

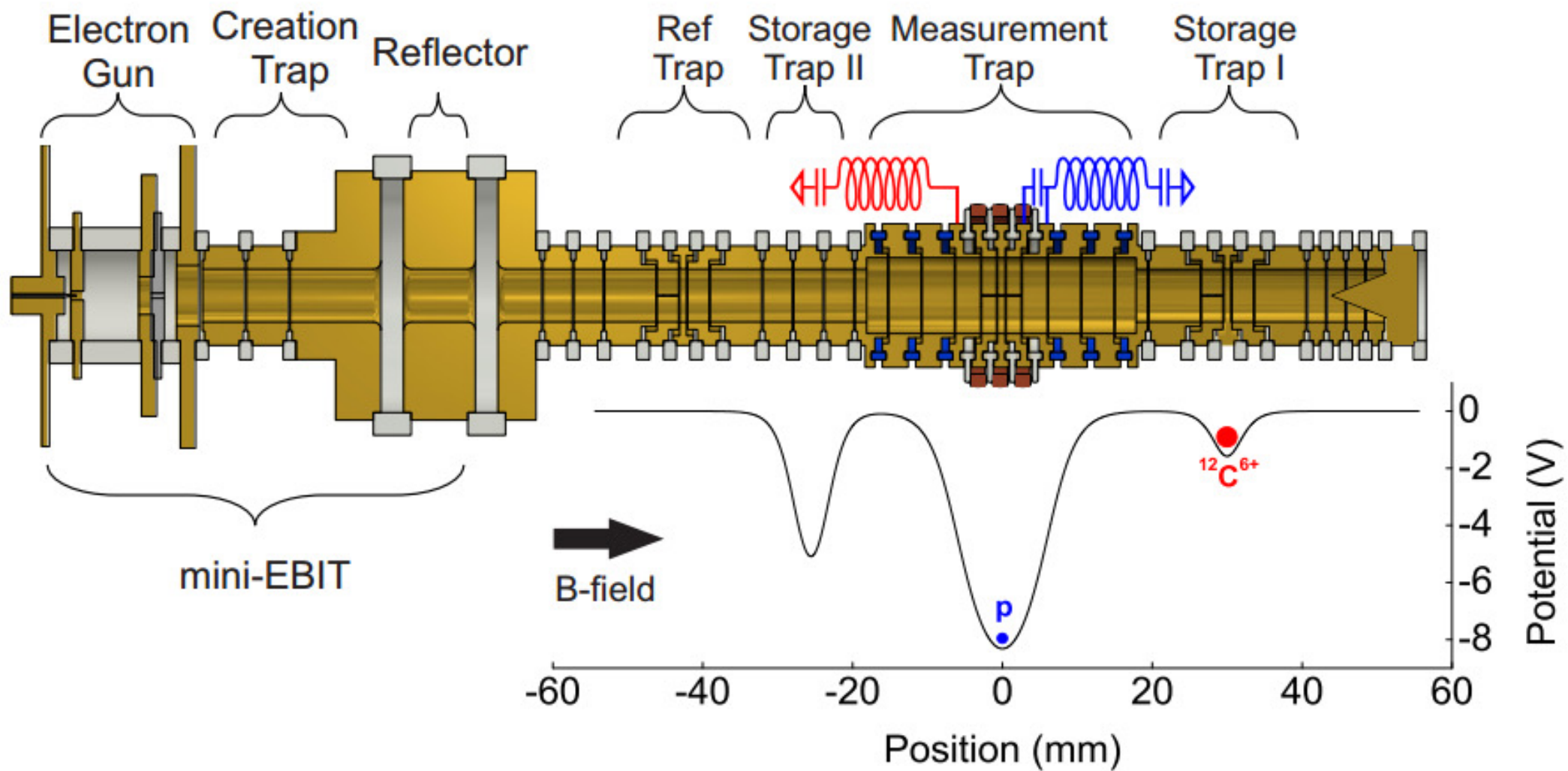
# DATA221 Intro Machine Learning

## 04 Gaussians

William Trimble  
Winter 2023

# Plan

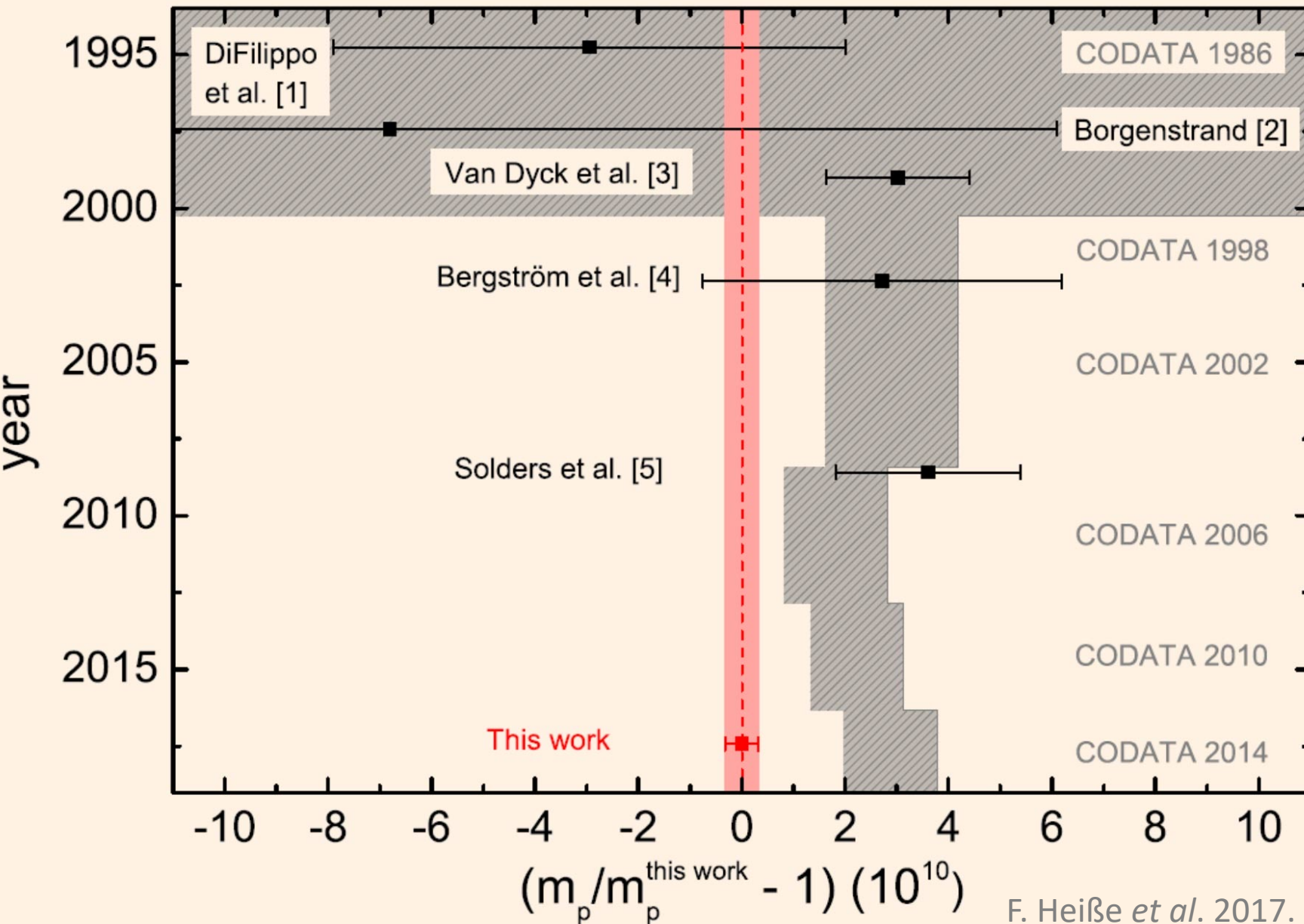
- Least Squares Adjustment (demo)
- Definitions
- Anatomy of the Gaussian distribution
- Fitting one-dimensional Gaussian mixture distribution



