# Mathematics of Big Data, I

# Lecture 8: Bayesian Learning, Bayesian Logistic and linear Regressions (review), Bayesian Inference, Intractable Integrals and Motivation for Approximate Methods, and Learning Theory

**Weiqing Gu**

Professor of Mathematics

Director of the Mathematics Clinic

Harvey Mudd College

Summer 2017

# Today's Topics

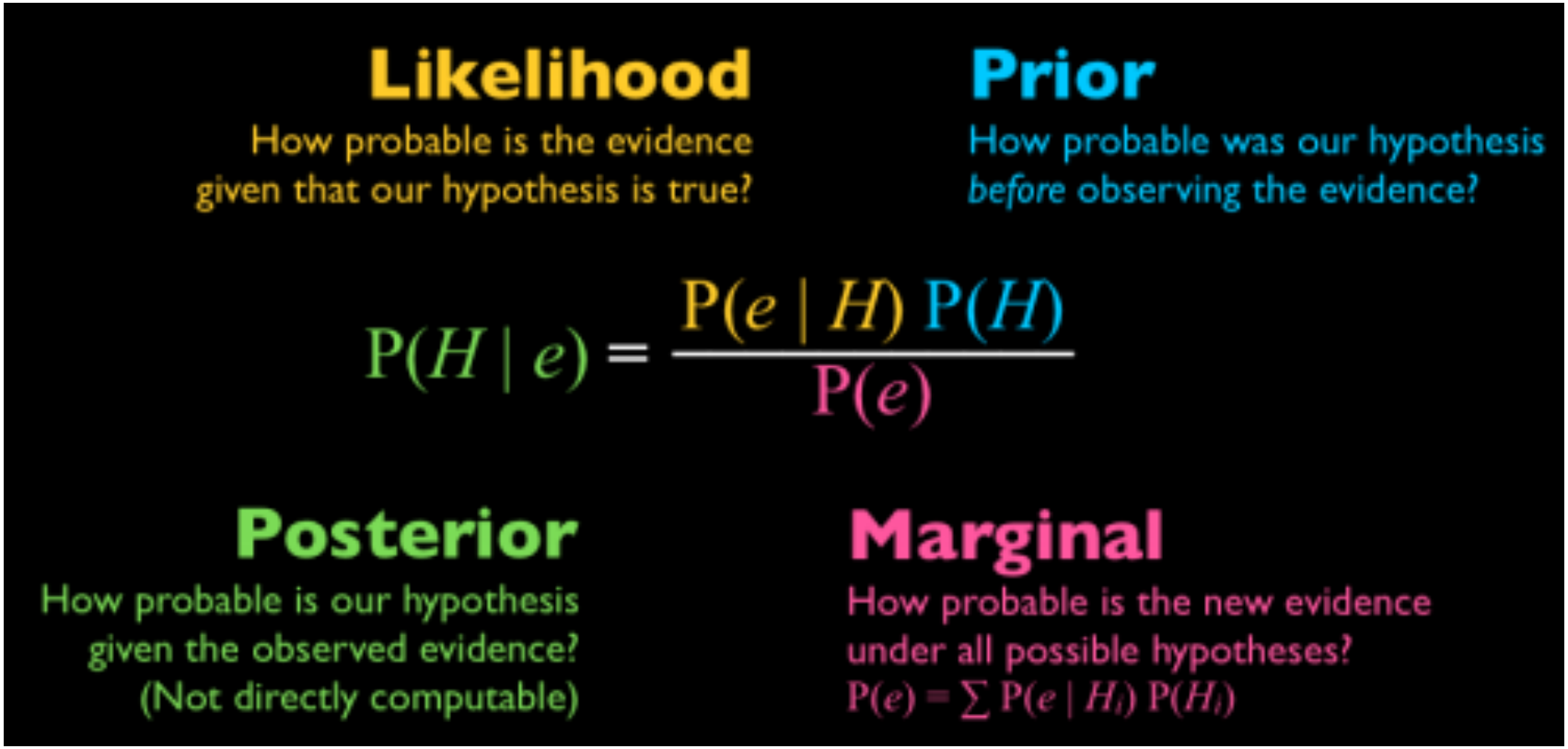
- Recap of Bayesian Reasoning
- Bayesian Linear Regression (which we've already seen)
- Bayesian Logistic Regression (Review)
- Bayesian Inference
- Intractable Integrals and Motivation for Approximate Methods (only if time permits)
- Learning Theory

# Today's Topics

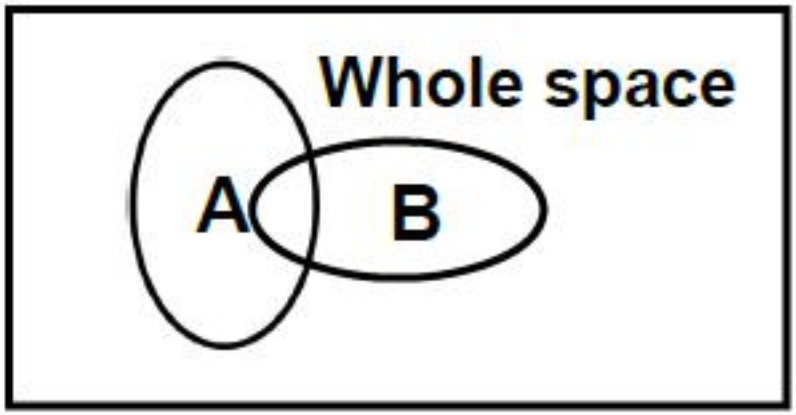
- **Recap of Bayesian Reasoning.**
- Bayesian Linear Regression (which we've already seen).
- Bayesian Logistic Regression (Review).
- Bayesian Inference.
- Intractable Integrals and Motivation for Approximate Methods (only if time permits).
- Learning Theory

# Let's Recap on
# Bayesian Reasoning/Bayesian Inference

- Key: Put distributions on everything and then use rules of probability!
- Recall again: **Bayes' Theorem**



# Visualize Bayes' Theorem




$$P(A) = \frac{\text{A}}{\text{Whole space}} \qquad P(B) = \frac{\text{B}}{\text{Whole space}}$$

$$P(A|B) = \frac{\text{A} \cap \text{B}}{\text{B}} \qquad P(B|A) = \frac{\text{A} \cap \text{B}}{\text{A}}$$

