Lecture 4

Frequentist Modelling And Regression

Last Time:

- Monte Carlo for Integrals
- Monte Carlo Variance
- Coin toss means, variance, CLT
- Numerical Integration vs Monte-Carlo Integration
- Frequentist Statistics
- Maximum Likelihood Estimation
- Sampling Distribution

Today

- Small World vs Big World
- MLE and Sampling
- Gaussian MLE
- Fitting without Noise
- What is noise?
- Fitting with Noise
- Test sets
- Validation and X-validation
- Regularization

Frequentist Statistics

Answers the question: **What is Data?** with

"data is a **sample** from an existing **population**"

- data is stochastic, variable
- model the sample. The model may have parameters
- find parameters for our sample. The parameters are considered **FIXED**.

Point Estimates

If we want to calculate some quantity of the population, like say the mean, we estimate it on the sample by applying an estimator F to the sample data D, so $\hat{\mu} = F(D)$.

Remember, **The parameter is viewed as fixed and the data as random, which is the exact opposite of the Bayesian approach which you will learn later in this class.**

True vs estimated

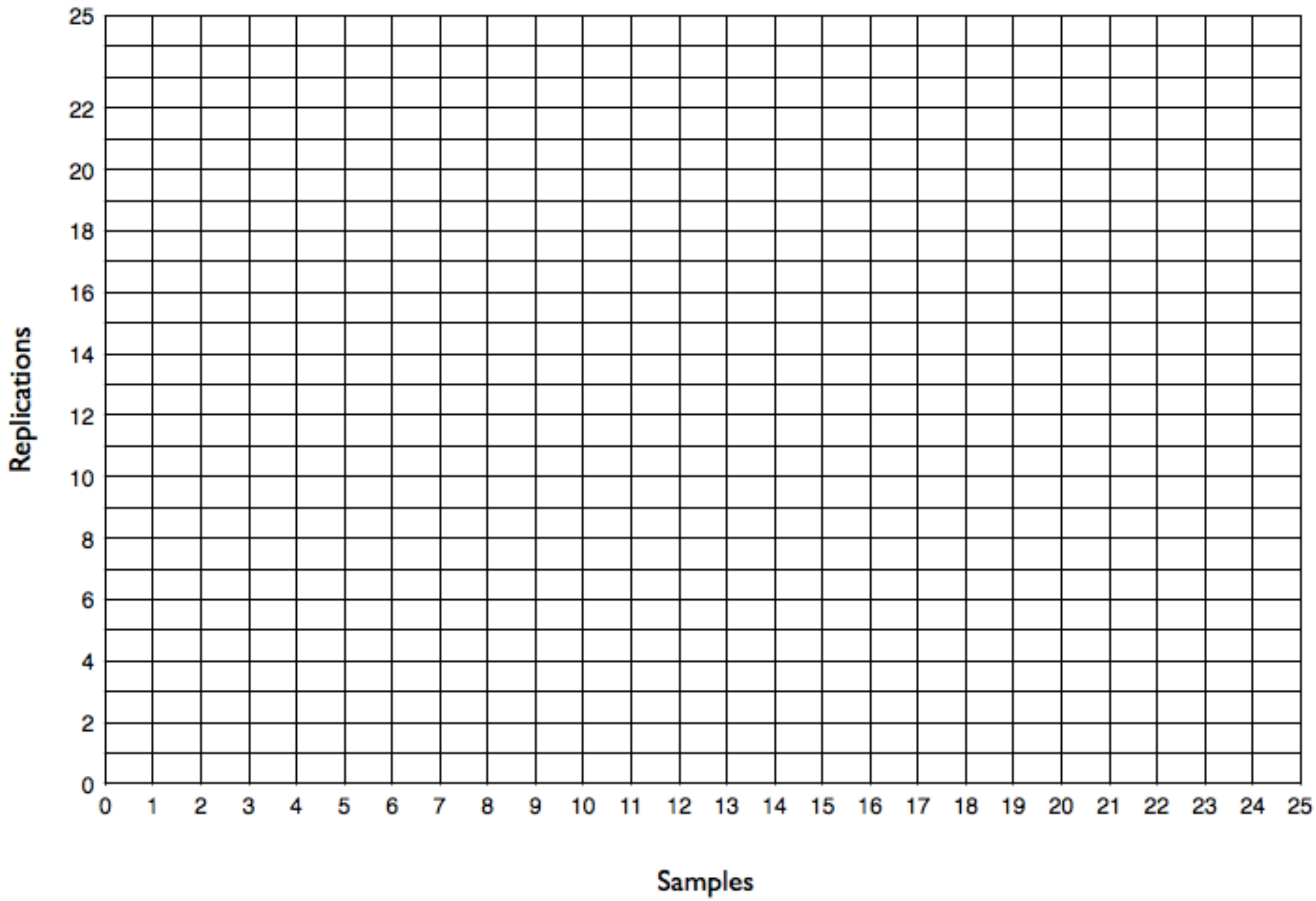
If your model describes the true generating process for the data, then there is some true μ^*.

We dont know this. The best we can do is to estimate $\hat{\mu}$.

Now, imagine that God gives you some M data sets **drawn** from the population, and you can now find μ on each such dataset.

So, we'd have M estimates.

M samples of N data points



Sampling distribution

As we let $M \to \infty$, the distribution induced on $\hat{\mu}$ is the empirical **sampling distribution of the estimator**.

μ could be λ, our parameter, or a mean, a variance, etc

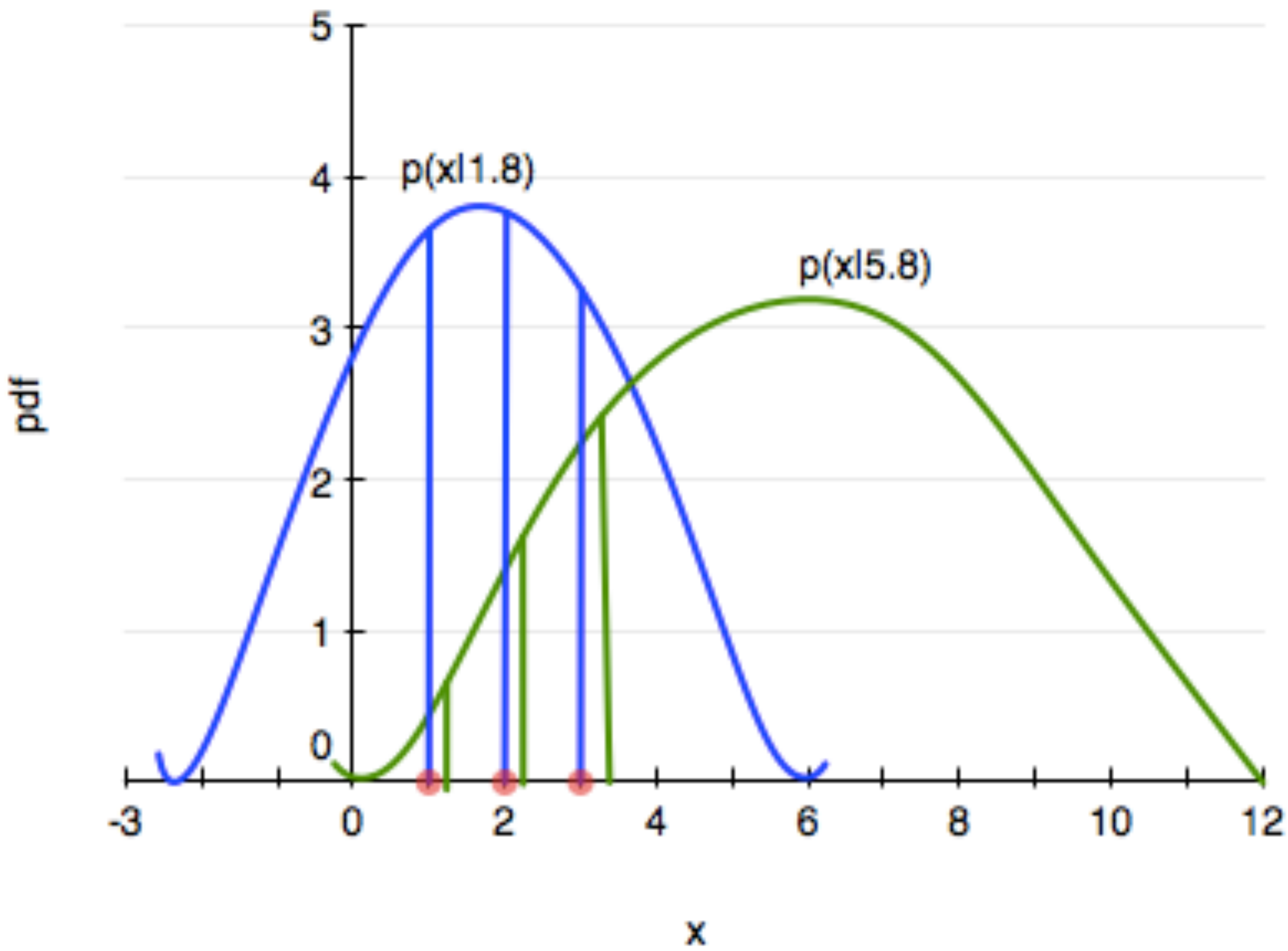
We could use the sampling distribution to get confidence intervals on λ.

But we dont have M samples. What to do?

Resampling

- if we want to estimate the SIZE of the effect we use bootstrap
- if we want to estimate the SIGNIFICANCE of the effect, we do PERMUTATION

Maximum Likelihood estimation



We have data on the wing length in millimeters of a nine members of a particular species of moth. We wish to make inferences from those measurements on the population quantities μ and σ.

Y = [16.4, 17.0, 17.2, 17.4, 18.2, 18.2, 18.2, 19.9, 20.8]

Let us assume a gaussian pdf:

$$p(y|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{y-\mu}{2\sigma}\right)^2}$$

MLE Estimators

LIKELIHOOD: $p(y_1, \ldots, y_n | \mu, \sigma^2) = \prod_{i=1}^{n} p(y_i | \mu, \sigma^2)$

$$= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{(y_i - \mu)^2}{2\sigma^2}\right)} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{1}{2} \sum_i \frac{(y_i - \mu)^2}{\sigma^2} \right\}$$

Take partials for $\hat{\mu}_{MLE}$ and $\hat{\sigma}^2_{MLE}$

From Likelihood to Predictive Distribution

- likelihood as a function of parameters is NOT a probability distribution, rather, its a function
- $p(y|\mu_{MLE}, \sigma^2_{MLE})$ on the other hand is a probability distribution
- think of it as $p(y^*|\{y_i\}, \mu_{MLE}, \sigma^2_{MLE})$ (`norm.rvs` with MLE parameters), "communicating with existing data" thru the parameters
- We'll call such a distribution a predictive distribution for as yet unseen data y^*, or the sampling distribution for data, or the data-generating distribution

MLE for Moth Wing

$$\hat{\mu}_{MLE} = \frac{1}{N}\sum_i y_i = \bar{Y};\ \hat{\sigma}^2_{MLE} = \frac{1}{N}\sum_i (Y_i - \bar{Y}^2)$$

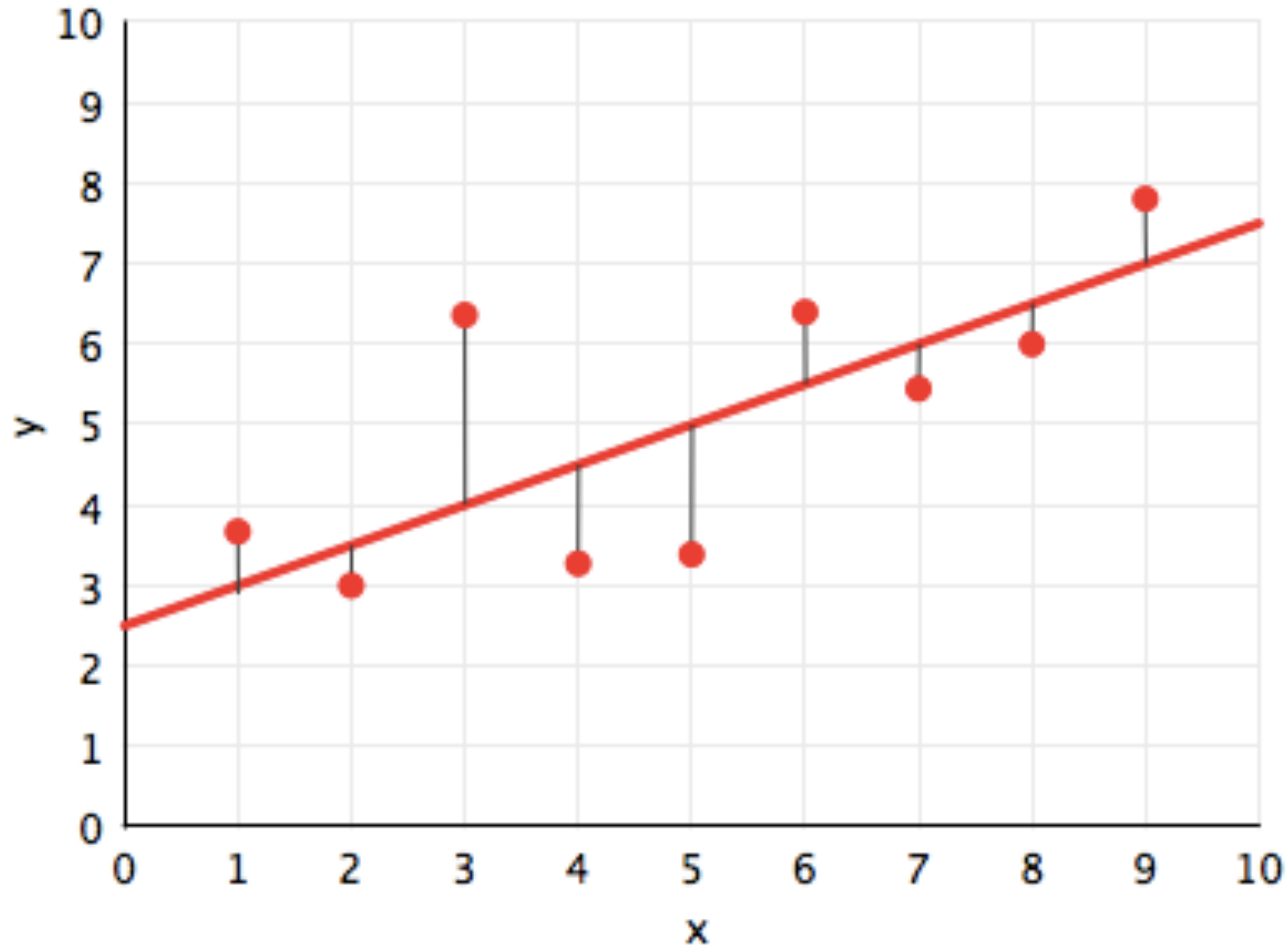
$\hat{\sigma}^2_{MLE}$ is a biased estimator of the population variance, while $\hat{\mu}_{MLE}$ is an unbiased estimator.

That is, $E_D[\hat{\mu}_{MLE}] = \mu$, where the D subscripts means the expectation with respect to the predictive, or data-sampling, or data generating distribution.

