Cmput 466 / 551

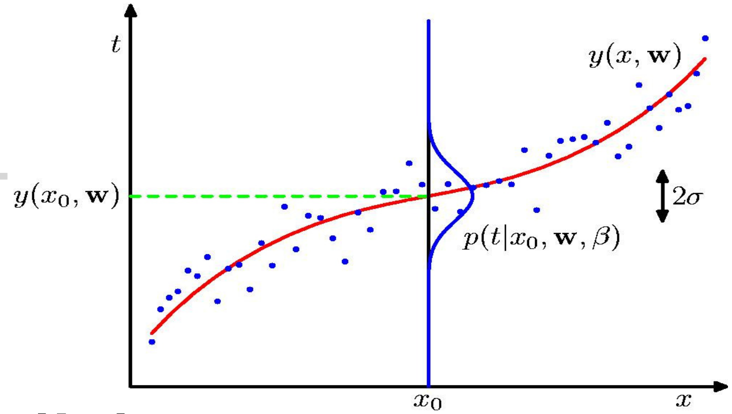
Evaluation of Learned Predictors

Covering chapters *HTF: Ch3, 7*

R Greiner
Department of Computing Science
University of Alberta

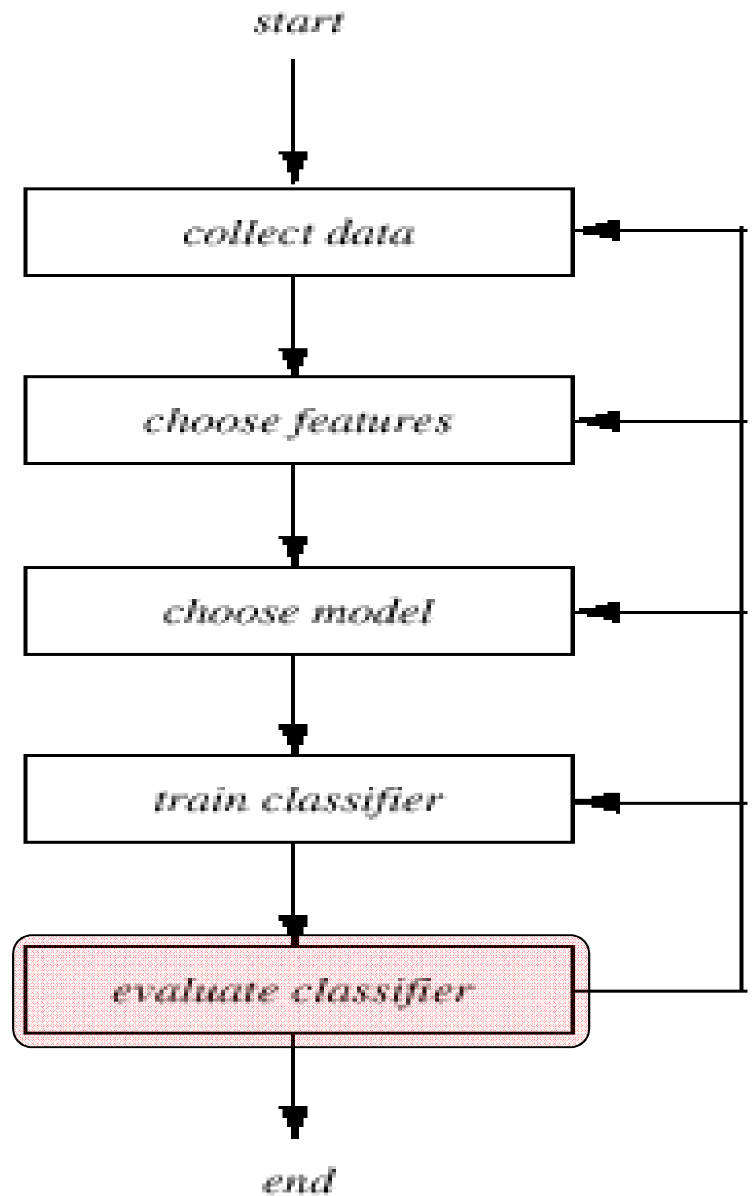
Thanks to: C Guestrin, T Dietterich, R Parr, H L Størvold, R Salakhutdinov