$$P(A \cap B) = \frac{\text{A} \cap \text{B}}{\text{Whole space}}$$

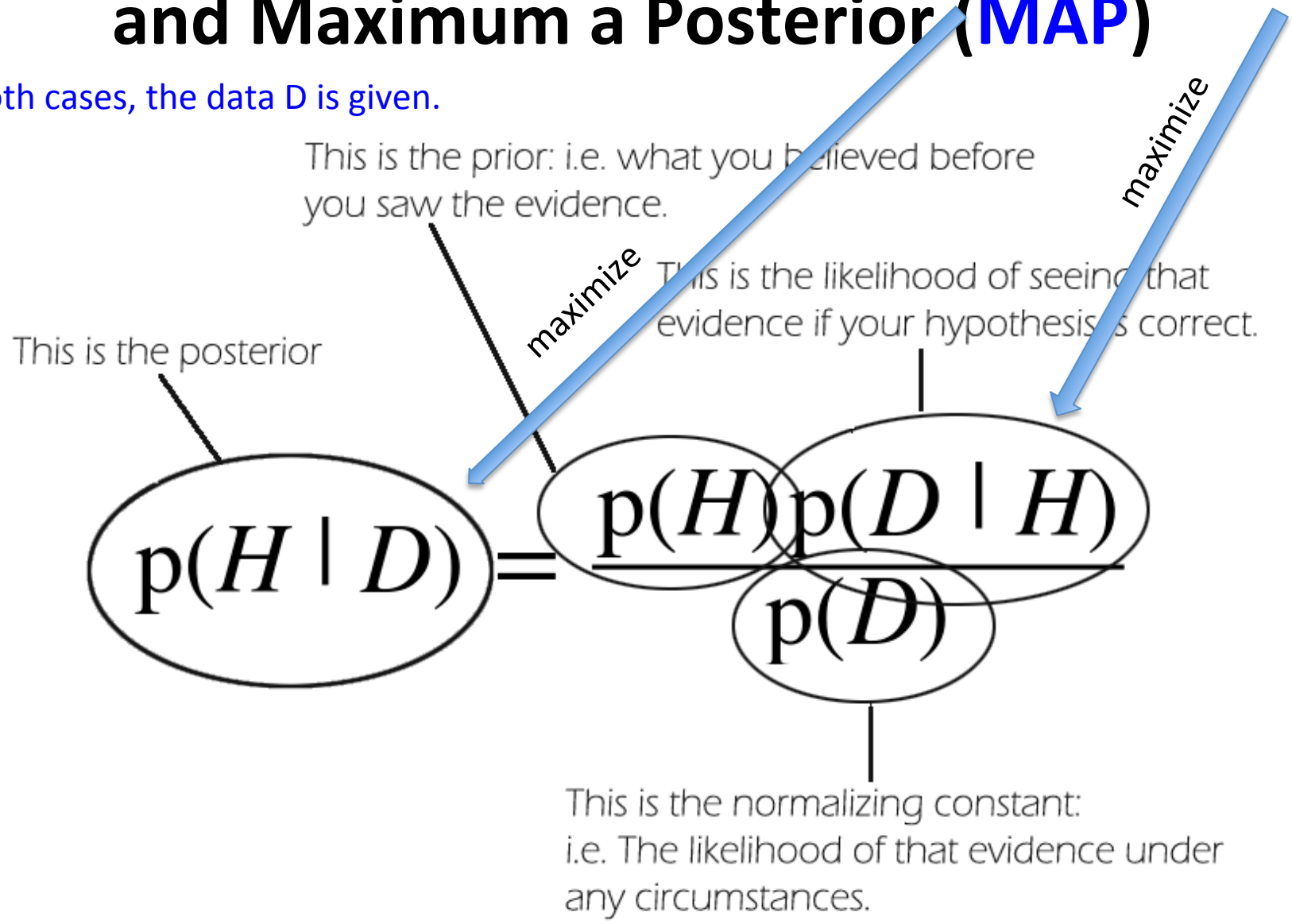
$$P(A) \times P(B|A) = \frac{\text{A}}{\text{Whole space}} \times \frac{\text{A} \cap \text{B}}{\text{A}} = \frac{\text{A} \cap \text{B}}{\text{Whole space}} = P(A \cap B)$$

$$P(B) \times P(A|B) = \frac{\text{B}}{\text{Whole space}} \times \frac{\text{A} \cap \text{B}}{\text{B}} = \frac{\text{A} \cap \text{B}}{\text{Whole space}} = P(A \cap B)$$

$$\Rightarrow P(B|A) = P(A|B) \times P(B) / P(A)$$

# Recall: Maximum Likelihood Estimation (MLE) and Maximum a Posterior (MAP)

In both cases, the data D is given.



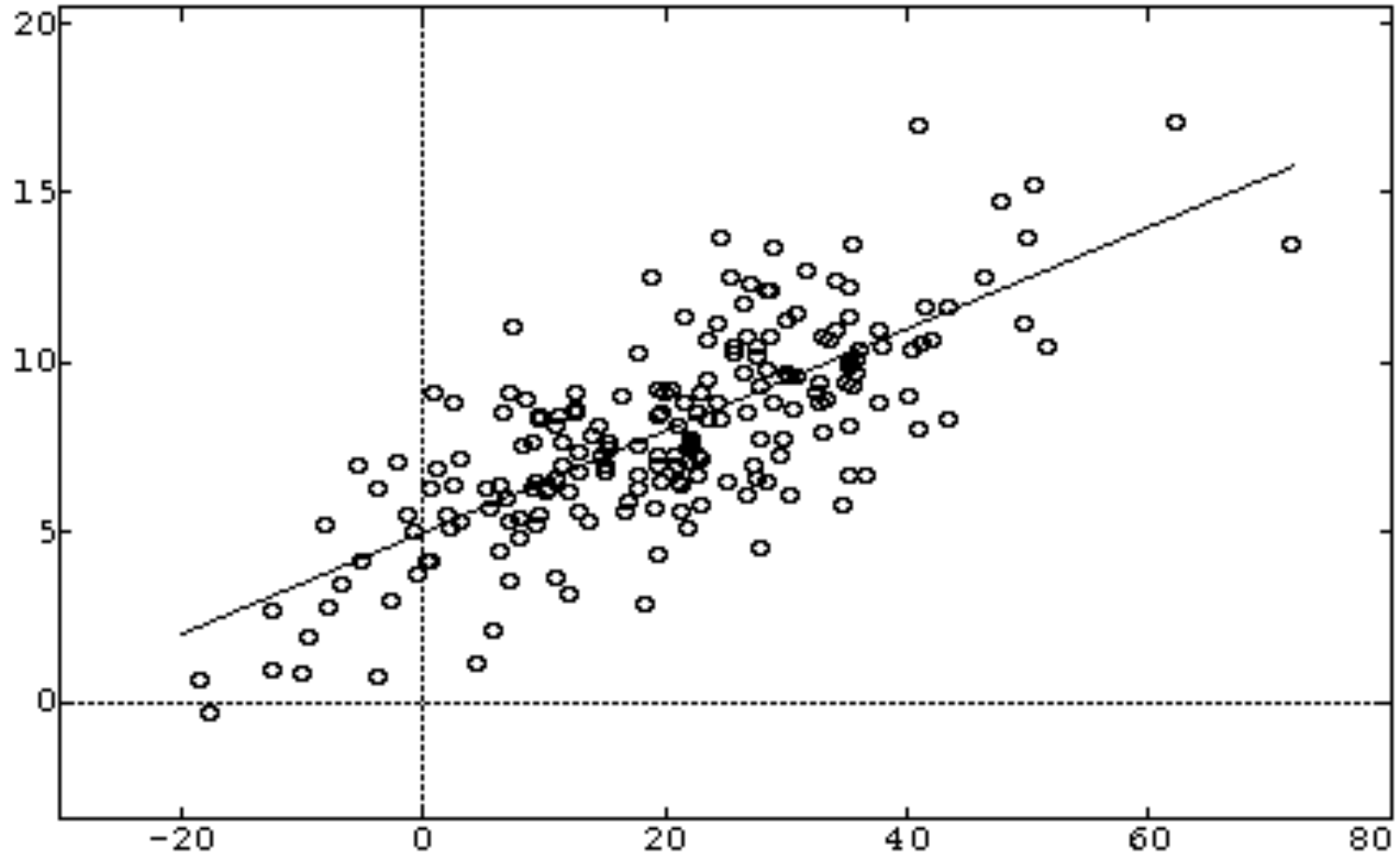
# Work out details with the students on the board

- **Maximum Likelihood Estimation (MLE)**
- Key: Find a good values of H such that P(D|H) is maximized.