VALUES: `sigma 1.33 mu 18.14`

REGRESSION

- how many dollars will you spend?
- what is your creditworthiness
- how many people will vote for Bernie t days before election
- use to predict probabilities for classification
- causal modeling in econometrics



HYPOTHESIS SPACES

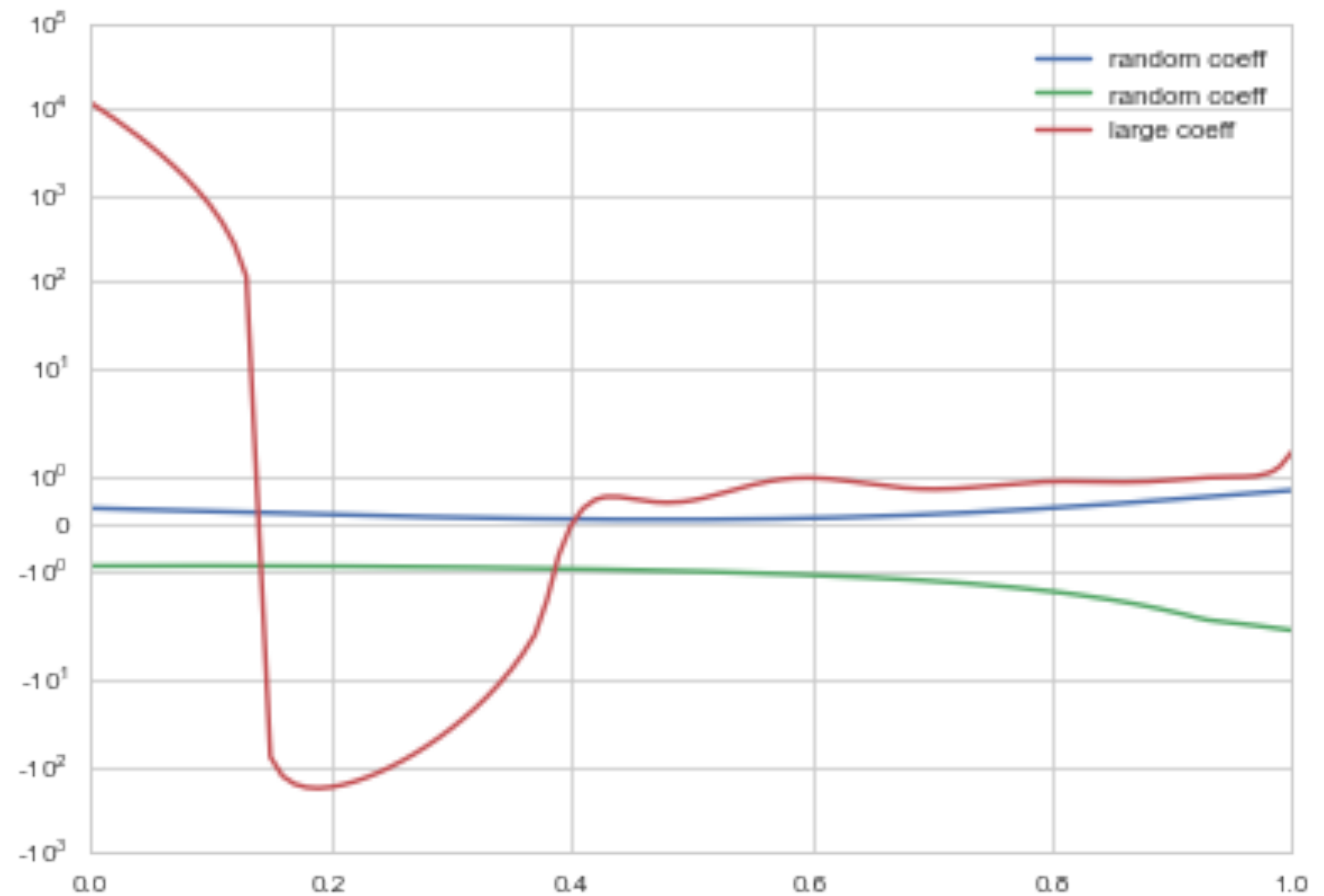
A polynomial looks so:

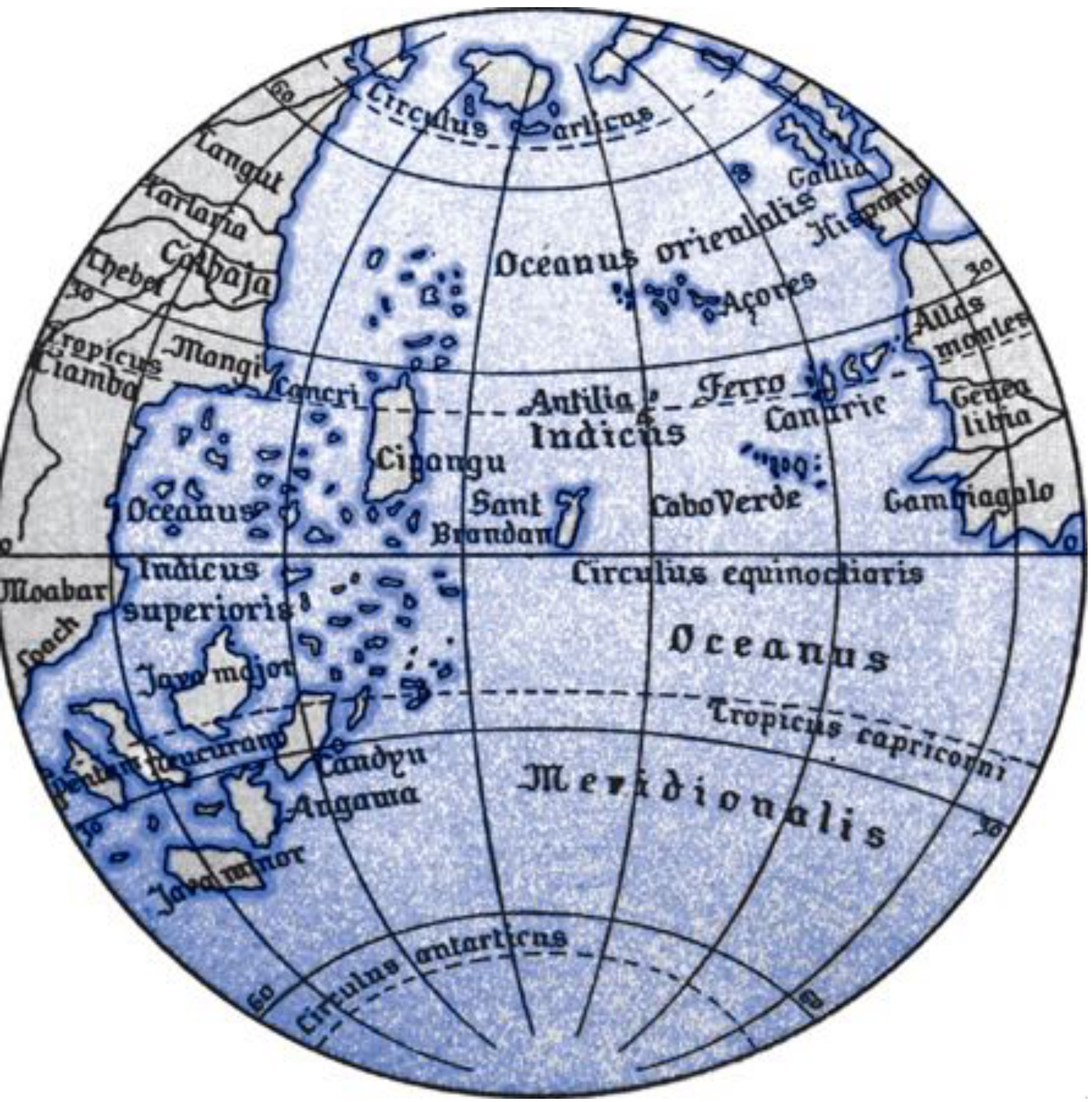
$$h(x) = \theta_0 + \theta_1 x^1 + \theta_2 x^2 + \ldots + \theta_n x^n = \sum_{i=0}^{n} \theta_i x^i$$

All polynomials of a degree or complexity d constitute a hypothesis space.

$$\mathcal{H}_1 : h_1(x) = \theta_0 + \theta_1 x$$

$$\mathcal{H}_{20} : h_{20}(x) = \sum_{i=0}^{20} \theta_i x^i$$



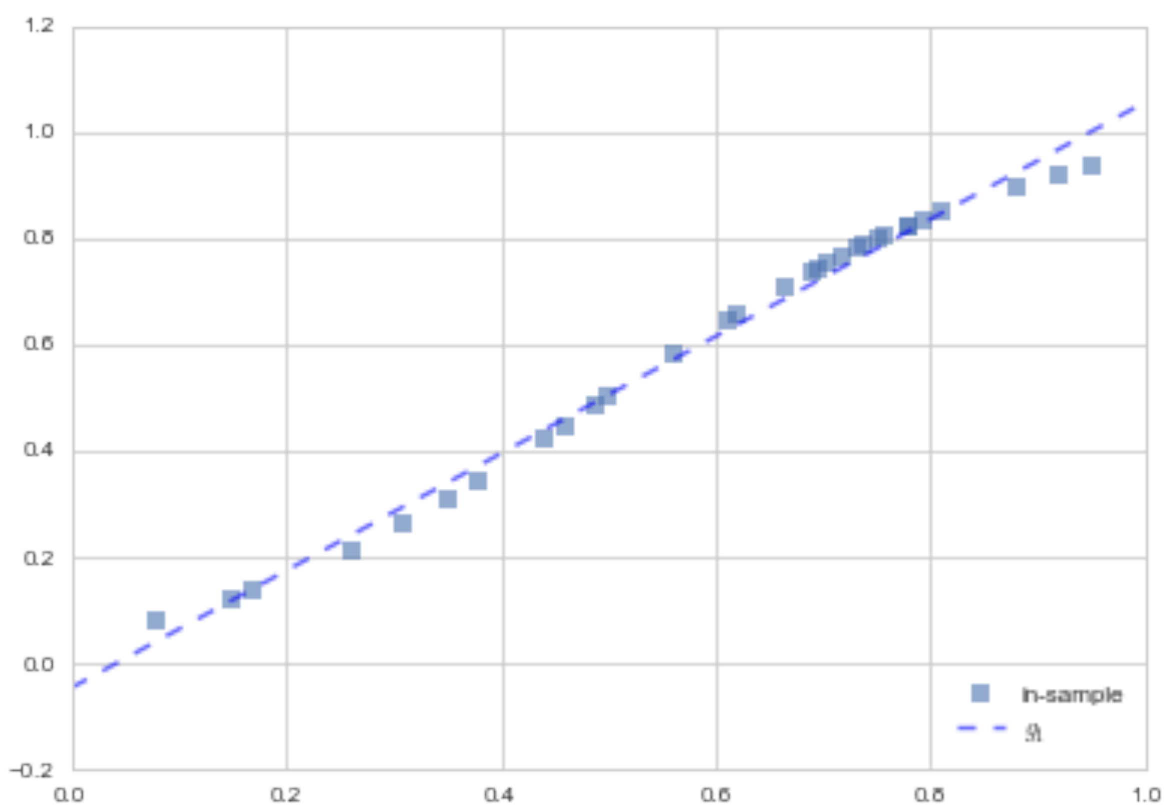


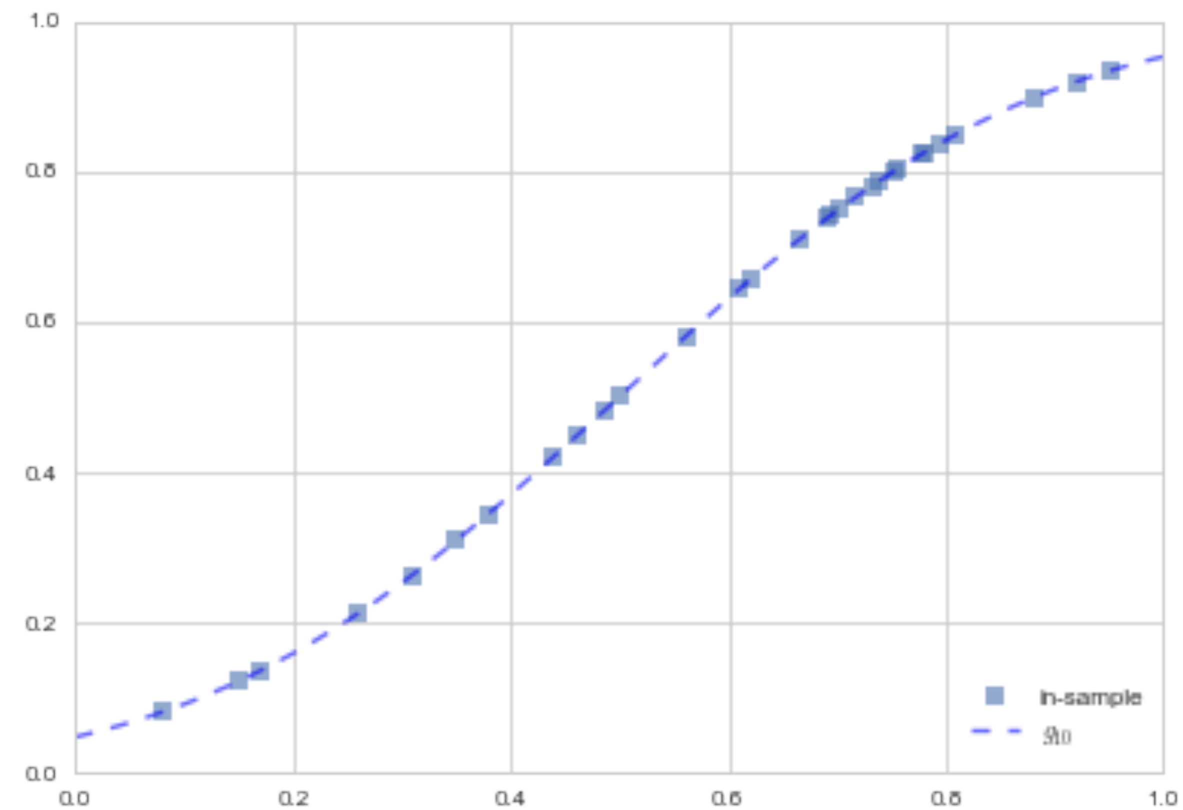
SMALL World vs BIG World

- *Small World* answers the question: given a model class (i.e. a Hypothesis space, whats the best model in it). It involves parameters. Its model checking.
- *BIG World* compares model spaces. Its model comparison with or without "hyperparameters".

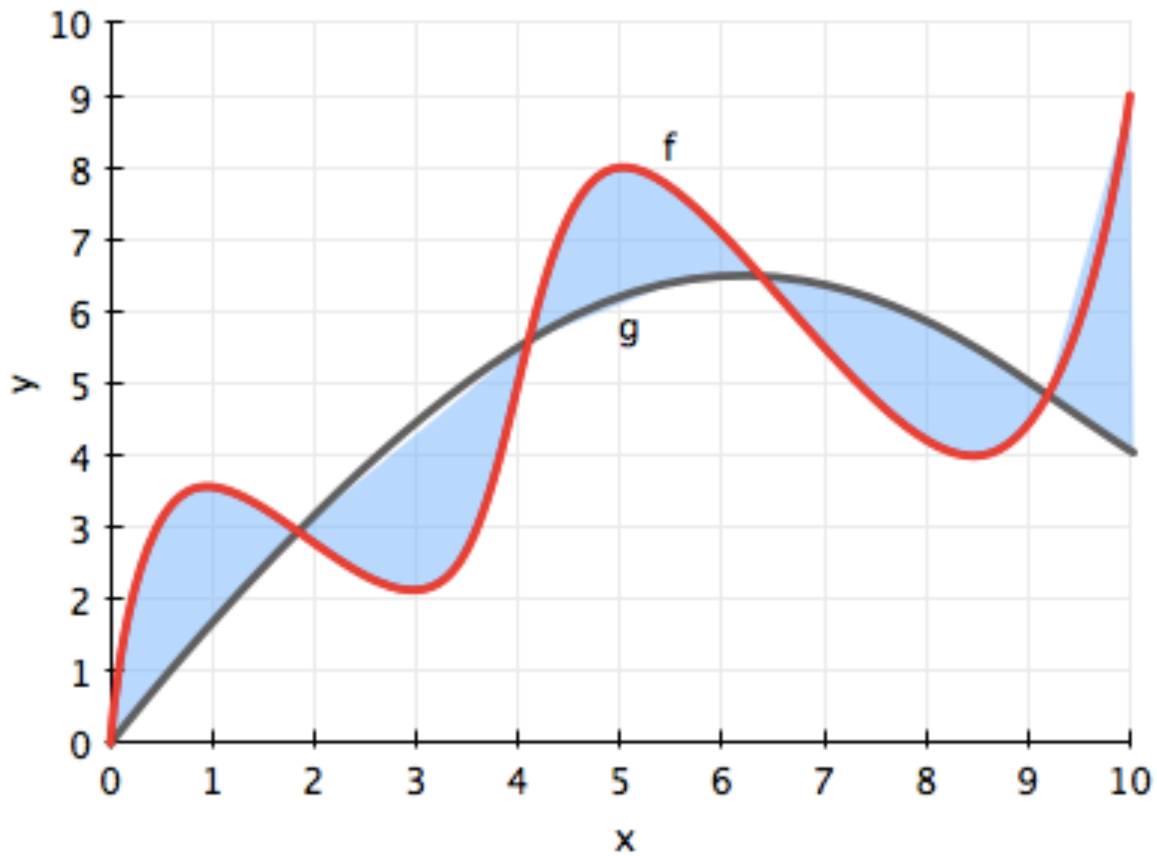
Approximation: Learning without noise

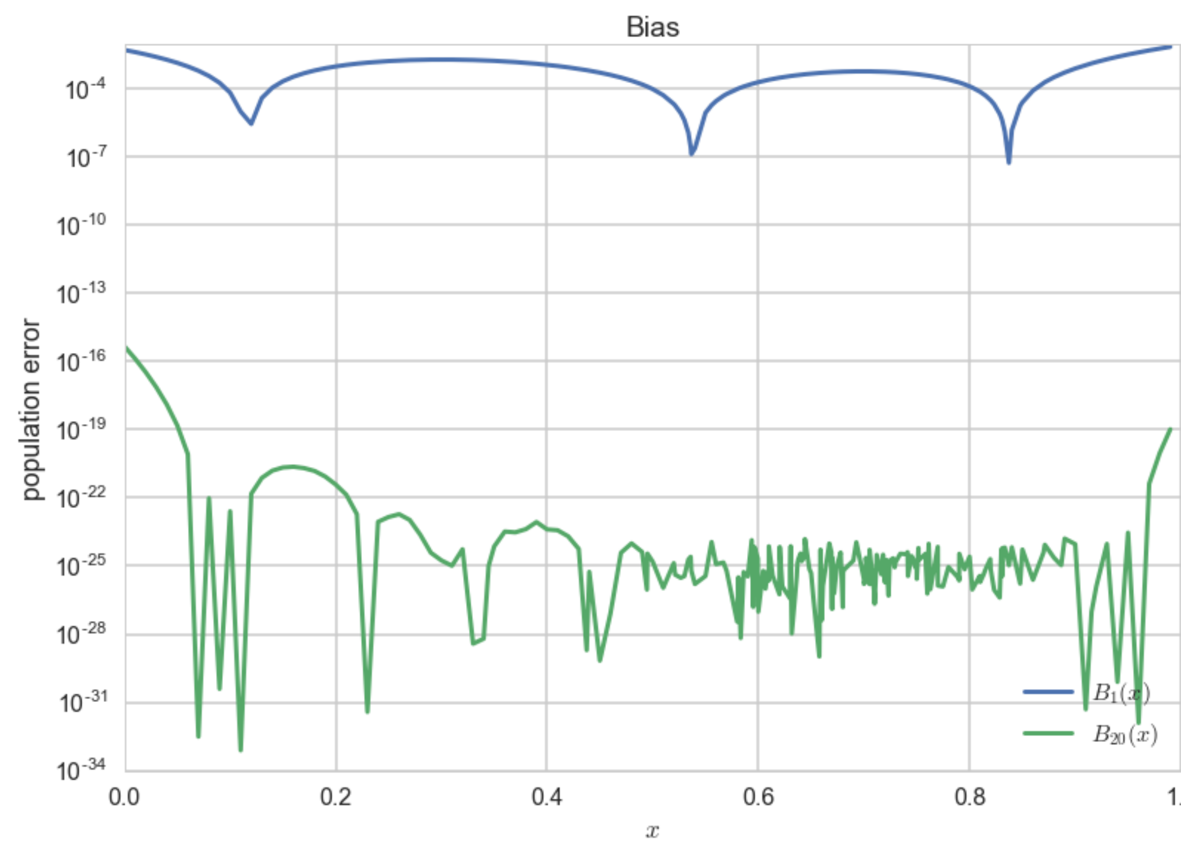
30 points of data. Which fit is better? Line in $\mathcal{H}_1$ or curve in $\mathcal{H}_{20}$?



