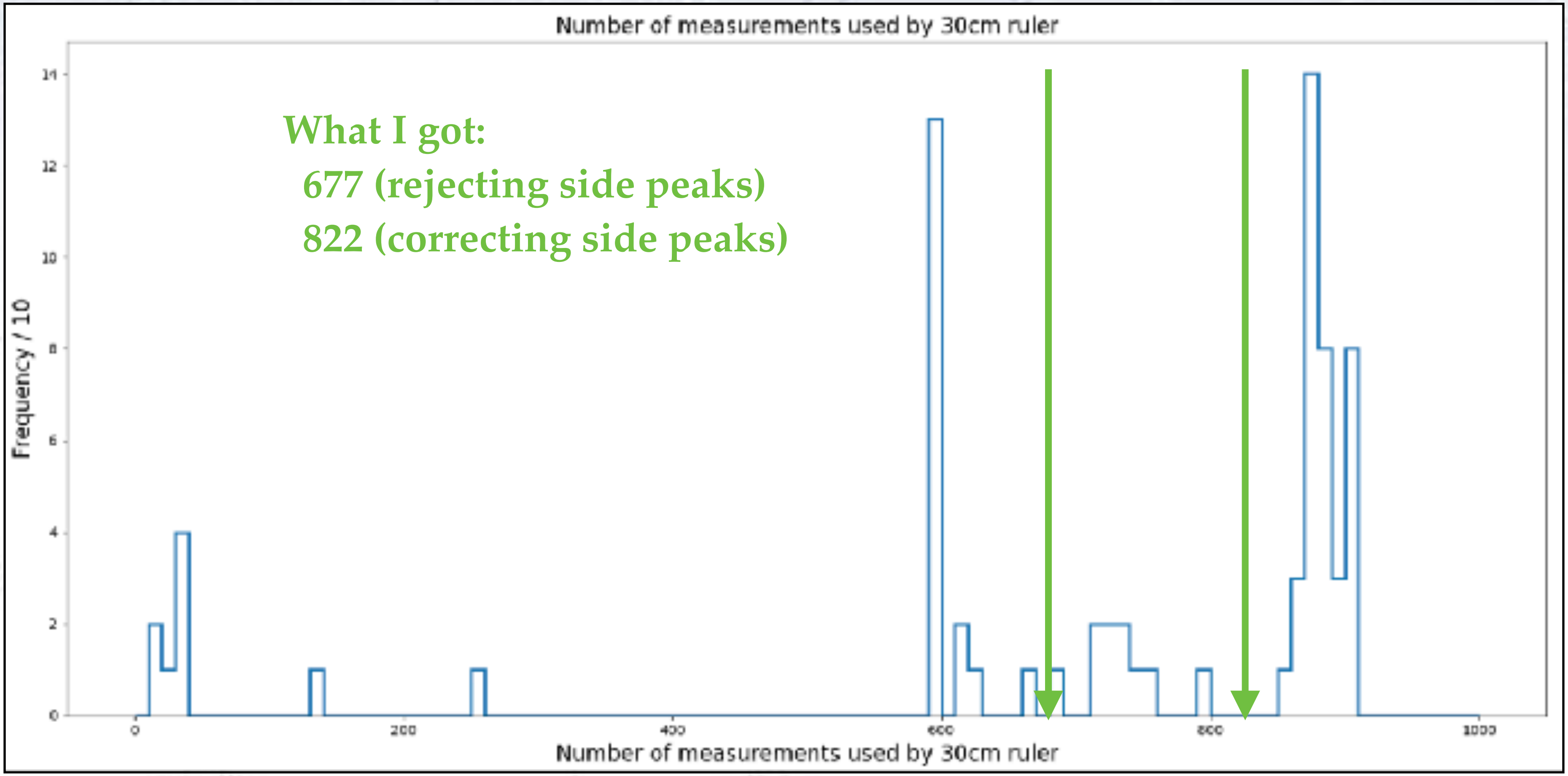
The measurement uncertainties varied even more wildly!!!