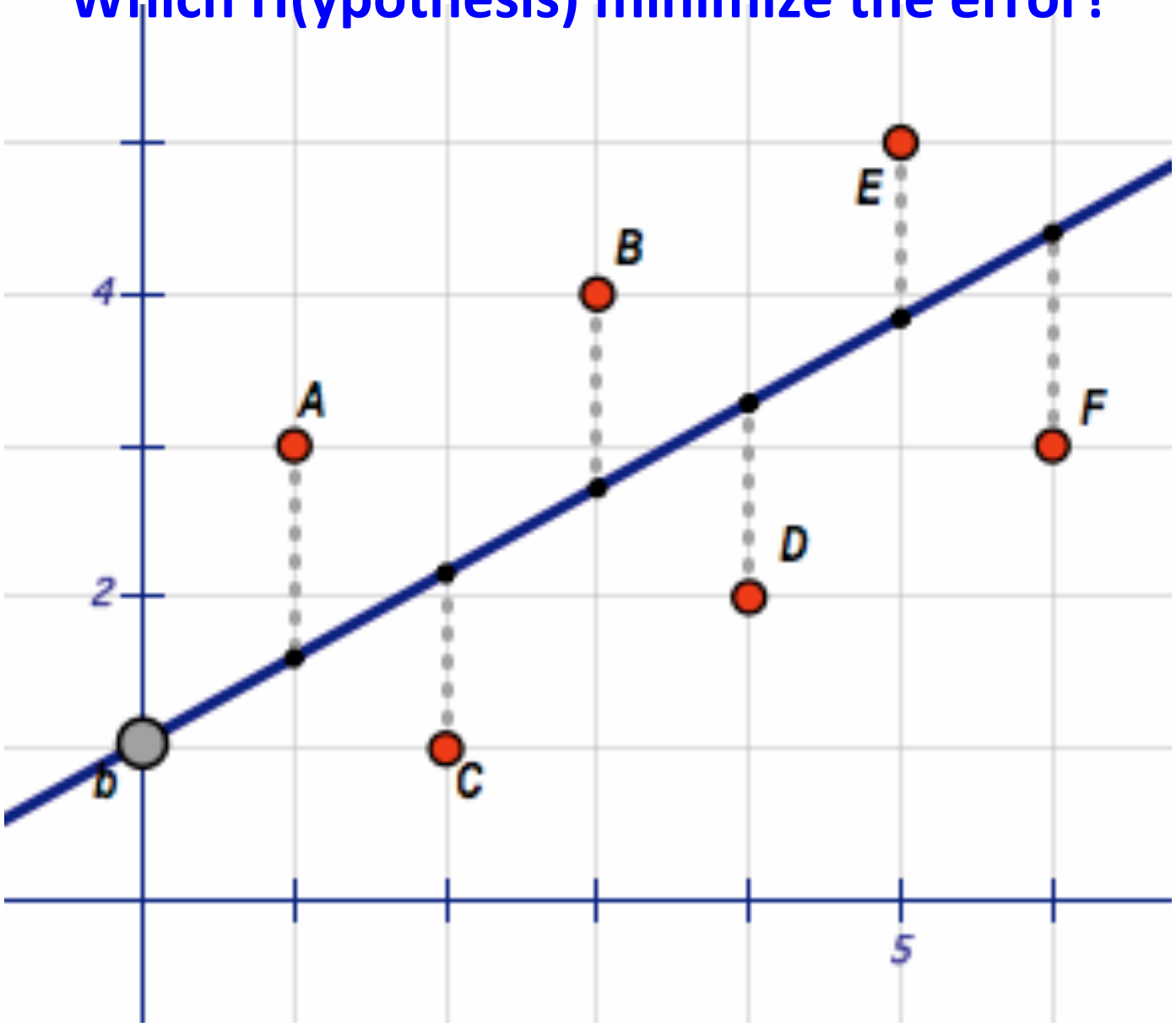
That is: Estimate the true H (hypothesis/model parameters) that the data D came from.

# Recall: Linear Regression

Given some data: $D = \{x_i, y_i\}$



# Which H(ypothesis) minimize the error?



# Assume a linear model

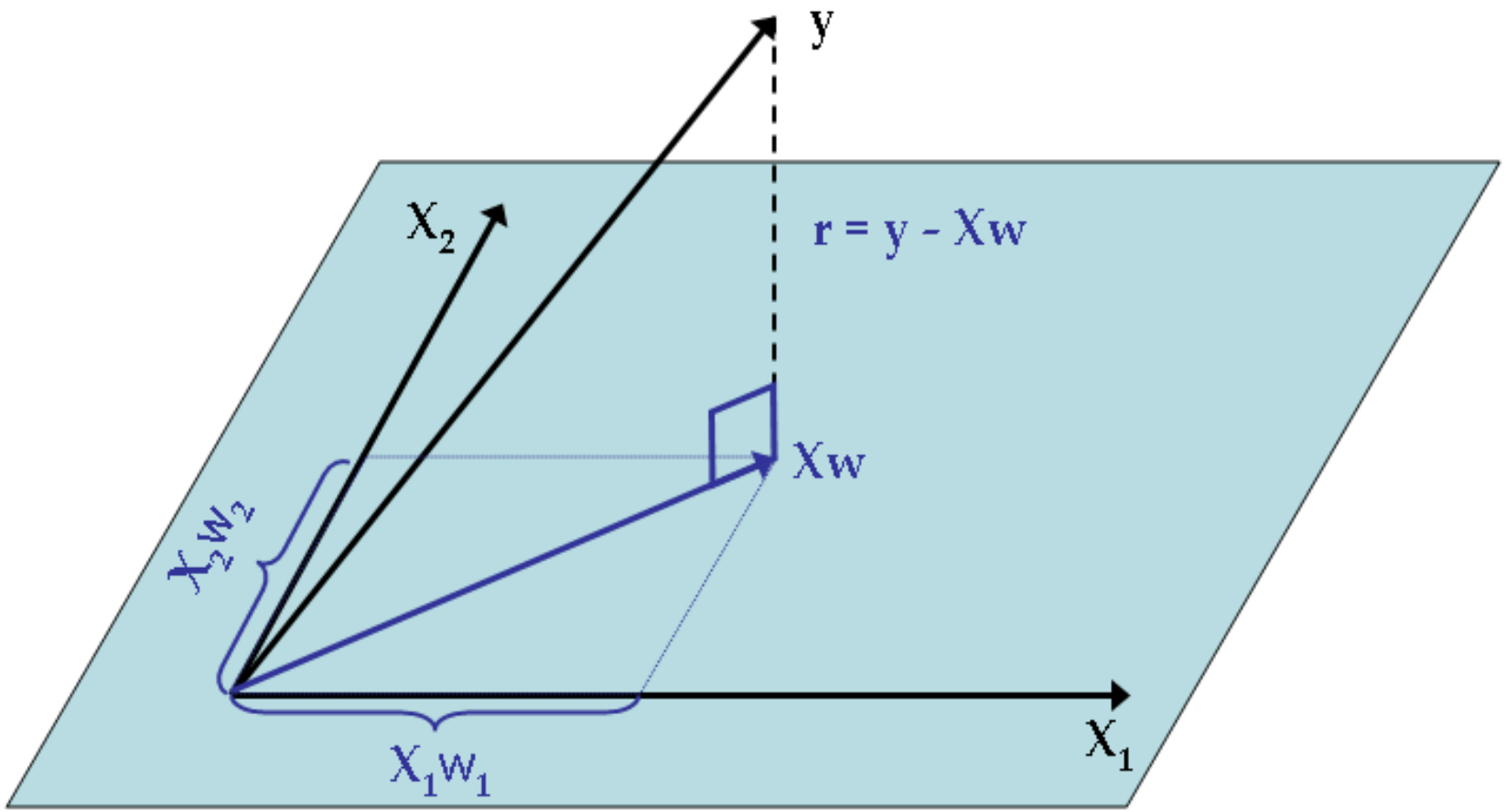
$$\begin{pmatrix} r_1 \\ \vdots \\ r_n \end{pmatrix} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} - \begin{pmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \vdots & \vdots \\ x_{n1} & \cdots & x_{nm} \end{pmatrix} \begin{pmatrix} w_1 \\ \vdots \\ w_m \end{pmatrix}$$

$$\rightarrow \quad \mathbf{r} = (\mathbf{y} - \mathbf{X}\mathbf{w})$$

This is equivalent to

$$y_i = \sum_j w_j x_{ij} + \mathcal{N}(0, \sigma^2) = \mathbf{x}_i \mathbf{w} + \mathcal{N}(0, \sigma^2)$$

# Geometrically you can see the solution!




$$\mathbf{w}_{MLE} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$$

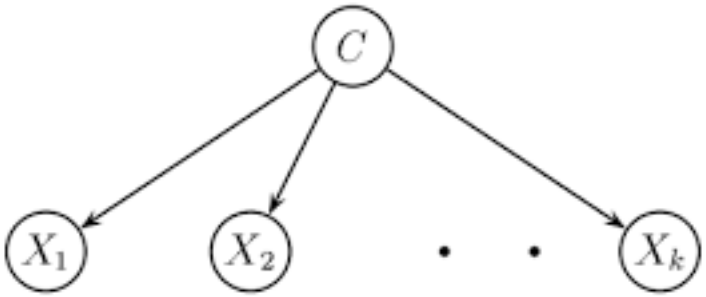
# Work out details with the students on the board

- **Maximum a Posterior (MAP)**

# Compare MLE with MAP

- Details on board.