This is a balance that compares the mass of the proton to the mass of  $^{12}\text{C}^{6+}$

F. Heiße *et al.* 2017. High-Precision Measurement of the Proton's Atomic Mass. *Phys. Rev. Lett* 119 (3): 033001; doi: 10.1103/PhysRevLett.119.033001

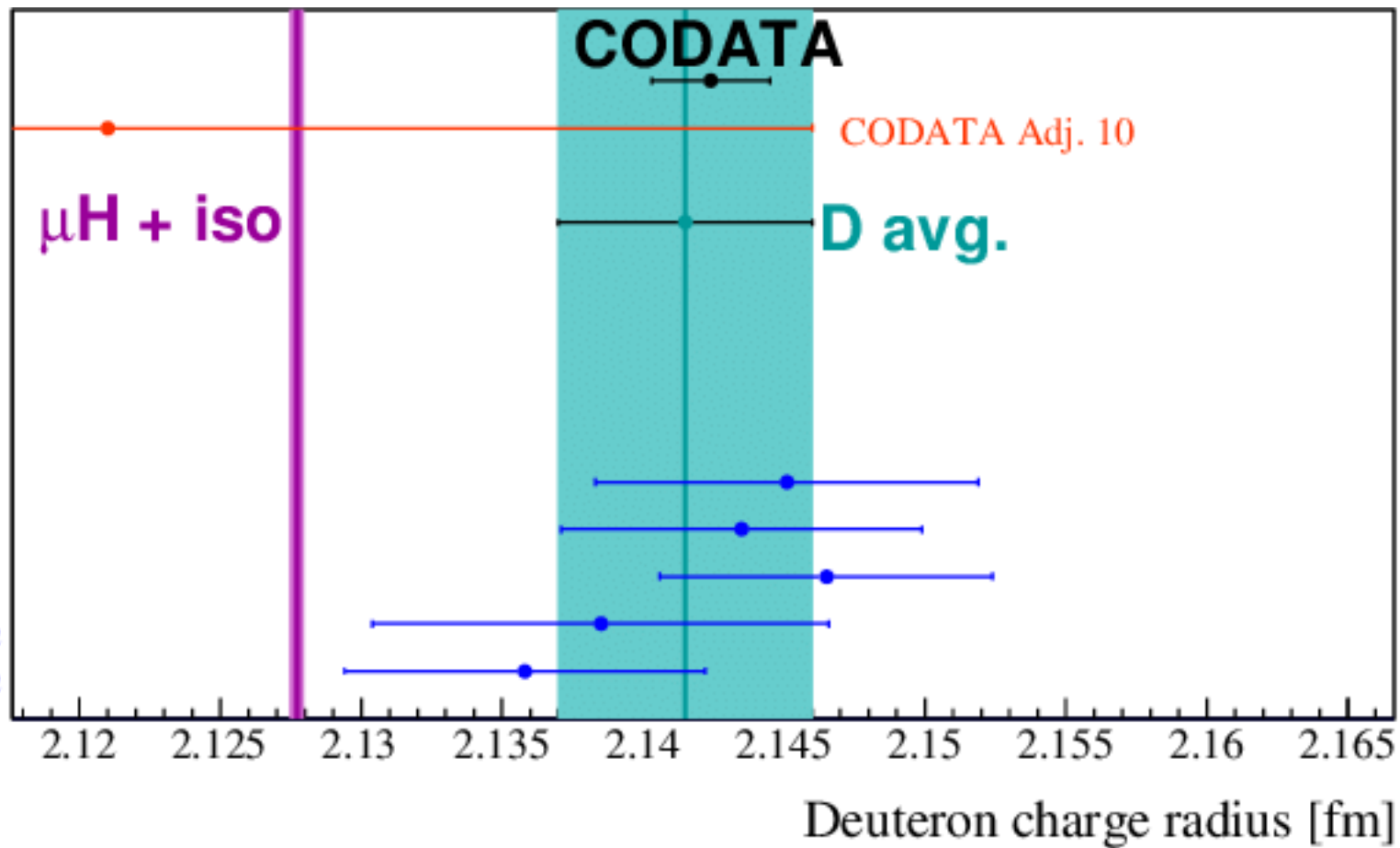




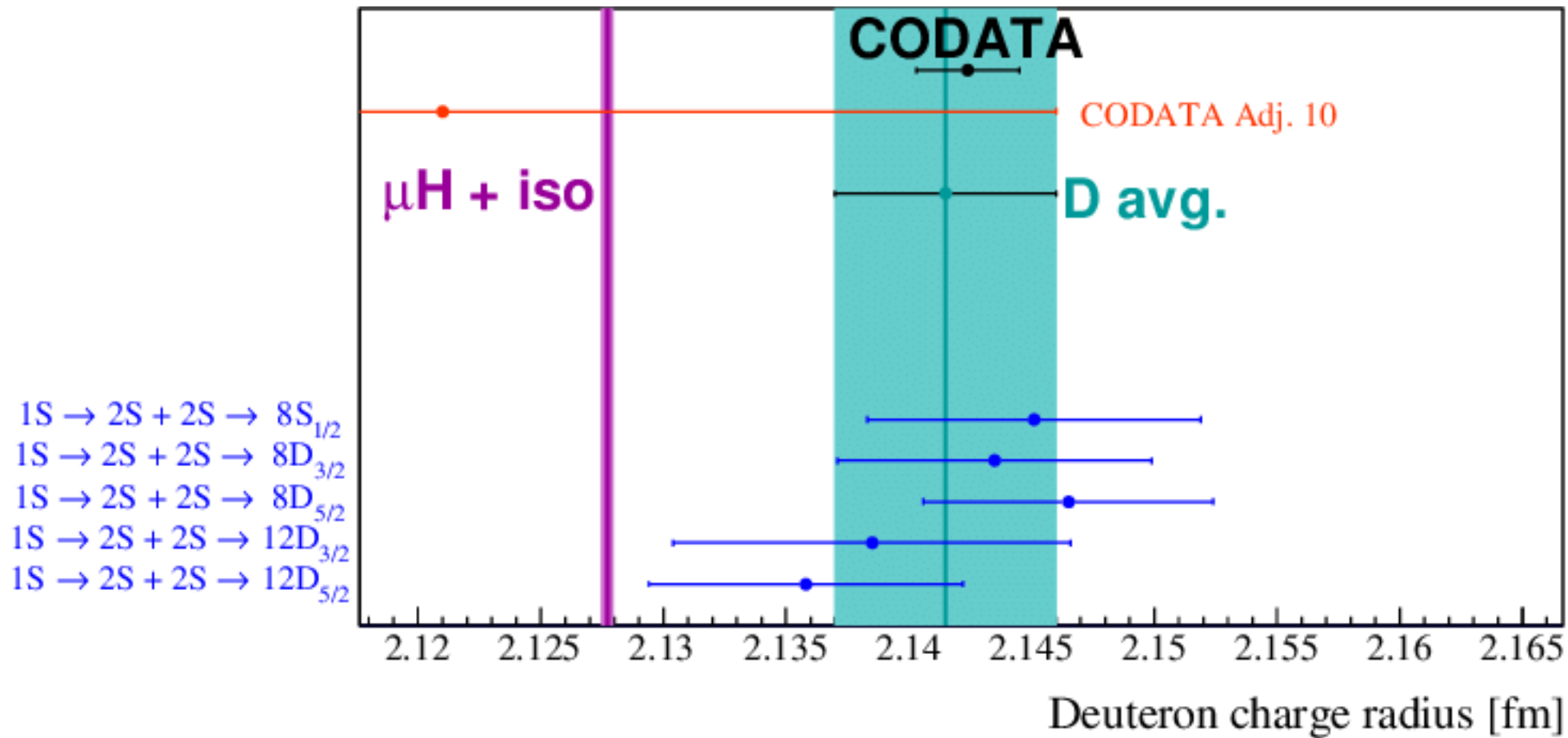
Multiple researchers, with advancing technology, have taken cracks at measuring the mass of the proton.

F. Heiße *et al.* 2017. High-Precision Measurement of the Proton's Atomic Mass. *Phys. Rev. Lett* 119 (3): 033001; doi: 10.1103/PhysRevLett.119.033001