The lowest was 0.0001, while the highest was 0.02 (two orders of magnitude).

Your number of measurements

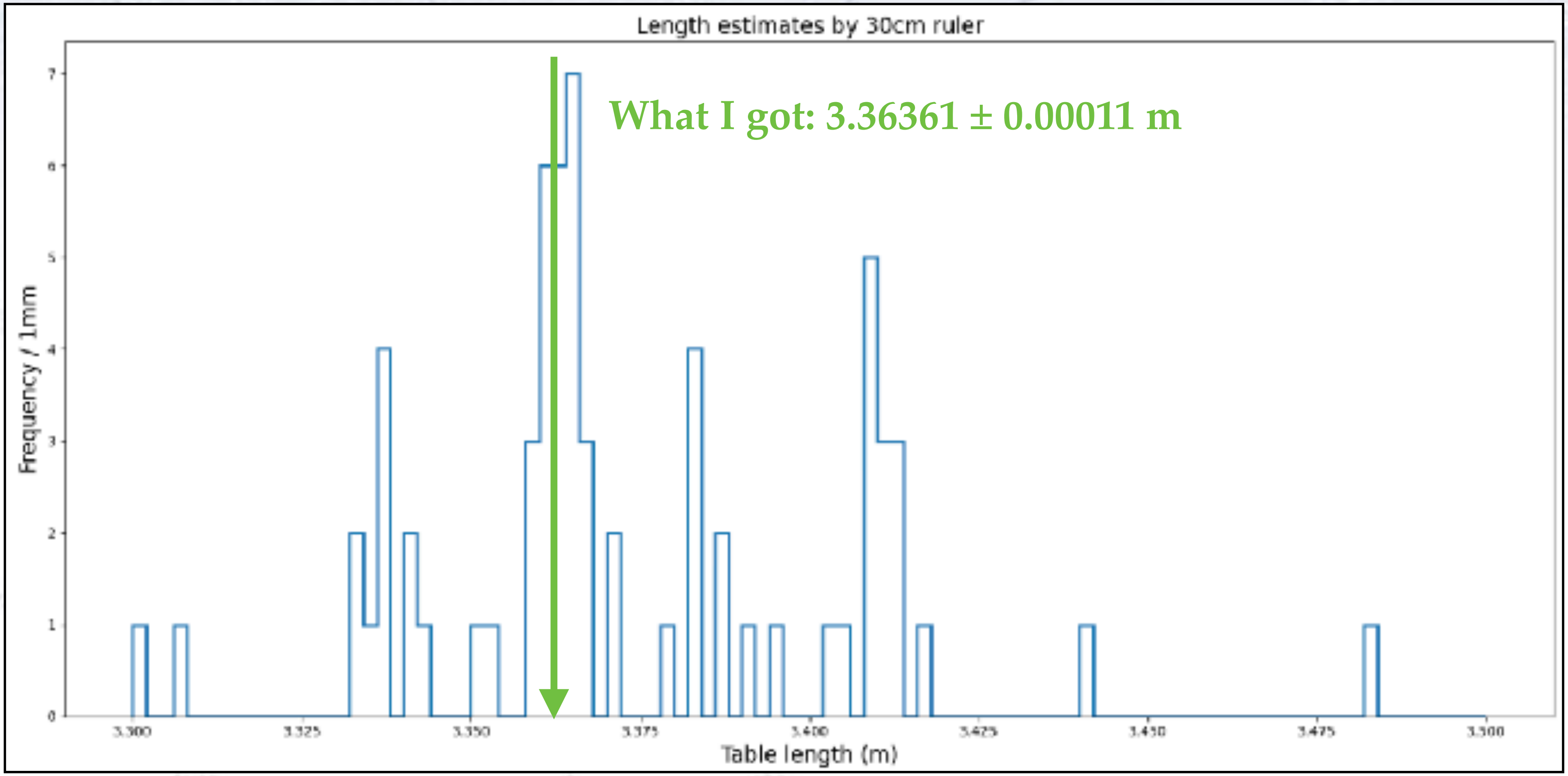
The number of measurements also varied, but some were in the right ballpark.