# Naïve Bayes




Class Prior Probability

Likelihood

$$P(c \mid x) = \frac{P(x \mid c)P(c)}{P(x)}$$

Posterior Probability

Predictor Prior Probability

$$P(c \mid X) \propto P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

To maximize this product, we take log of it

# Today's Topics

- Recap of Bayesian Reasoning.
- **Bayesian Linear Regression (which we've already seen).**
- Bayesian Logistic Regression (Review).
- Bayesian Inference.
- Intractable Integrals and Motivation for Approximate Methods (only if time permits).
- Learning Theory

# Bayesian Linear Regression

- Why not use MEL?
- Since it is often over-fitting.
- How can we address this?
- Why not use MAP?  We put a prior
- But we do not have representation of our uncertainty.

# Let's see an example

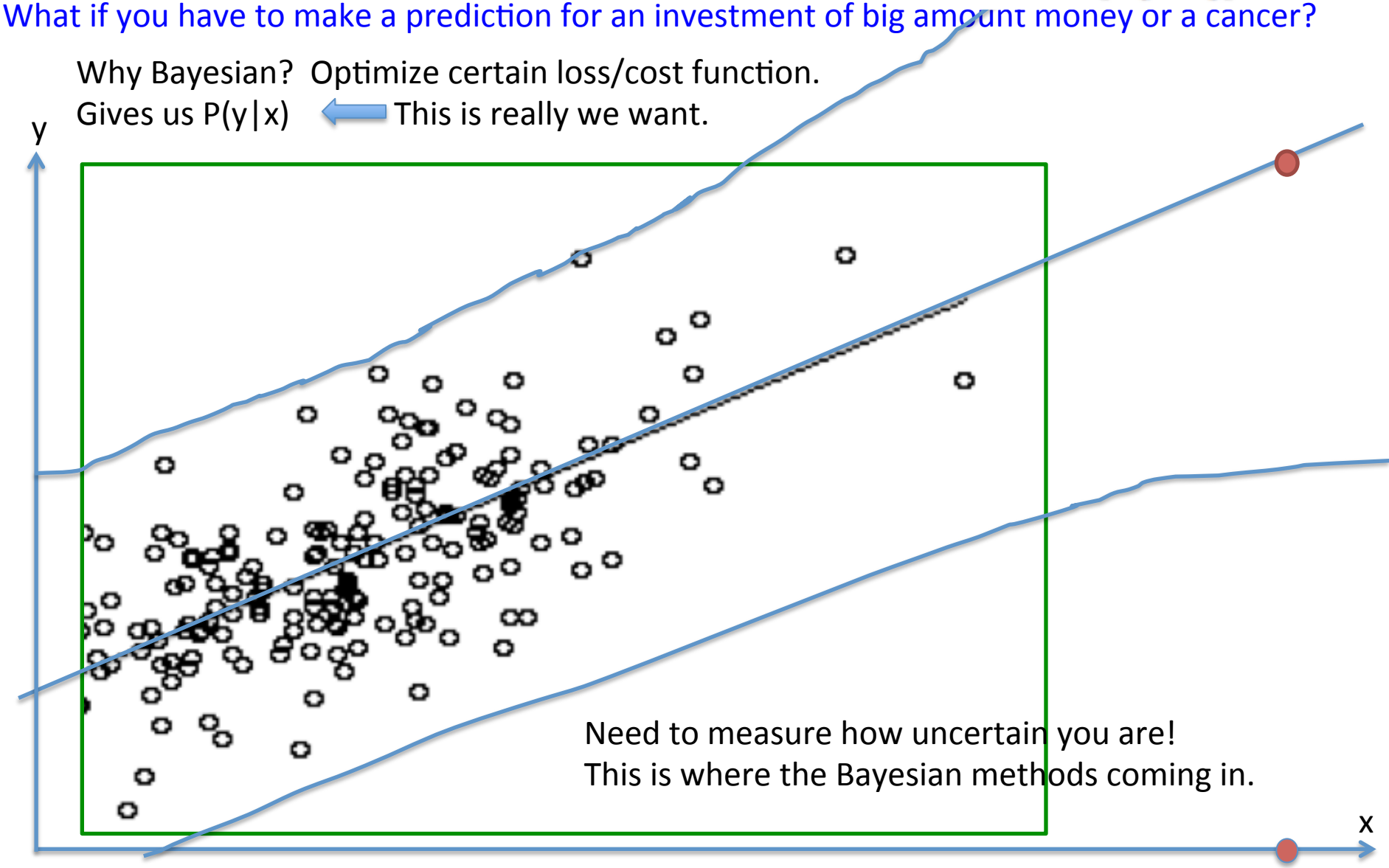
Given some data:

$$D = \{x_i, y_i\}$$

What if you have to make a prediction for an investment of big amount money or a cancer?

Why Bayesian? Optimize certain loss/cost function.

Gives us P(y|x) ⟸ This is really we want.



# Work out details with the students

# Be careful with the notations! Sometimes we use A for the design matrix and x as parameter vector!

- Where $y = (y_1, y_2, \ldots, y_n)^T$
- Where A is the design matrix.

- $A = \begin{bmatrix} — & (x^{(1)})^T & — \\ — & (x^{(2)})^T & — \\ & \vdots & \\ — & (x^{(m)})^T & — \end{bmatrix}$

# Important fact for Bayesian Linear Regression

- Keys:
- If we put Gaussian distributions for both likelihood and for the prior, the the posterior will be another Gaussian distribution!
- Its predictive distribution is again Gaussian!
- Both are closed solutions!

# Today's Topics

- Recap of Bayesian Reasoning.
- Bayesian Linear Regression (which we've already seen).
- **Bayesian Logistic Regression (Review).**
- Bayesian Inference.
- Intractable Integrals and Motivation for Approximate Methods (only if time permits).
- Learning Theory

# Bayesian Logistic Regression

- We will see that
- There is no analytic closed formula solutions (the integration involved is not integratable, usual approximation method using grids is exponential (#p, something as NP-hard).
- We call such an **integration is intractable.**
- We will have to smart approximation method called **Monte Carol approximation**.

# Bayesian Logistic Regression

- Work out details with students on the board.