$1S \rightarrow 2S + 2S \rightarrow 8S_{1/2}$   
 $1S \rightarrow 2S + 2S \rightarrow 8D_{3/2}$   
 $1S \rightarrow 2S + 2S \rightarrow 8D_{5/2}$   
 $1S \rightarrow 2S + 2S \rightarrow 12D_{3/2}$   
 $1S \rightarrow 2S + 2S \rightarrow 12D_{5/2}$



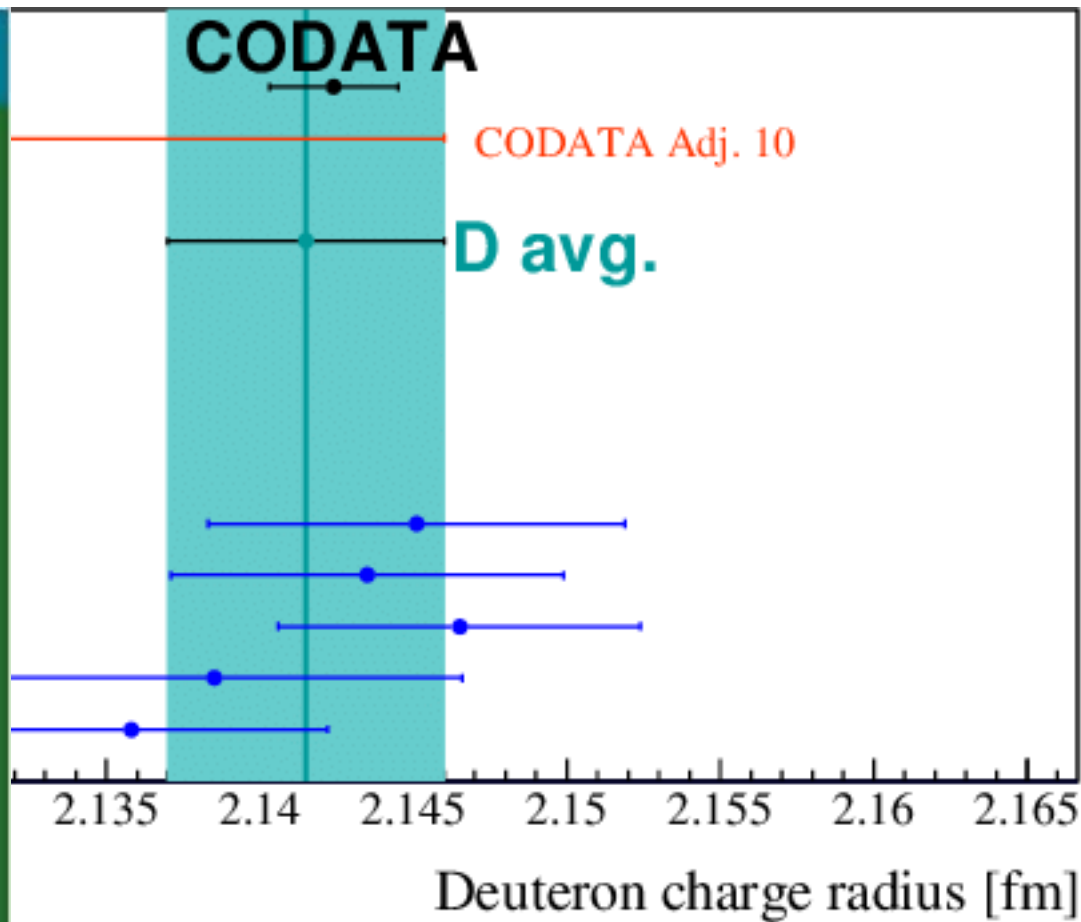
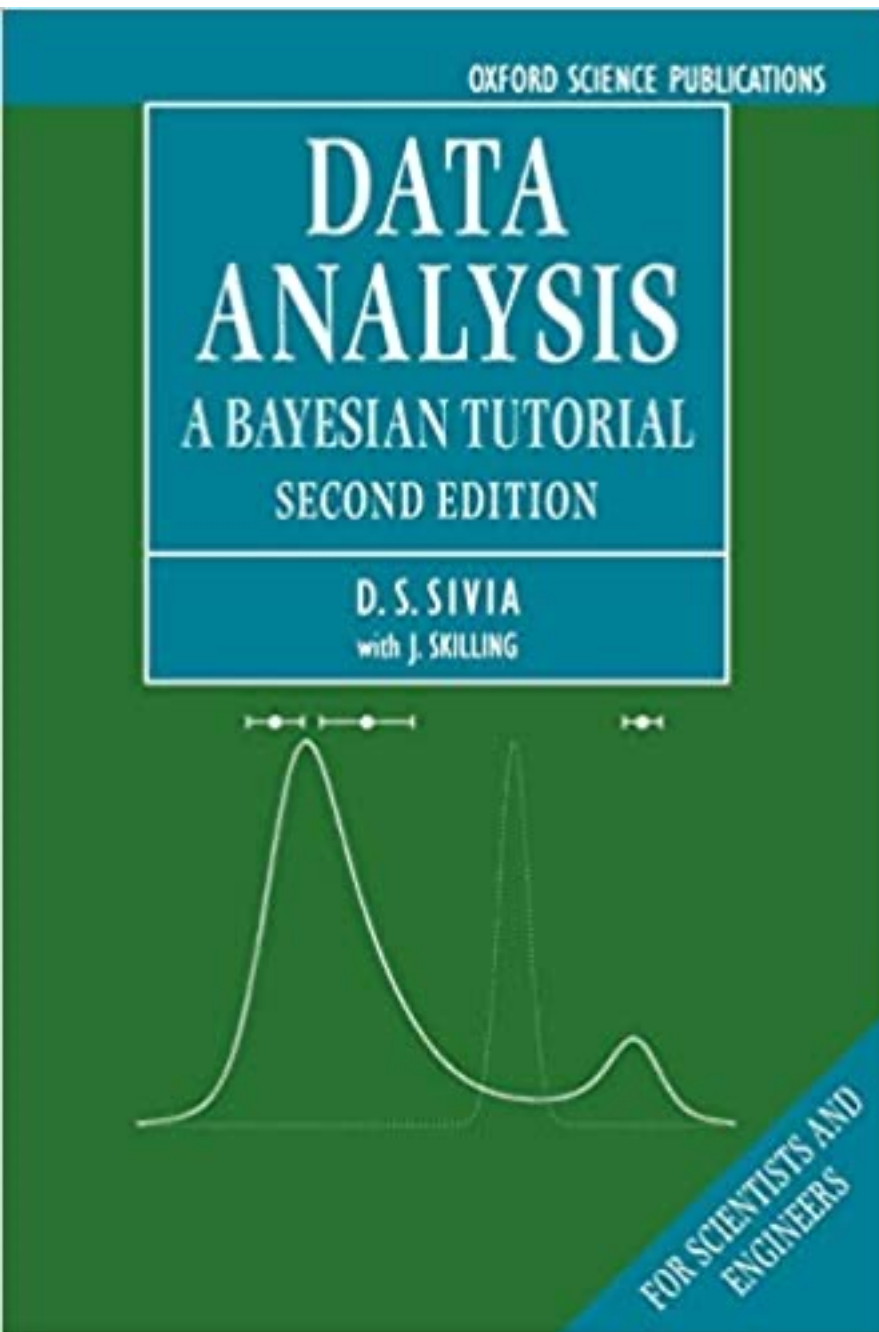
In a different measurement campaign, there are multiple (technically different) measurement approaches to the deuteron charge radius.