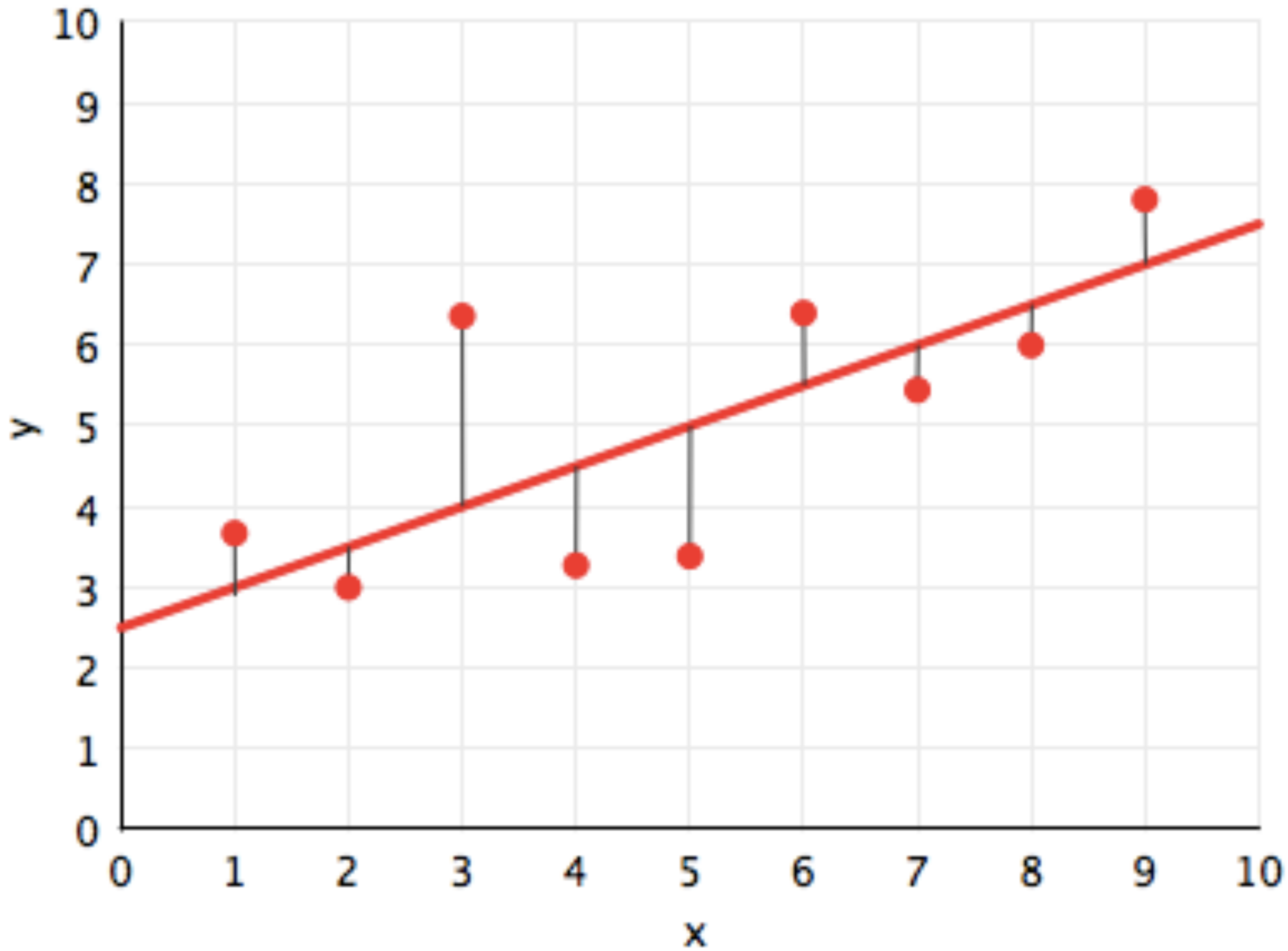


Bias or Mis-specification Error





RISK: What does it mean to FIT?



Minimize distance from the line?

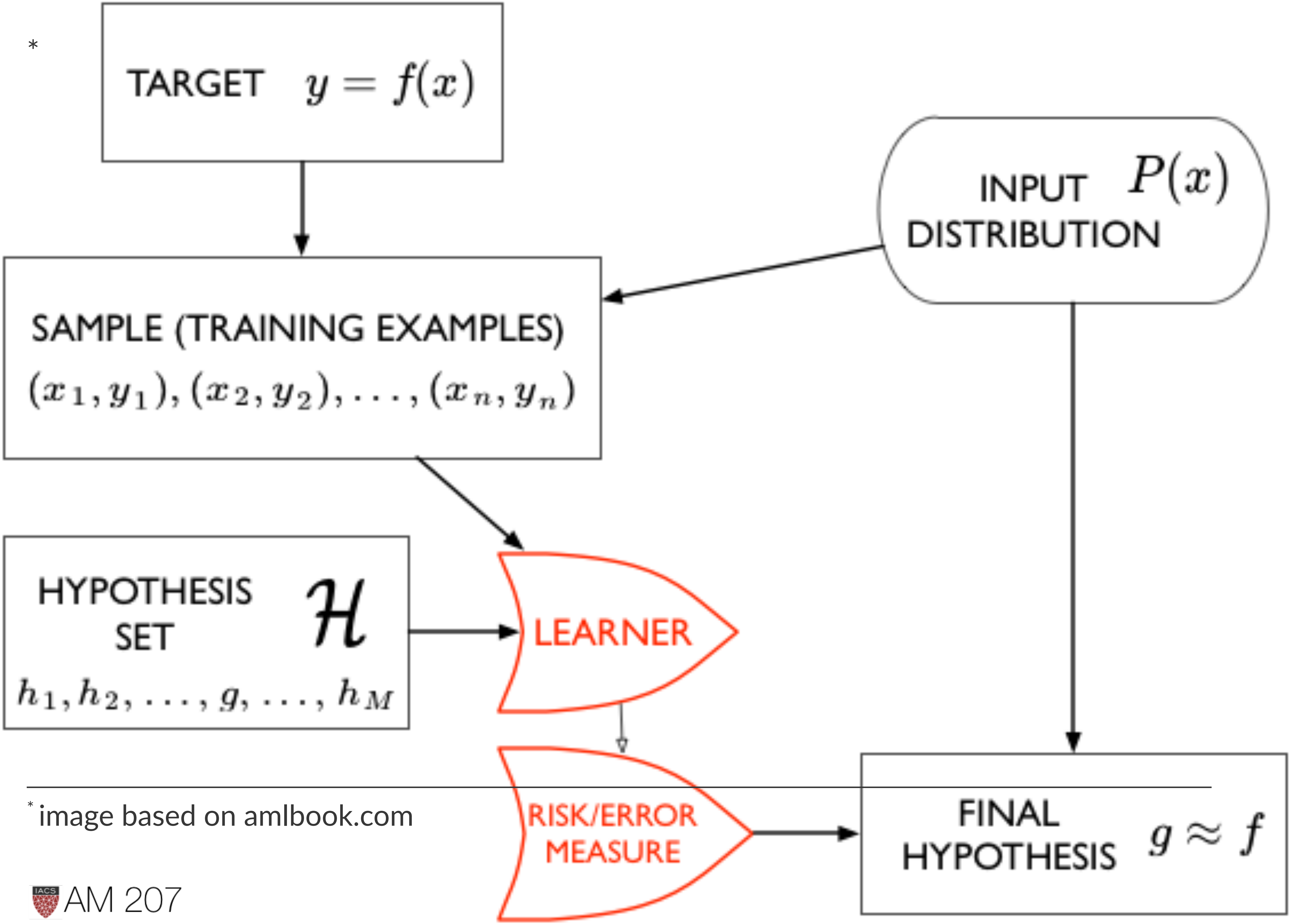
$$R_{\mathcal{D}}(h_1(x)) = \frac{1}{N} \sum_{y_i \in \mathcal{D}} (y_i - h_1(x_i))^2$$

Minimize squared distance from the line. Empirical Risk Minimization.

$$g_1(x) = \arg \min_{h_1(x) \in \mathcal{H}} R_{\mathcal{D}}(h_1(x)).$$

Get intercept w_0 and slope w_1.

*



* image based on amlbook.com

What is noise?

- even in an approximation problem, sampling can be a source of noise
- noise comes from measurement error, missing features, etc
- sometimes it can be systematic as well, but its mostly random on account of being a combination of many small things...

SAMPLE vs POPULATION

Want:

$$R_{out}(h) = E_{p(x)}[(h(x) - f(x))^2] = \int dx p(x)(h(x) - f(x))^2$$

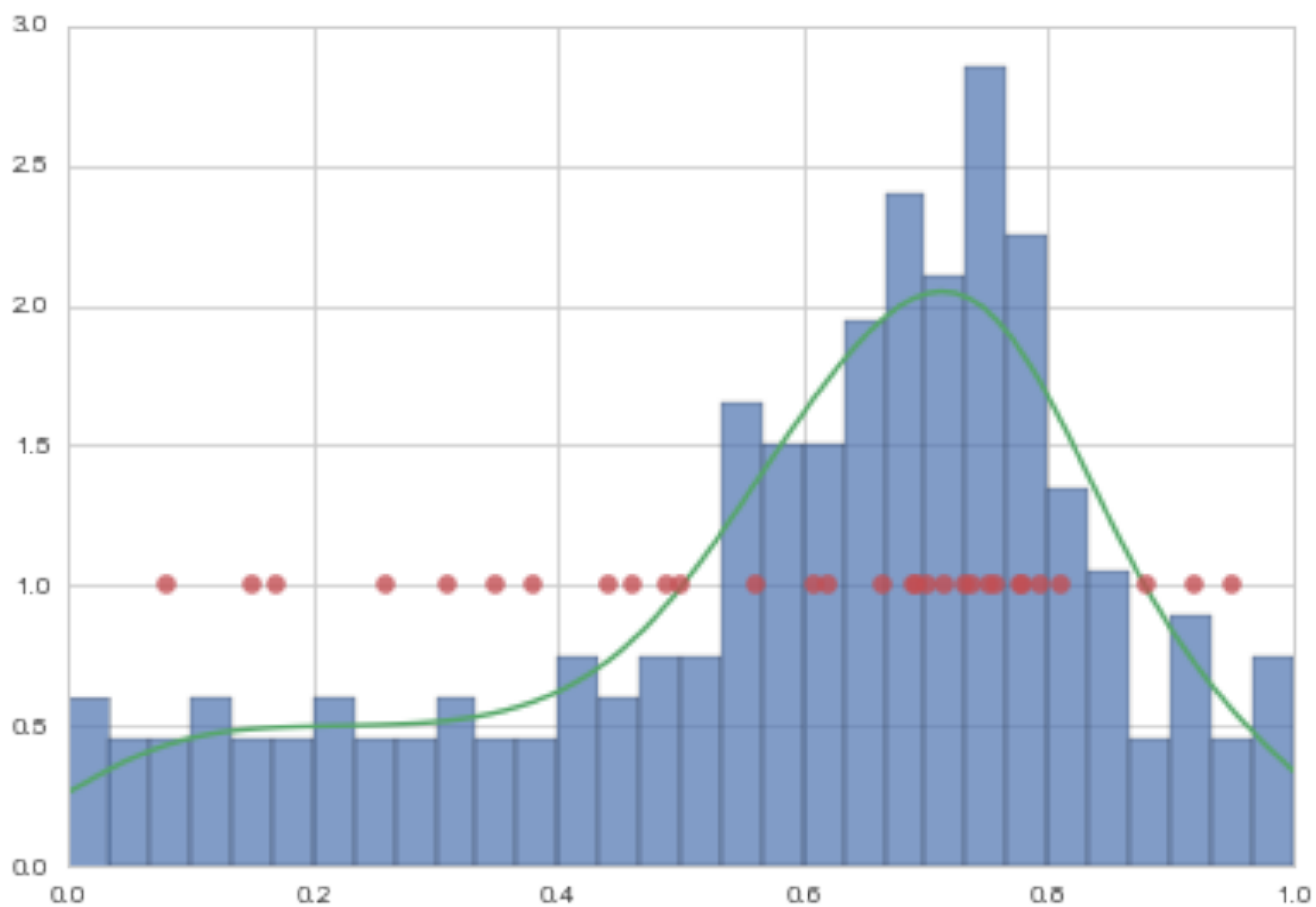
LLN:

$$R_{out}(h) = \lim_{n \to \infty} \frac{1}{n} \sum_{x_i \sim p(x)} (h(x_i) - f(x_i))^2 = \lim_{n \to \infty} \frac{1}{n} \sum_{x_i \sim p(x)} (h(x_i) - y_i)^2$$

$\mathcal{D}$ representative

$$(\mathcal{D} \sim p(x)) \implies \mathcal{R}_{\mathcal{D}}(h) = \sum_{x_i \in \mathcal{D}} (h(x_i) - y_i)^2$$

Statement of the Learning Problem

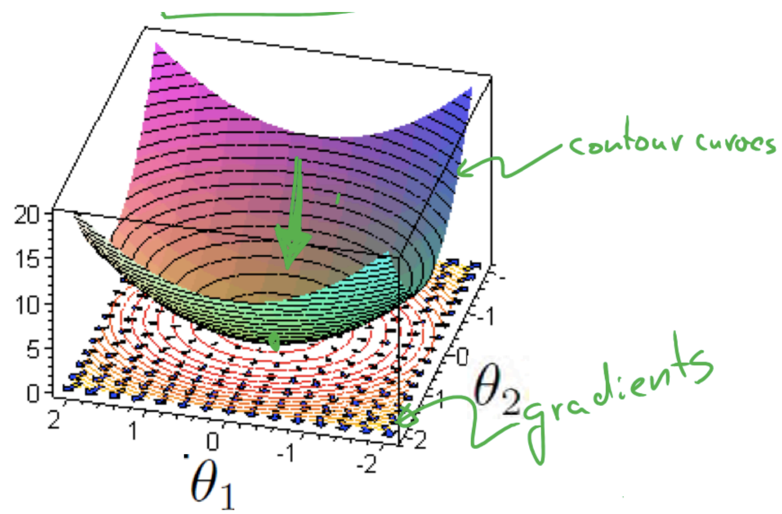


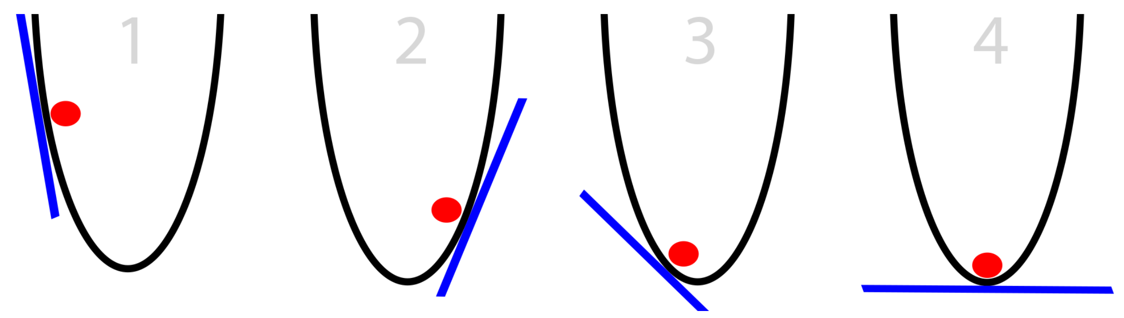
The sample must be representative of the population!

$$A : R_{\mathcal{D}}(g)\ \ smallest\, on\, \mathcal{H}$$
$$B : R_{out}(g) \approx R_{\mathcal{D}}(g)$$

A: Empirical risk estimates in-sample risk.
B: Thus the out of sample risk is also small.