Remember that the impact is only sqrt(N), and thus not overly important!

Your measurement uncertainty

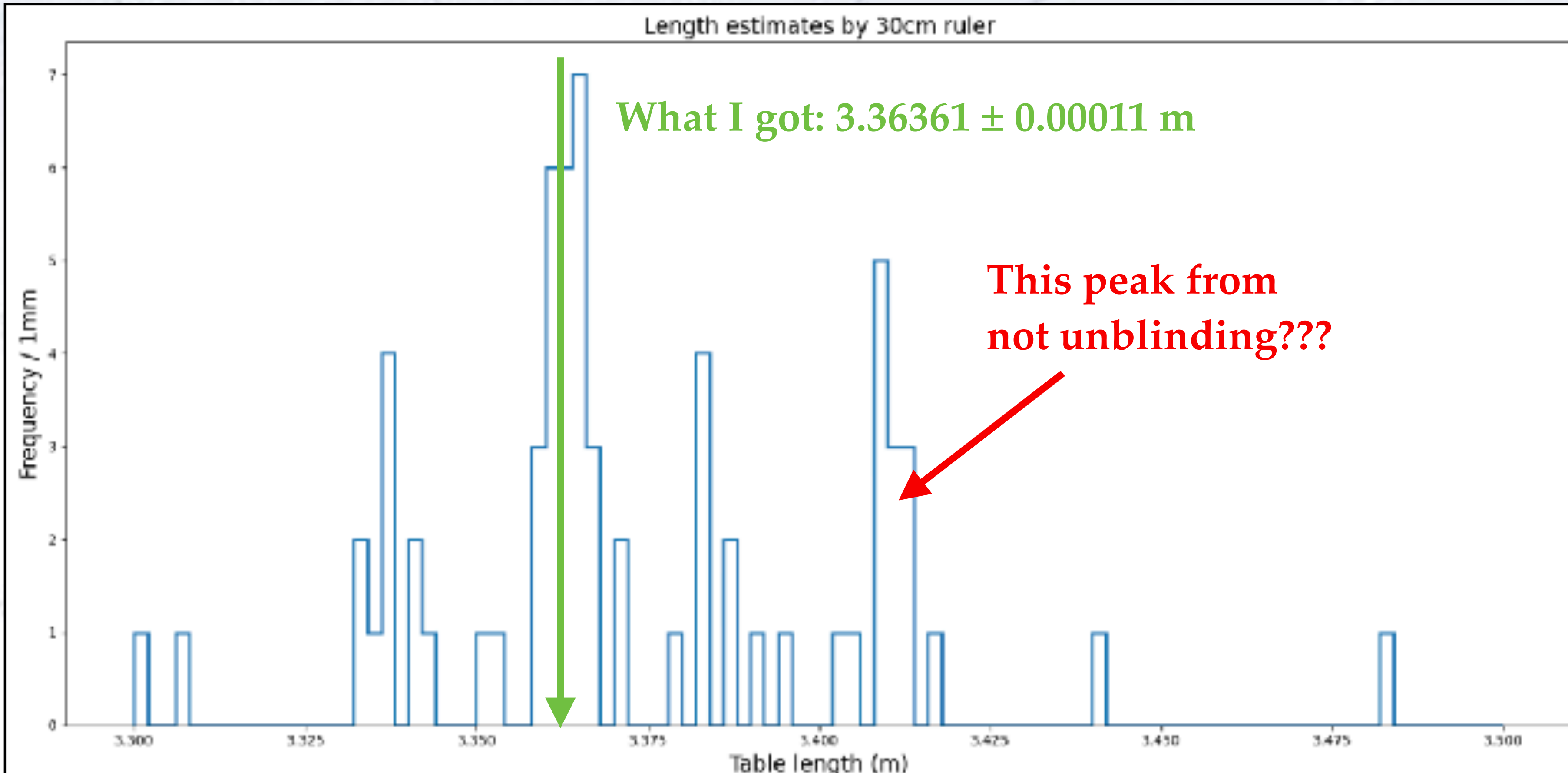
The uncertainties varied quite a bit…. from 3.3 to beyond 3.4.