In a different measurement campaign, there are multiple (technically different) measurement approaches to the deuteron charge radius.

In a measurement-evaluation doctrine called “Least Squares Adjustment,” information about the physical constant (deuteron charge radius) is combined from multiple experiments using a very lightweight version of Bayesian inference.

Goes by the term “metaanalysis” in other fields.

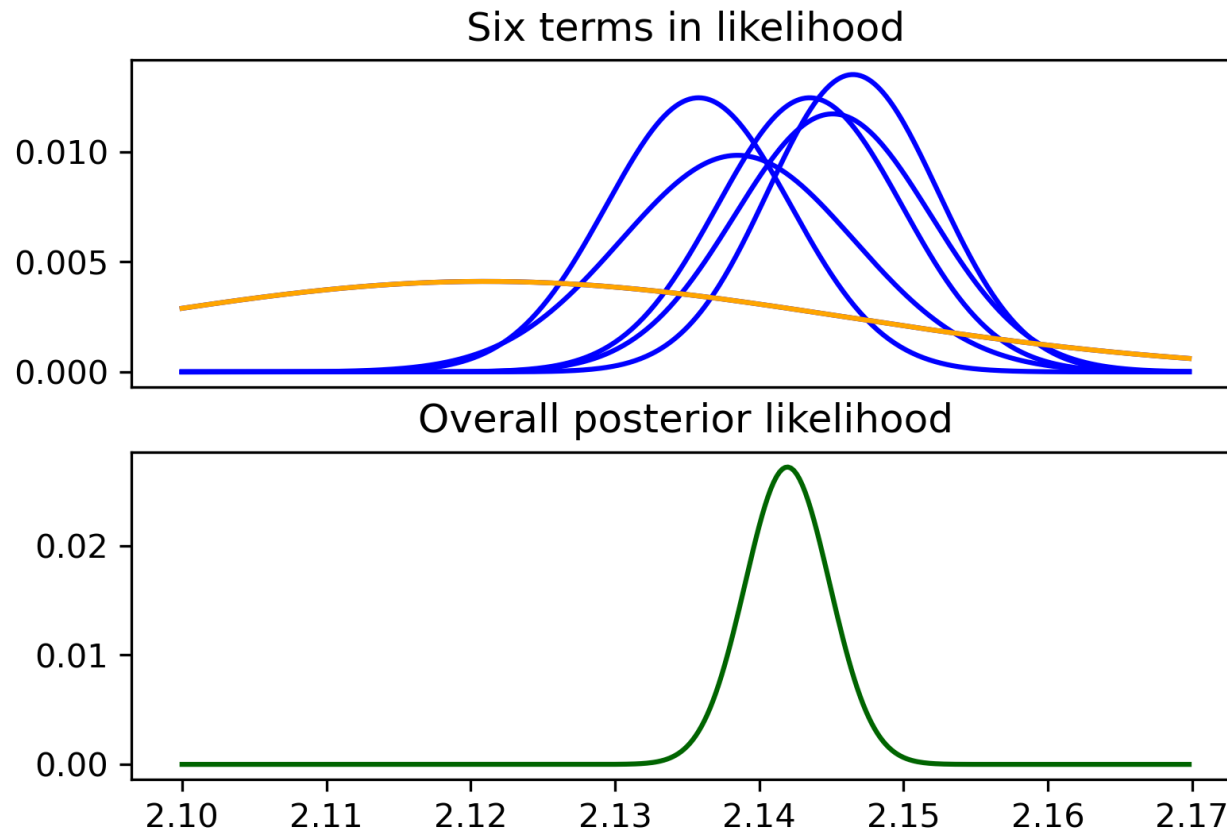


In a different measurement campaign, there are multiple (technically different) measurement approaches to the deuteron charge radius.

ine called “Least Squares Adjustment,” information (on charge radius) is combined from multiple  
nt version of Bayesian inference.

other fields.