# Today's Topics

- Recap of Bayesian Reasoning.
- Bayesian Linear Regression (which we've already seen).
- Bayesian Logistic Regression (Review).
- **Bayesian Inference.**
- Intractable Integrals and Motivation for Approximate Methods (only if time permits).
- Learning Theory

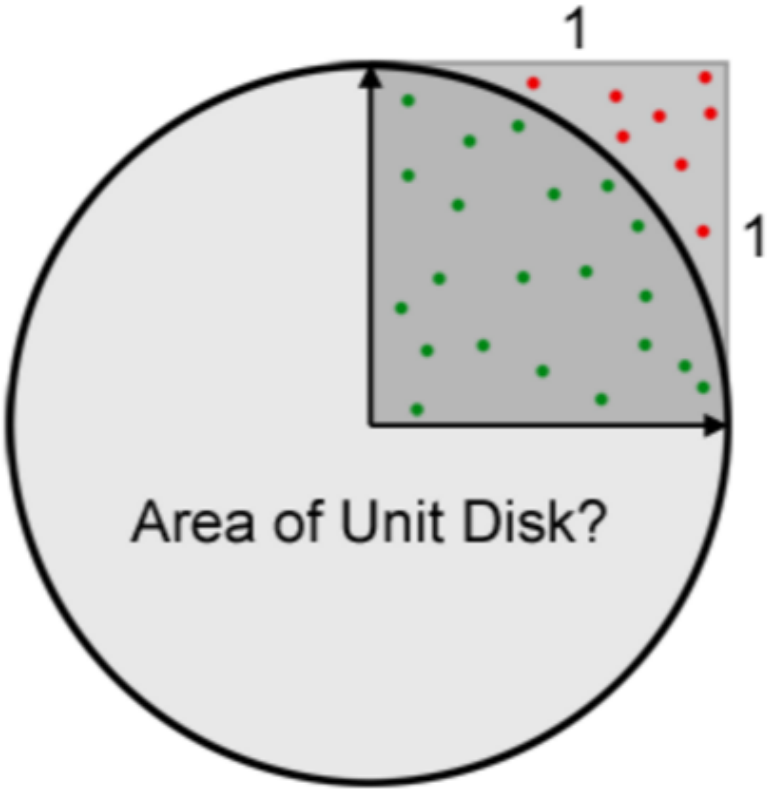
# Bayesian Inference

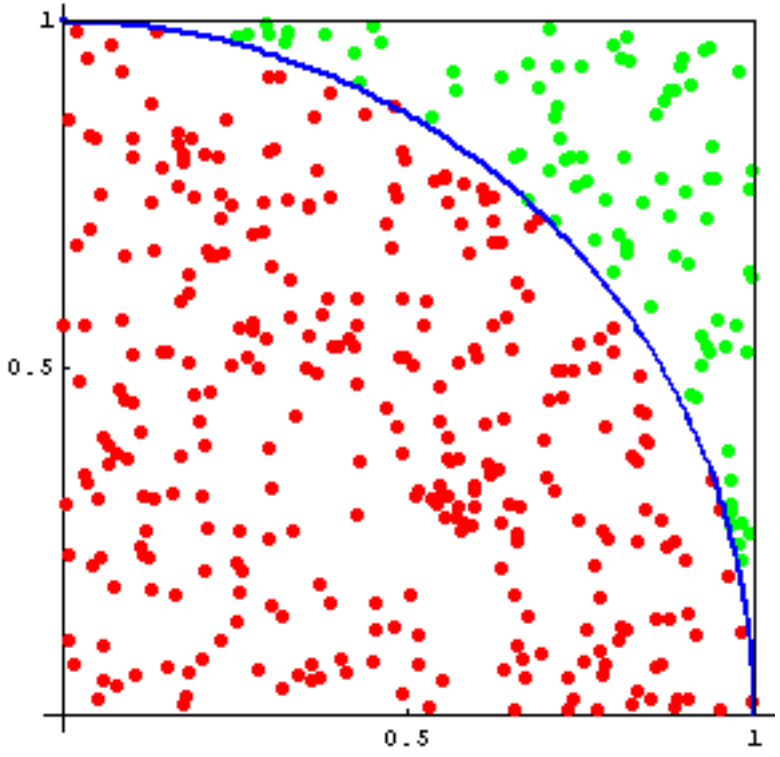
- Work out detail with students on the board.

# Today's Topics

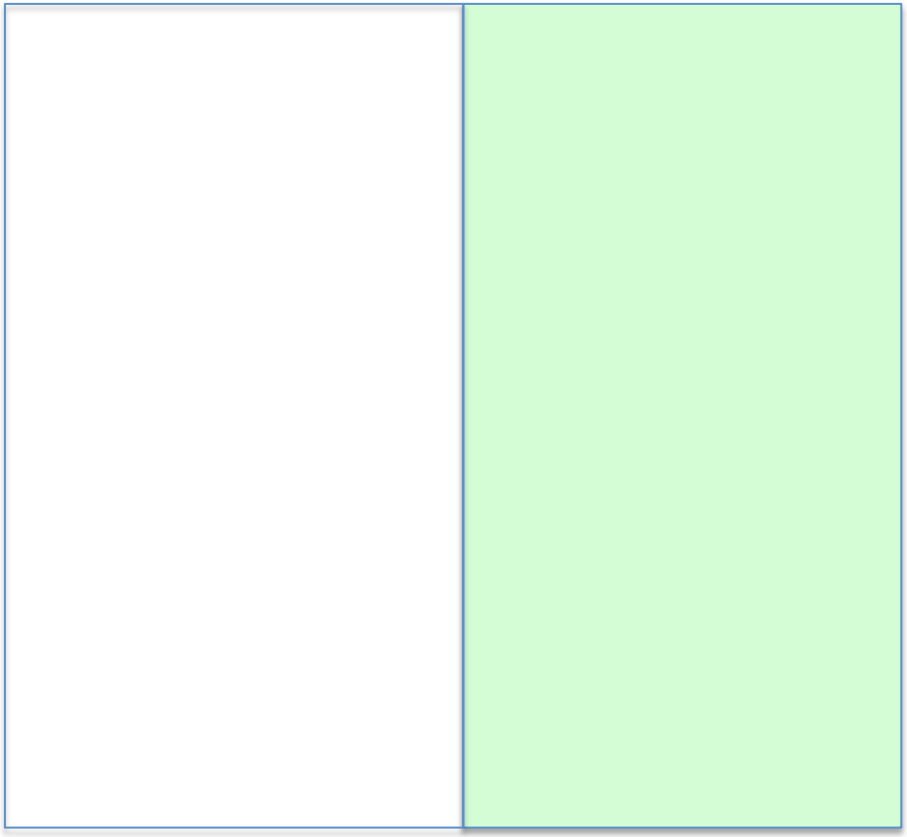
- Recap of Bayesian Reasoning.
- Bayesian Linear Regression (which we've already seen).
- Bayesian Logistic Regression (Review).
- Bayesian Inference.
- **Intractable Integrals and Motivation for Approximate Methods (only if time permits).**
- Learning Theory