Outline



- Linear Regression
- Evaluating Learned Predictors
    - Training set error vs True error
    - Test set error
    - Cross Validation
- Linear Classification
- Overfitting
    - Bias-Variance analysis
    - Regularization, Internal C-V, Bayesian Model

The Design Cycle



Training Set Error

- Given a labeled dataset S (training data),
  - learn optimal predictor $\boldsymbol{\theta}_{\mathbf{S}}$

$$\boldsymbol{\theta}_{\mathbf{S}} = \theta^*(\mathbf{S}) = \arg\min_{\theta} \frac{1}{|\mathbf{S}|} \sum_{(x,y)\in\mathbf{S}} \left(y - \sum_i \theta_i \, h_i(x) \right)^2$$

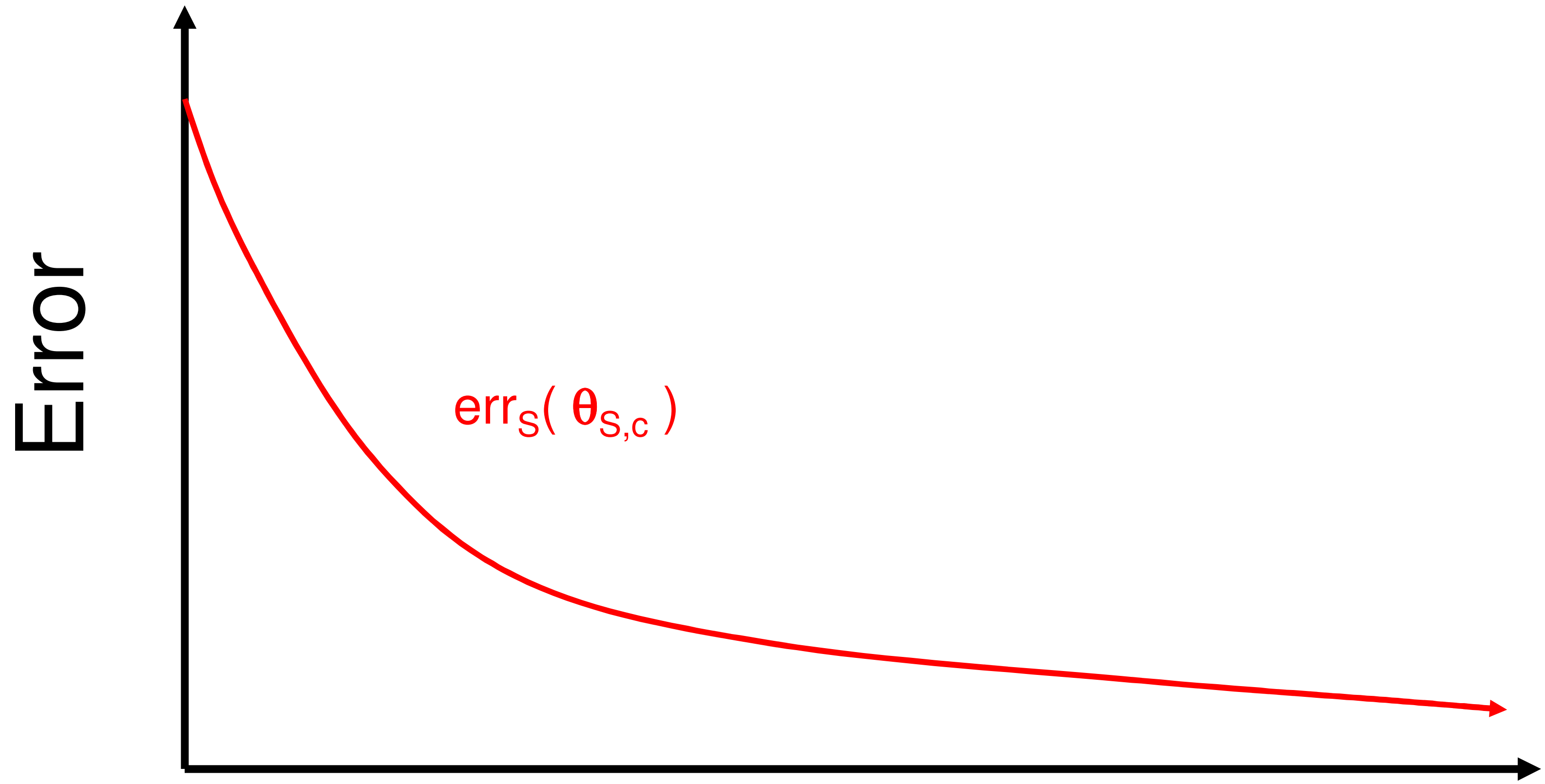
  - compute empirical error (of any $\boldsymbol{\theta}$)