CONVEX MINIMIZATION



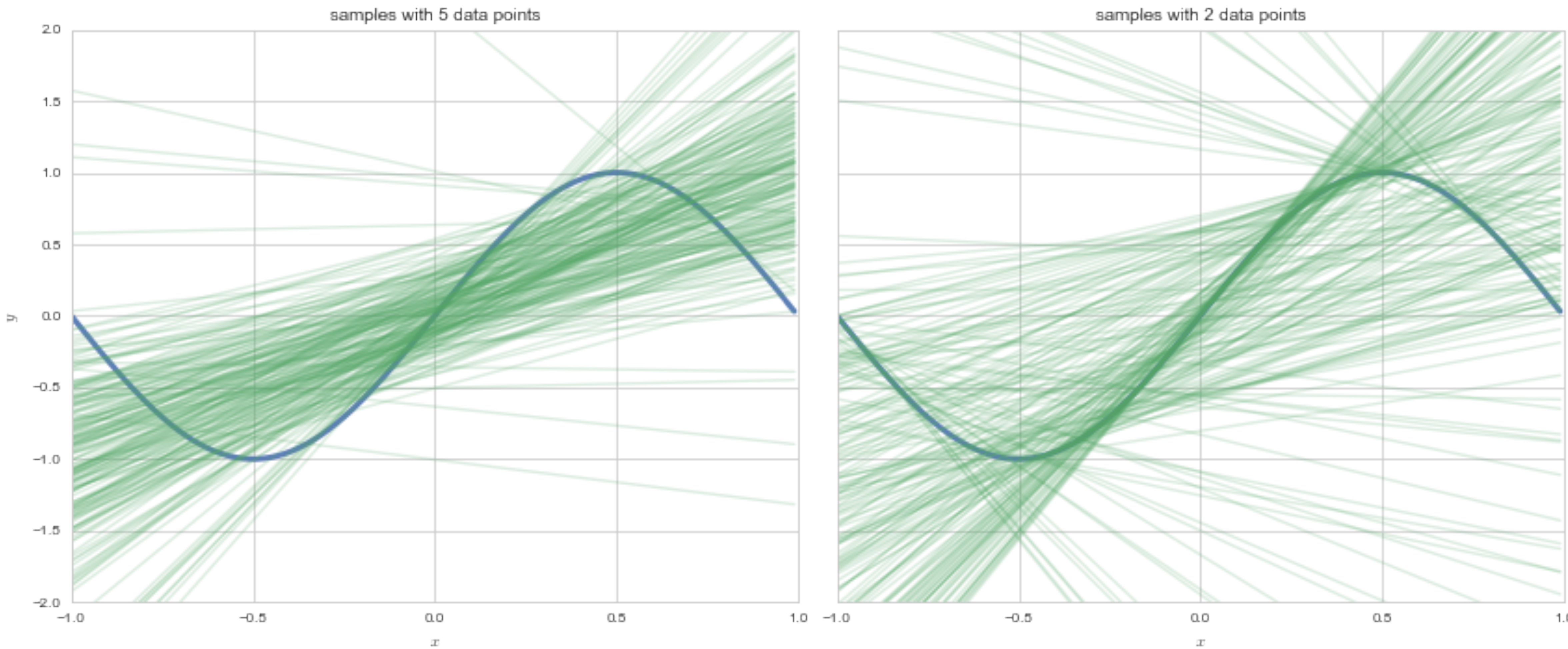


In general one can use gradient descent .

For linear-regression, one can however just do this using matrix algebra.

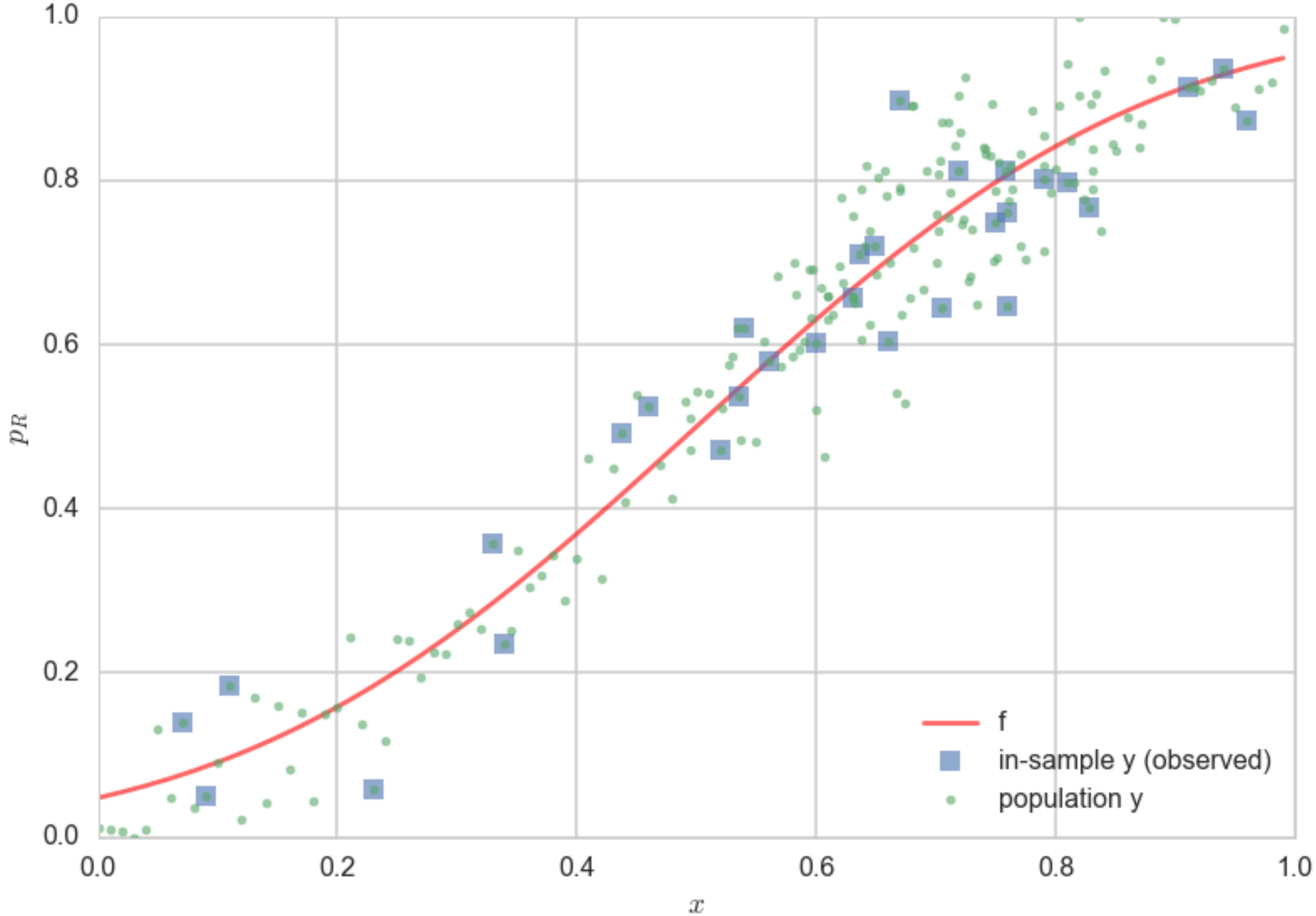
Image From Nando-deFreitas Deep Learning Course 2015

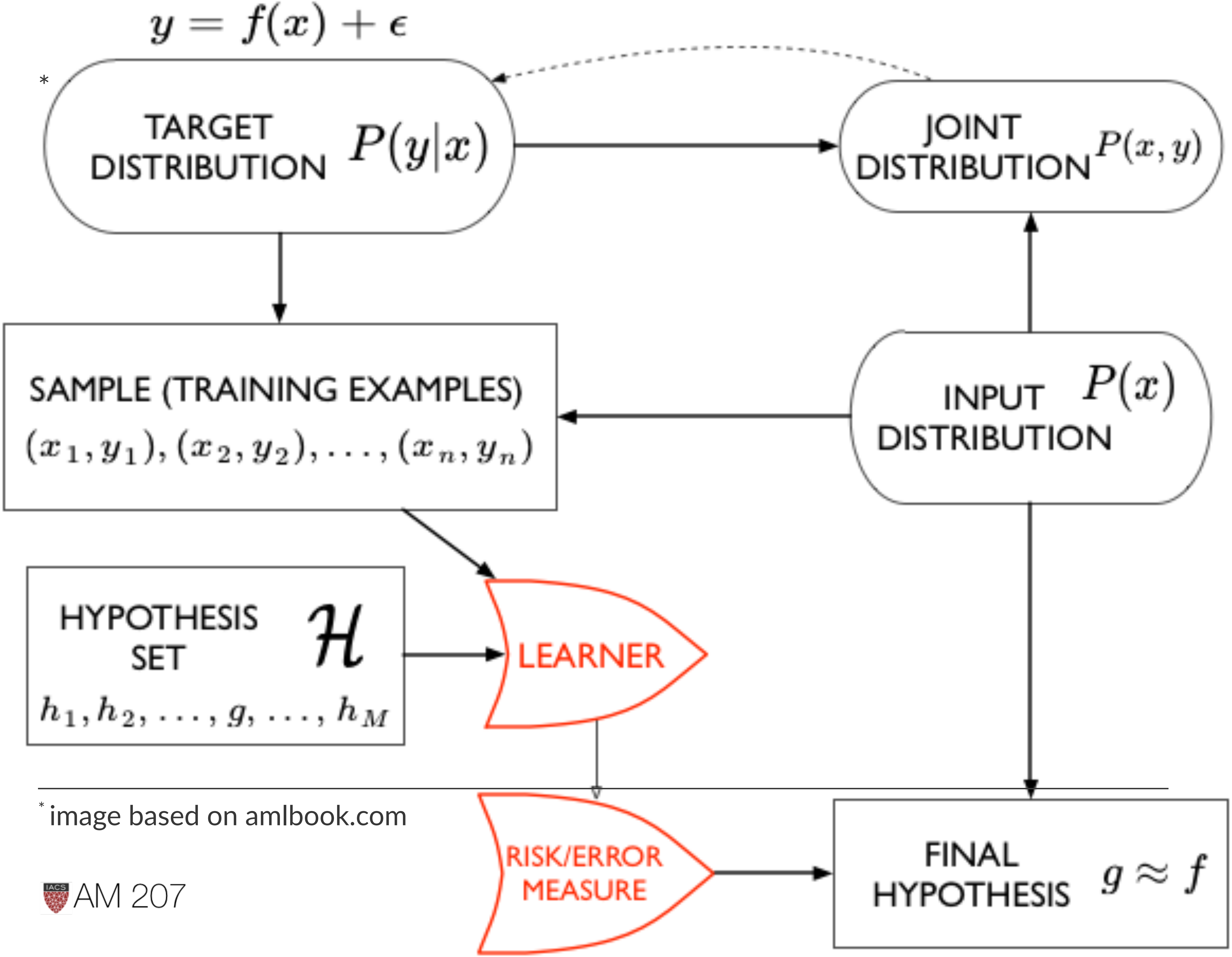
DATA SIZE MATTERS: straight line fits to a sine curve



Corollary: Must fit simpler models to less data!

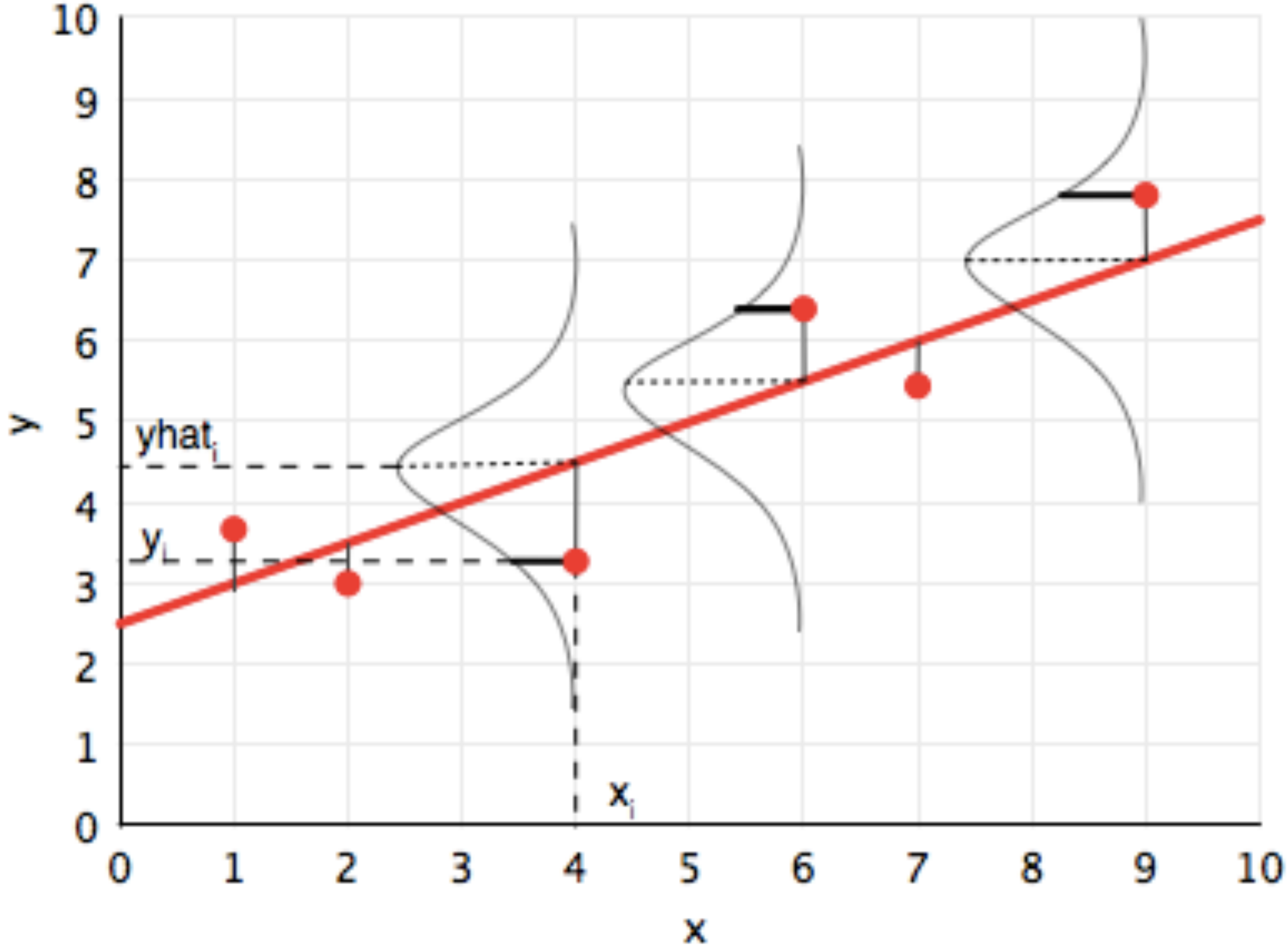
THE REAL WORLD HAS NOISE





* image based on amlbook.com

Linear Regression MLE



Gaussian Distribution assumption

Each y_i is gaussian distributed with "mean" $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x}_i$ (the regression line) and there is noise ϵ with variance σ^2:

$$y_i \sim N(\mathbf{w} \cdot \mathbf{x}_i, \sigma^2).$$

$$N(\mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(y-\mu)^2/2\sigma^2},$$

We can then write the likelihood:

$$\mathcal{L} = p(\mathbf{y}|\mathbf{x}, \mathbf{w}, \sigma) = \prod_i p(\mathbf{y}_i|\mathbf{x}_i, \mathbf{w}, \sigma)$$

$$\mathcal{L} = (2\pi\sigma^2)^{(-n/2)} e^{\frac{-1}{2\sigma^2} \sum_i (y_i - \mathbf{w}\cdot\mathbf{x}_i)^2}.$$

The log likelihood ℓ then is given by:

$$\ell = \frac{-n}{2} log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_i (y_i - \mathbf{w} \cdot \mathbf{x}_i)^2.$$

Maximizing gives:

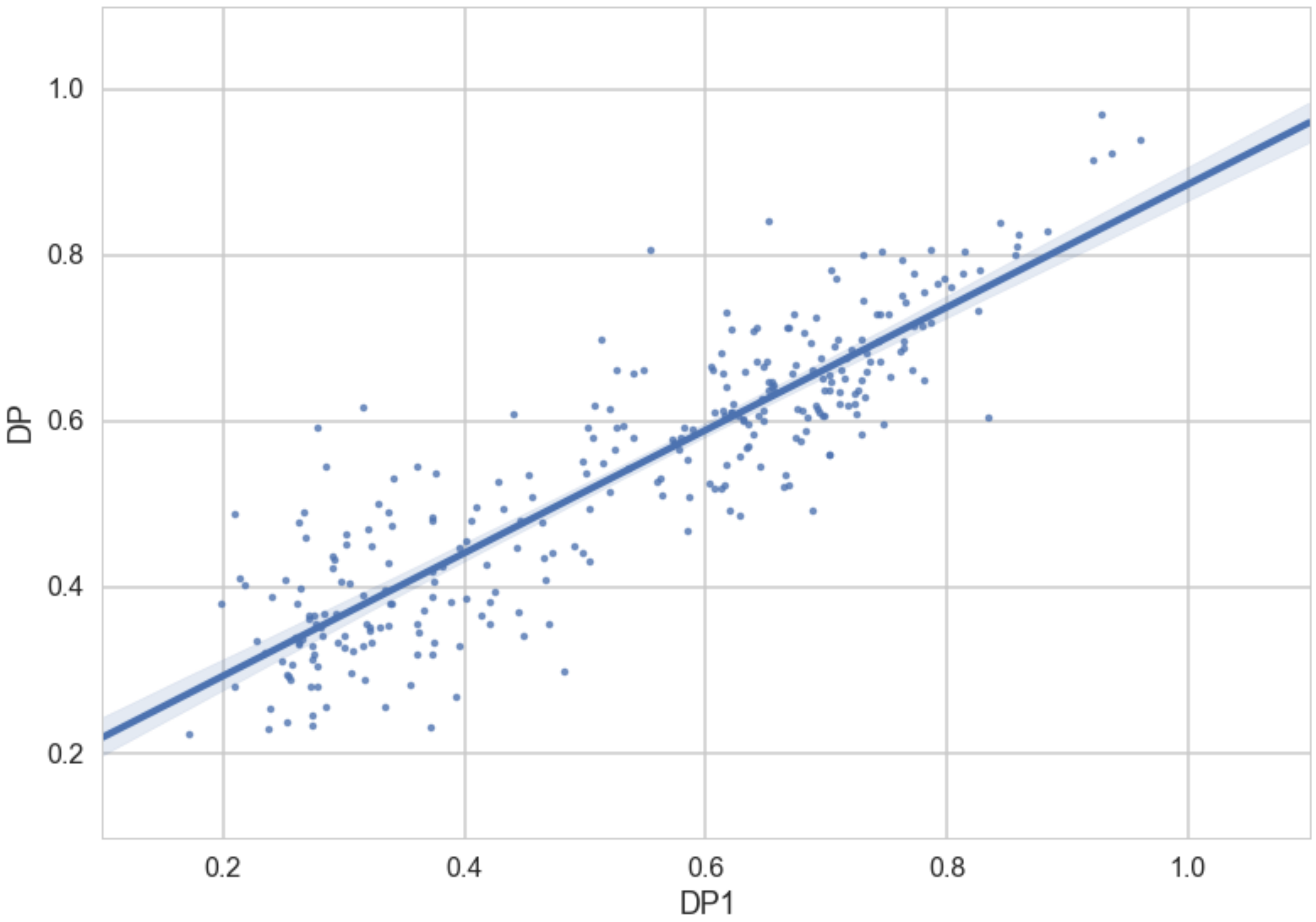
$$\mathbf{w}_{MLE} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y},$$

where we stack rows to get:

$$\mathbf{X} = stack(\{\mathbf{x}_i\})$$

$$\sigma^2_{MLE} = \frac{1}{n}\sum_i (y_i - \mathbf{w}\cdot\mathbf{x}_i)^2.$$

Example: House Elections



From Likelihood to Predictive Distribution

- the band on the previous graph is the sampling distribution of the regression line, or a representation of the sampling distribution of the $\mathbf{w}$.
- $p(y|\mathbf{x}, \mu_{MLE}, \sigma^2_{MLE})$ is a probability distribution
- thought of as $p(y^*|\mathbf{x}^*, \{\mathbf{x}_i, y_i\}, \mu_{MLE}, \sigma^2_{MLE})$, it is a predictive distribution for as yet unseen data y^* at $\mathbf{x}^*$, or the sampling distribution for data, or the data-generating distribution, at the new covariates $\mathbf{x}^*$. This is a wider band.

```
Dem_Perc(t) ~
Dem_Perc(t-2) + I
```

- done in statsmodels
- From Gelman and Hwang

Dep. Variable:	DP	**R-squared:**	0.806
Model:	OLS	**Adj. R-squared:**	0.804
Method:	Least Squares	**F-statistic:**	612.0
Date:	Tue, 13 Oct 2015	**Prob (F-statistic):**	1.04e-105
Time:	16:33:01	**Log-Likelihood:**	368.81
No. Observations:	298	**AIC:**	-731.6
Df Residuals:	295	**BIC:**	-720.5
Df Model:	2		
Covariance Type:	nonrobust		

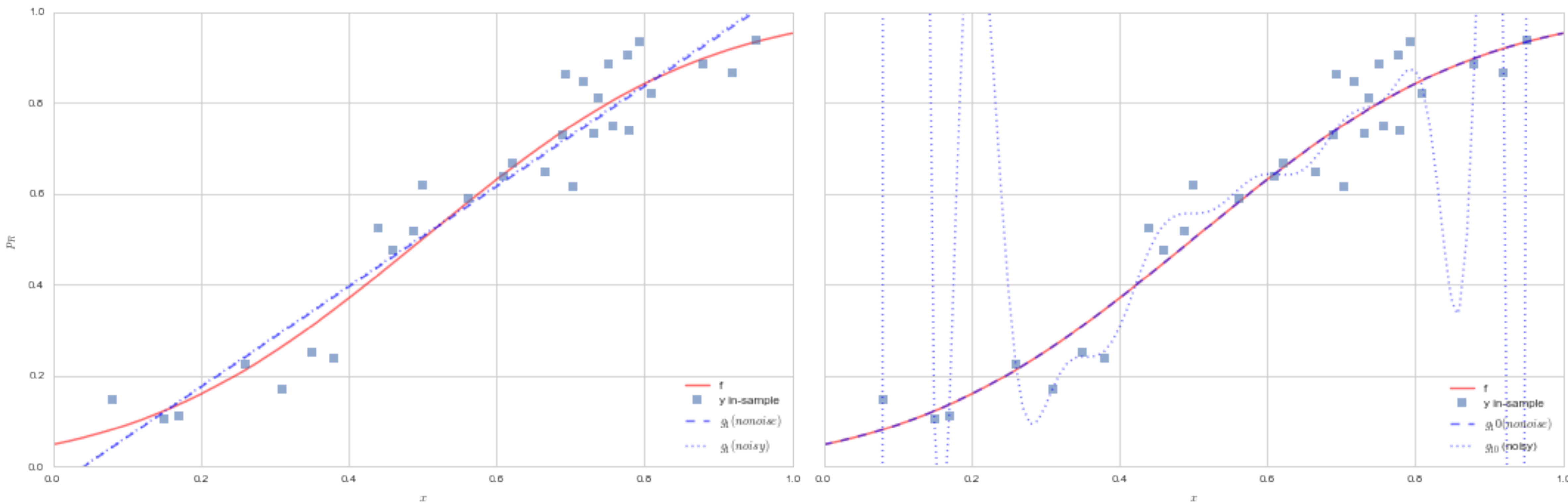
	coef	**std err**	**t**	**P>|t|**	**[95.0% Conf. Int.]**
Intercept	0.2326	0.020	11.503	0.000	0.193 0.272
DP1	0.5622	0.040	14.220	0.000	0.484 0.640
I	0.0429	0.008	5.333	0.000	0.027 0.059

Omnibus:	7.465	**Durbin-Watson:**	1.728
Prob(Omnibus):	0.024	**Jarque-Bera (JB):**	7.316
Skew:	0.374	**Prob(JB):**	0.0258
Kurtosis:	3.174	**Cond. No.**	13.1

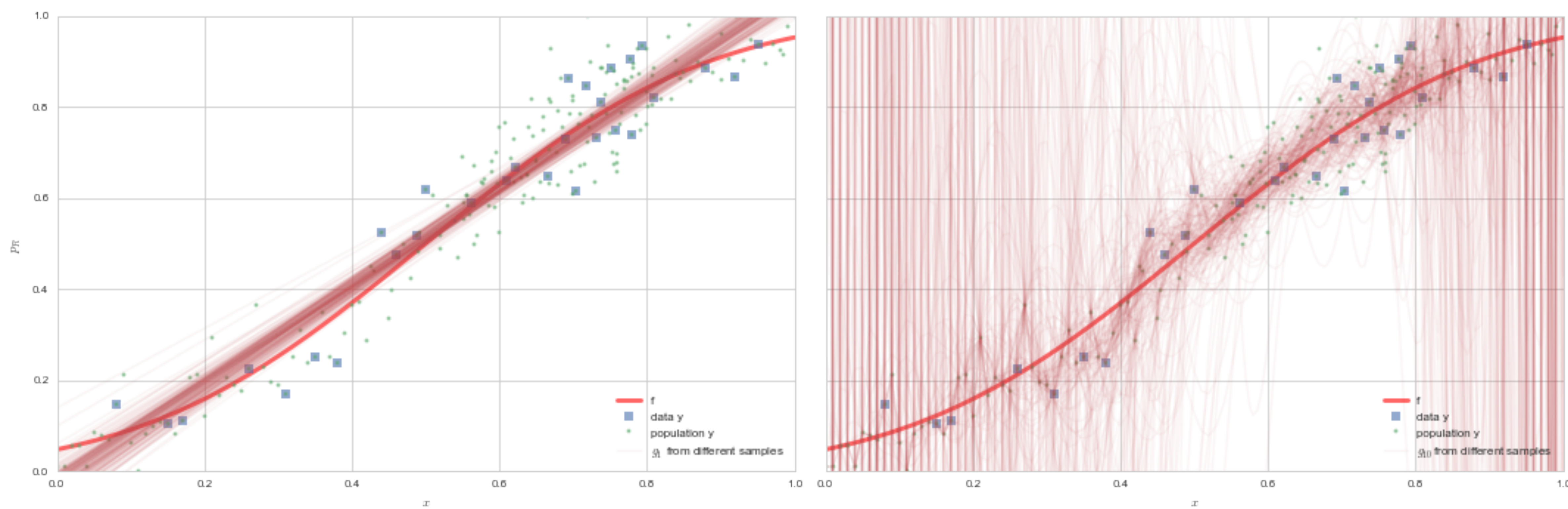
THE REAL WORLD HAS NOISE

Which fit is better now?

The line or the curve?



UNDERFITTING (Bias) vs OVERFITTING (Variance)



Every model has Bias and Variance

$$R_{out}(h) = E_{p(x)}[(h(x) - y)^2] = \int dx p(x)(h(x) - f(x) - \epsilon)^2.$$

Fit hypothesis $h = g_{\mathcal{D}}$, where $\mathcal{D}$ is our training sample.

Define:

$$\langle R \rangle = \int dy dx\, p(x,y)(h(x) - y)^2 = \int dy dx p(y \mid x) p(x)(h(x) - y)^2.$$

$$\langle R \rangle = E_{\mathcal{D}}[R_{out}(g_{\mathcal{D}})] = E_{\mathcal{D}} E_{p(x)}[(g_{\mathcal{D}}(x) - f(x) - \epsilon)^2]$$

$$\bar{g} = E_{\mathcal{D}}[g_{\mathcal{D}}] = (1/M) \sum_{\mathcal{D}} g_{\mathcal{D}}$$

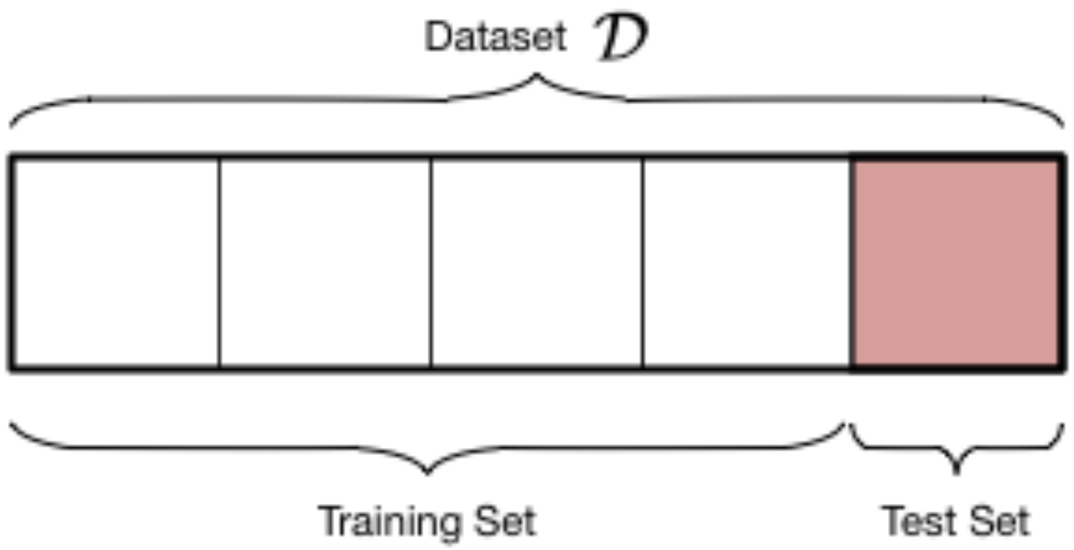
Then,

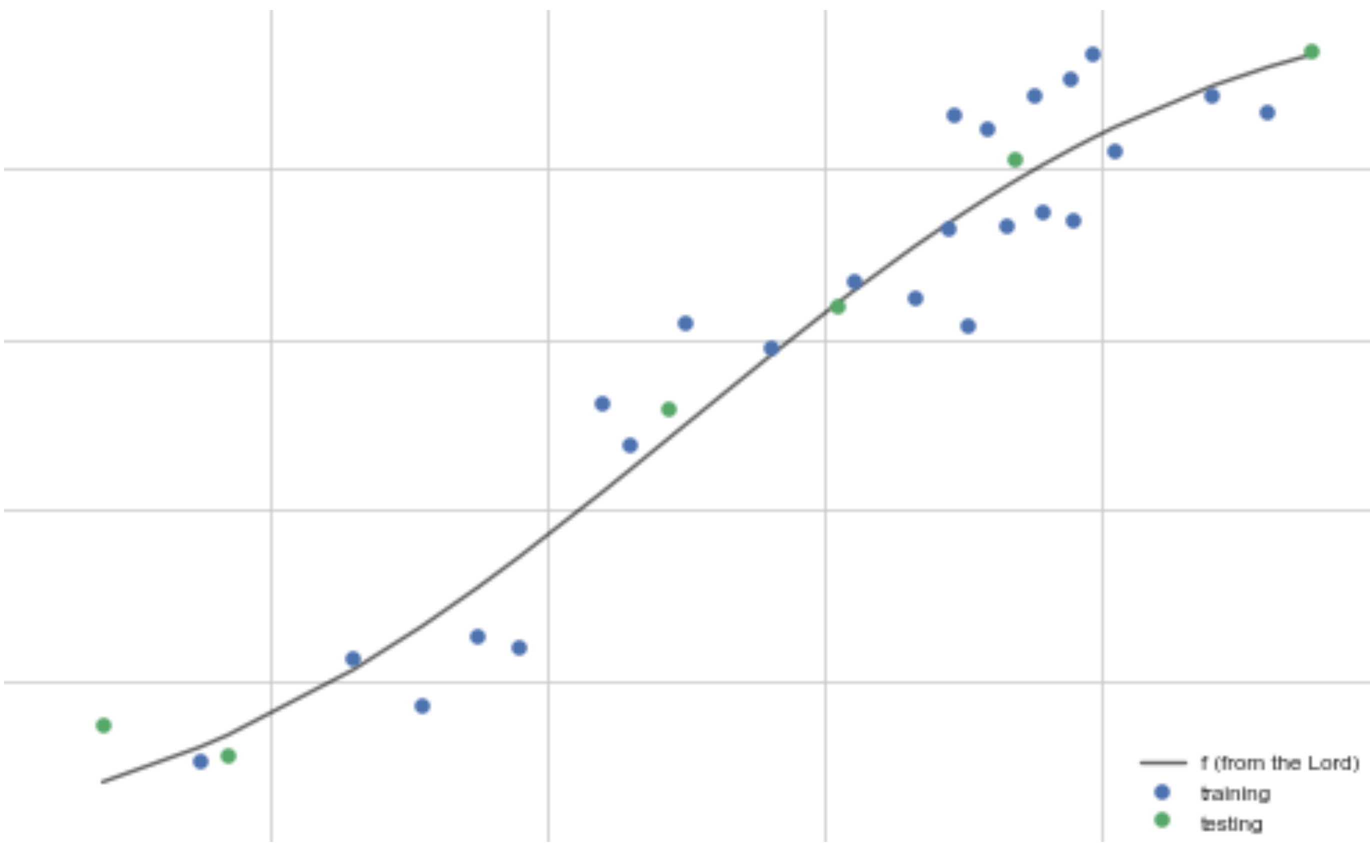
$$\langle R \rangle = E_{p(x)}[E_{\mathcal{D}}[(g_{\mathcal{D}} - \bar{g})^2]] + E_{p(x)}[(f - \bar{g})^2] + \sigma^2$$

This is the bias variance decomposition for regression.