# Least-squares adjustment with normal posteriors



This is the simplest Bayesian inference problem I can think of with a one-dimensional, continuous variable; the probability density at every stage is a well-behaved normal distribution.

The log-likelihoods are parabolas.

Caution: this weighted-average approach takes the standard deviations as known with certainty.

$$\text{Posterior}(x) = P_1(x) P_2(x) P_3(x) P_4(x) P_5(x) \text{Prior}(x)$$

LEASTSQADJUST.ipynb



# Least-squares adjustment

$$\begin{aligned}\text{Prior}(x) &= N(m_0, s_0) = \frac{1}{\sqrt{2\pi s_0^2}} e^{-\frac{(x-m_0)^2}{2s_0^2}} \\ \text{Likelihood}_1(x) &= N(m_1, s_1) = \frac{1}{\sqrt{2\pi s_1^2}} e^{-\frac{(x-m_1)^2}{2s_1^2}}\end{aligned}$$

In one dimension, imagine I have a variable  $x$ .

I want the best number I can get for  $P(x | \text{DATA})$  (the posterior)  
but all I have are a basket full of  $p(\text{DATA}_i | x)$ . (the likelihoods)

Bayes' rule to the rescue!

$$P(x | \text{DATA}) = P(x) P(\text{DATA} | x) / p(\text{DATA})$$

# Least-squares adjustment

$$\text{Prior}(x) = N(m_0, s_0) = \frac{1}{\sqrt{2\pi s_0^2}} e^{-\frac{(x-m_0)^2}{2s_0^2}}$$

$$\text{Likelihood}_1(x) = N(m_1, s_1) = \frac{1}{\sqrt{2\pi s_1^2}} e^{-\frac{(x-m_1)^2}{2s_1^2}}$$

$$\text{Likelihood}_2(x) = N(m_2, s_2) = \frac{1}{\sqrt{2\pi s_2^2}} e^{-\frac{(x-m_2)^2}{2s_2^2}}$$

$$\text{Likelihood}_3(x) = N(m_3, s_3) = \frac{1}{\sqrt{2\pi s_3^2}} e^{-\frac{(x-m_3)^2}{2s_3^2}}$$

...

$$\text{Posterior} = \text{Prior} * L1 * L2 * L3 \dots$$

Take prior to be constant for now; take logs of all the likelihoods:

Prior(x) = constant

$$\text{Likelihood}_1(x) = N(m_1, s_1) = \frac{1}{\sqrt{2\pi s_1^2}} e^{-\frac{(x-m_1)^2}{2s_1^2}}$$

$$\text{Likelihood}_2(x) = N(m_2, s_2) = \frac{1}{\sqrt{2\pi s_2^2}} e^{-\frac{(x-m_2)^2}{2s_2^2}}$$

$$\text{Likelihood}_3(x) = N(m_3, s_3) = \frac{1}{\sqrt{2\pi s_3^2}} e^{-\frac{(x-m_3)^2}{2s_3^2}}$$

...

Posterior = Prior \* L1 \* L2 \* L3 ...

# Least-squares adjustment

$$\log \text{Prior}(x) = \log C = C$$

$$\text{LLikelihood\_1}(x) = \log N(m_1, s_1) = -\frac{1}{2} \log(2\pi s_1^2) - \frac{(x-m_1)^2}{2s_1^2}$$

$$\text{LLikelihood\_2}(x) = \log N(m_2, s_2) = -\frac{1}{2} \log(2\pi s_2^2) - \frac{(x-m_2)^2}{2s_2^2}$$

$$\text{LLikelihood\_3}(x) = \log N(m_3, s_3) = -\frac{1}{2} \log(2\pi s_3^2) - \frac{(x-m_3)^2}{2s_3^2}$$

...

$$\log \text{Posterior} = \log \text{Prior} + \text{LL1} + \text{LL2} + \text{LL3} \dots$$



# Least-squares adjustment

$$\log \text{Prior}(x) = \log C = C$$

$$\text{LLikelihood\_1}(x) = \log N(m_1, s_1) = -\frac{1}{2} \log(2\pi s_1^2) - \frac{(x-m_1)^2}{2s_1^2}$$

$$\text{LLikelihood\_2}(x) = \log N(m_2, s_2) = -\frac{1}{2} \log(2\pi s_2^2) - \frac{(x-m_2)^2}{2s_2^2}$$

$$\text{LLikelihood\_3}(x) = \log N(m_3, s_3) = -\frac{1}{2} \log(2\pi s_3^2) - \frac{(x-m_3)^2}{2s_3^2}$$

...

While these are straightforward functions of the  $s_i$ , these are just parabolas as a function of  $x$ . Sum of parabolas is a ...

# Least-squares adjustment

$$\log \text{Prior}(x) = \log C = C$$

$$\text{Likelihood}_1(x) = \log N(m_1, s_1) = -\frac{1}{2} \log(2\pi s_1^2) - \frac{(x-m_1)^2}{2s_1^2}$$

reorganize terms in powers of x

$$\log \text{Posterior} = -\frac{1}{2} \sum \frac{1}{s_i^2} x^2 + \frac{2}{2} \sum \frac{m_i}{s_i^2} x - \frac{1}{2} \sum \frac{m_i^2}{s_i^2}$$

$$\frac{d \log(\text{posterior})}{dx} = -\sum \frac{1}{s_i^2} x + \sum \frac{m_i}{s_i^2} = 0$$

$$x_{\max} = \frac{\sum \frac{m_i}{s_i^2}}{\sum \frac{1}{s_i^2}} \quad S_{\text{tot}} = \frac{1}{\sqrt{\sum \frac{1}{s_i^2}}}$$

# Ta-da! (Undergrad-level combination of measurements)

$$x_{\max} = \frac{\sum \frac{m_i}{s_i^2}}{\sum \frac{1}{s_i^2}} \quad s_{\text{tot}} = \frac{1}{\sqrt{\sum \frac{1}{s_i^2}}}$$

When you have a collection of normally-distributed likelihoods with known means and standard deviations

(These represent measurements and someone's judgement of the uncertainty attached to each measurement.)

We have a procedure that turns likelihood functions into algebra

And the product of N normal distributions is a normal distribution that is much tighter.