# Monte Carlo Approximation

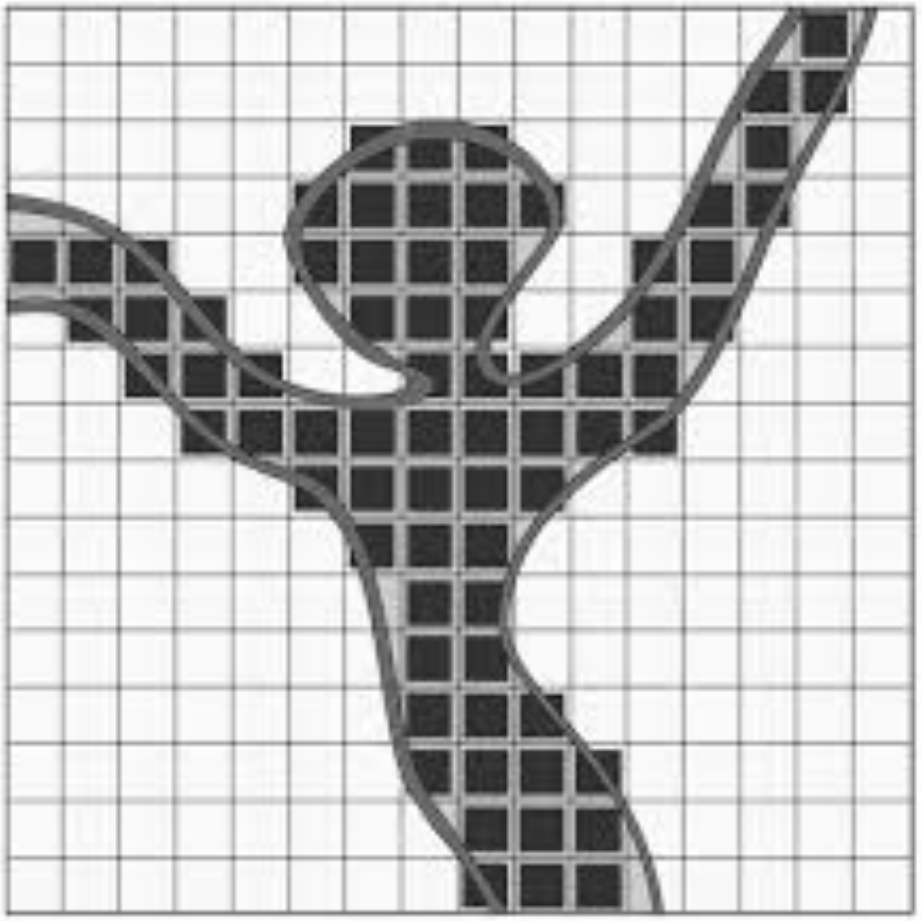


Close your eyes and throw a ball to it, what is the chance to get into the green area?



# Grid Approximation



# Today's Topics

- Recap of Bayesian Reasoning.
- Bayesian Linear Regression (which we've already seen).
- Bayesian Logistic Regression (Review).
- Bayesian Inference.
- Intractable Integrals and Motivation for Approximate Methods (only if time permits).
- **Learning Theory** ***(if time permits, otherwise, read only).***

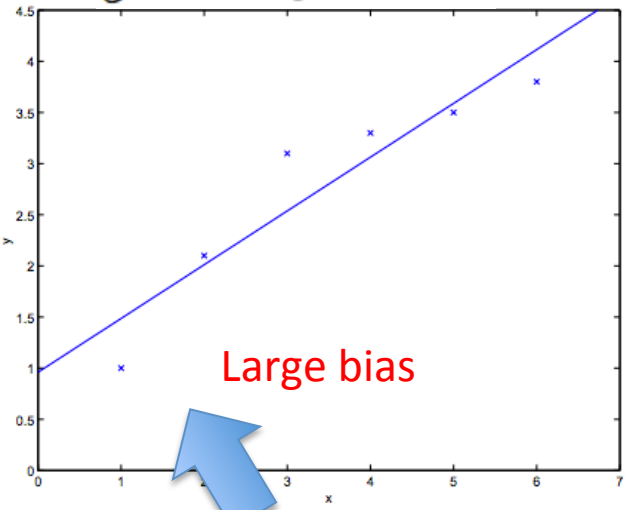
# Learning Theory

- **Bias/Variance Trade-off**
- **Union and Chernoff/Hoeffding Bounds**

This topic will be closely following Prof. Ng's notes on Learning Theory.

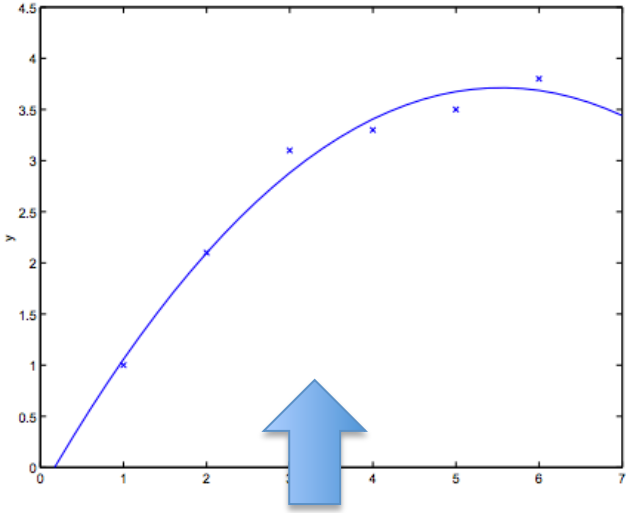
# Bias/variance tradeoff

$y = \theta_0 + \theta_1 x$



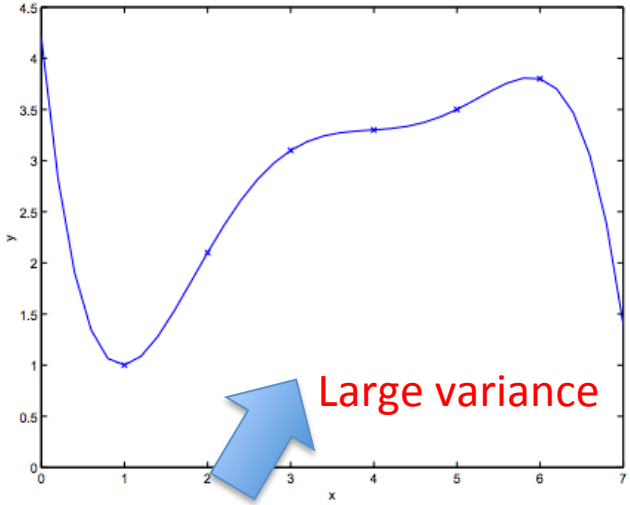

This linear model is too simple. it suffers from **large bias,** and may **underfit**,(i.e., **fail to capture structure exhibited by) the data.**

$y = \theta_0 + \theta_1 x + \theta_2 x^2$



This is a good One!

$y = \theta_0 + \theta_1 x + \cdots \theta_5 x^5$




Did not result in a good model! Specifically, even though the 5th order polynomial did a very **good** job predicting y (say, prices of houses) from x (say, living area) **for the examples in the training set**, we **do not** expect the model shown to be a **good** one **for** predicting the prices of houses **not in the training set.**

Does not generalize well → **Generalization error**

# Definition of Generalization Error

- Whatever errors you captured in your model, either fail to capture or "over" capture from your small set of training data, that do not reflect the wider pattern of the relationship between x and y on your testing data are called generation errors.
- **Generalization Error consists**
  - **Bias**
  - **Variance**

# Variance and Bias

- For example, when fitting a 5th order polynomial as in the rightmost figure, there is a large risk that we're fitting patterns in the data that happened to be present in our small, finite training set, but that do not reflect the wider pattern of the relationship between x and y.
- This could be, say, because in the training set we just happened by chance to get a slightly more-expensive-than-average house here, and a slightly less-expensive-than-average house there, and so on.
- By fitting these "spurious" patterns in the training set, we might again obtain a model with large generalization error. In this case, we say the model has large variance.
- We define the bias of a model to be the expected generalization error even if we were to fit it to a very (say, infinitely) large training set.

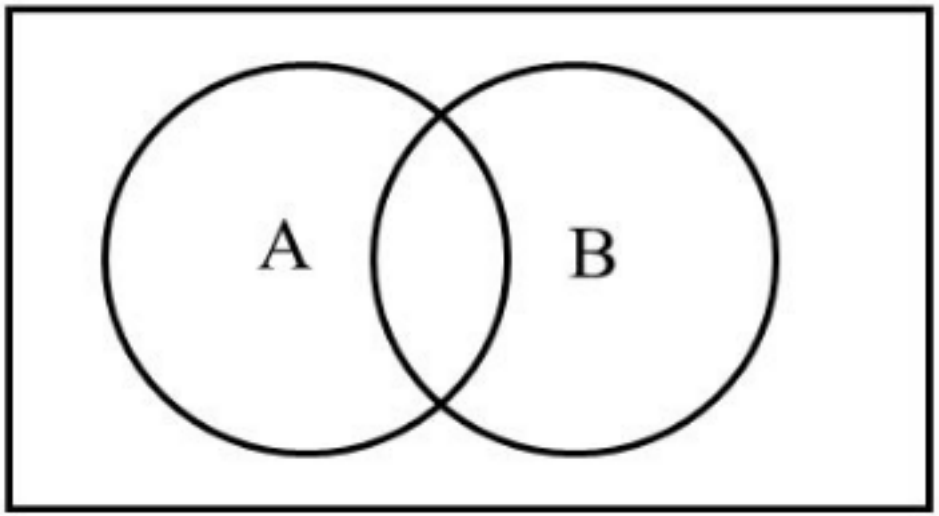
# There is a tradeoff between bias & variance.

