

Lecture 5

Regression Modelling And Information Theory

Last Time:

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

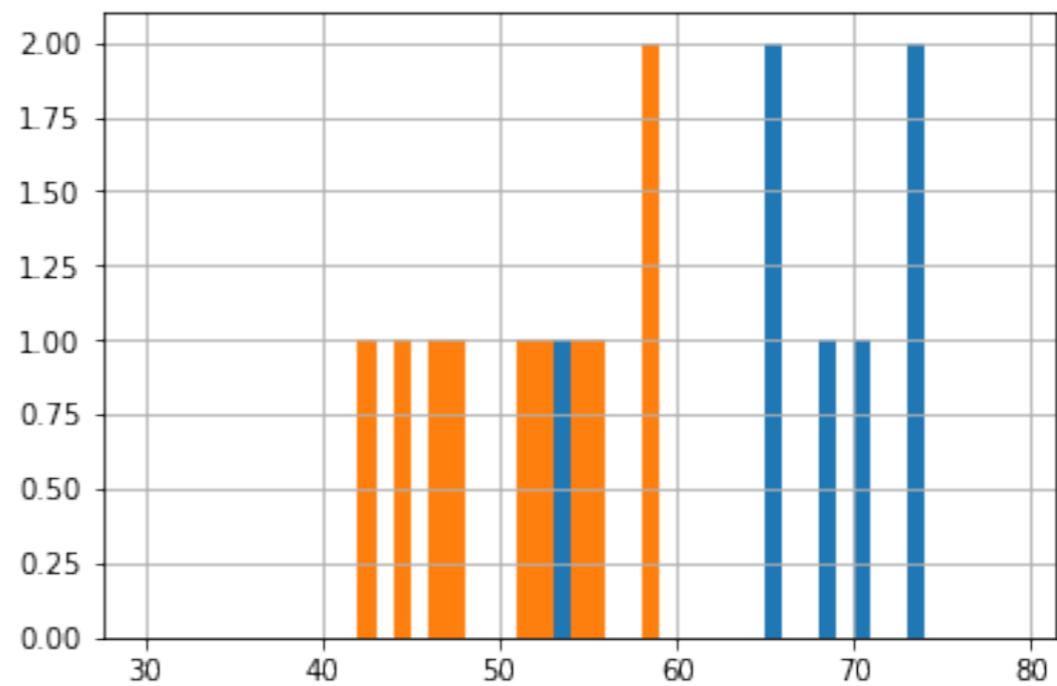
Today

- More on significance
- Test Sets
- Validation and X-validation
- Regularization
- The KL Divergence and Deviance
- In-sample penalties: the AIC

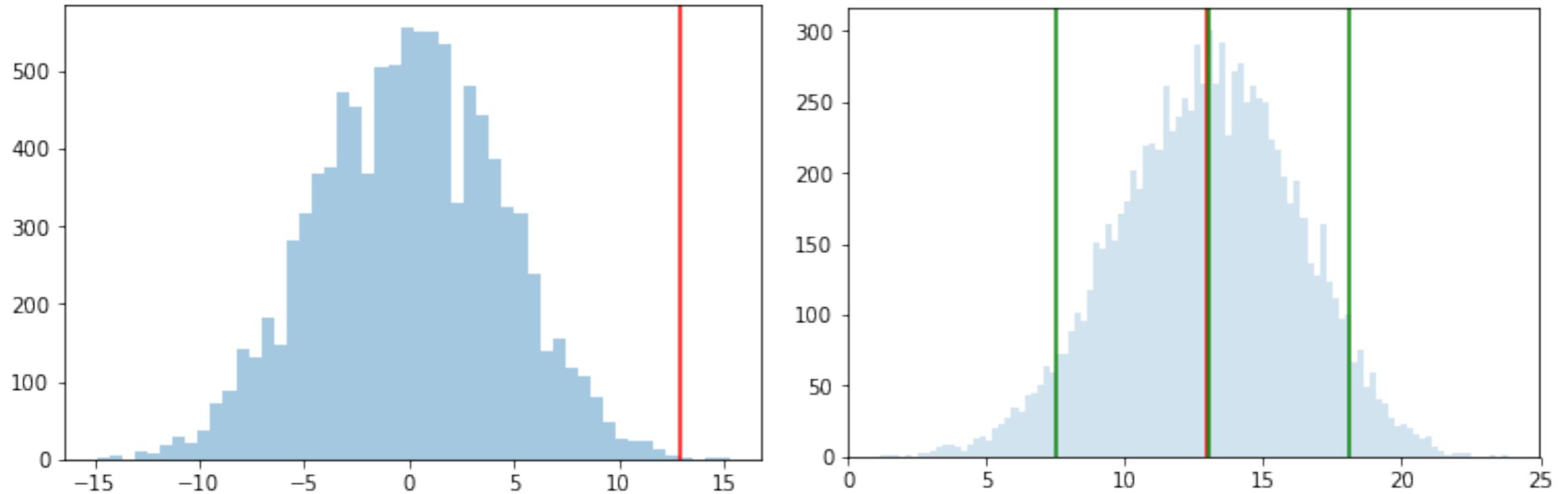
dosage	label
0	P
1	P
2	P
3	P
4	P
5	P
6	P
7	P
8	P
9	P
10	D
11	D
12	D
13	D
14	D
15	D
16	D
17	D
18	D

Dose vs Placebo

Actual mean effect is about 13.



Significance vs Size of Effect



Left, permute all labels. Right, sample with replacement within groups.

HYPOTHESIS SPACES

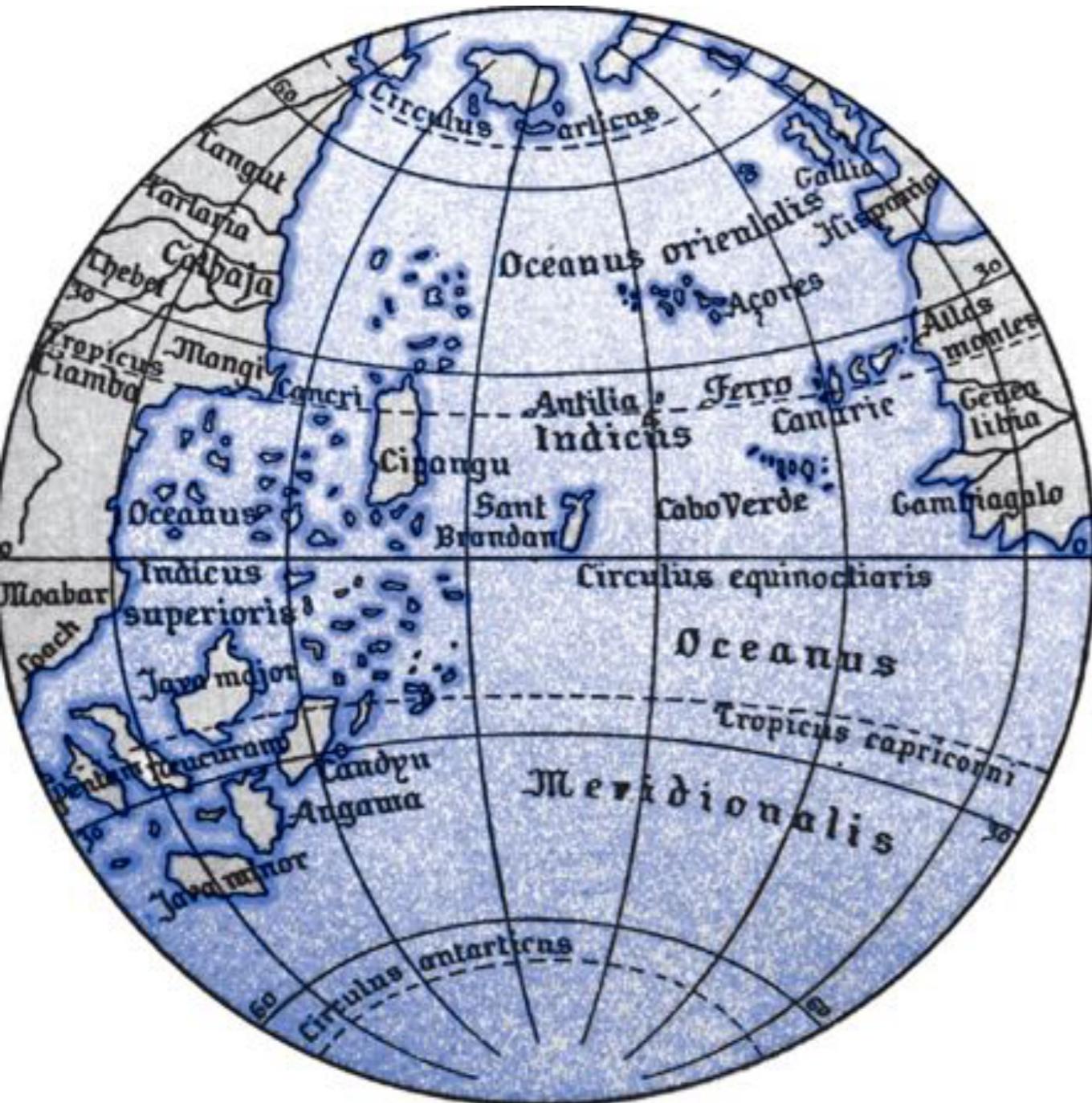
A polynomial looks so:

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

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

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

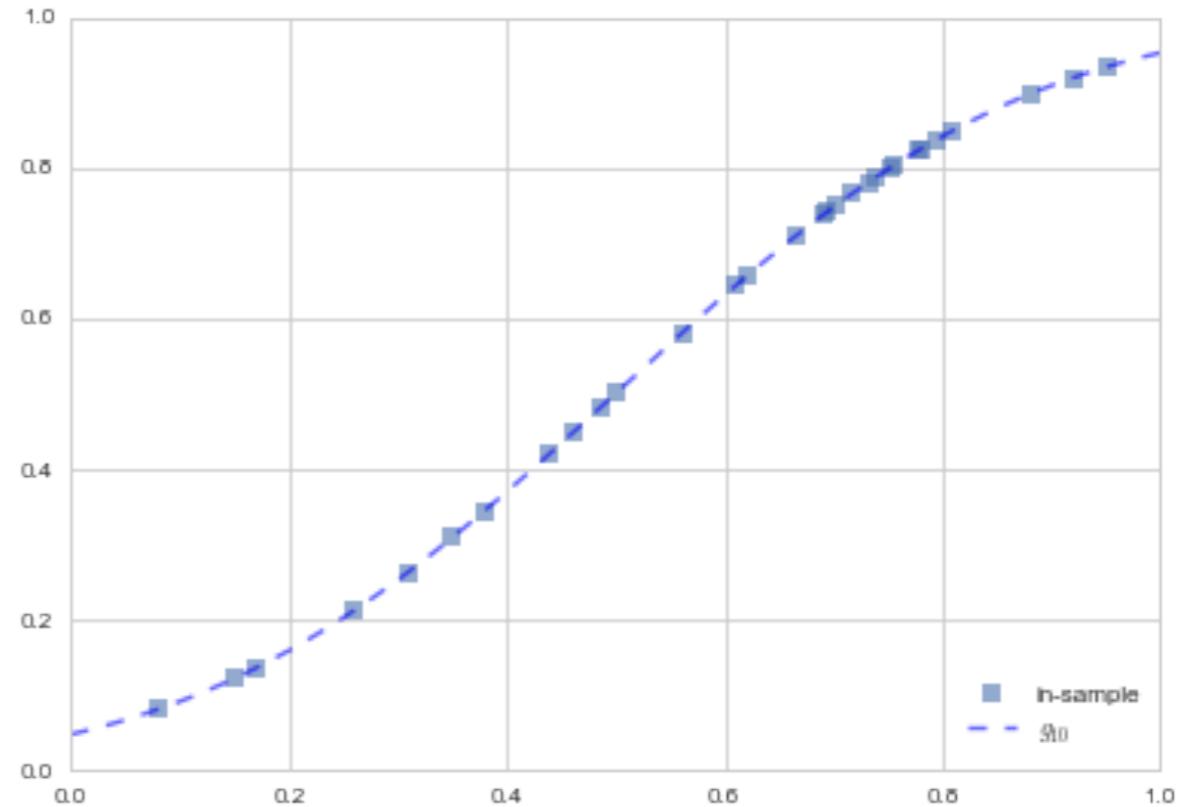
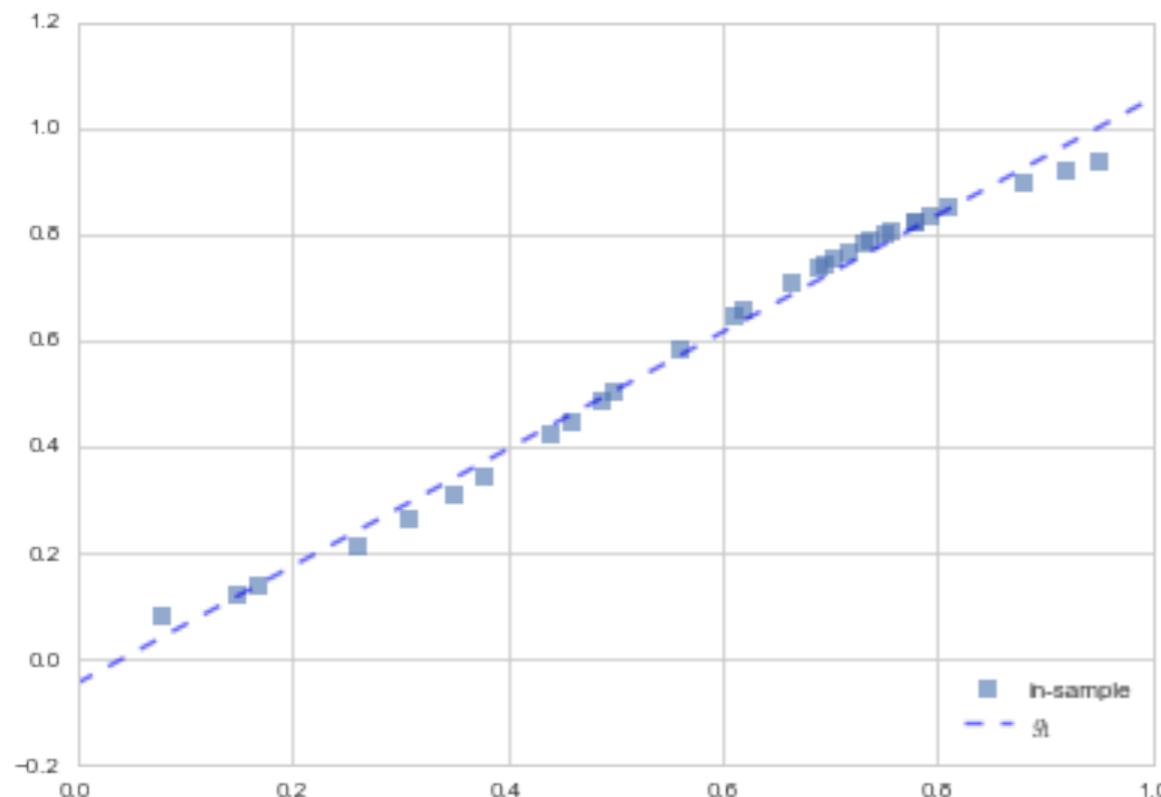
SMALL World vs BIG World