Estimating uncertainties is (still) hard.

Your measurement uncertainty

The uncertainties varied quite a bit…. from 3.3 to beyond 3.4.



Estimating uncertainties is (still) hard.

The Quick & Dirty

The quick and dirty solution(s):

The above analysis is some work, but once you get the hang of it, and have previous produced (or copied understood) code for the task, it is less cumbersome. Once you see a plot of the data and understand what is happening, **the essense of this data analysis is to only consider the reasonable measurements** and extract a value from these.

The below code does that in a quick and dirty manner, which fails to do all the checks that are needed, if the data is important and the situation calls for it.

```
# Looking at the initial (30cm in particular) plots, it is clear that there is a central peak of valid measurements.
# By eye, the 30cm / 2m peak is at 3.415m / 3.350m (blinded!) and the width about 2.5cm / 1cm, so discarting all measurements
# outside +- 7.5cm / 3cm is a crude but fast way forward.
m30cm = np.abs(L30cm - 3.415) < 0.075
mu30cm = np.mean(np.array(L30cm)[m30cm])
sig30cm = np.std(np.array(L30cm)[m30cm])
print(f"  The crude (unweighted) mean = {mu30cm:7.5f} +- {sig30cm/np.sqrt(len(L30cm)):7.5f} m" +
      f"  (brutally raw (gu)estimate) from {len(np.array(L30cm)[m30cm]):d} measurements")

m2m = np.abs(L2m - 3.350) < 0.030
mu2m = np.mean(np.array(L2m)[m2m])
sig2m = np.std(np.array(L2m)[m2m])
print(f"  The crude (unweighted) mean = {mu2m:7.5f} +- {sig2m/np.sqrt(len(L2m)):7.5f} m" +
      f"  (brutally raw (gu)estimate) from {len(np.array(L2m)[m2m]):d} measurements")
```

```
  The crude (unweighted) mean = 3.41055 +- 0.00057 m  (brutally raw (gu)estimate) from 705 measurements
  The crude (unweighted) mean = 3.34932 +- 0.00022 m  (brutally raw (gu)estimate) from 801 measurements
```

However, this is approach is not advicable. Give it a little more consideration, and **promise yourself that you'll at the very minimum always do the following three key things**:

1. Blind the data
2. Plot the data
3. Make reassuring cross check and info print statements throughout your code.

By doing these things, you might only have to debug your way out of the unforeseen (e.g. negative uncertainties) to get to a decent result, that you can convince yourself and others is in the right ballpark. Good luck.

Notice that even though there has been a lot of analysis, comparison, and discussion of the result, the actual value of the table length has not yet been unblinded!

Conclusions

<u>Specifically on the analysis:</u>

- Greatest improvement came from simply removing mis-measurements!
- Weighted result was a further improvement, but required good uncertainties.
- The uncertainties are accepted as "reasonable", as they have good pull distributions, and improve the result.
- The 30cm and 2m results match, giving credibility to the stated precision.

<u>More generally:</u>

- What appears to be a trivial task, turns out to require some thought anyhow. (Ask yourself how many fellow students would have been able to get a good result and error?)
- There were several choices to be made in the analysis:
  1. Which measurements to accept.
  2. Which uncertainties to accept.
  3. To correct or discard understood mis-measurements.
- All this can be solved with simple Python code.