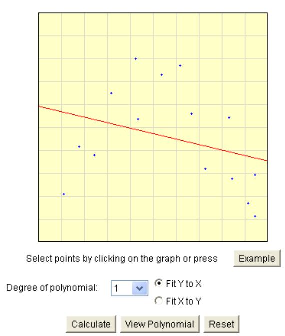
$$\mathrm{err}_{\mathbf{S}}(\, \boldsymbol{\theta} \,) = \frac{1}{|\mathbf{S}|} \sum_{(x,y)\in\mathbf{S}} \left(y - \sum_i \theta_i \, h_i(x) \right)^2$$

- **Training set error:** $\mathrm{err}_{S}(\, \theta_S \,)$

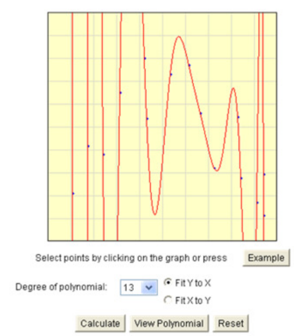
Note $\mathrm{err}_{\alpha}(\, \theta_\beta \,)$ is related to $J(\, \theta_\beta \,)$

Training Set Error as a function of Model Complexity

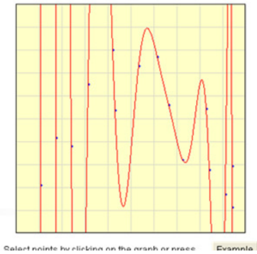




"Model Complexity", c
… eg, #basis functions;
degree of poly, …



True Prediction Error



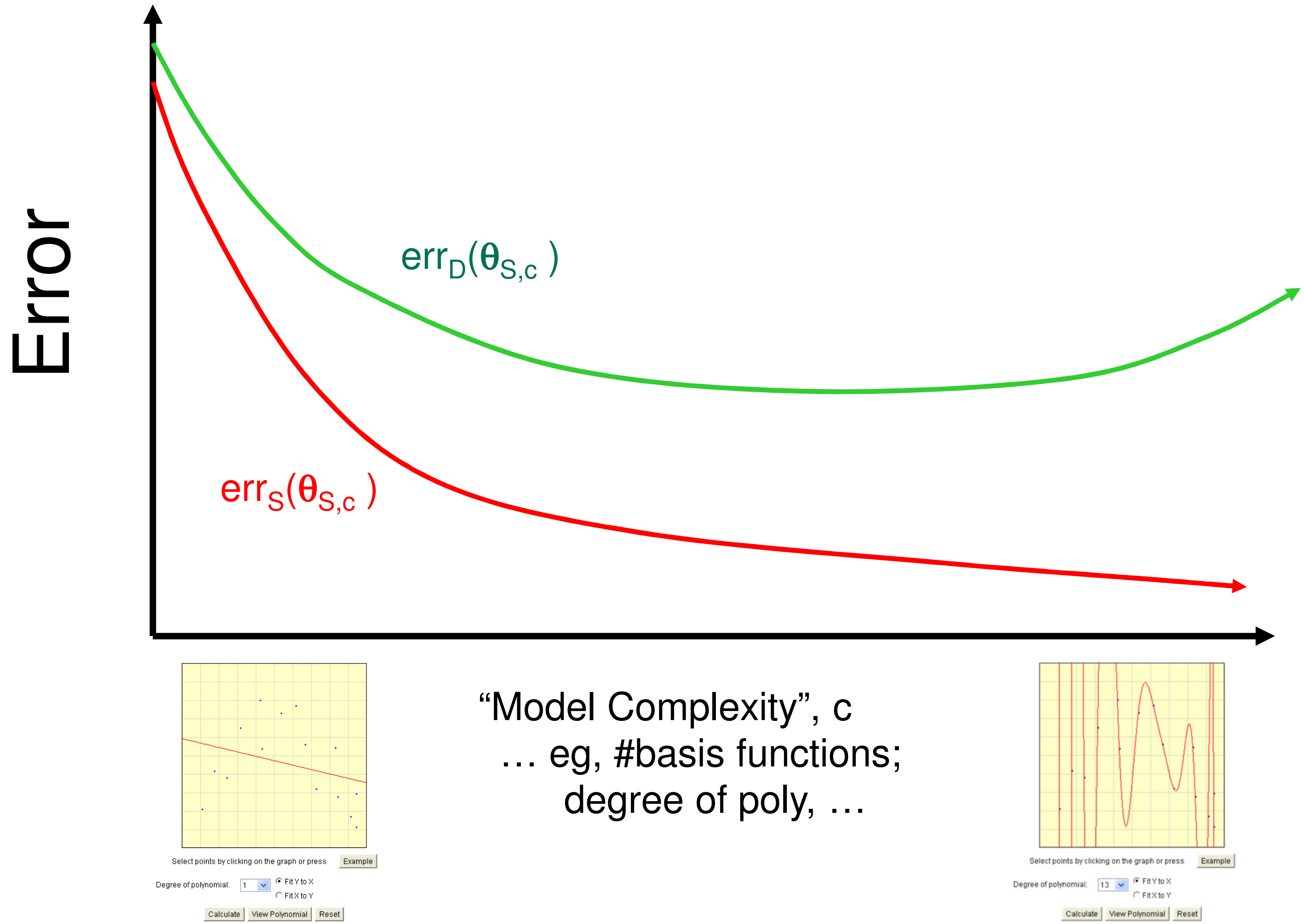
- **Goal:** *small error over all likely input points,* from D(**x**,y) … **not just wrt training data**:
  - should use **prediction error:**