# Ta-da! (Undergrad-level combination of measurements)

$$x_{\max} = \frac{\sum \frac{m_i}{s_i^2}}{\sum \frac{1}{s_i^2}} \quad s_{\text{tot}} = \frac{1}{\sqrt{\sum \frac{1}{s_i^2}}} \quad \text{Posterior (x)} = N(x_{\max}, s_{\text{tot}})$$

When you have a collection of normally-distributed likelihoods with known means and standard deviations

(These represent measurements and someone's judgement of the uncertainty attached to each measurement.)

We have a procedure that turns likelihood functions into algebra

And the product of N normal distributions is a normal distribution that is much tighter.

Caution: this procedure does not know or care whether the measurements agree with each other.



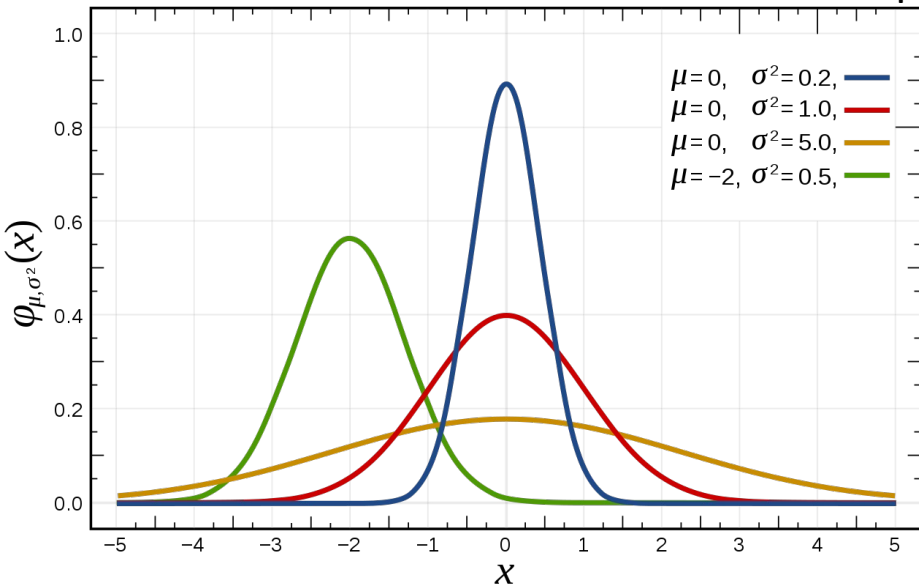
# Anatomy of the normal distribution

$$\mathcal{N}(x|\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp -\frac{(x-\mu)^2}{2\sigma^2}$$

“The normalizing constant”

x- dependence  
shape and position  
of distribution

peak density proportional to  $1/\sigma$



# Anatomy of the normal distribution

$$\log \mathcal{N}(x|\mu, \sigma) = -\frac{1}{2} \log(2\pi) - \log \sigma - \frac{(x-\mu)^2}{2\sigma^2}$$

Constant term,  
not going to give  
us trouble

Term depends  
only on  $\sigma$

Term depends on  
 $\sigma$  and  $\mu$

# Anatomy of the normal distribution

$$\log \mathcal{N}(x|\mu, \sigma) = -\frac{1}{2} \log(2\pi) - \log \sigma - \frac{(x-\mu)^2}{2\sigma^2}$$

Constant term,  
not going to give  
us trouble

Term depends  
only on  $\sigma$

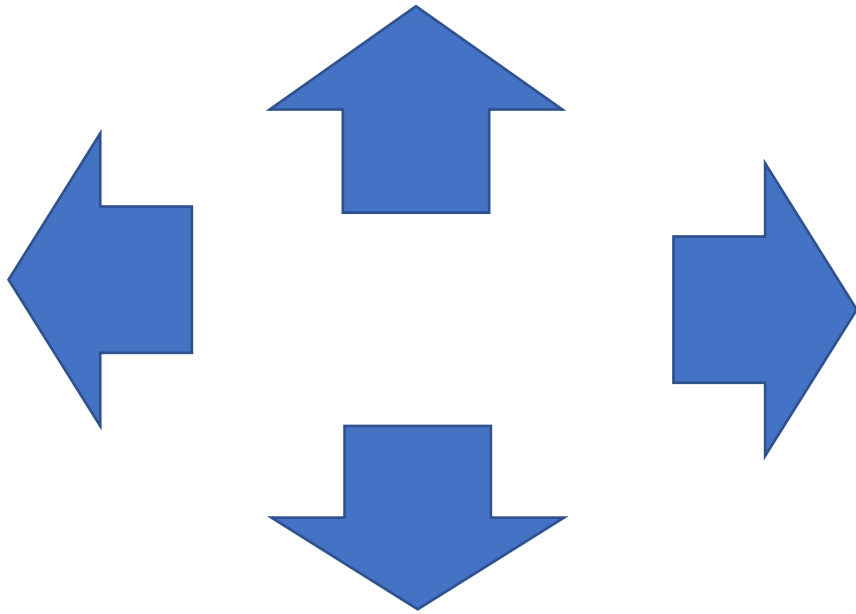
Term depends on  
 $\sigma$  and  $\mu$

Penalizes  
distributions that  
are too spread  
out

Tells us where  
the center is

# Anatomy of the normal distribution

$$\log \mathcal{N}(x|\mu, \sigma) = -\frac{1}{2} \log(2\pi) - \log \sigma - \frac{(x-\mu)^2}{2\sigma^2}$$



When  $\sigma$  is too large, this term makes likelihood small

When  $\sigma$  is too small, this term penalizes the likelihood because the distribution will miss the data.



Normal( $x_1$ ) x Normal ( $x_2$ ) ??

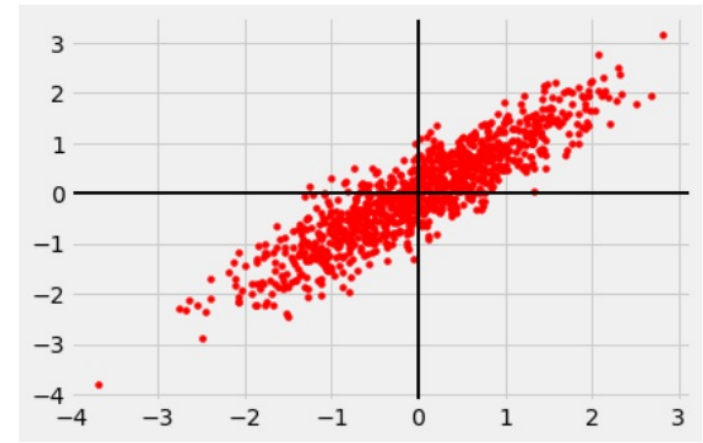
$$\mathcal{N}(x_1, x_2 | \mu_1, \mu_2, \sigma_{11}, \sigma_{22}) = \frac{1}{(2\pi)} \frac{1}{\sigma_{11}\sigma_{22}} \exp -\frac{1}{2} \left( \frac{(x_1 - \mu_1)^2}{\sigma_{11}} + \frac{(x_2 - \mu_2)^2}{\sigma_{22}} \right)$$

If I take the product of two 1-D normal distributions, I get a perfectly good probability distribution, but it isn't quite flexible enough for my data-modeling needs.