- Meaning: If our model is too "simple" and has very few parameters, then it may have large bias (but small variance).
- But if it is too "complex" and has very many parameters, then it may suffer from large variance (but have smaller bias).
- In the example above, **fitting a quadratic function does better** than either of the extremes of a first or a fifth order polynomial.

# The Union Bound

Let $A_1, A_2, \ldots, A_k$ be $k$ different events (that may not be independent). Then

$$P(A_1 \cup \cdots \cup A_k) \leq P(A_1) + \ldots + P(A_k).$$




$P(A \cup B) <= P(A) + P(B)$

# Hoeffding Inequality / Chernoff Bound

Let $Z_1, \ldots, Z_m$ be $m$ independent and identically distributed (iid) random variables drawn from a Bernoulli($\phi$) distribution.

I.e., $P(Z_i = 1) = \phi$, and $P(Z_i = 0) = 1 - \phi$.

Let $\hat{\phi} = (1/m) \sum_{i=1}^{m} Z_i$ be the mean of these random variables,

and let any $\gamma > 0$ be fixed. Then

$$P(|\phi - \hat{\phi}| > \gamma) \leq 2 \exp(-2\gamma^2 m)$$

# Meaning of Hoeffding inequality (also called Chernoff bound)

The Chernoff bound says that if we take $\hat{\phi}$—the average of m Bernoulli($\phi$) random variables—to be our estimate of $\phi$, then the probability of our being far from the true value is small, so long as m is large.

Another way of saying this is that if you have a biased coin whose chance of landing on heads is $\phi$, then if you toss it m times and calculate the fraction of times that it came up heads, that will be a good estimate of $\phi$ with high probability (if m is large).

- Using the Union Bound and Chernoff Bound, we will be able to prove some of the deepest and most important results in learning theory.

- To simplify our exposition, let's restrict our attention to binary classification in which the labels are $y \in \{0, 1\}$.

- Everything we'll say here generalizes to other, including regression and multi-class classification, problems.

We assume we are given a training set $S = \{(x^{(i)}, y^{(i)}); i = 1, \ldots, m\}$ of size $m$, where the training examples $(x^{(i)}, y^{(i)})$ are drawn iid from some probability distribution $\mathcal{D}$. For a hypothesis $h$, we define the **training error** (also called the **empirical risk** or **empirical error** in learning theory) to be

$$\hat{\varepsilon}(h) = \frac{1}{m} \sum_{i=1}^{m} 1\{h(x^{(i)}) \neq y^{(i)}\}.$$

This is just the fraction of training examples that $h$ misclassifies. When we want to make explicit the dependence of $\hat{\varepsilon}(h)$ on the training set $S$, we may also write this a $\hat{\varepsilon}_S(h)$. We also define the generalization error to be

$$\varepsilon(h) = P_{(x,y)\sim\mathcal{D}}(h(x) \neq y).$$

I.e. this is the probability that, if we now draw a new example $(x, y)$ from the distribution $\mathcal{D}$, $h$ will misclassify it.

Note that we have assumed that the training data was drawn from the *same* distribution $\mathcal{D}$ with which we're going to evaluate our hypotheses (in the definition of generalization error). This is sometimes also referred to as one of the **PAC** assumptions.[2]

PAC = Probably Approximately Correct

# Important Results of Learning Theorem

- In certain sense, training error will be close to generalization error with high probability, assuming m is large.

**Theorem.** Let $|\mathcal{H}| = k$, and let any $m, \delta$ be fixed. Then with probability at least $1 - \delta$, we have that

$$\varepsilon(\hat{h}) \leq \left( \min_{h \in \mathcal{H}} \varepsilon(h) \right) + 2\sqrt{\frac{1}{2m} \log \frac{2k}{\delta}}.$$

**Corollary.** Let $|\mathcal{H}| = k$, and let any $\delta, \gamma$ be fixed. Then for $\varepsilon(\hat{h}) \leq \min_{h \in \mathcal{H}} \varepsilon(h) + 2\gamma$ to hold with probability at least $1 - \delta$, it suffices that

$$\begin{aligned} m &\geq \frac{1}{2\gamma^2} \log \frac{2k}{\delta} \\ &= O\left( \frac{1}{\gamma^2} \log \frac{k}{\delta} \right), \end{aligned}$$

Consider the setting of linear classification, and let $h_\theta(x) = 1\{\theta^T x \geq 0\}$. What's a reasonable way of fitting the parameters $\theta$? One approach is to try to minimize the training error, and pick

$$\hat{\theta} = \arg\min_\theta \hat{\varepsilon}(h_\theta).$$

How to auto select model?

We call this process **empirical risk minimization** (ERM), and the resulting hypothesis output by the learning algorithm is $\hat{h} = h_{\hat{\theta}}$. We think of ERM as the most "basic" learning algorithm, and it will be this algorithm that we focus on in these notes. (Algorithms such as logistic regression can also be viewed as approximations to empirical risk minimization.)

We define the **hypothesis class** $\mathcal{H}$

$$\mathcal{H} = \{h_\theta : h_\theta(x) = 1\{\theta^T x \geq 0\}, \theta \in \mathbb{R}^{n+1}\}$$

Empirical risk minimization can now be thought of as a minimization over the class of functions $\mathcal{H}$, in which the learning algorithm picks the hypothesis:

$$\hat{h} = \arg\min_{h \in \mathcal{H}} \hat{\varepsilon}(h)$$

In this course, we assume $\mathcal{H}$ is finite.

Let's start by considering a learning problem in which we have a finite hypothesis class $\mathcal{H} = \{h_1, \ldots, h_k\}$ consisting of $k$ hypotheses. Thus, $\mathcal{H}$ is just a set of $k$ functions mapping from $\mathcal{X}$ to $\{0, 1\}$, and empirical risk minimization selects $\hat{h}$ to be whichever of these $k$ functions has the smallest training error.

We would like to give guarantees on the generalization error of $h$. Our strategy for doing so will be in two parts: First, we will show that $\hat{\varepsilon}(h)$ is a reliable estimate of $\varepsilon(h)$ for all $h$. Second, we will show that this implies an upper-bound on the generalization error of $\hat{h}$.