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

Without Noise...

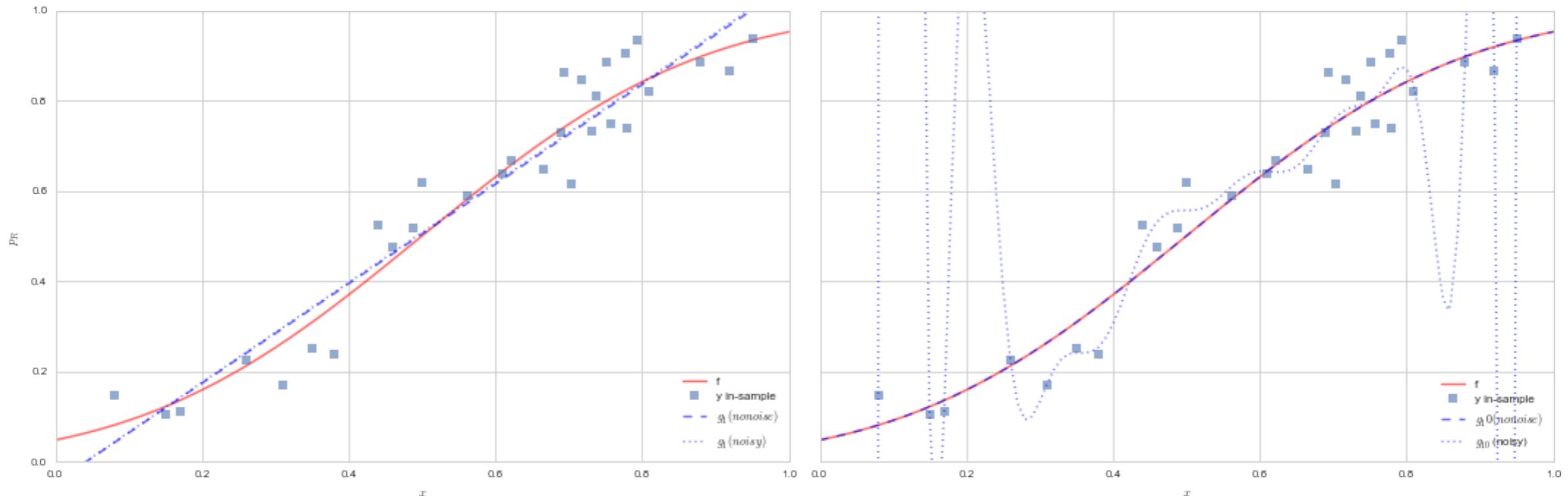
30 points of data. Which fit is better? Line in \mathcal{H}_1 or curve in \mathcal{H}_{20} ?



THE REAL WORLD HAS NOISE

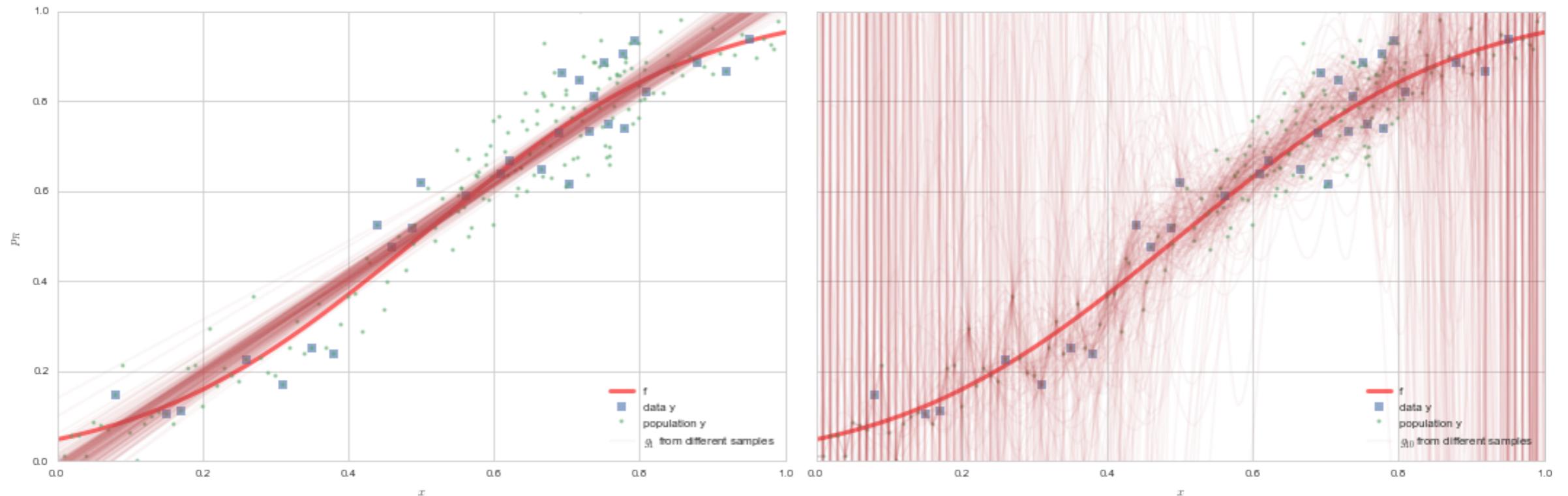
Which fit is better now?

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:

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

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

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

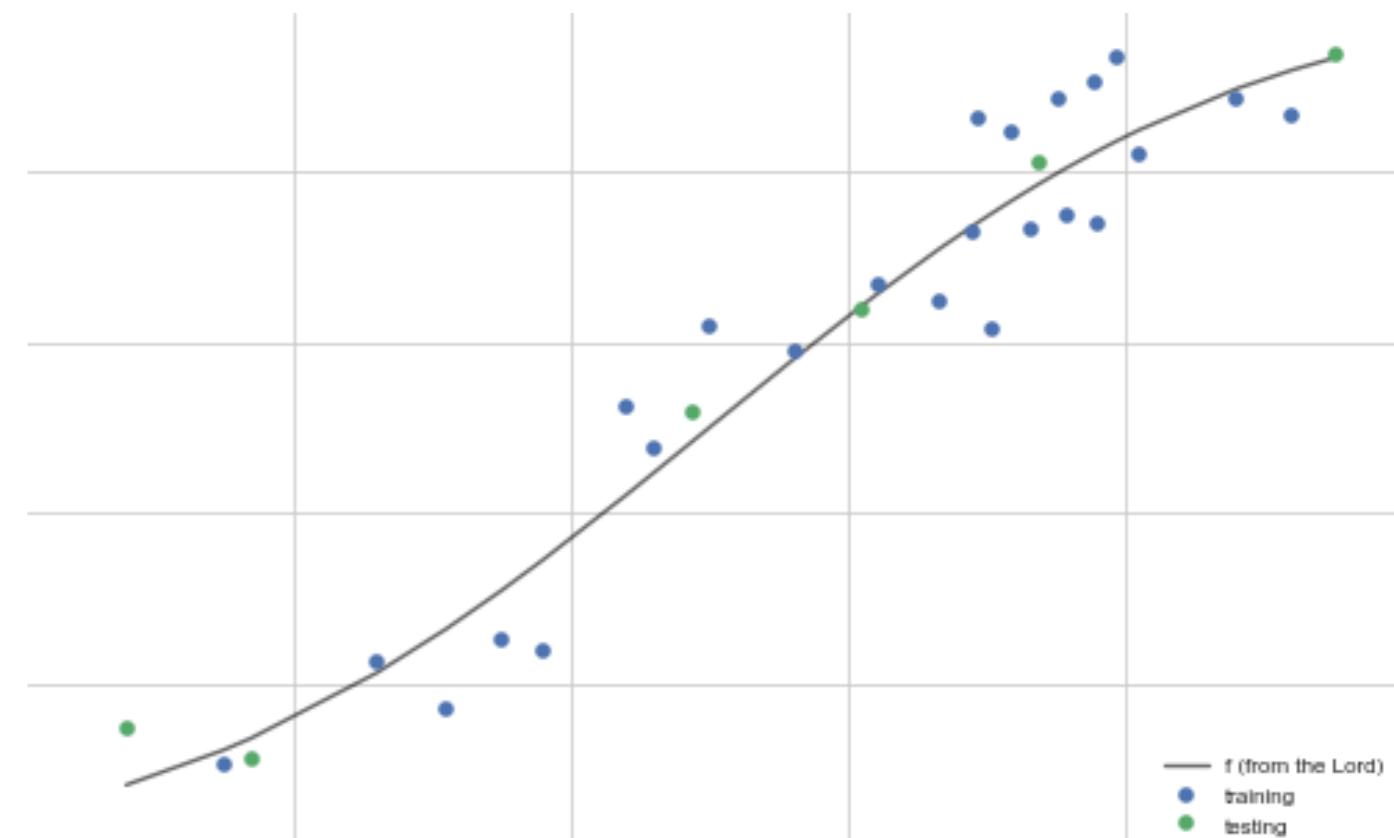
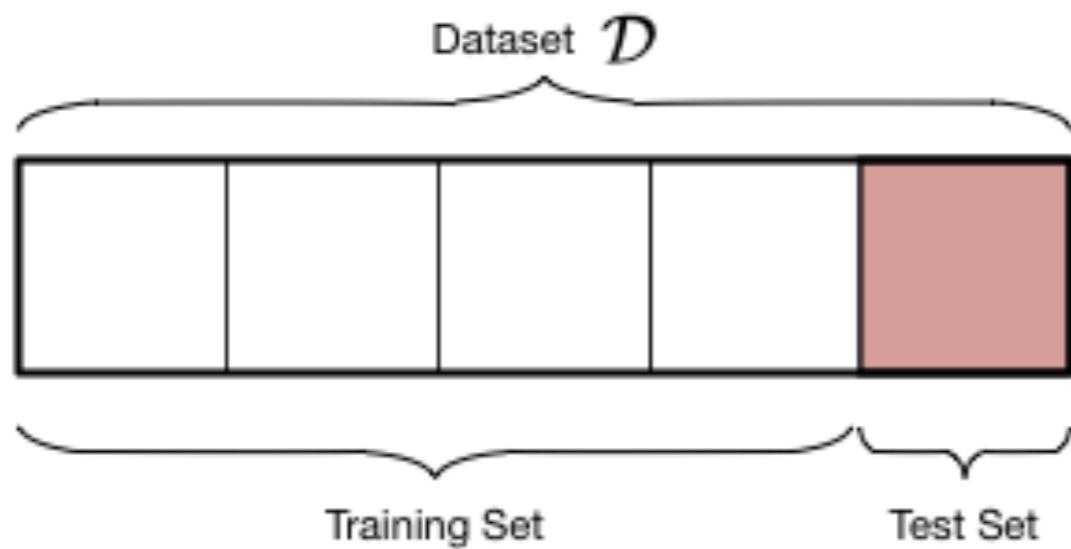
$$\begin{aligned}
 &= EdEp(gd^2 + f^2 + e^2 - 2gd*f + 2f*e - 2e*gd) \\
 &= EdEp(gd^2 + f^2 - 2gd*f) + \sigma^2 \\
 &=
 \end{aligned}$$