This factorizes into the product of a term that depends only on  $x_1$  and  $\sigma_1$  and a term only in  $x_2$  and  $\sigma_2$ .

This expression doesn't permit me to make distributions where the probability density of  $x_1$  depends on the value of  $x_2$

# Multivariate normal



$$\Sigma = \begin{bmatrix} \sigma_{11}^2 & \sigma_{12}^2 \\ \sigma_{12}^2 & \sigma_{22}^2 \end{bmatrix}$$

$$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{(\sigma_{11}^2 \sigma_{22}^2 - \sigma_{12}^2)} \exp -\frac{1}{2} \left( \begin{bmatrix} x_0 - \mu_0 & x_1 - \mu_1 \end{bmatrix} \begin{bmatrix} \sigma_{11}^2 & \sigma_{12}^2 \\ \sigma_{12}^2 & \sigma_{22}^2 \end{bmatrix}^{-1} \begin{bmatrix} x_0 - \mu_0 \\ x_1 - \mu_1 \end{bmatrix} \right)$$

Constant term,  
not going to give  
us trouble

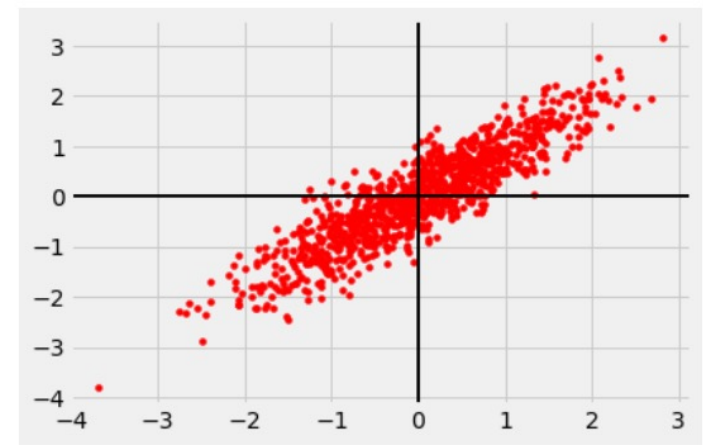
Term depends  
only on  $\Sigma$

Two powers of  $\mathbf{x}-\boldsymbol{\mu}$

# Multivariate normal

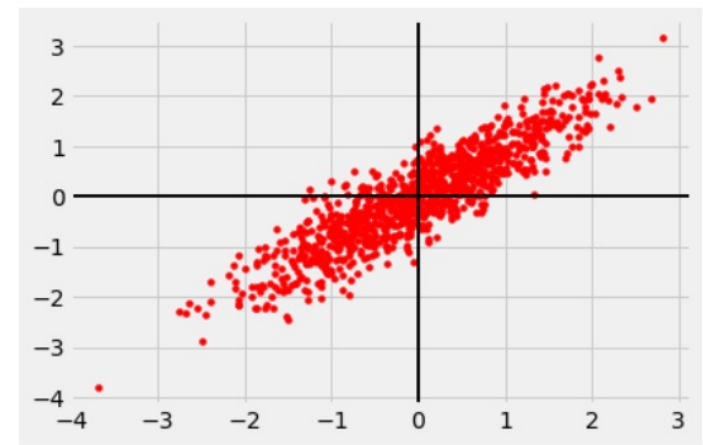
$$\Sigma = \begin{bmatrix} \sigma_{11}^2 & \cdots & \sigma_{1D}^2 \\ \vdots & \ddots & \vdots \\ \sigma_{D1}^2 & \cdots & \sigma_{DD}^2 \end{bmatrix}$$

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}|^{D/2}} \exp -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})$$



# Multivariate normal

$$\Sigma = \begin{bmatrix} \sigma_{11}^2 & \cdots & \sigma_{1D}^2 \\ \vdots & \ddots & \vdots \\ \sigma_{D1}^2 & \cdots & \sigma_{DD}^2 \end{bmatrix}$$

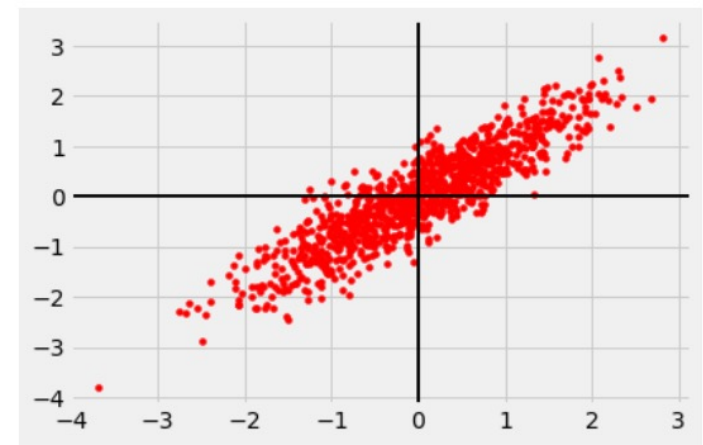


$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}|^{D/2}} \exp -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})$$

How many parameters do I need for a D-dimensional multivariate Gaussian ?

# Multivariate normal

$$\Sigma = \begin{bmatrix} \sigma_{11}^2 & \cdots & \sigma_{1D}^2 \\ \vdots & \ddots & \vdots \\ \sigma_{D1}^2 & \cdots & \sigma_{DD}^2 \end{bmatrix}$$

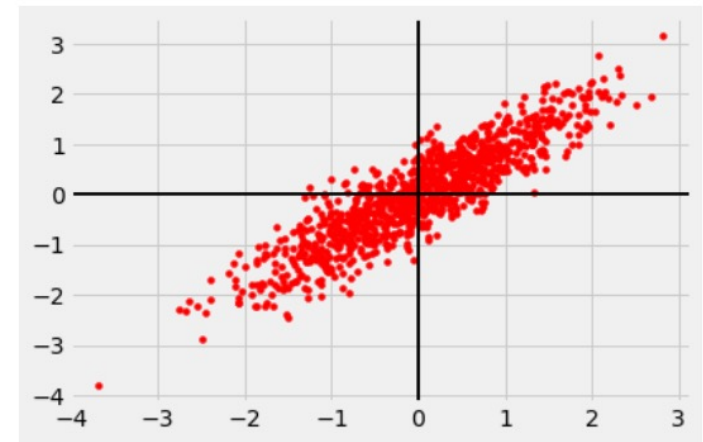


$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}|^{D/2}} \exp -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})$$