Thus, $\hat{\varepsilon}(h_i)$ is exactly the mean of the $m$ random variables $Z_j$ that are drawn iid from a Bernoulli distribution with mean $\varepsilon(h_i)$. Hence, we can apply the Hoeffding inequality, and obtain

$$P(|\varepsilon(h_i) - \hat{\varepsilon}(h_i)| > \gamma) \leq 2 \exp(-2\gamma^2 m).$$

This shows that, for our particular $h_i$, training error will be close to generalization error with high probability, assuming $m$ is large. But we don't just want to guarantee that $\varepsilon(h_i)$ will be close to $\hat{\varepsilon}(h_i)$ (with high probability) for just only one particular $h_i$. We want to prove that this will be true for simultaneously for *all* $h \in \mathcal{H}$. To do so, let $A_i$ denote the event that $|\varepsilon(h_i) - \hat{\varepsilon}(h_i)| > \gamma$. We've already show that, for any particular $A_i$, it holds true that $P(A_i) \leq 2\exp(-2\gamma^2 m)$. Thus, using the union bound, we have that

$$\begin{aligned} P(\exists\, h \in \mathcal{H}. |\varepsilon(h_i) - \hat{\varepsilon}(h_i)| > \gamma) &= P(A_1 \cup \cdots \cup A_k) \\ &\leq \sum_{i=1}^{k} P(A_i) \\ &\leq \sum_{i=1}^{k} 2\exp(-2\gamma^2 m) \\ &= 2k\exp(-2\gamma^2 m) \end{aligned}$$

If we subtract both sides from 1, we find that

$$\begin{aligned} P(\neg\exists\, h \in \mathcal{H}.|\varepsilon(h_i) - \hat{\varepsilon}(h_i)| > \gamma) &= P(\forall h \in \mathcal{H}.|\varepsilon(h_i) - \hat{\varepsilon}(h_i)| \leq \gamma) \\ &\geq 1 - 2k\exp(-2\gamma^2 m) \end{aligned}$$

(The "$\neg$" symbol means "not.") So, with probability at least $1-2k\exp(-2\gamma^2 m)$, we have that $\varepsilon(h)$ will be within $\gamma$ of $\hat{\varepsilon}(h)$ for all $h \in \mathcal{H}$. This is called a *uniform convergence* result, because this is a bound that holds simultaneously for all (as opposed to just one) $h \in \mathcal{H}$.

In the discussion above, what we did was, for particular values of $m$ and $\gamma$, give a bound on the probability that for some $h \in \mathcal{H}$, $|\varepsilon(h) - \hat{\varepsilon}(h)| > \gamma$. There are three quantities of interest here: $m$, $\gamma$, and the probability of error; we can bound either one in terms of the other two.

For instance, we can ask the following question: Given $\gamma$ and some $\delta > 0$, how large must $m$ be before we can guarantee that with probability at least $1 - \delta$, training error will be within $\gamma$ of generalization error? By setting $\delta = 2k \exp(-2\gamma^2 m)$ and solving for $m$, [you should convince yourself this is the right thing to do!], we find that if

$$m \geq \frac{1}{2\gamma^2} \log \frac{2k}{\delta},$$

then with probability at least $1 - \delta$, we have that $|\varepsilon(h) - \hat{\varepsilon}(h)| \leq \gamma$ for all $h \in \mathcal{H}$. (Equivalently, this shows that the probability that $|\varepsilon(h) - \hat{\varepsilon}(h)| > \gamma$ for some $h \in \mathcal{H}$ is at most $\delta$.) This bound tells us how many training examples we need in order make a guarantee. The training set size $m$ that a certain method or algorithm requires in order to achieve a certain level of performance is also called the algorithm's **sample complexity**.

The key property of the bound above is that the number of training examples needed to make this guarantee is only *logarithmic* in $k$, the number of hypotheses in $\mathcal{H}$. This will be important later.

Similarly, we can also hold $m$ and $\delta$ fixed and solve for $\gamma$ in the previous equation, and show [again, convince yourself that this is right!] that with probability $1 - \delta$, we have that for all $h \in \mathcal{H}$,

$$|\hat{\varepsilon}(h) - \varepsilon(h)| \leq \sqrt{\frac{1}{2m} \log \frac{2k}{\delta}}.$$

Now, let's assume that uniform convergence holds, i.e., that $|\varepsilon(h) - \hat{\varepsilon}(h)| \leq \gamma$ for all $h \in \mathcal{H}$. What can we prove about the generalization of our learning algorithm that picked $\hat{h} = \arg\min_{h \in \mathcal{H}} \hat{\varepsilon}(h)$?

Take any one, fixed, $h_i \in \mathcal{H}$. Consider a Bernoulli random variable $Z$ whose distribution is defined as follows. We're going to sample $(x, y) \sim \mathcal{D}$. Then, we set $Z = 1\{h_i(x) \neq y\}$. I.e., we're going to draw one example, and let $Z$ indicate whether $h_i$ misclassifies it. Similarly, we also define $Z_j = 1\{h_i(x^{(j)}) \neq y^{(j)}\}$. Since our training set was drawn iid from $\mathcal{D}$, $Z$ and the $Z_j$'s have the same distribution.

We see that the misclassification probability on a randomly drawn example—that is, $\varepsilon(h)$—is exactly the expected value of $Z$ (and $Z_j$). Moreover, the training error can be written

$$\hat{\varepsilon}(h_i) = \frac{1}{m} \sum_{j=1}^{m} Z_j.$$

Define $h^* = \arg\min_{h \in \mathcal{H}} \varepsilon(h)$ to be the best possible hypothesis in $\mathcal{H}$. Note that $h^*$ is the best that we could possibly do given that we are using $\mathcal{H}$, so it makes sense to compare our performance to that of $h^*$. We have:

$$\begin{aligned} \varepsilon(\hat{h}) &\leq \hat{\varepsilon}(\hat{h}) + \gamma \\ &\leq \hat{\varepsilon}(h^*) + \gamma \\ &\leq \varepsilon(h^*) + 2\gamma \end{aligned}$$

The first line used the fact that $|\varepsilon(\hat{h}) - \hat{\varepsilon}(\hat{h})| \leq \gamma$ (by our uniform convergence assumption). The second used the fact that $\hat{h}$ was chosen to minimize $\hat{\varepsilon}(h)$, and hence $\hat{\varepsilon}(\hat{h}) \leq \hat{\varepsilon}(h)$ for all $h$, and in particular $\hat{\varepsilon}(\hat{h}) \leq \hat{\varepsilon}(h^*)$. The third line used the uniform convergence assumption again, to show that $\hat{\varepsilon}(h^*) \leq \varepsilon(h^*) + \gamma$. So, what we've shown is the following: If uniform convergence occurs, then the generalization error of $\hat{h}$ is at most $2\gamma$ worse than the best possible hypothesis in $\mathcal{H}$!

# Let's put all this together into a theorem.

**Theorem.** Let $|\mathcal{H}| = k$, and let any $m, \delta$ be fixed. Then with probability at least $1 - \delta$, we have that

$$\varepsilon(\hat{h}) \leq \left( \min_{h \in \mathcal{H}} \varepsilon(h) \right) + 2\sqrt{\frac{1}{2m} \log \frac{2k}{\delta}}.$$

This is proved by letting $\gamma$ equal the $\sqrt{\cdot}$ term, using our previous argument that uniform convergence occurs with probability at least $1 - \delta$, and then noting that uniform convergence implies $\varepsilon(h)$ is at most $2\gamma$ higher than $\varepsilon(h^*) = \min_{h \in \mathcal{H}} \varepsilon(h)$ (as we showed previously).

This also quantifies what we were saying previously saying about the bias/variance tradeoff in model selection. Specifically, suppose we have some hypothesis class $\mathcal{H}$, and are considering switching to some much larger hypothesis class $\mathcal{H}' \supseteq \mathcal{H}$. If we switch to $\mathcal{H}'$, then the first term $\min_h \varepsilon(h)$ can only decrease (since we'd then be taking a min over a larger set of functions). Hence, by learning using a larger hypothesis class, our "bias" can only decrease. However, if k increases, then the second $2\sqrt{\cdot}$ term would also increase. This increase corresponds to our "variance" increasing when we use a larger hypothesis class.

**Corollary.** Let $|\mathcal{H}| = k$, and let any $\delta, \gamma$ be fixed. Then for $\varepsilon(\hat{h}) \leq \min_{h \in \mathcal{H}} \varepsilon(h) + 2\gamma$ to hold with probability at least $1 - \delta$, it suffices that

$$\begin{aligned} m &\geq \frac{1}{2\gamma^2} \log \frac{2k}{\delta} \\ &= O\left(\frac{1}{\gamma^2} \log \frac{k}{\delta}\right), \end{aligned}$$