Then,

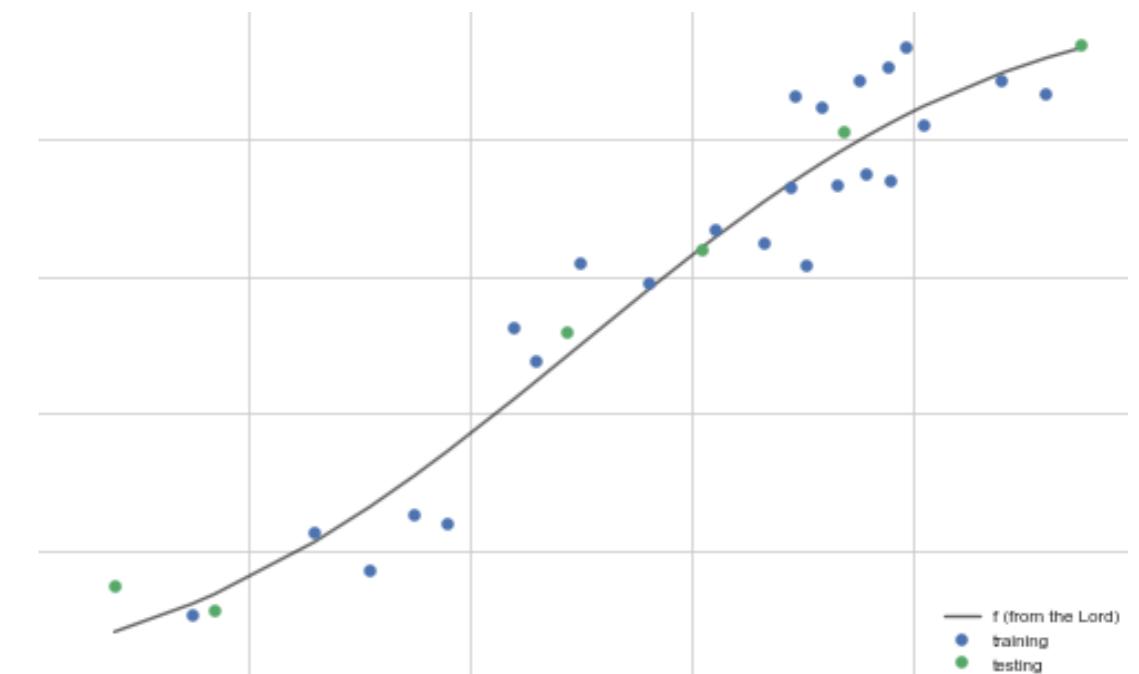
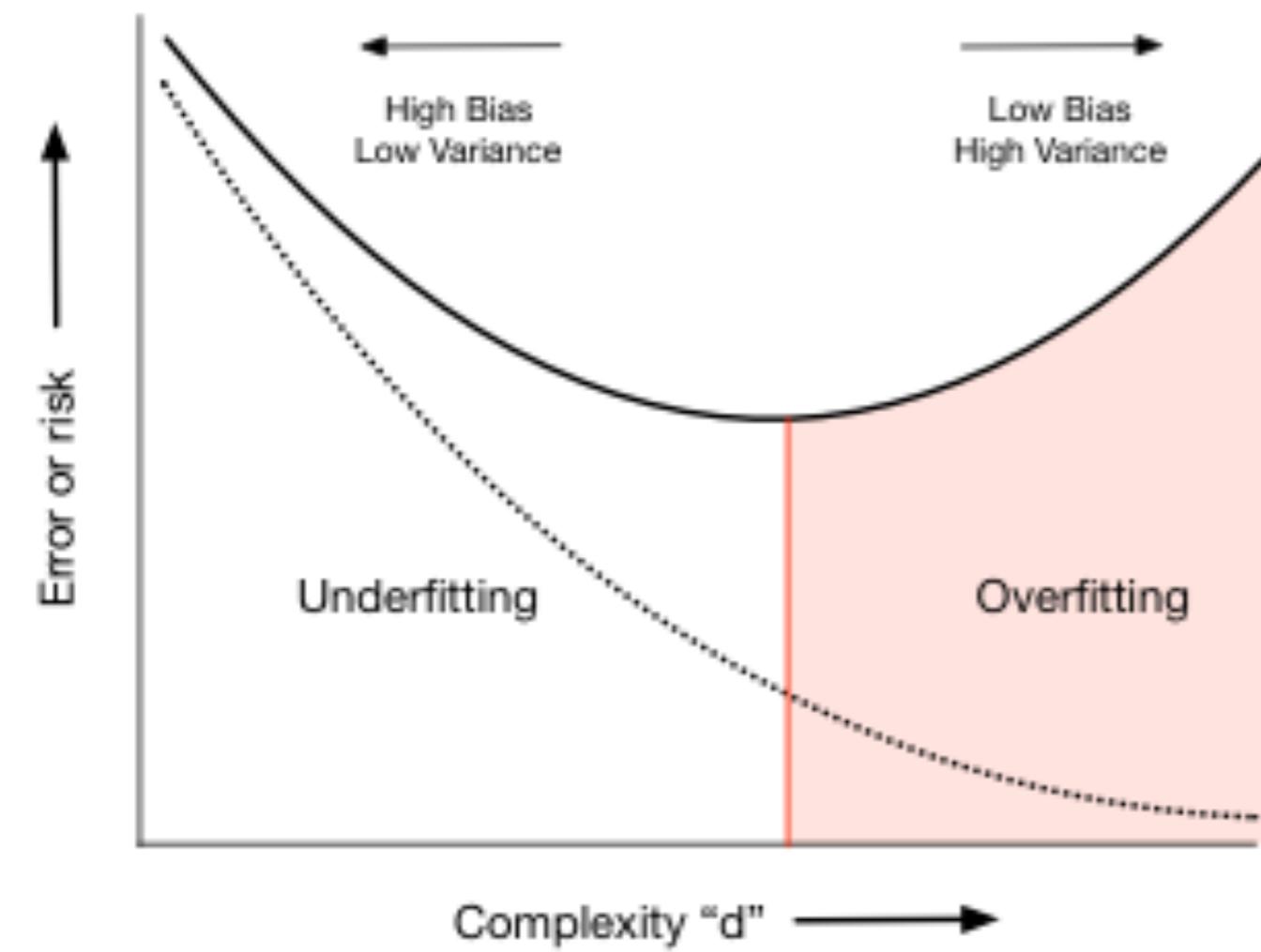
This is the bias variance decomposition for regression.

- 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.

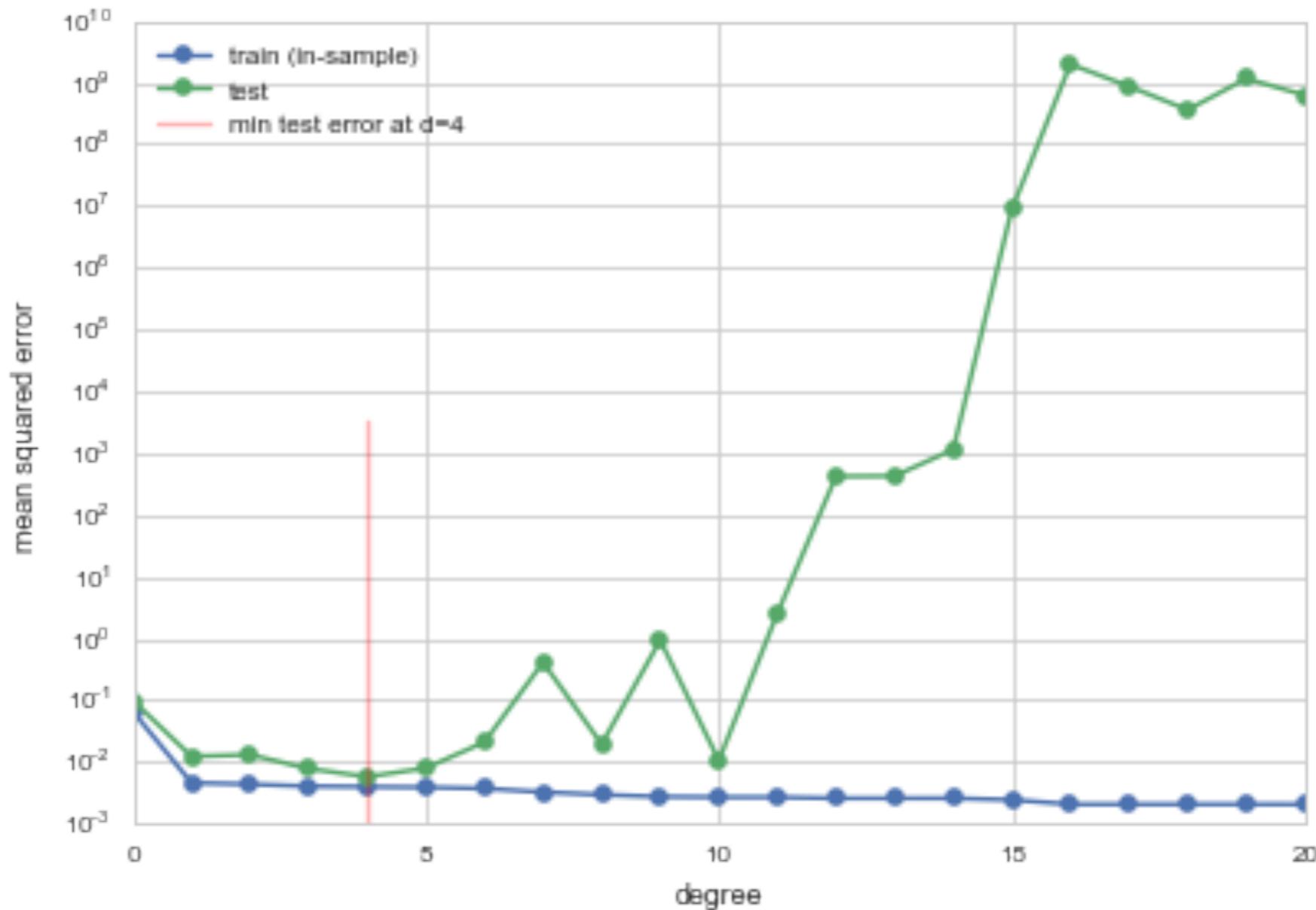
TRAIN AND TEST



MODEL COMPARISON: A Large World approach



We "fit" for d



Do we still have 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.**

Hoeffding's inequality

population fraction μ , sample drawn with replacement, fraction ν :

$$P(|\nu - \mu| > \epsilon) \leq 2e^{-2\epsilon^2 N}$$

For hypothesis h , identify 1 with $h(x_i) \neq f(x_i)$ at sample x_i . Then μ, ν are population/sample error rates. Then,

$$P(|R_{in}(h) - R_{out}(h)| > \epsilon) \leq 2e^{-2\epsilon^2 N}$$

Interpretation: We need large N to have small errors.