$$\mathrm{err}_D(\boldsymbol{\theta}) = E_{(\mathbf{x},y)\sim D}\left[\left(y - \sum_i \theta_i\, h_i(\mathbf{x})\right)^2\right]$$

$$= \int_{\mathbf{x},y} \left(y - \sum_i \theta_i\, h_i(\mathbf{x})\right)^2 D(\mathbf{x},y)\, d\mathbf{x}\, dy$$

- Note: $\mathrm{err}_D(\theta_S) \neq$ Training error $\mathrm{err}_S(\theta_S)$
  - $\mathrm{err}_S(\theta_S)$ can be poor measure of "quality" of solution

Prediction Error as a function of Model Complexity



Computing Prediction Error

- Computing true prediction error

$$err_D(\boldsymbol{\theta}) = \int_{x,y} \left(y - \sum_i \theta_i h_i(\mathbf{x})\right)^2 D(\mathbf{x}, \mathrm{y})\ d\mathbf{x}\, dy$$

  - Depends on $D(\mathbf{x}, y)$ – typically not known
  - Estimate (parameterized form) can be difficult integral
- New sample: a set of i.i.d. points

$$S' = \{ (\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_M, y_M) \} \text{ from } D(\mathbf{x}, y)$$

$$\mathrm{err}_D(\boldsymbol{\theta}_S) \approx \mathrm{err}_{S'}(\boldsymbol{\theta}_S) = \frac{1}{|S'|} \sum_{(x,y)\in S'} \left(y - \sum_i \theta_{S,i}\, h_i(x)\right)^2$$

Training Error ≠ Prediction Error

- Sampling approximation of prediction error:

  $$\mathrm{err}_{S'}(\ \theta_S\ ) \approx \mathrm{err}_D(\ \theta_S\ )$$

- Training error :

  $$\mathrm{err}_{S}(\ \theta_S\ ) \not\approx \mathrm{err}_D(\ \theta_S\ )$$

- Very similar equations!!!
  - Why is *training error* a bad measure of *prediction error*?

Training Error ≠ Prediction Error

- ...or:

Because you cheated!!!

Training error is good estimate for a single θ,
but you optimized θ with respect to the training error,
and found θ that is good for *this set of instances*

Training error is a (optimistically) biased estimate of prediction error

-

- Very similar equations!!!
  - Why is *training error* a bad measure of *prediction error*?