How many parameters do I need for a D-dimensional multivariate Gaussian ?

$$\boldsymbol{\mu} : D \quad \boldsymbol{\Sigma} : (D^2 + D)/2$$

# Multivariate normal



$$\mathcal{N}(\mathbf{x}|\mathbf{x}_0, \mathbf{\Sigma}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\mathbf{\Sigma}|^{D/2}} \exp -\frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T \mathbf{\Sigma}^{-1}(\mathbf{x} - \mathbf{x}_0)$$

Constant term,  
not going to give  
us trouble

Term depends  
only on  $\Sigma$

Two powers of  $\mathbf{x}-\mathbf{x}_0$

$\Sigma$  covariance matrix

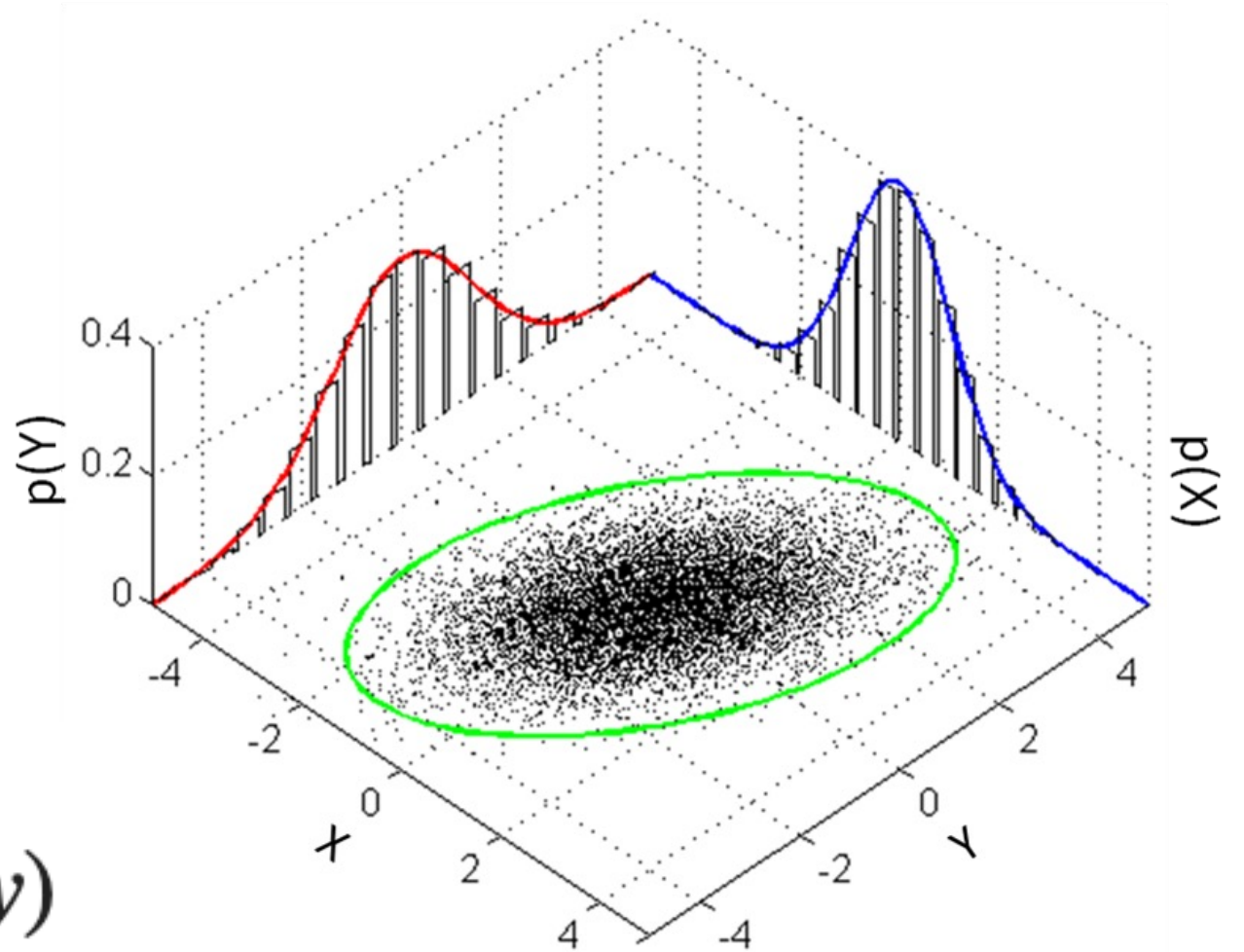
$\Sigma^{-1}$  “precision” matrix

# Jargon: Marginalization

To “marginalize” is to take a distribution of many variables and turn it into a distribution of fewer variables by integrating the product of the density and the prior probability of each of the unwanted variables.

When we are able to do this, this gives us the one-dimensional probability distribution (for the parameter of interest) that we seek, in essence averaging over all likely values of the variables not of interest.

$$P(x) = \int dy P(x|y)P(y)$$



# Unsupervised learning

I don't have labels, just data  $\{\mathbf{x}_i\}_{i=1}^N$

My model is a function that does something useful with new data.

Tasks may include

**detecting outliers** (points where the density is low; **density estimation**)

**dimensionality reduction** (produce coordinates in a low-dimensional space; compress the data while retaining its essence)

**clustering** (output is a cluster id)



# Supervised learning

Supervised learning: the **dataset** is a collection of **labeled examples**  $\{ \mathbf{x}_i, y_i \}_{i=1}^N$

N examples

$\mathbf{x}$  is called a **feature vector** and

$y$  is the set of **labels**

$\mathbf{x}$ ,  $y$  can be in  $\mathbb{R}^n$  or can be discrete, categorical labels, or some more complex kind of input and output

Goal of supervised learning is to produce a model that turns a feature vector into an approximation of the label.

# Jargon

parameter = numbers that determine “the model.”

hyperparameter = number that controls the behavior of the learning algorithm

**regression** = prediction of a real-valued label

**fitting** = determining the parameters of the model

**classification** = prediction of a label from a finite set of labels

**shallow learning algorithm** – fits the parameters of the model from the features of the training data.

**deep learning algorithm** fits the parameters of the model from the output of layers that are fed by the training features.

# Generative vs. discriminative

If I fit a probability distribution to my data, I can evaluate

$P(\text{newdata} \mid \text{model})$  and

$P(\text{label} \mid \text{model}, \text{newdata})$

but I also can draw samples from the distribution. Models of this sort are called “generative” models because they permit me to “generate” new data consistent with the model parameters.

Models that just find the function of  $x$  that gives the right answer  $y$  (say, by fitting weights directly rather than finding weights from fitted distribution parameters) are called “discriminative” models.