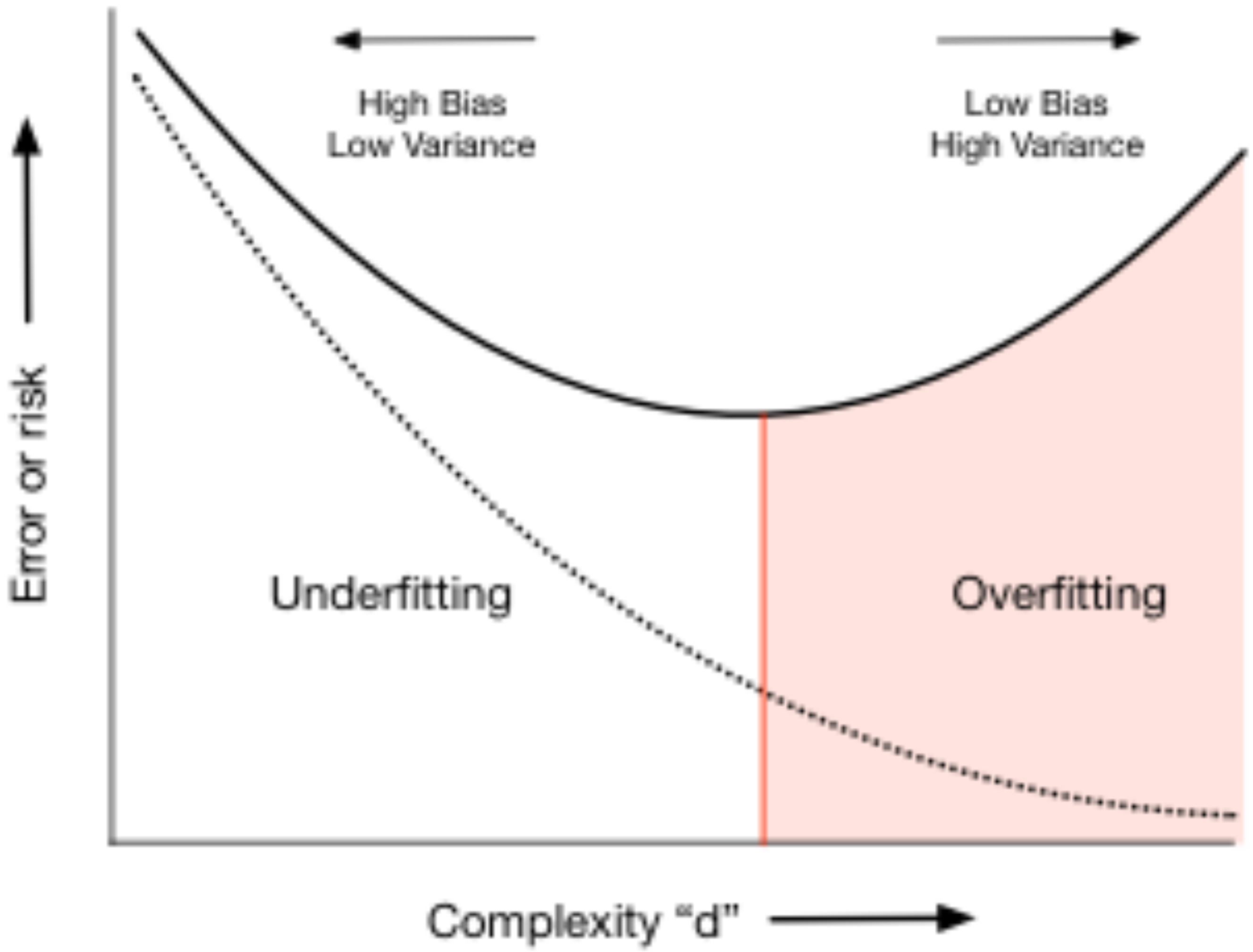
- first term is **variance**, squared error of the various fit g's from the average g, the hairiness.
- second term is **bias**, how far the average g is from the original f this data came from.
- third term is the **stochastic noise**, minimum error that this model will always have.

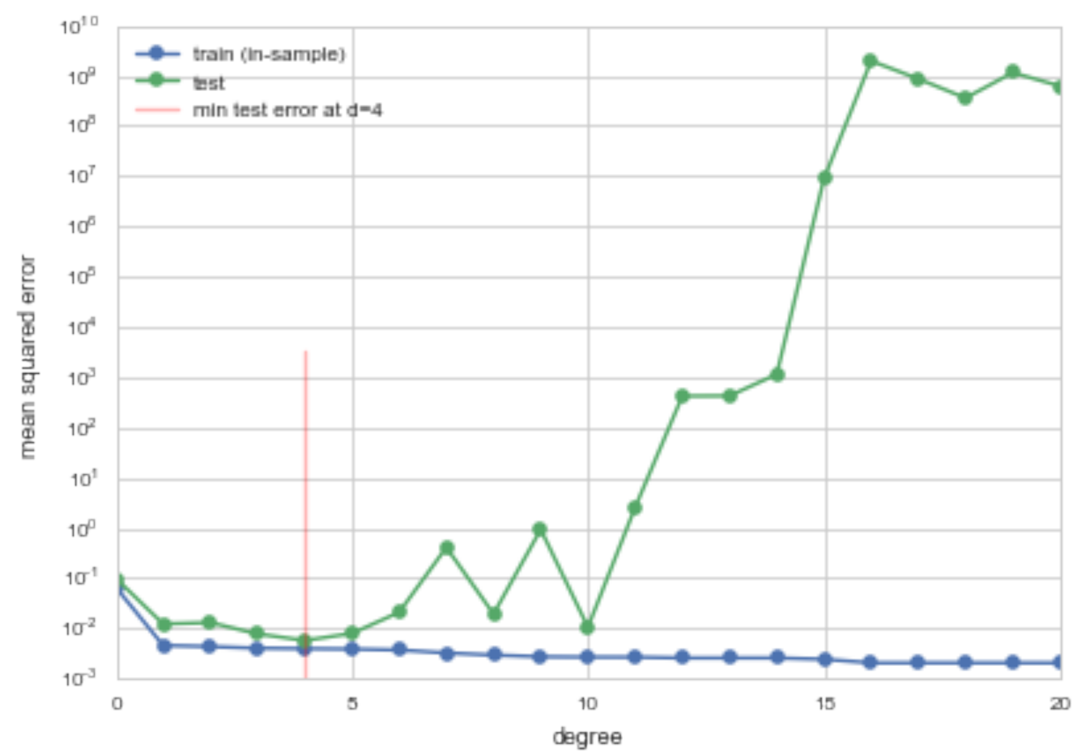
TRAIN AND TEST





BALANCE THE COMPLEXITY: A LARGE WORLD APPROACH





Is this still a test set?

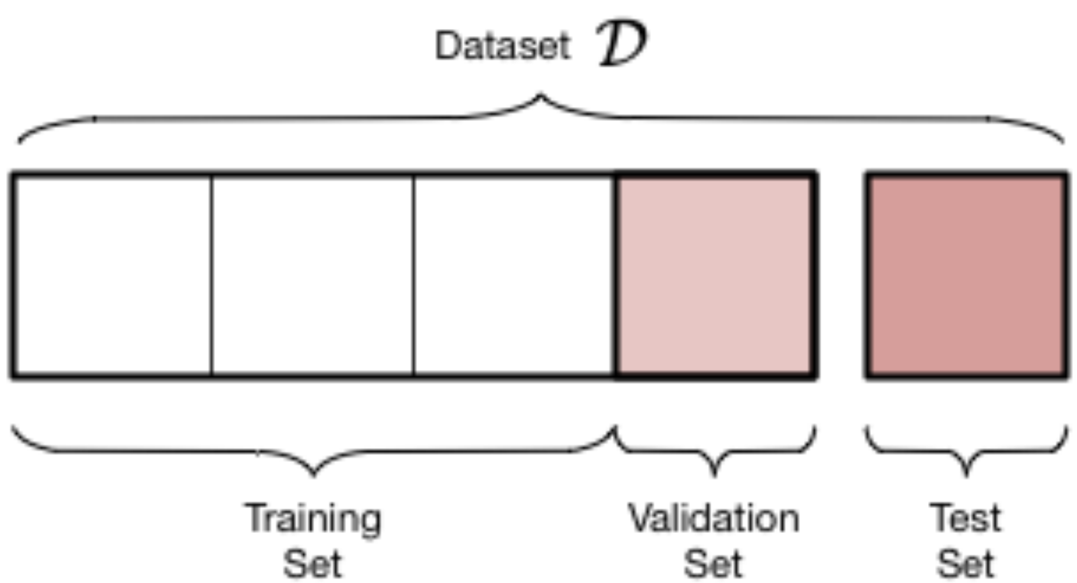
Trouble:

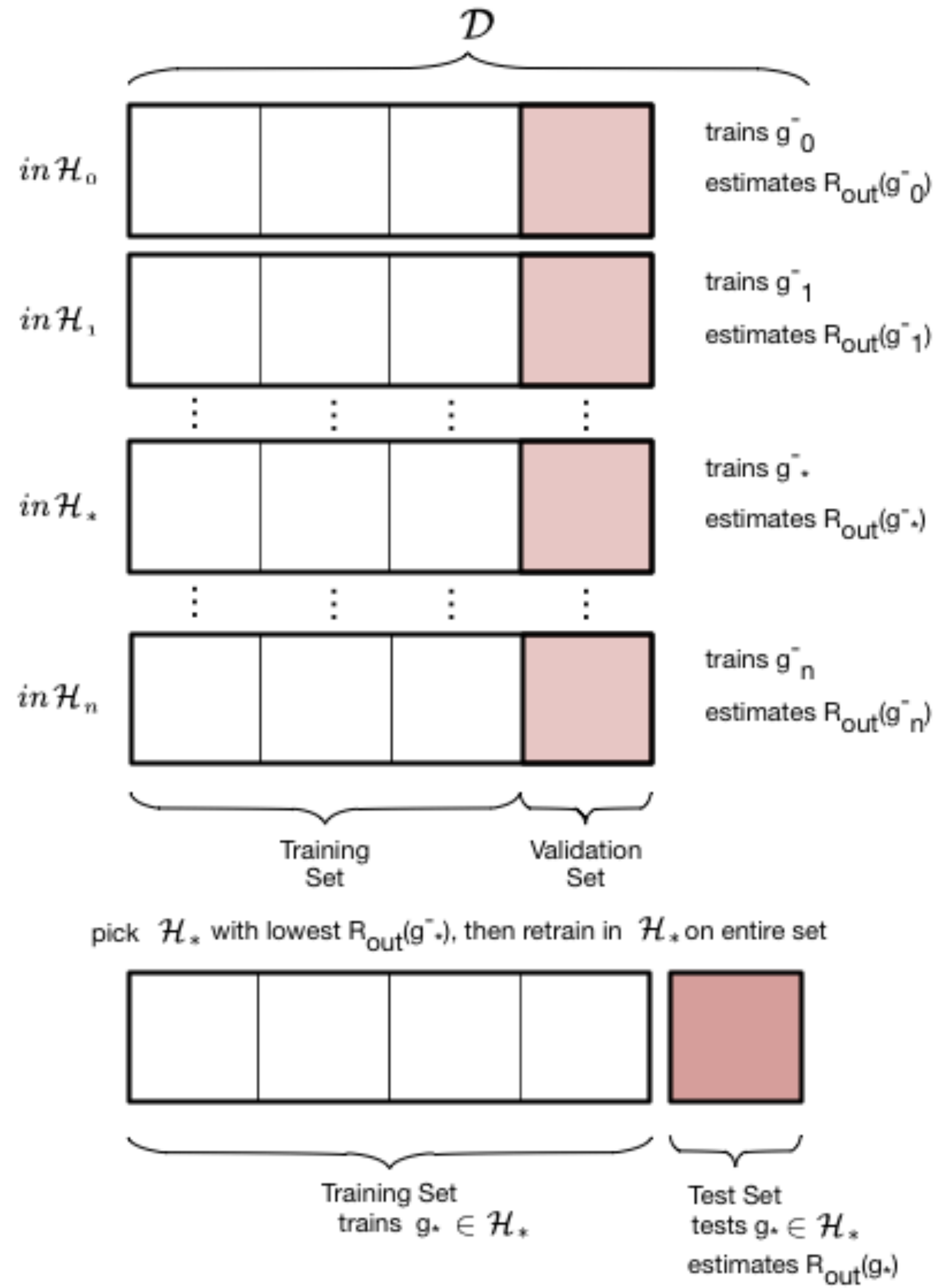
- no discussion on the error bars on our error estimates
- "visually fitting" a value of $d \implies$ contaminated test set.

The moment we **use it in the learning process, it is not a test set**.

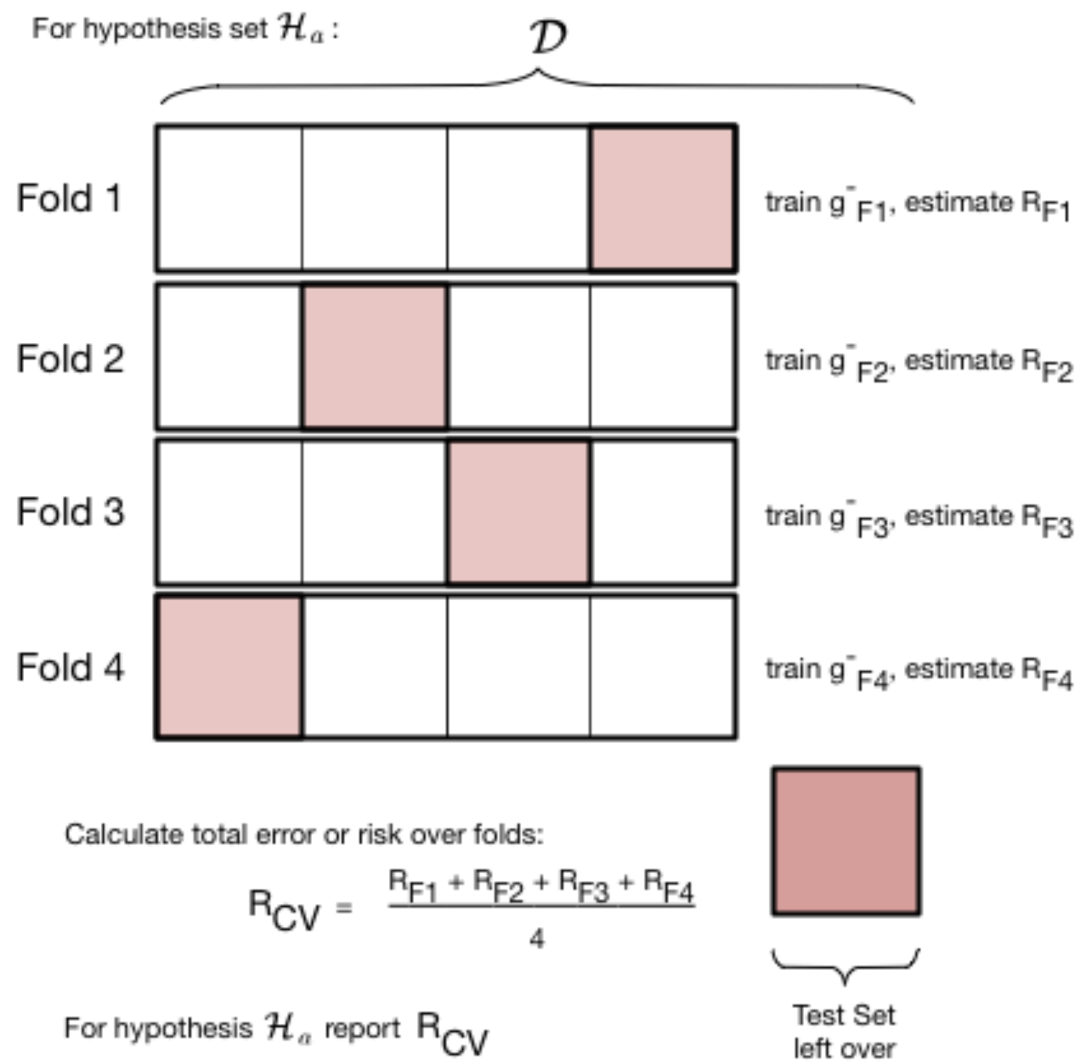
VALIDATION

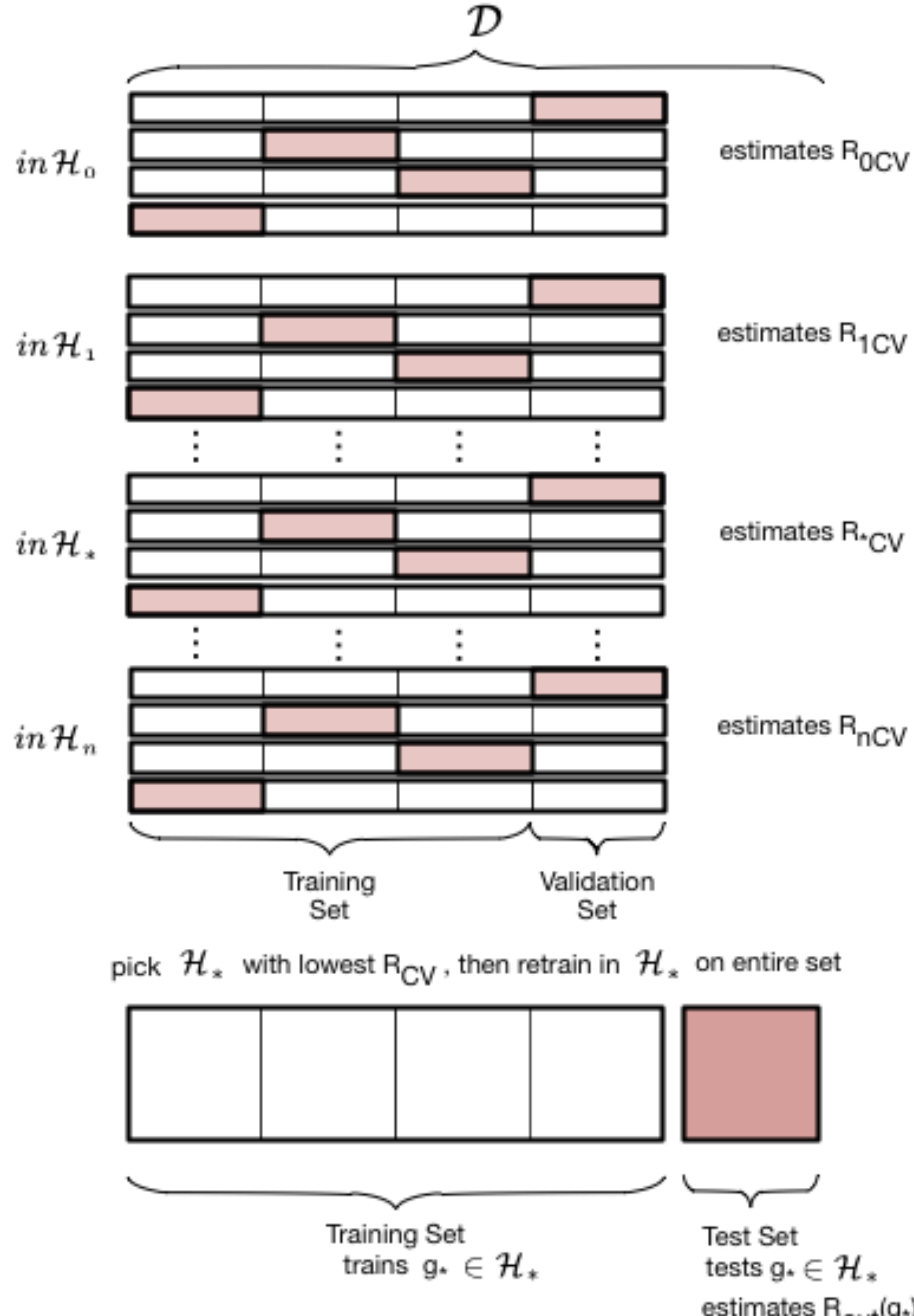
- train-test not enough as we *fit* for d on test set and contaminate it
- thus do train-validate-test