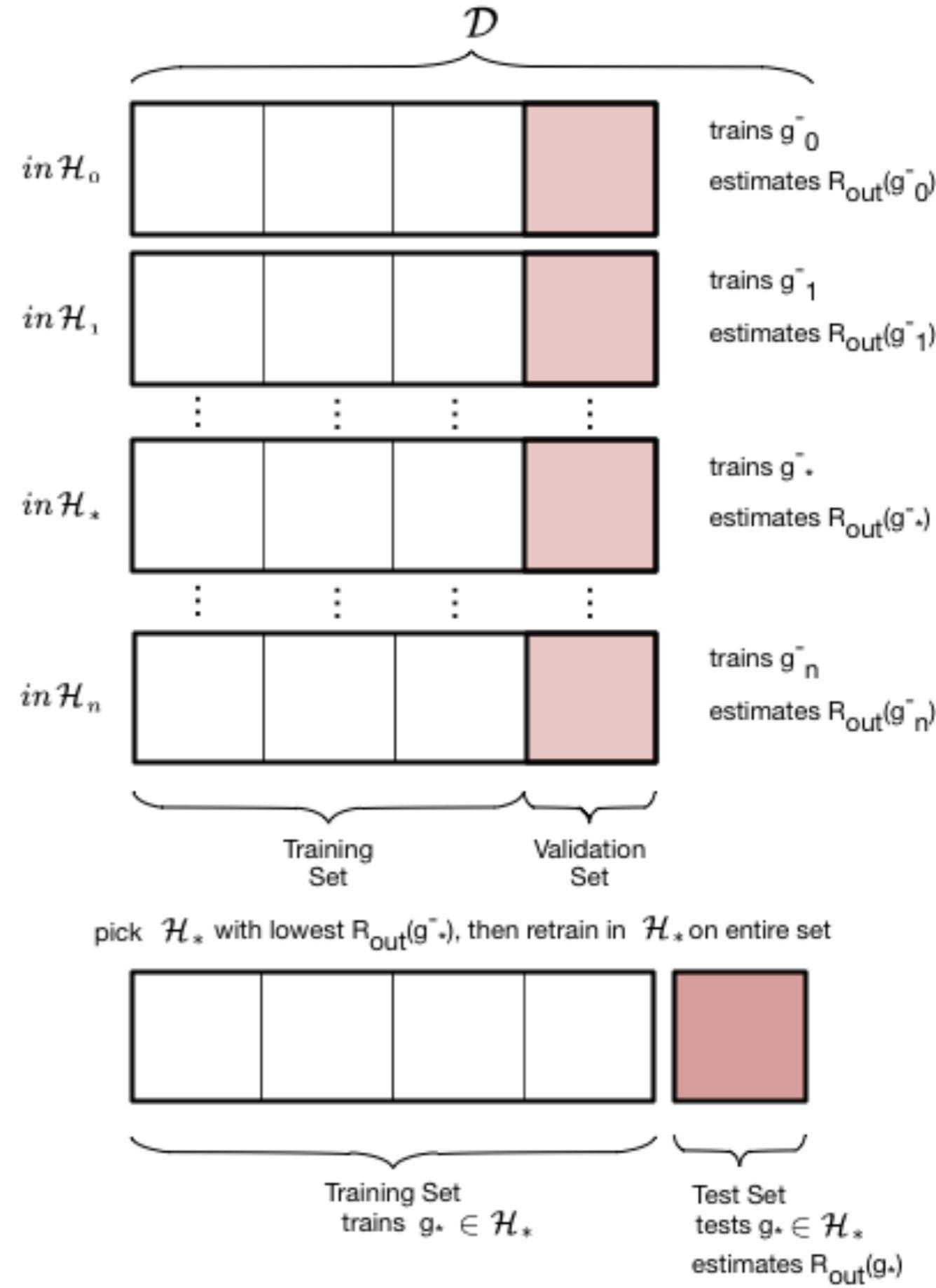
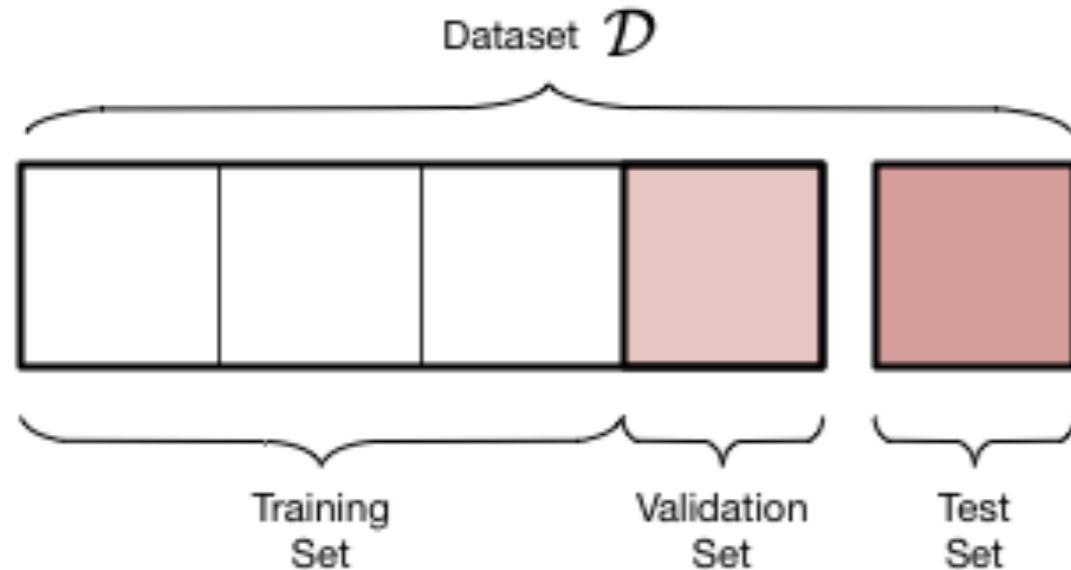
- Hoeffding inequality holds ONCE we have picked a hypothesis h , as we need it to label the 1 and 0s.
- But over the training set we one by one pick all the models in the hypothesis space
- best fit g is among the h in \mathcal{H} , g must be h_1 OR h_2
OR....Say **effectively** M such choices:

$M = \# \text{ hypotheses we are testing/comparing with}$

$$P(|R_{in}(g) - R_{out}(g)| \geq \epsilon) \leq \sum_{h_i \in \mathcal{H}} P(|R_{in}(h_i) - R_{out}(h_i)| \geq \epsilon) \leq 2M e^{-2\epsilon^2 N}$$

VALIDATION

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



usually we want to fit a hyperparameter

- we **wrongly** already attempted to fit d on our previous test set.
- choose the d, g^* combination with the lowest validation set risk.
- $R_{val}(g^{-*}, d^*)$ has an optimistic bias since d effectively fit on validation set
- its Hoeffding bound must now take into account the grid-size as the effective size of the hypothesis

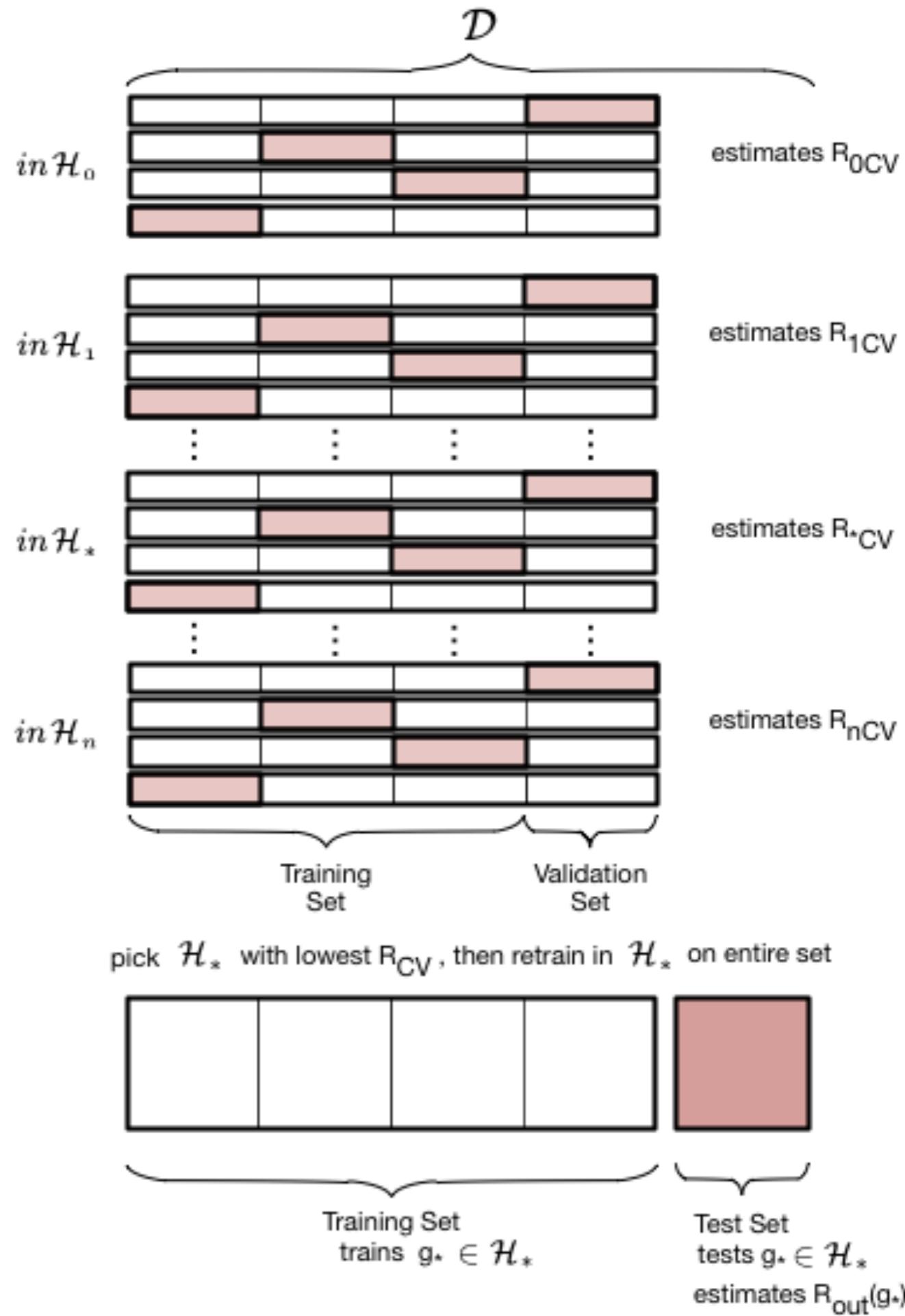
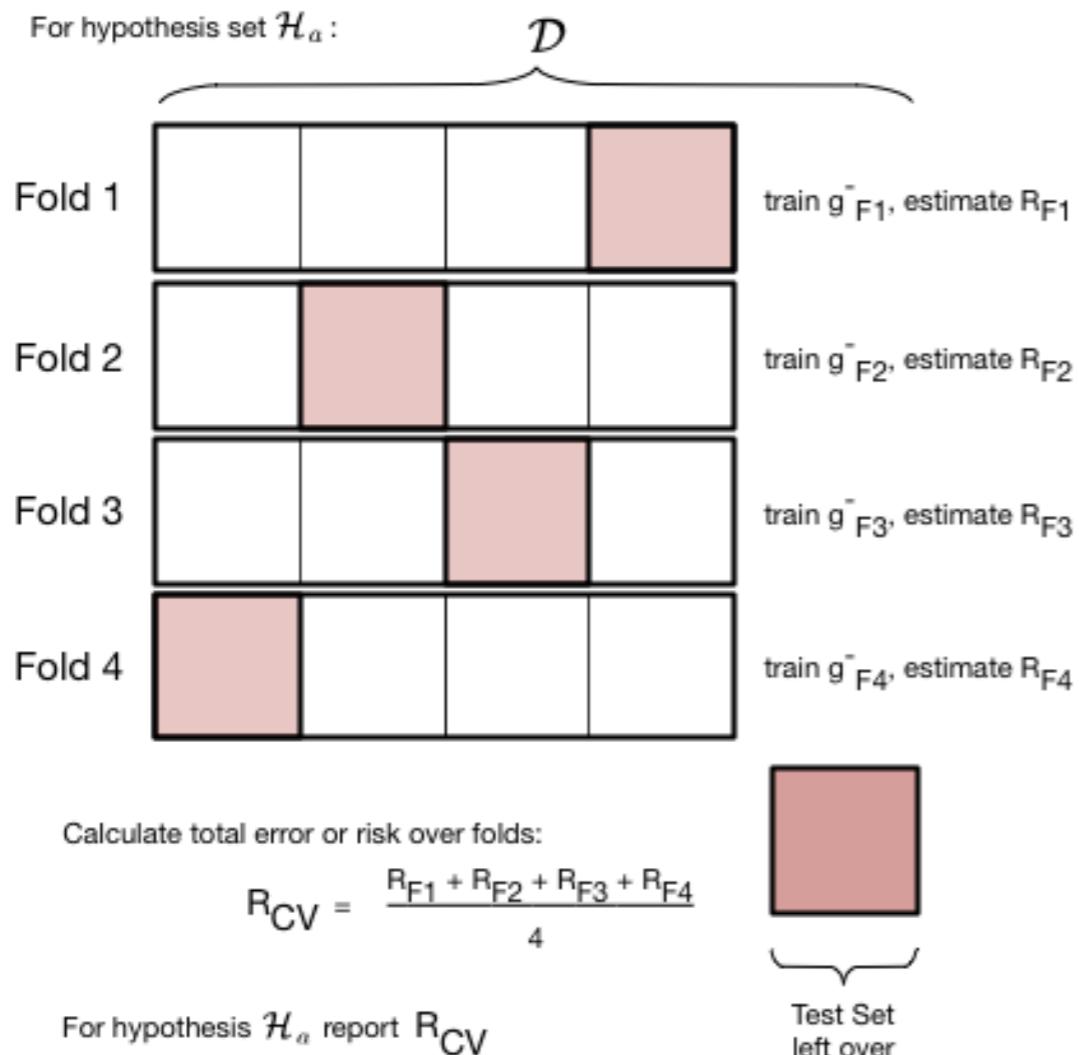
- this size from hyperparameters is typically a smaller size than that from parameters.