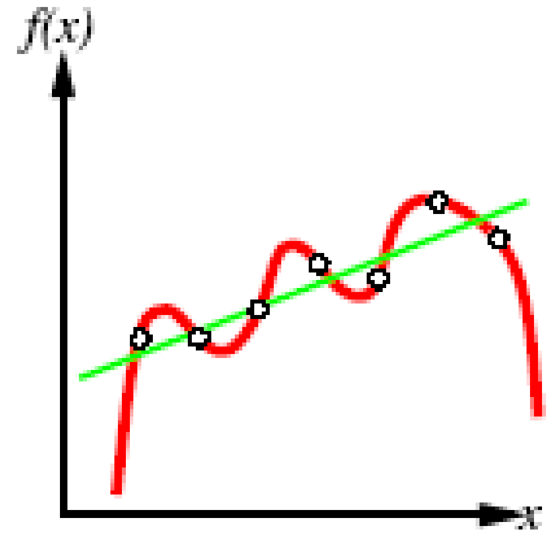
Example …

- $\widehat{\mathrm{err}}_S(\ \theta_S\ ) \neq \mathrm{err}_D(\ \theta_S\ )$

- $\widehat{\mathrm{err}}_S(\ \theta_S\ ) \equiv$
  Eval θ_S on training set S
  - only approximation to $\mathrm{err}_D(\ \theta_S\ )$
  - ... can be TOO optimistic!

- "Cheating"

  Like being evaluated on test after seeing SAME test ...



$\widehat{\mathrm{err}}_S(\ \theta_r\ ) = 0$
$\widehat{\mathrm{err}}_S(\ \theta_g\ ) > 0$

Fit-to-Data ≠ Generalization

- "Overfitting"

  Best "fit-to-data" can find "meaningless regularity" in data

  (coincidences in the noise)

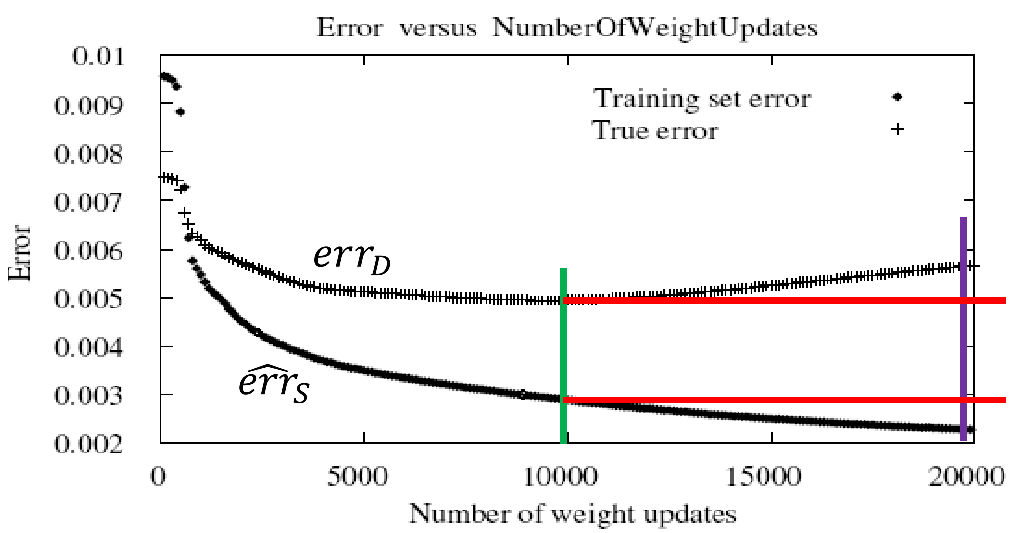
  $\Rightarrow$ bad generalization behavior
- Defn: Hypothesis $h \in H$ *overfits* training data if

  $\exists$ alternative hypothesis $h' \in H$ s.t.