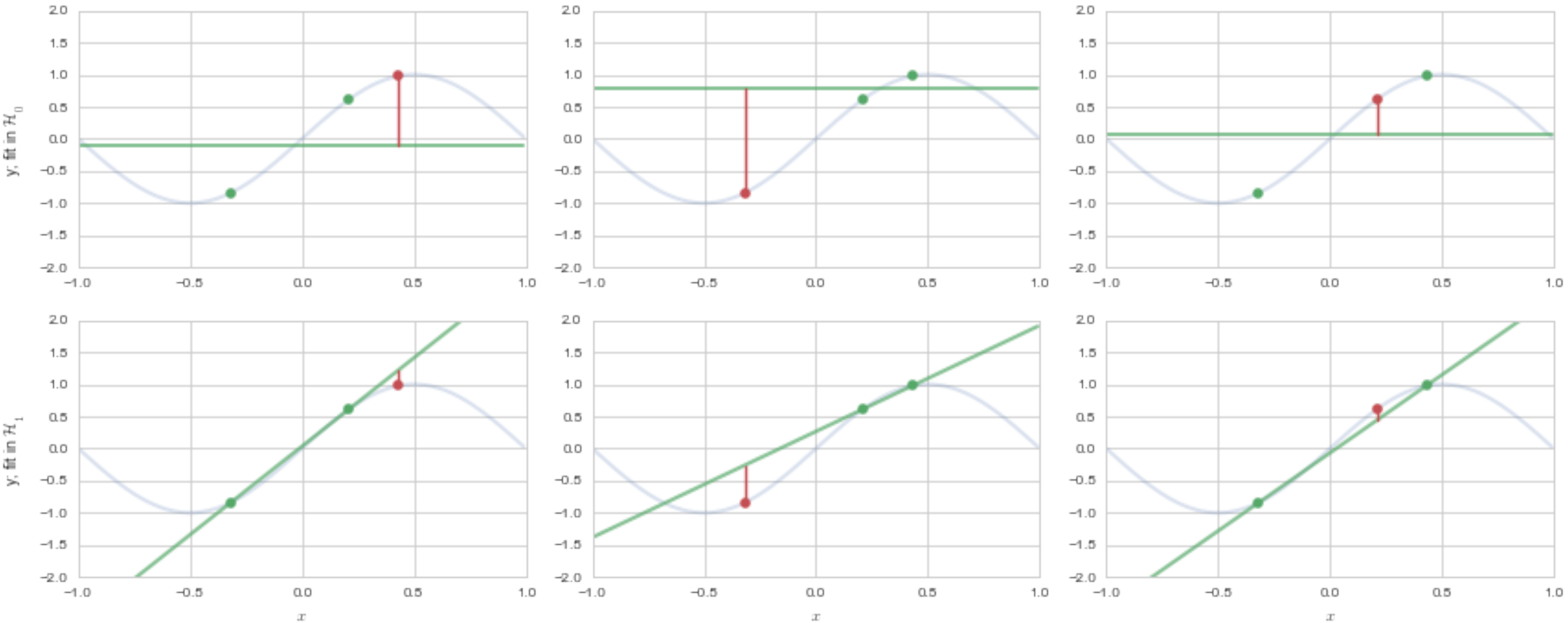




CROSS-VALIDATION





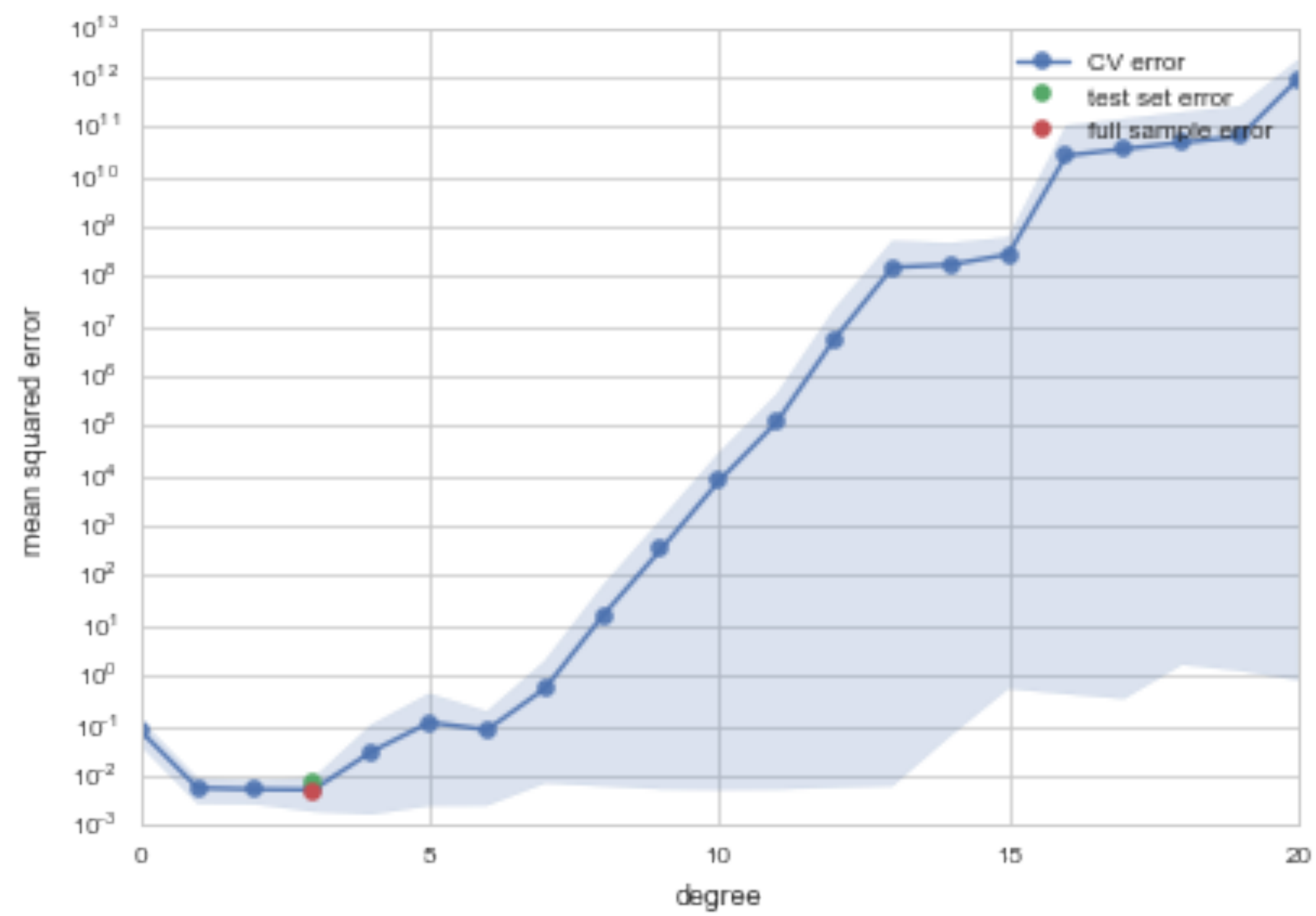

y; fit in $\mathcal{H}_0$
y; fit in $\mathcal{H}_1$
x

CROSS-VALIDATION

is

- a resampling method
- robust to outlier validation set
- allows for larger training sets
- allows for error estimates

Here we find $d = 3$.



Cross Validation considerations

- validation process as one that estimates R_{out} directly, on the validation set. It's critical use is in the model selection process.
- once you do that you can estimate R_{out} using the test set as usual, but now you have also got the benefit of a robust average and error bars.
- key subtlety: in the risk averaging process, you are actually averaging over different g^- models, with different parameters.

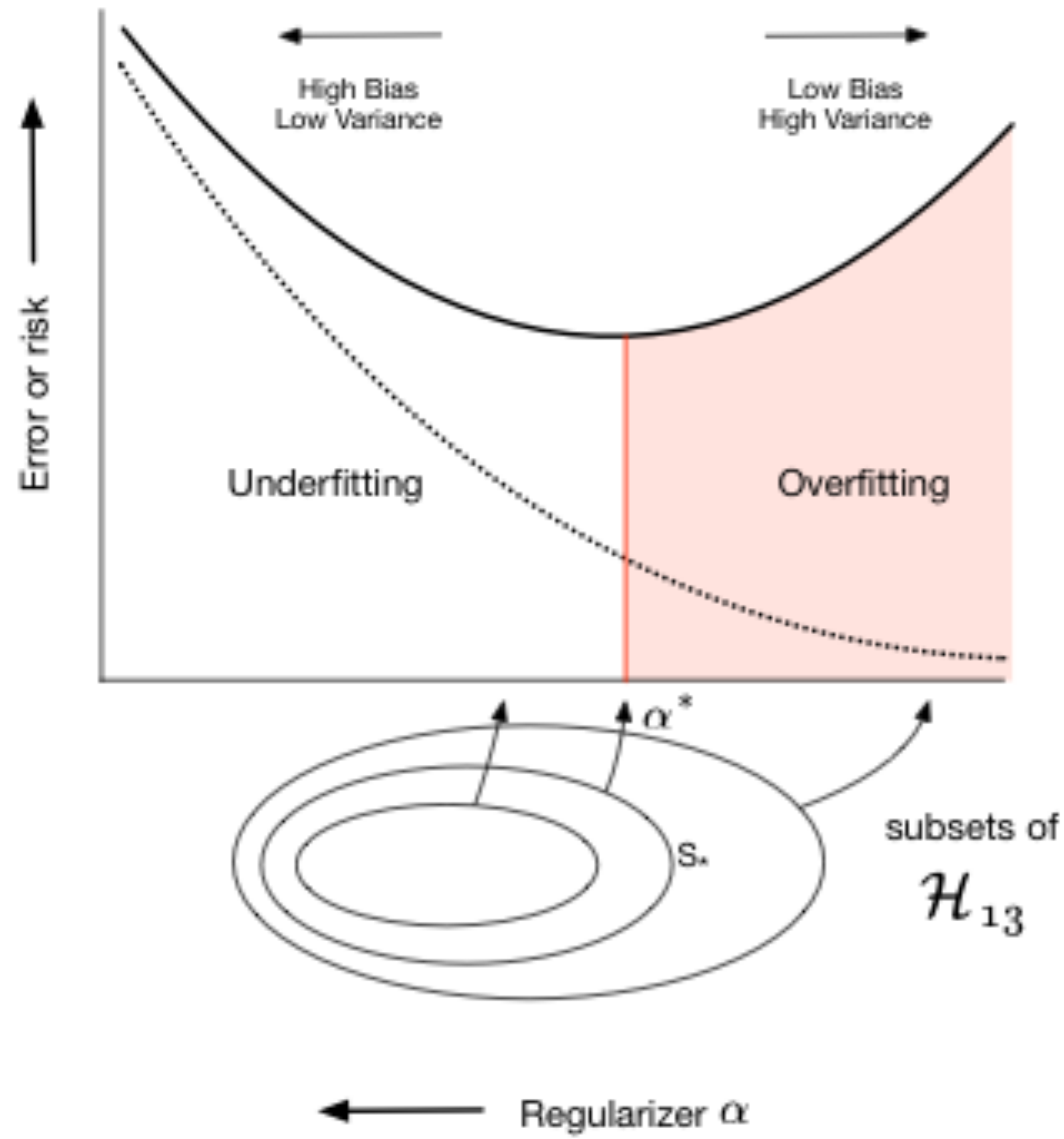
REGULARIZATION: A SMALL WORLD APPROACH

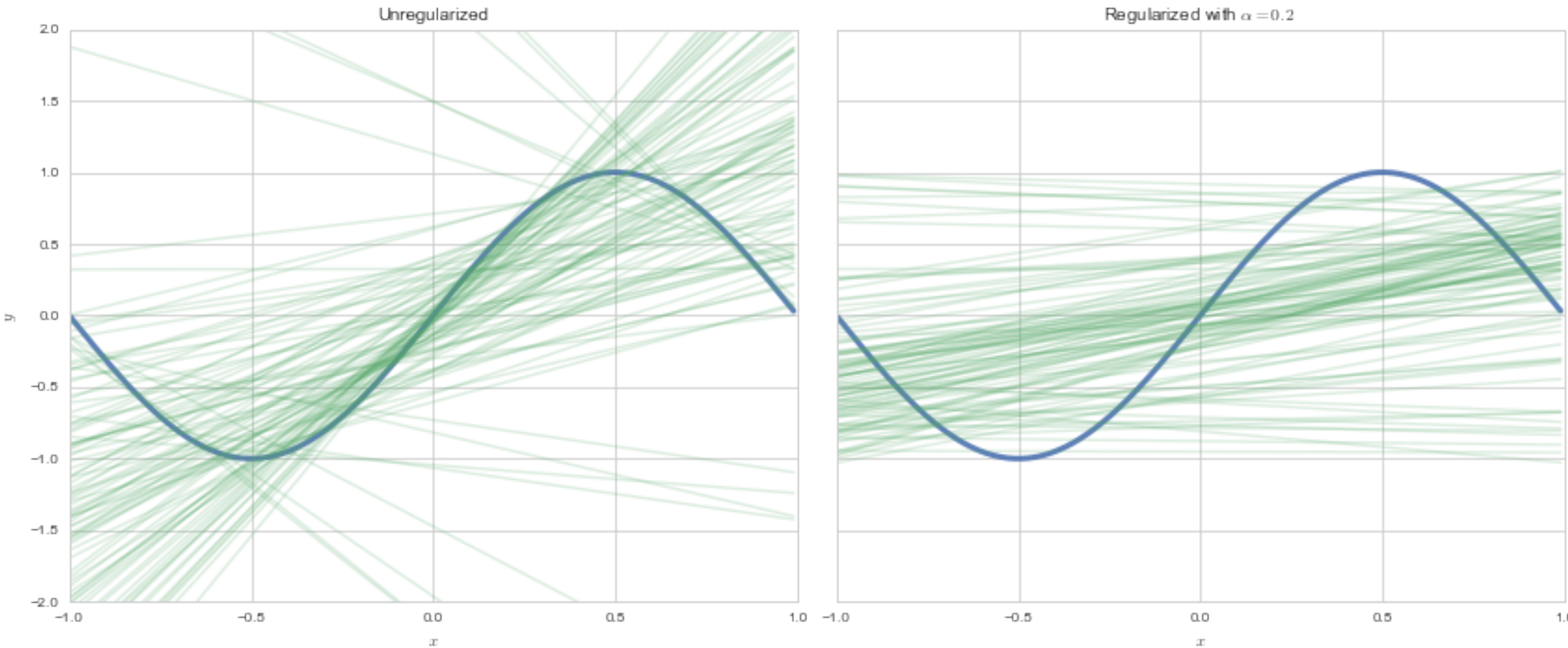
Keep higher a-priori complexity and impose a