Retrain on entire set!

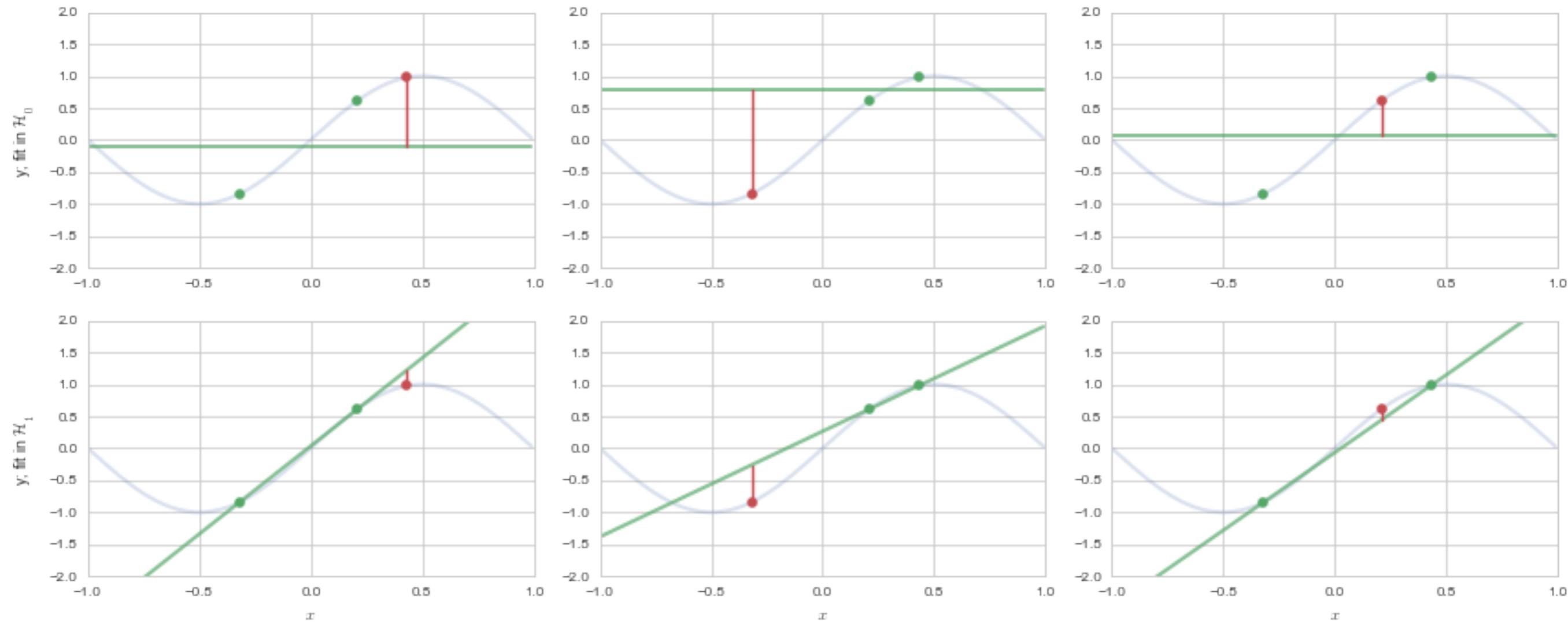
- finally retrain on the entire train+validation set using the appropriate (g^{-*}, d^*) combination.
- works as training for a given hypothesis space with more data typically reduces the risk even further.
- test set has a M of 1!

CROSS-VALIDATION

* Shuffle the data before data chunk splits



"leave-one-out validation"



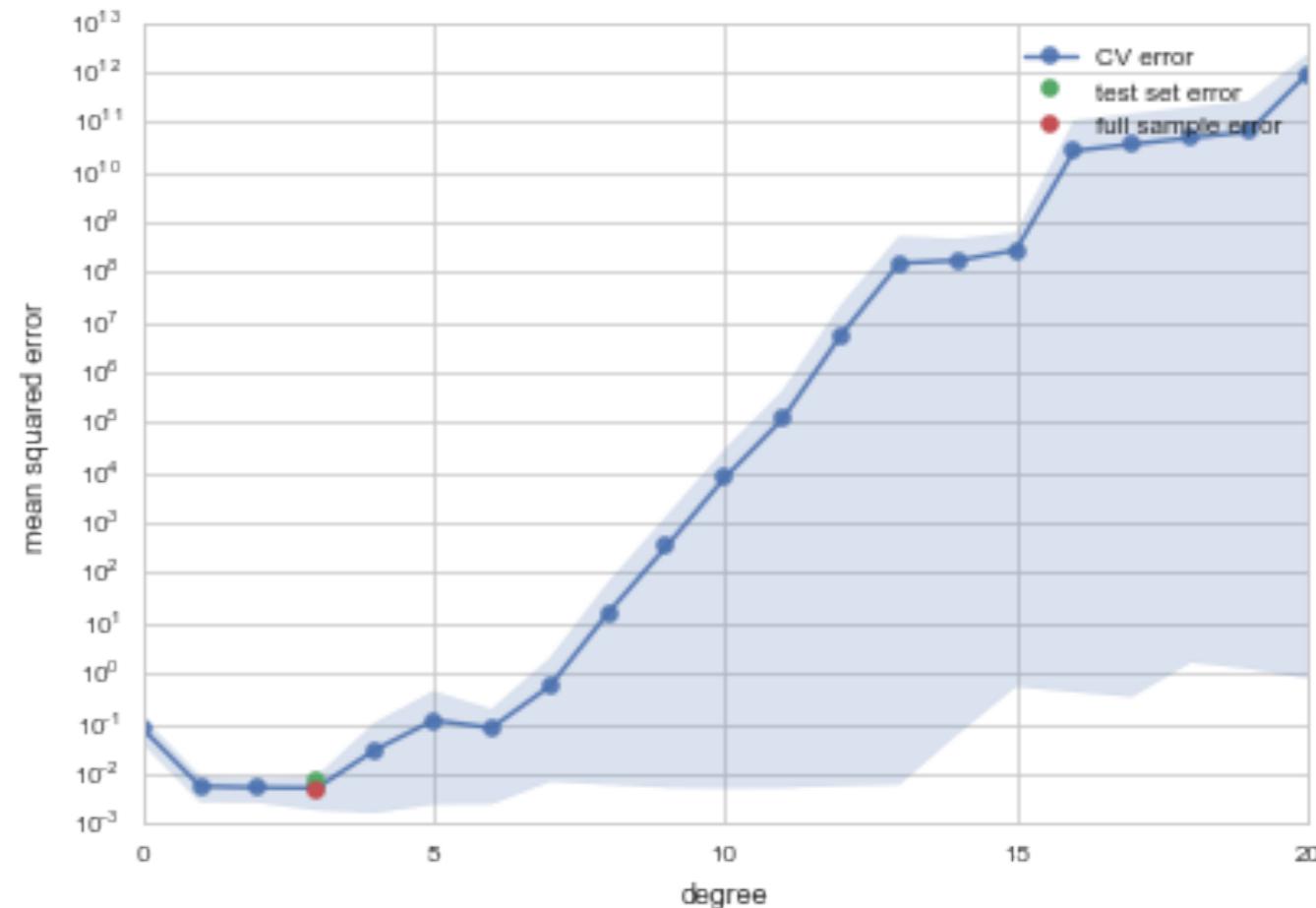
CROSS-VALIDATION

is

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

Here we find $d = 3$.

****Compare****
Random forest uses bootstrapped datasets
for model validation.



Cross Validation considerations

out-of-sample residuals

- 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

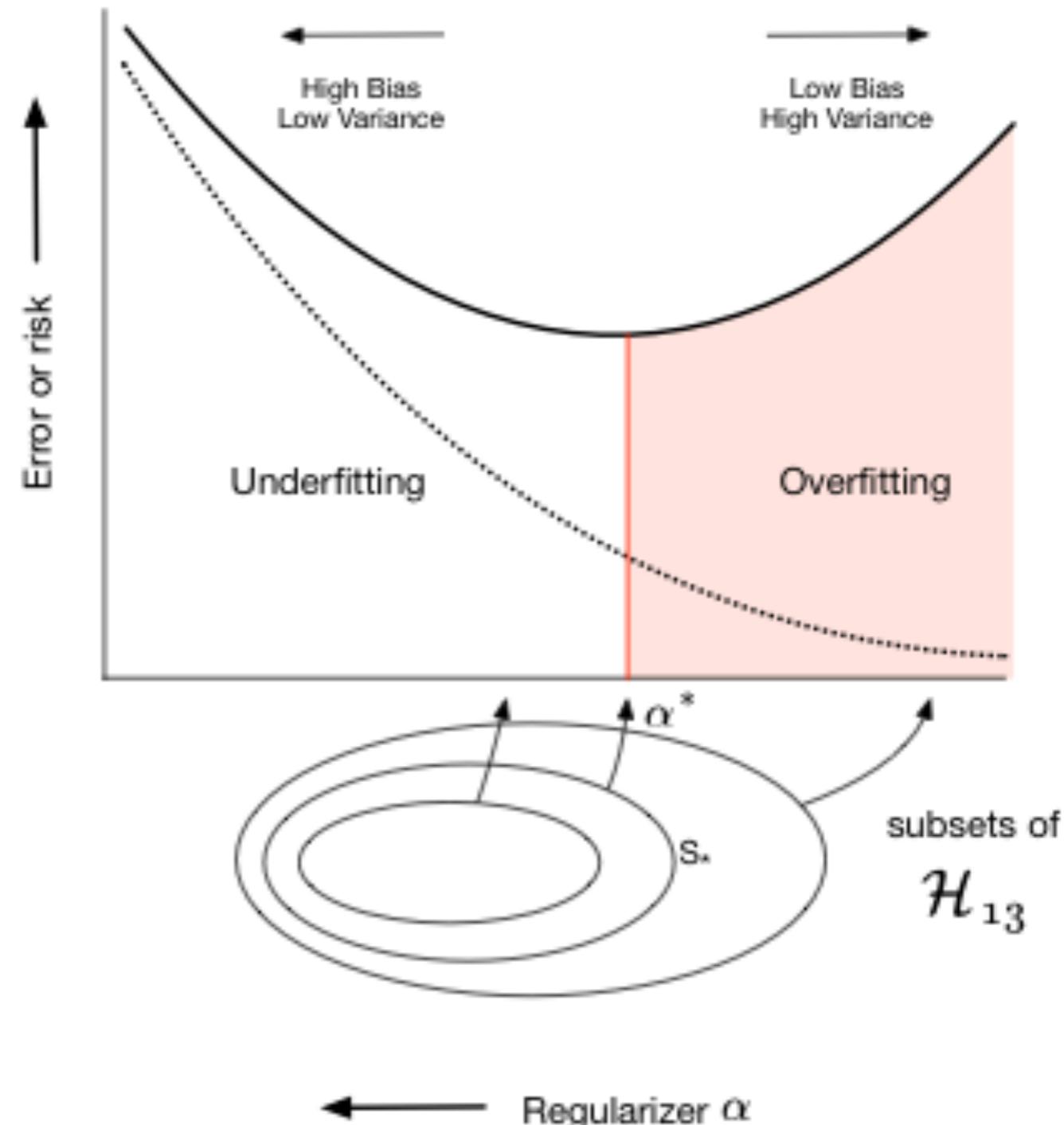
Keep higher a-priori complexity and impose a

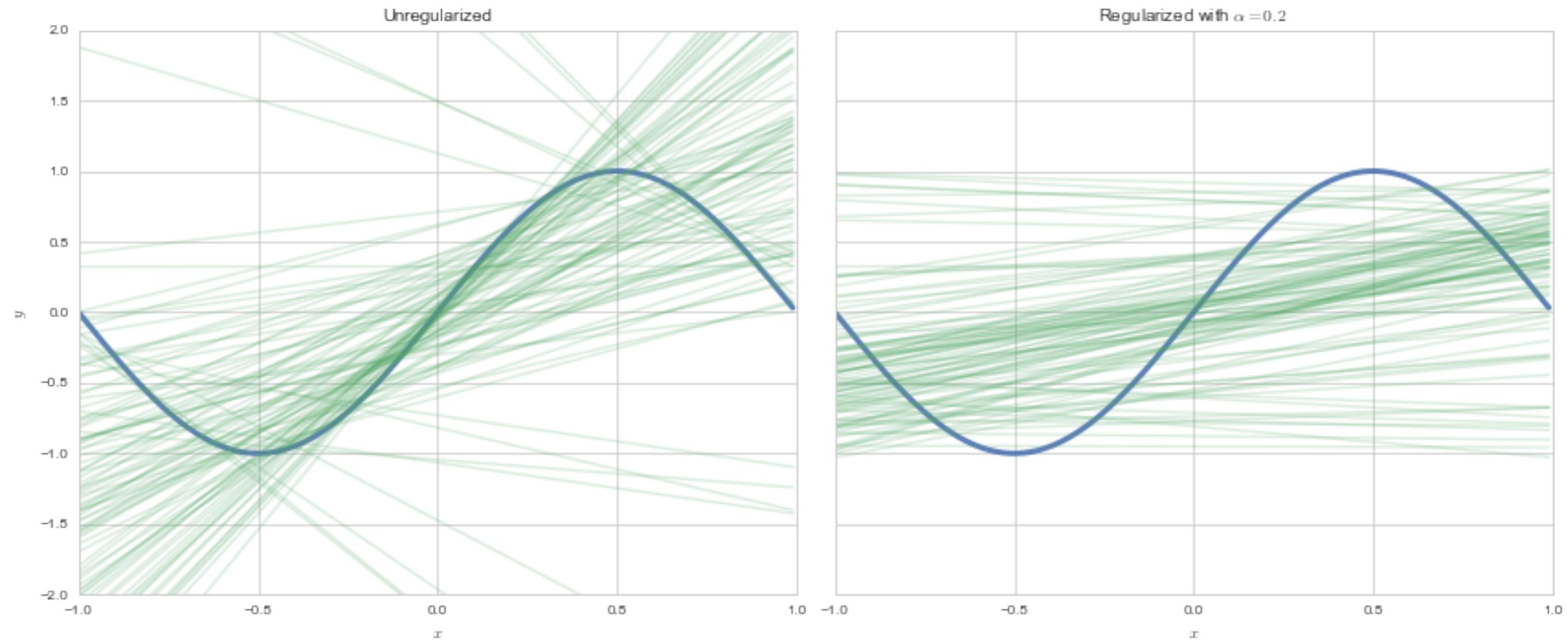
complexity penalty

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

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

Ridge



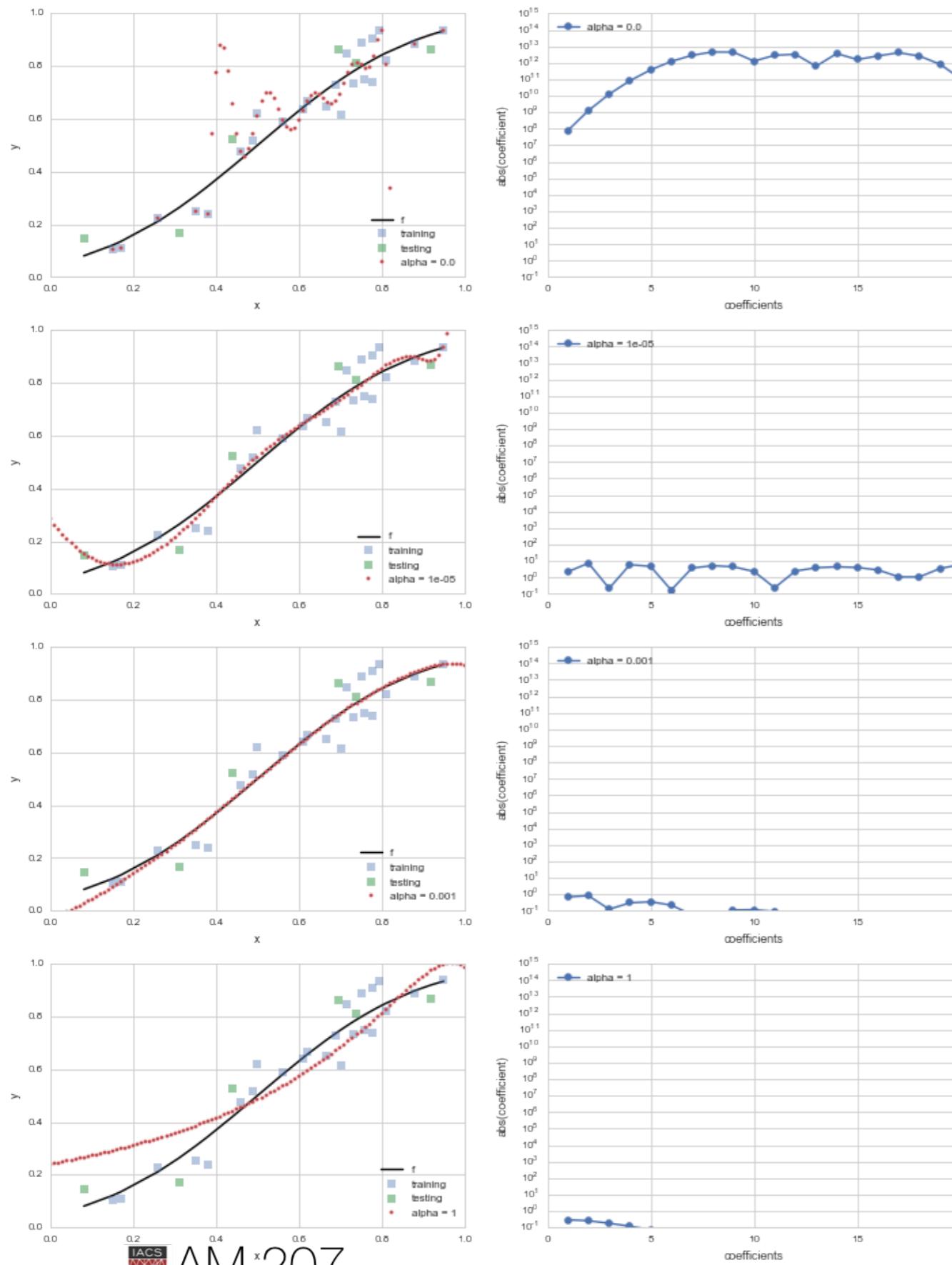


REGULARIZATION

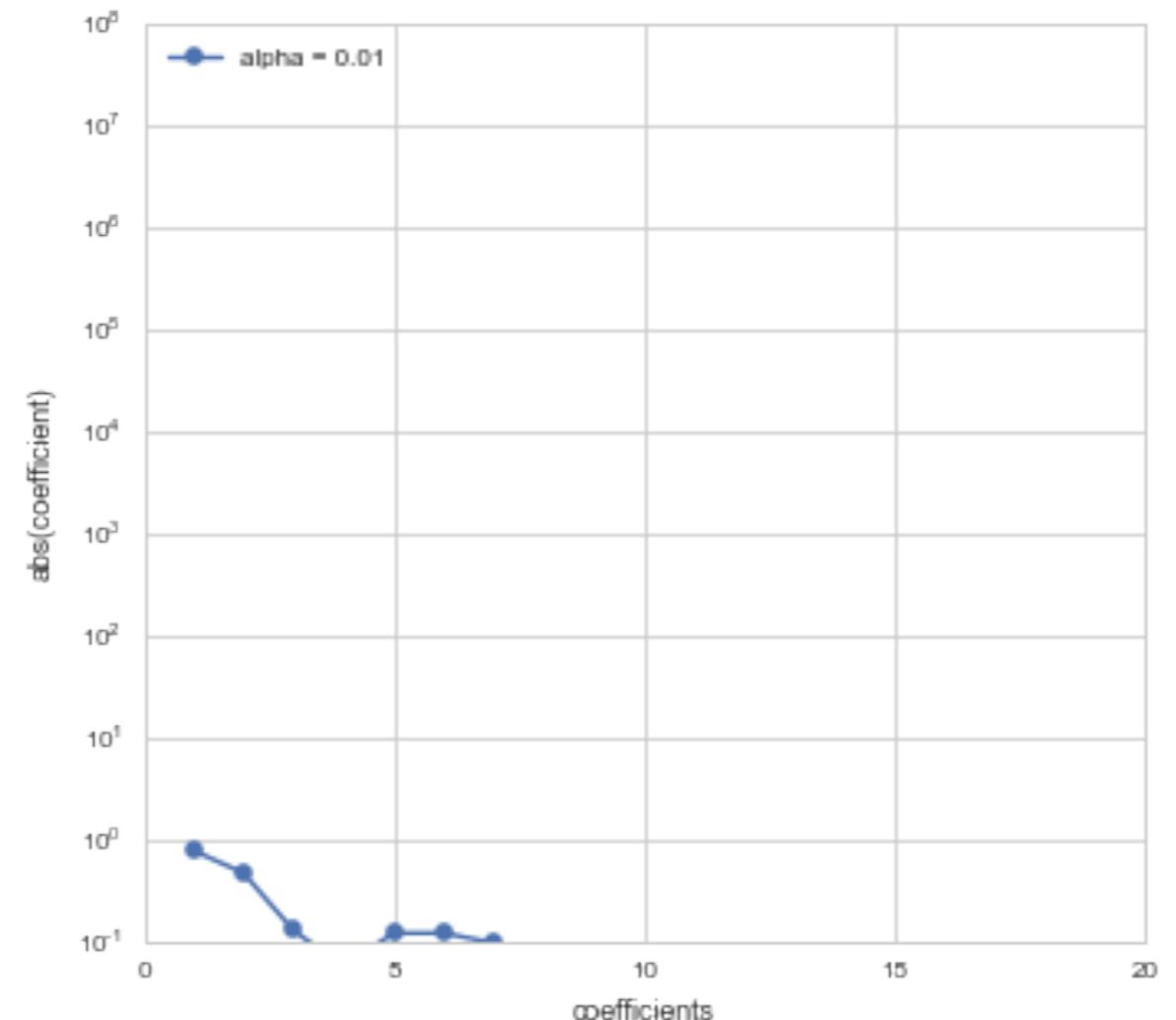
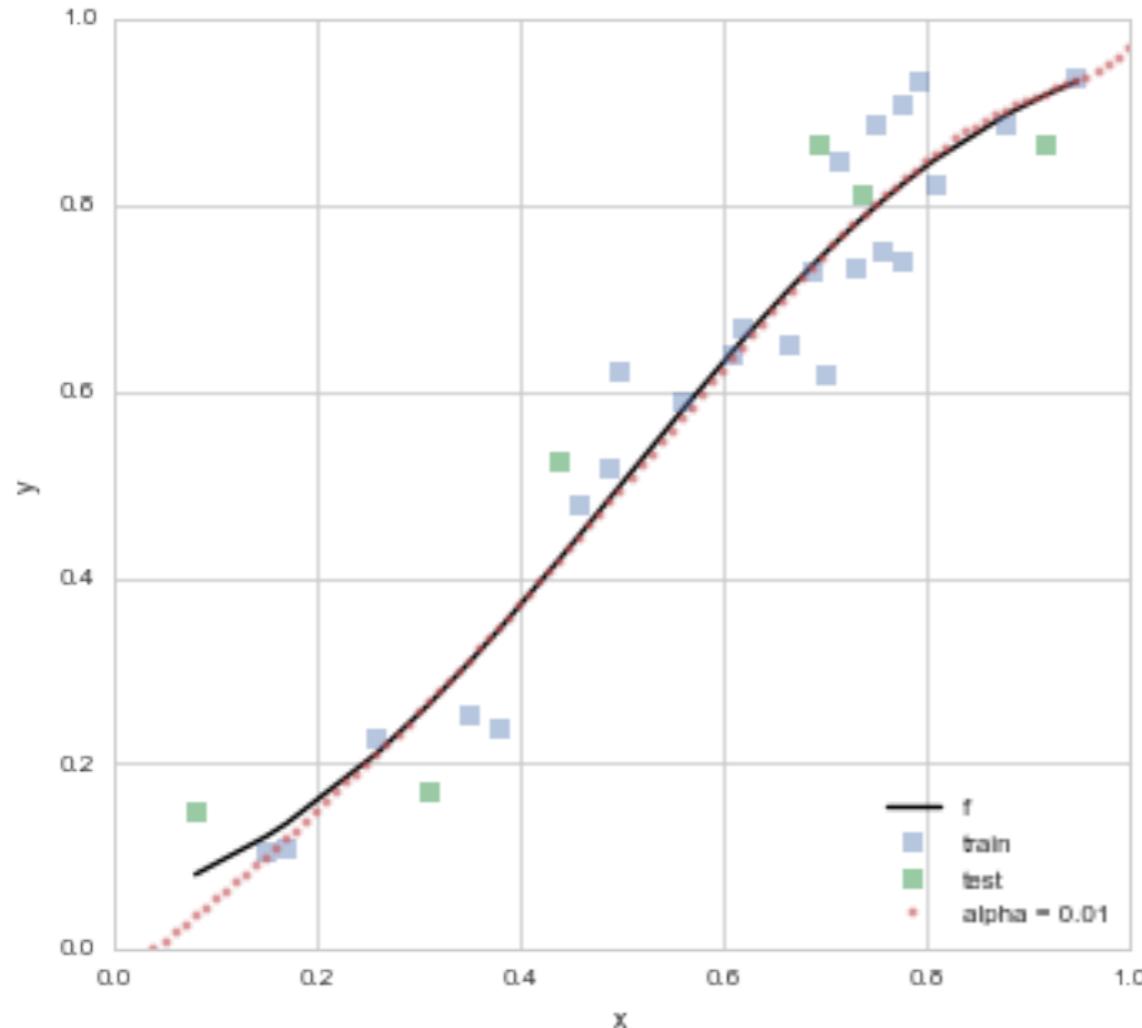
$$\mathcal{R}(h_j) = \sum_{y_i \in \mathcal{D}} (y_i - h_j(x_i))^2 + \alpha \sum_{i=0}^j \theta_i^2.$$