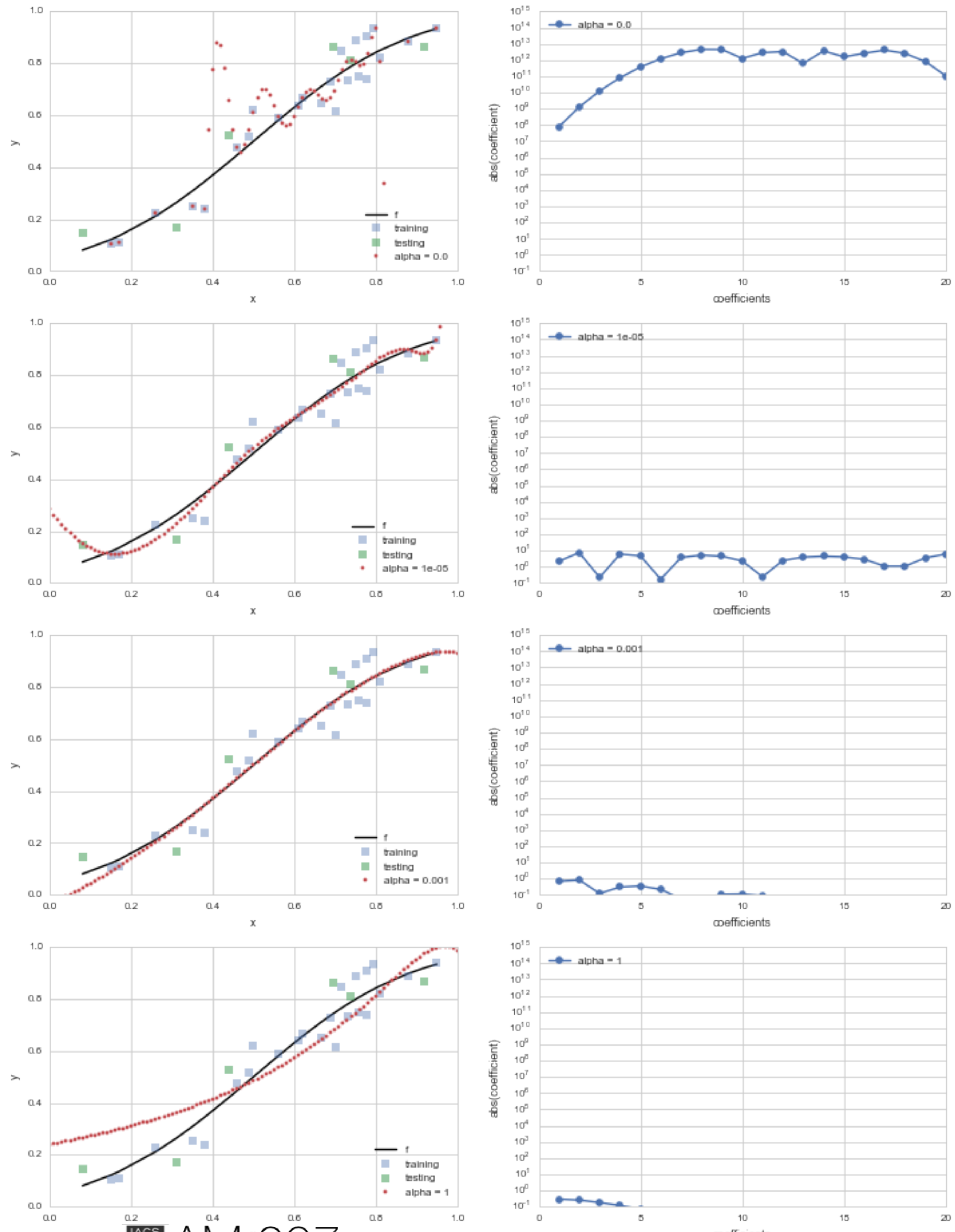
complexity penalty

on risk instead, to choose a SUBSET of $\mathcal{H}_{big}$. We'll make the coefficients small:

$$\sum_{i=0}^{j} \theta_i^2 < C.$$




Unregularized
Regularized with $\alpha = 0.2$
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−1.0
−0.5
0.0
0.5
1.0
x
y



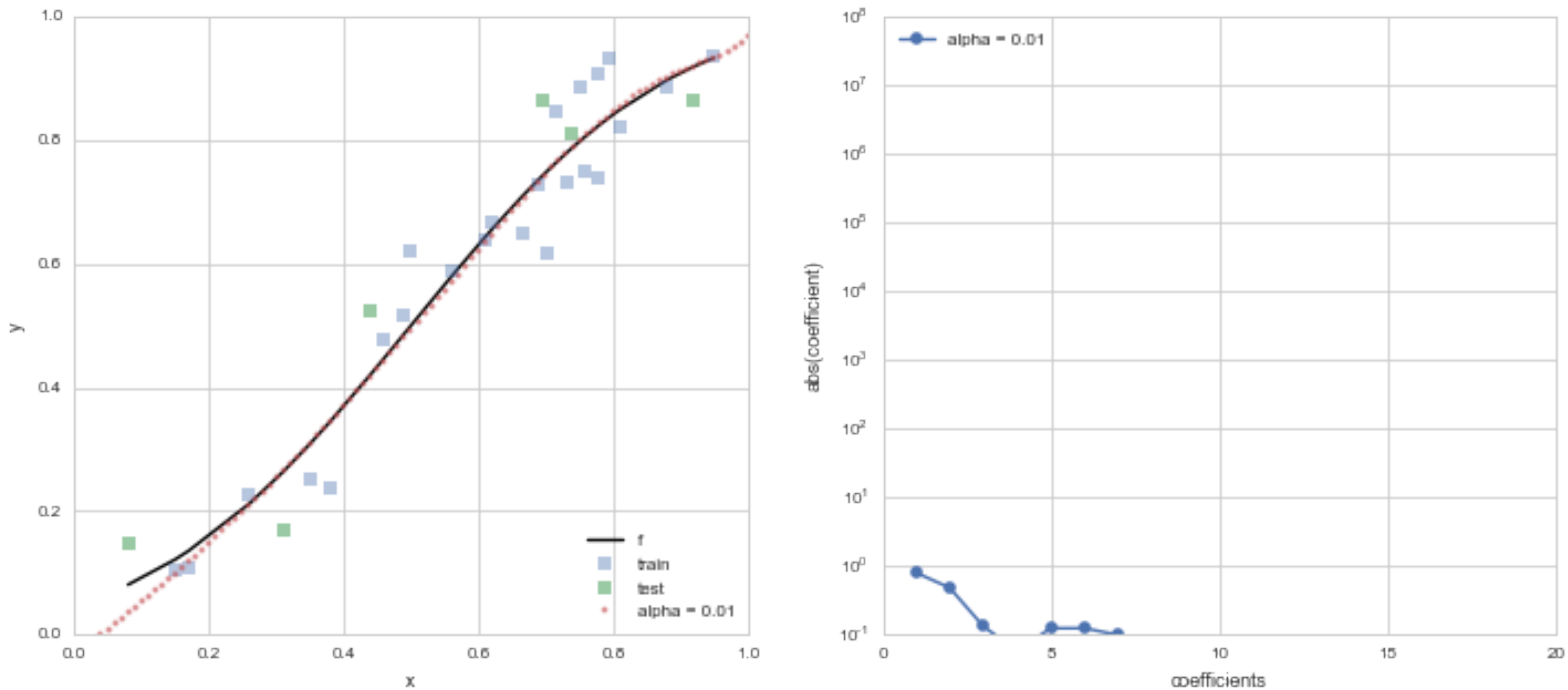
REGULARIZATION

$$\mathcal{R}(h_j) = \sum_{y_i \in \mathcal{D}} (y_i - h_j(x_i))^2 + \alpha \sum_{i=0}^{j} \theta_i^2 .$$

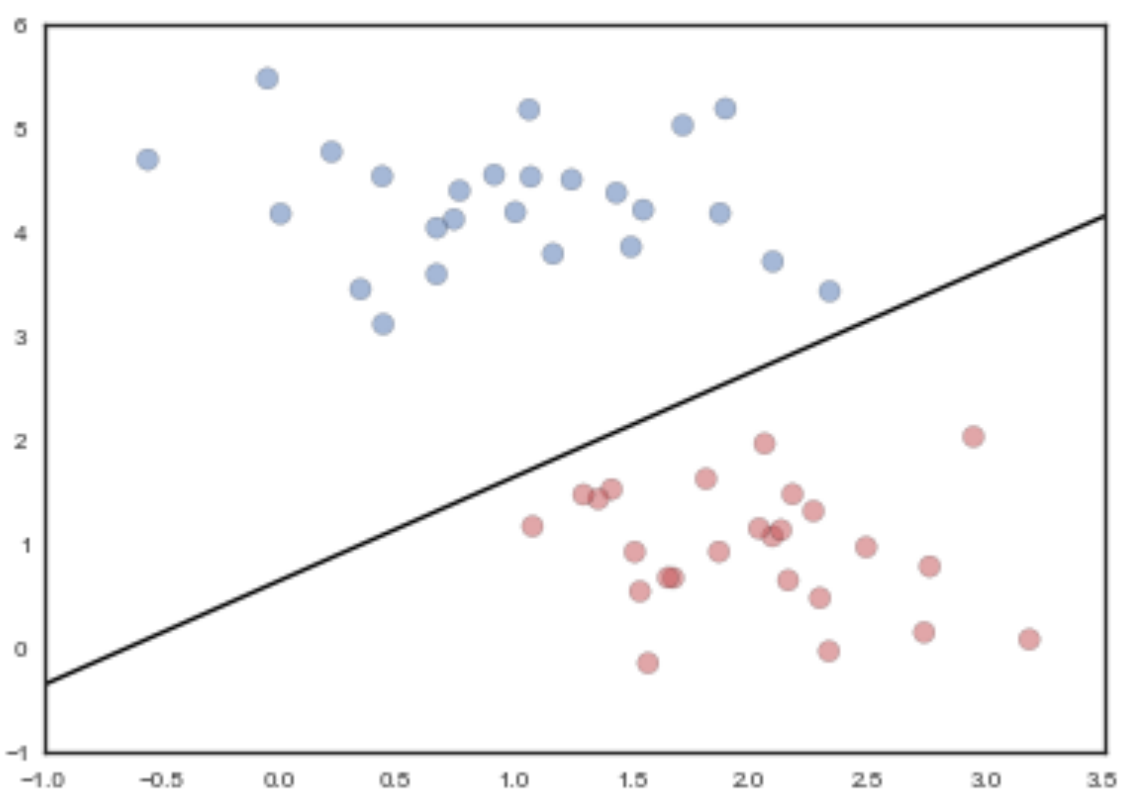
As we increase α, coefficients go towards 0.

Lasso uses $\alpha \sum_{i=0}^{j} |\theta_i|$, sets coefficients to exactly 0.

Regularization with Cross-Validation



CLASSIFICATION



- will a customer churn?
- is this a check? For how much?
- a man or a woman?
- will this customer buy?
- do you have cancer?
- is this spam?
- whose picture is this?
- what is this text about?[j]

---

[j] image from code in http://bit.ly/1Azg29G

MLE for Logistic Regression

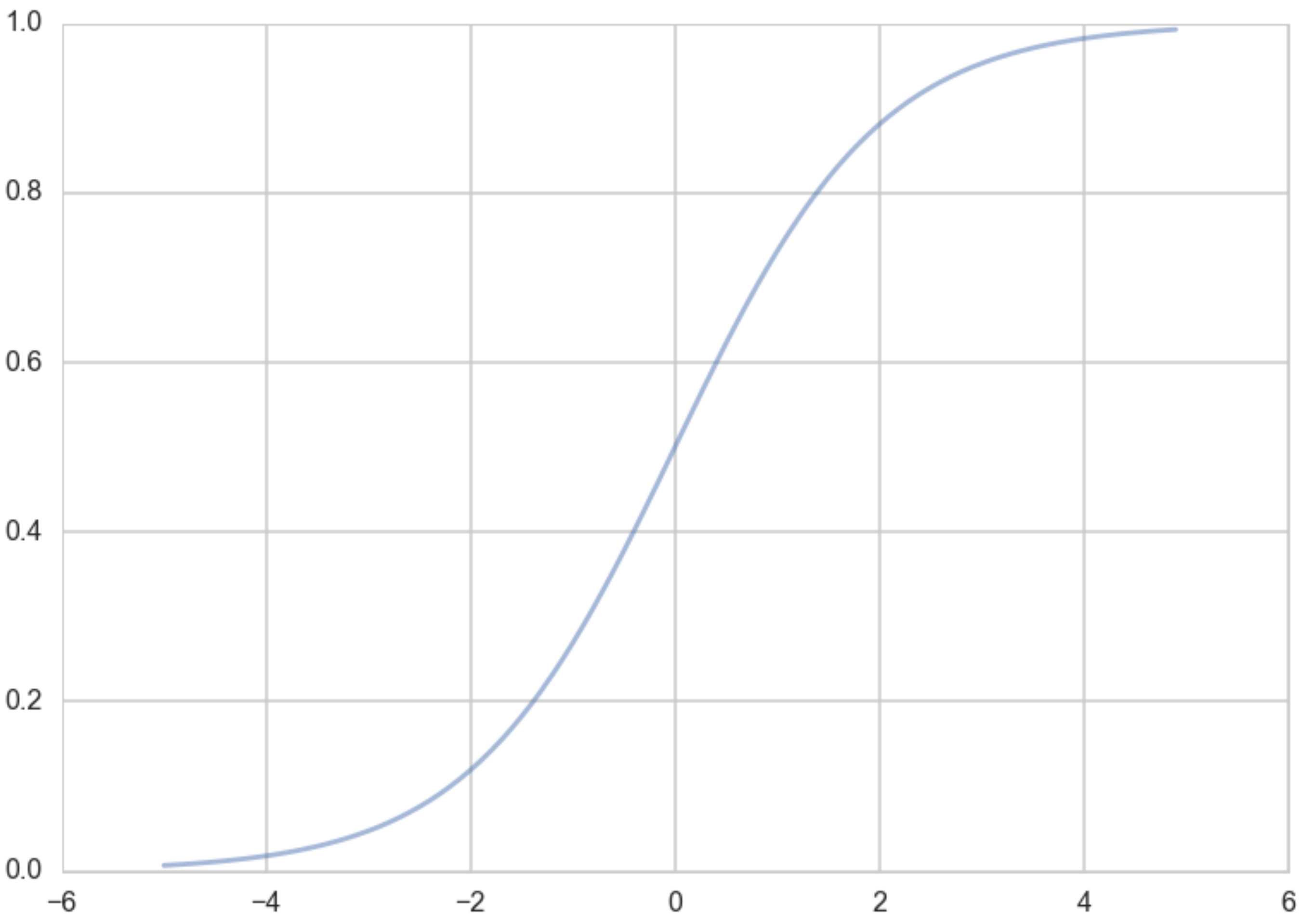
- example of a Generalized Linear Model (GLM)
- "Squeeze" linear regression through a **Sigmoid** function
- this bounds the output to be a probability
- What is the sampling Distribution?

Sigmoid function

This function is plotted below:

```
h = lambda z: 1./(1+np.exp(-z))
zs=np.arange(-5,5,0.1)
plt.plot(zs, h(zs), alpha=0.5);
```

Identify: $z = \mathbf{w} \cdot \mathbf{x}$ and $h(\mathbf{w} \cdot \mathbf{x})$ with the probability that the sample is a '1' ($y = 1$).



Then, the conditional probabilities of $y = 1$ or $y = 0$ given a particular sample's features $\mathbf{x}$ are:

$$P(y = 1|\mathbf{x}) = h(\mathbf{w} \cdot \mathbf{x})$$
$$P(y = 0|\mathbf{x}) = 1 - h(\mathbf{w} \cdot \mathbf{x}).$$

These two can be written together as

$$P(y|\mathbf{x}, \mathbf{w}) = h(\mathbf{w} \cdot \mathbf{x})^y (1 - h(\mathbf{w} \cdot \mathbf{x}))^{(1-y)}$$

BERNOULLI!!

Multiplying over the samples we get:

$$P(y|\mathbf{x},\mathbf{w}) = P(\{y_i\}|\{\mathbf{x}_i\},\mathbf{w}) = \prod_{y_i \in \mathcal{D}} P(y_i|\mathbf{x}_i,\mathbf{w}) = \prod_{y_i \in \mathcal{D}} h(\mathbf{w}\cdot\mathbf{x}_i)^{y_i}\left(1 - h(\mathbf{w}\cdot\mathbf{x}_i)\right)^{(1-y_i)}$$

A noisy y is to imagine that our data $\mathcal{D}$ was generated from a joint probability distribution $P(x,y)$. Thus we need to model y at a given x, written as $P(y \mid x)$, and since $P(x)$ is also a probability distribution, we have:

$$P(x,y) = P(y \mid x)P(x),$$

Indeed its important to realize that a particular sample can be thought of as a draw from some "true" probability distribution.

maximum likelihood estimation maximises the **likelihood of the sample y**,

$$\mathcal{L} = P(y \mid \mathbf{x}, \mathbf{w}).$$

Again, we can equivalently maximize

$$\ell = log(P(y \mid \mathbf{x}, \mathbf{w}))$$

Thus

$$
\begin{aligned}
\ell &= log\left(\prod_{y_i \in \mathcal{D}} h(\mathbf{w}\cdot\mathbf{x}_i)^{y_i}(1-h(\mathbf{w}\cdot\mathbf{x}_i))^{(1-y_i)}\right)\\
&= \sum_{y_i \in \mathcal{D}} log\left(h(\mathbf{w}\cdot\mathbf{x}_i)^{y_i}(1-h(\mathbf{w}\cdot\mathbf{x}_i))^{(1-y_i)}\right)\\
&= \sum_{y_i \in \mathcal{D}} log\, h(\mathbf{w}\cdot\mathbf{x}_i)^{y_i} + log\,(1-h(\mathbf{w}\cdot\mathbf{x}_i))^{(1-y_i)}\\
&= \sum_{y_i \in \mathcal{D}} (y_i log(h(\mathbf{w}\cdot\mathbf{x})) + (1-y_i)log(1-h(\mathbf{w}\cdot\mathbf{x})))
\end{aligned}
$$

Use Convex optimization! (soon, hw)