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



MODEL COMPARISON: In-sample estimation

- Suppose we have a large-world subset of nested models.
- .. thus the models have the same likelihood form
- would be nice to not have to spend data on validation sets
- and exploit the notion that a negative log likelihood is a loss
- we could use strength of effects
- but not really needed for prediction

KL-Divergence

$$\begin{aligned} D_{KL}(p, q) &= E_p[\log(p) - \log(q)] = E_p[\log(p/q)] \\ &= \sum_i p_i \log\left(\frac{p_i}{q_i}\right) \text{ or } \int dP \log\left(\frac{p}{q}\right) \end{aligned}$$

$$D_{KL}(p, p) = 0$$

KL divergence measures distance/dissimilarity of the two distributions $p(x)$ and $q(x)$.

Divergence:
*The additional uncertainty
induced by using probabilities
from one distribution to
describe another distribution*

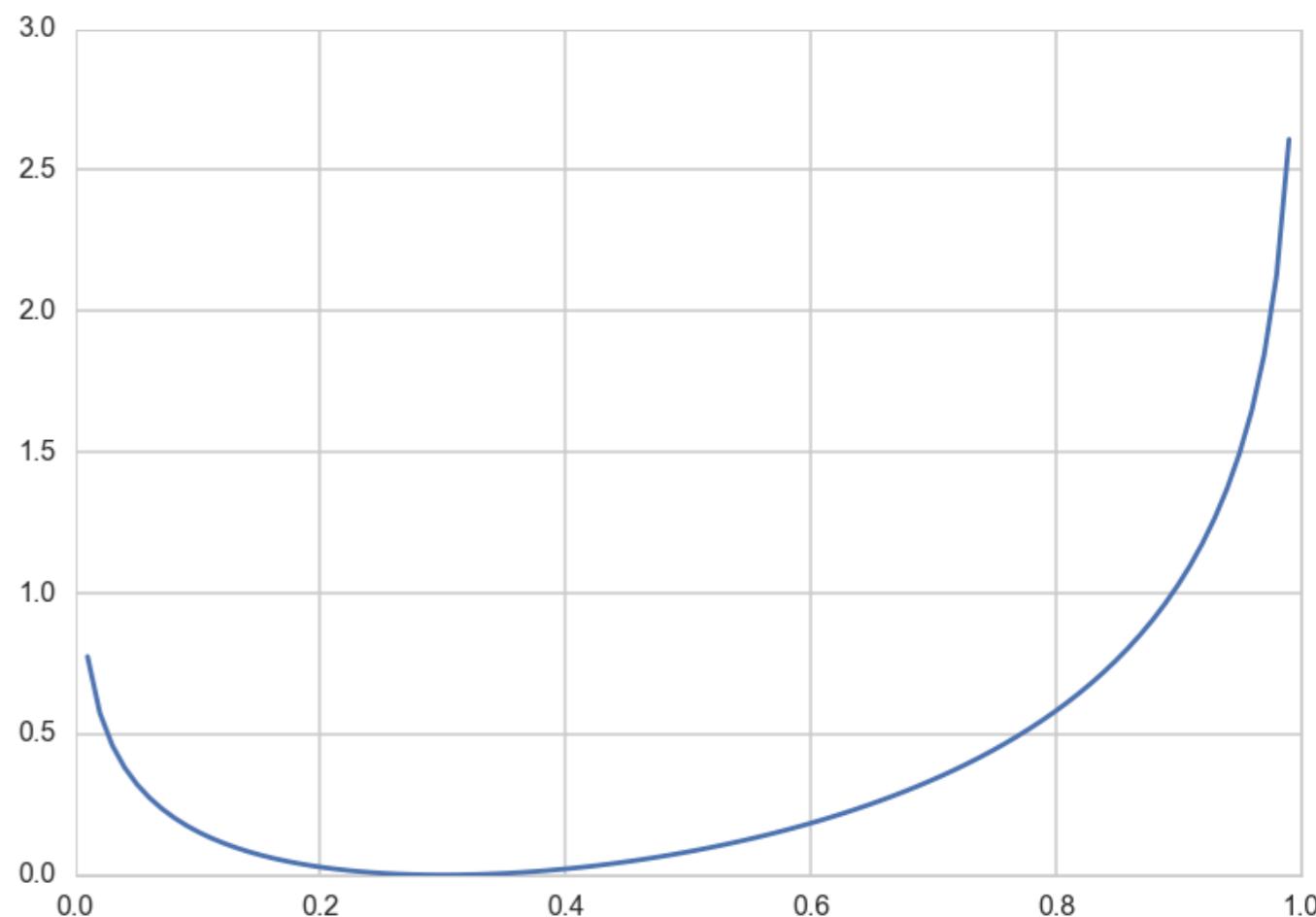
- McElreath page 179

KL example

Bernoulli Distribution p with
 $p = 0.3$.

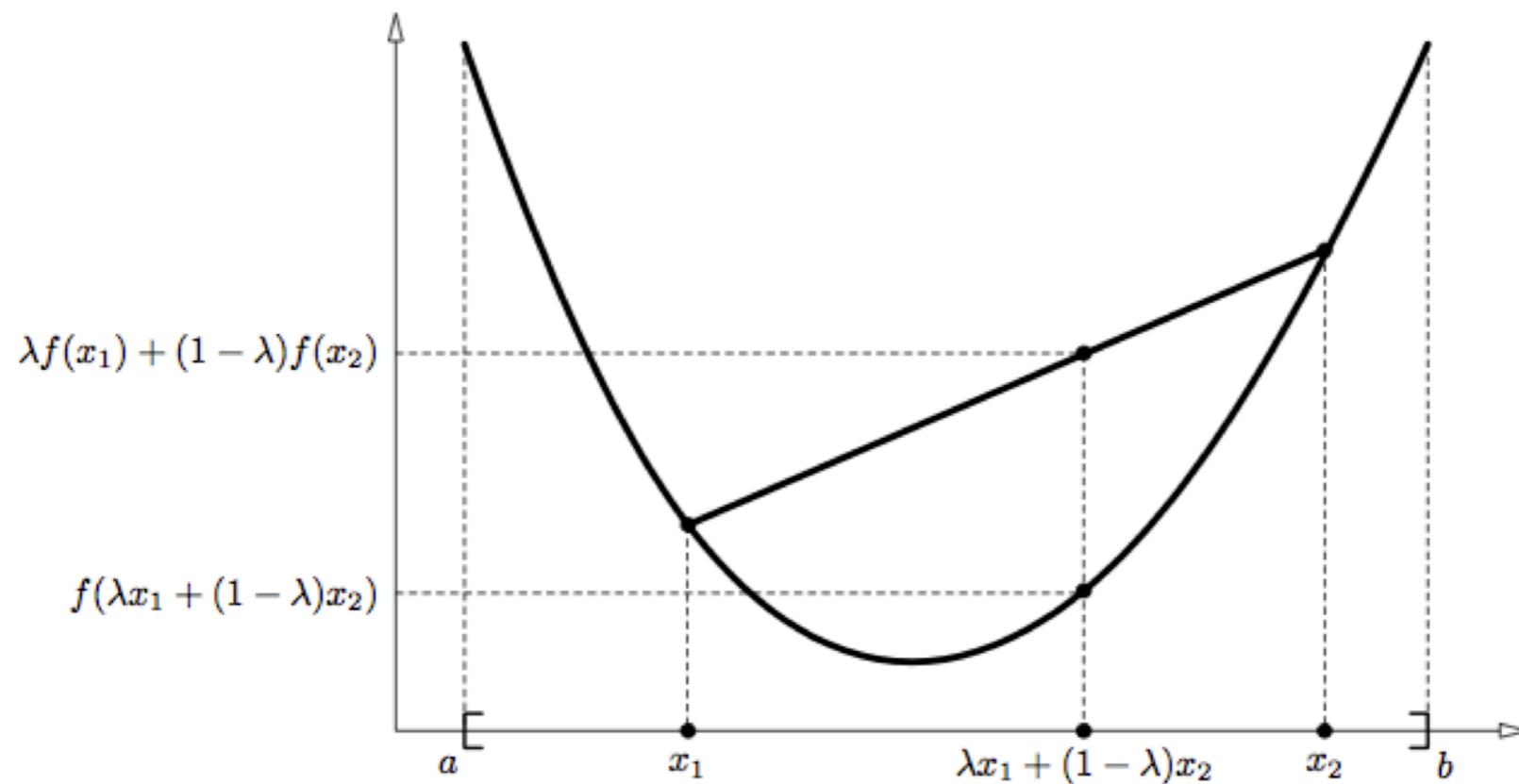
Try to approximate by q . What parameter?

```
def kld(p,q):  
    return p*np.log(p/q) + (1-p)*np.log((1-p)/(1-q))
```



Jensen's Inequality for convex $f(x)$:

$$E[f(X)] \geq f(E[X])$$



KL-Divergence is always non-negative

Jensen's inequality:

$$\implies D_{KL}(p, q) \geq 0 \text{ (0 iff } q = p \forall x).$$

$$\begin{aligned} D_{KL}(p, q) &= E_p[\log(p/q)] = E_p[-\log(q/p)] \geq -\log(E_p[q/p]) = \\ &\quad -\log\left(\int dQ\right) = 0 \end{aligned}$$

MARS ATTACKS (Topps, 1962; Burton 1996)

$\text{Earth} : q = \{0.7, 0.3\}$, $\text{Mars} : p = \{0.01, 0.99\}$.



Earth to predict Mars, less surprise on landing: $D_{KL}(p, q) = 1.14$, $D_{KL}(q, p) = 2.62$.

PROBLEM: we dont know distribution p . If we did, why do inference?

SOLUTION: Use the empirical distribution

That is, approximate population expectations by sample averages.

$$\implies D_{KL}(p, q) = E_p[\log(p/q)] = \frac{1}{N} \sum_i \log(p_i/q_i)$$

Maximum Likelihood justification

$$D_{KL}(p, q) = E_p[\log(p/q)] = \frac{1}{N} \sum_i (\log(p_i) - \log(q_i))$$

Minimizing KL-divergence \implies maximizing

$$\sum_i \log(q_i)$$

Which is exactly the log likelihood! MLE!

Model Comparison: Likelihood Ratio

$$D_{KL}(p, q) - D_{KL}(p, r) = E_p[\log(r) - \log(q)] = E_p[\log\left(\frac{r}{q}\right)]$$

In the sample approximation we have:

$$D_{KL}(p, q) - D_{KL}(p, r) = \frac{1}{N} \sum_i \log\left(\frac{r_i}{q_i}\right) = \frac{1}{N} \log\left(\frac{\prod_i r_i}{\prod_i q_i}\right) = \frac{1}{N} \log\left(\frac{\mathcal{L}_r}{\mathcal{L}_q}\right)$$

MODEL COMPARISON: Deviance

You only need the sample averages of the logarithm of r and q :

$$D_{KL}(p, q) - D_{KL}(p, r) = \langle \log(r) \rangle - \langle \log(q) \rangle$$

Define the deviance: $D(q) = -2 \sum_i \log(q_i)$, a **LOSS** ...

$$D_{KL}(p, q) - D_{KL}(p, r) = \frac{2}{N} (D(q) - D(r))$$

Example

Generate data from:

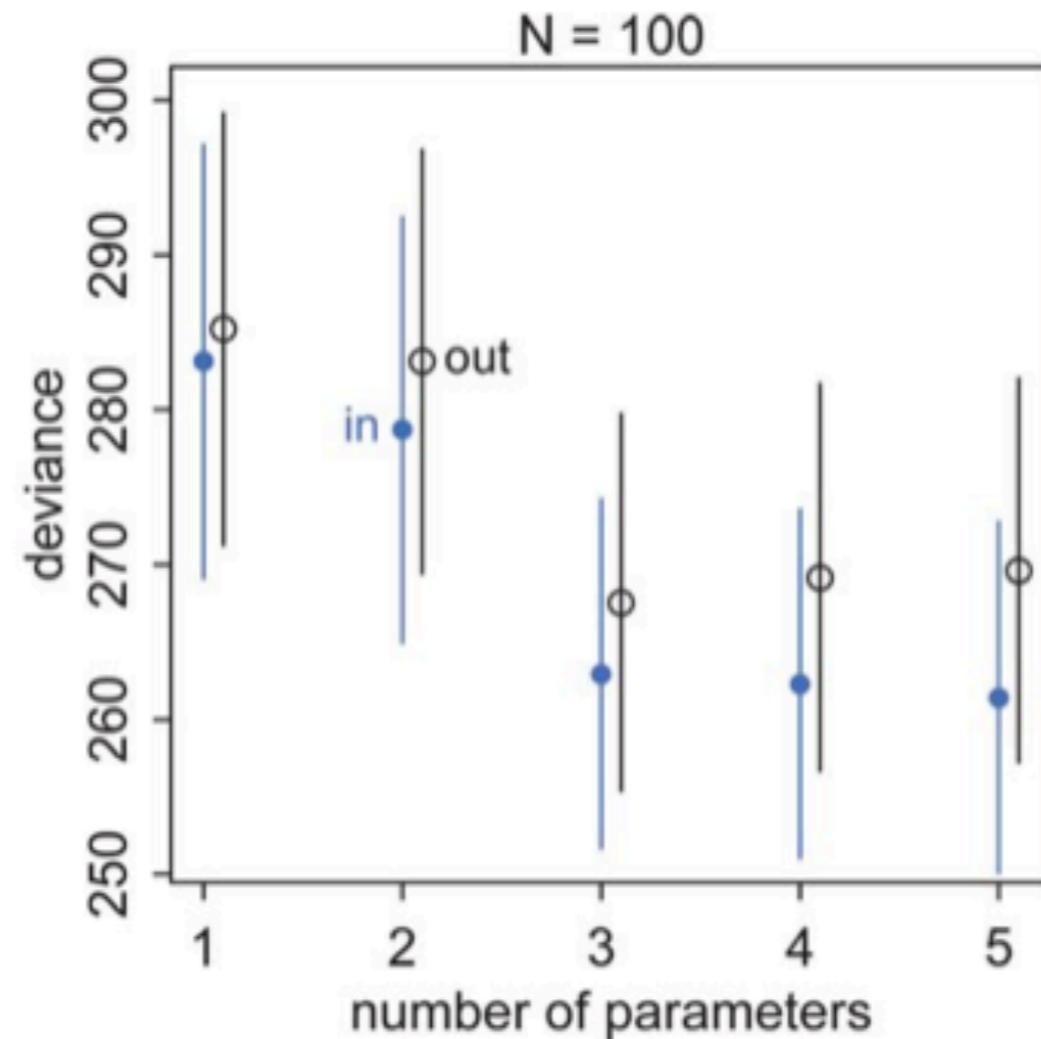
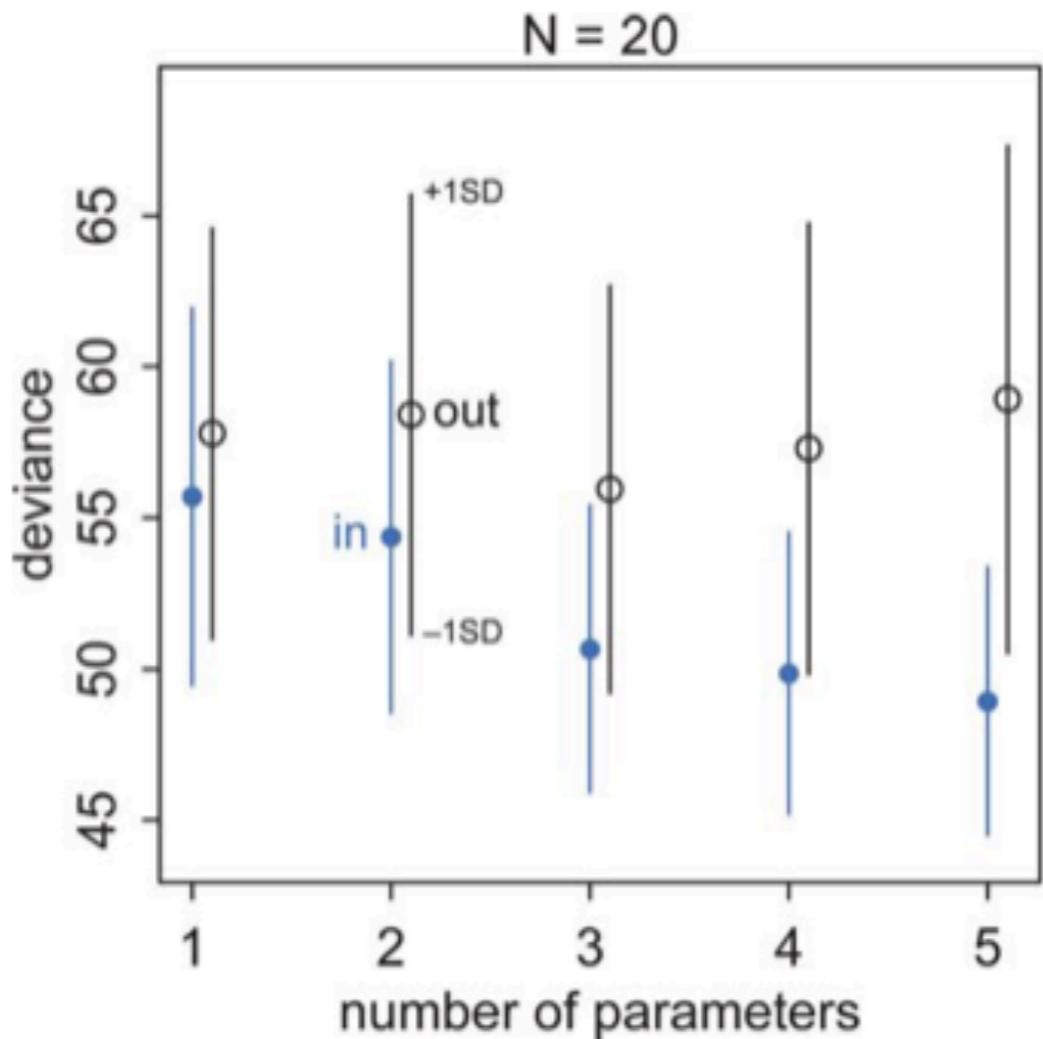
$$\mu_i = 0.15x_{1,i} - 0.4x_{2,i}, \quad y \sim N(\mu, 1)$$

2 parameter model.

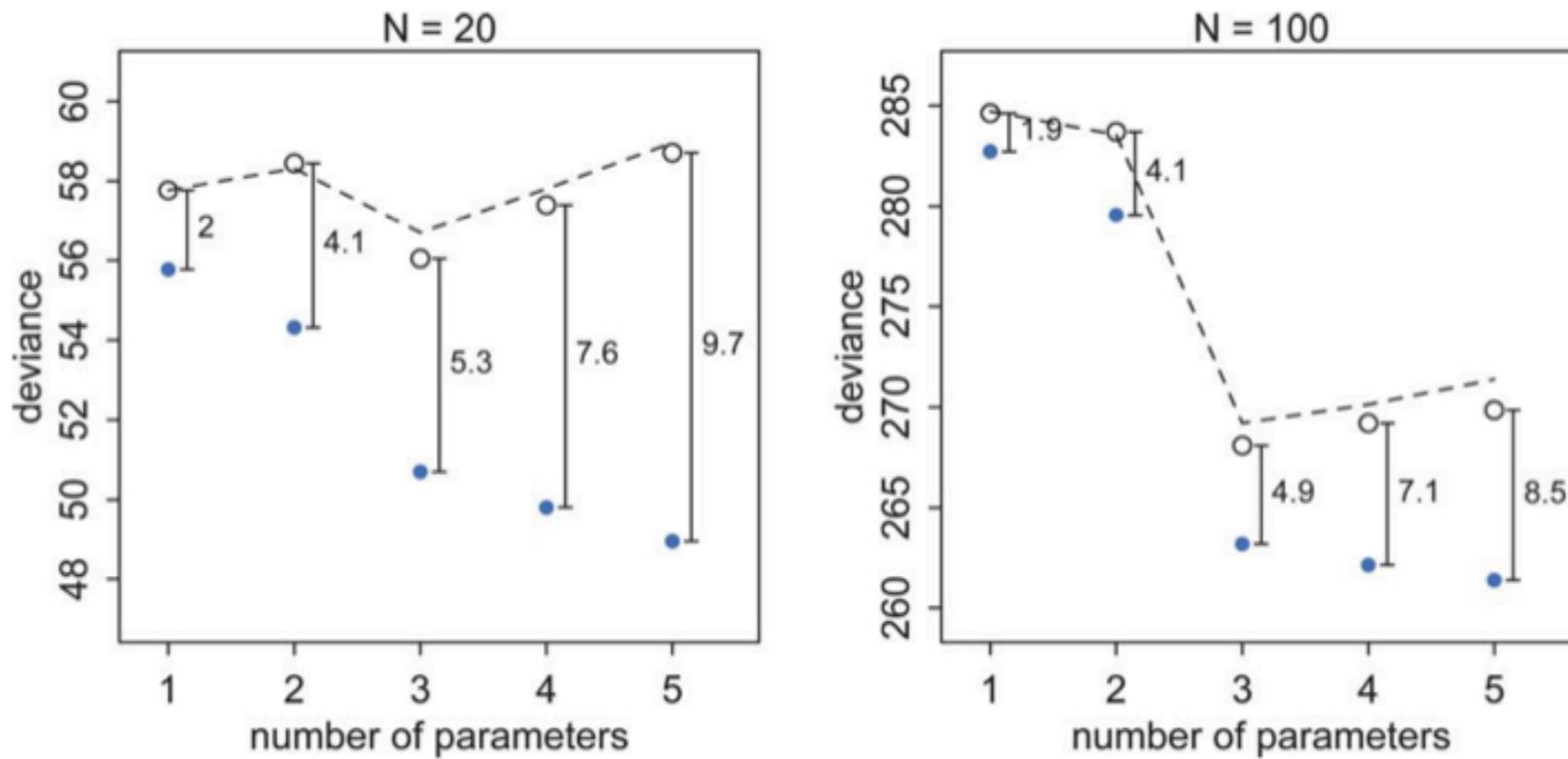
Generate 10,000 realizations, for 1-5 parameters, 20 data points and 100 data points.

Split into train and test, and do OLS.

Train and Test Deviances



Train and Test Deviances



The test set deviances are $2 * p$ above the training set ones.

Akaike Information Criterion:

AIC estimates out-of-sample deviance

$$AIC = D_{train} + 2p$$

- Assumption: likelihood is approximately multivariate gaussian.
- penalized log-likelihood or risk if we choose to identify our distribution with the likelihood:
REGULARIZATION

AIC for Linear Regression

$$AIC = D_{train} + 2p \text{ where}$$

$$D(q) = -2 \sum_i \log(q_i) = -2\ell$$

$$\sigma_{MLE}^2 = \frac{1}{N} SSE$$

$$AIC = -2\left(-\frac{N}{2}(\log(2\pi) + \log(\sigma^2)) - 2\left(-\frac{1}{2\sigma_{MLE}^2} \times SSE\right)\right) + 2p$$

$$AIC = N \log(SSE/N) + 2p + \text{constant}$$