$$\widehat{\mathrm{err}}_S(h) < \widehat{\mathrm{err}}_S(h')$$

but

$$\mathrm{err}_D(h) > \mathrm{err}_D(h')$$

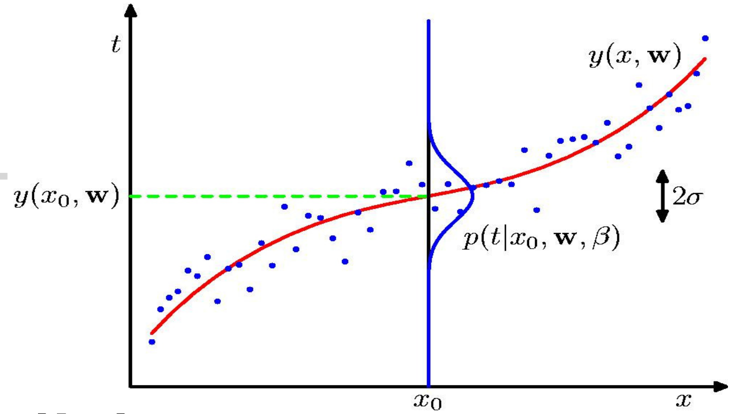


$$\widehat{\mathrm{err}}_S(h_{20000}) < \widehat{\mathrm{err}}_S(h_{10000})$$

but

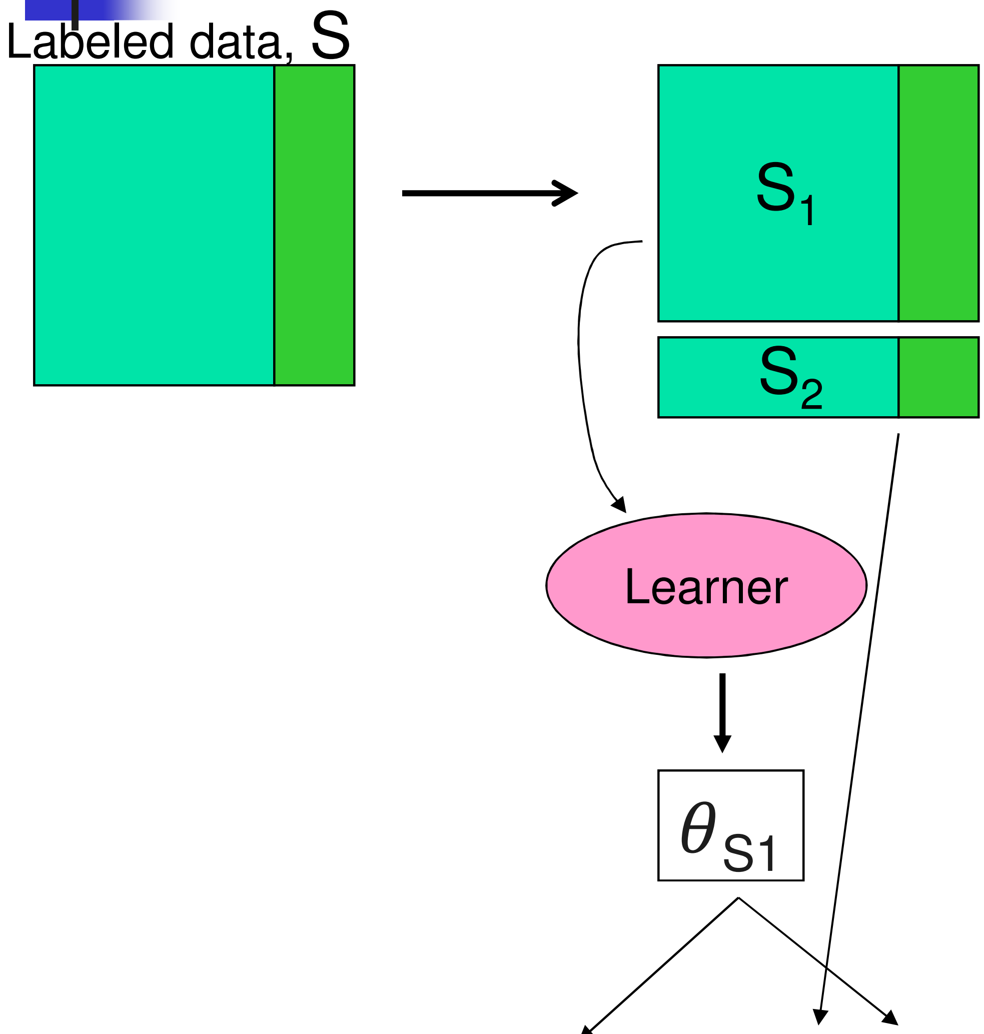
$$\mathrm{err}_D(h_{20000}) > \mathrm{err}_D(h_{10000})$$

Outline



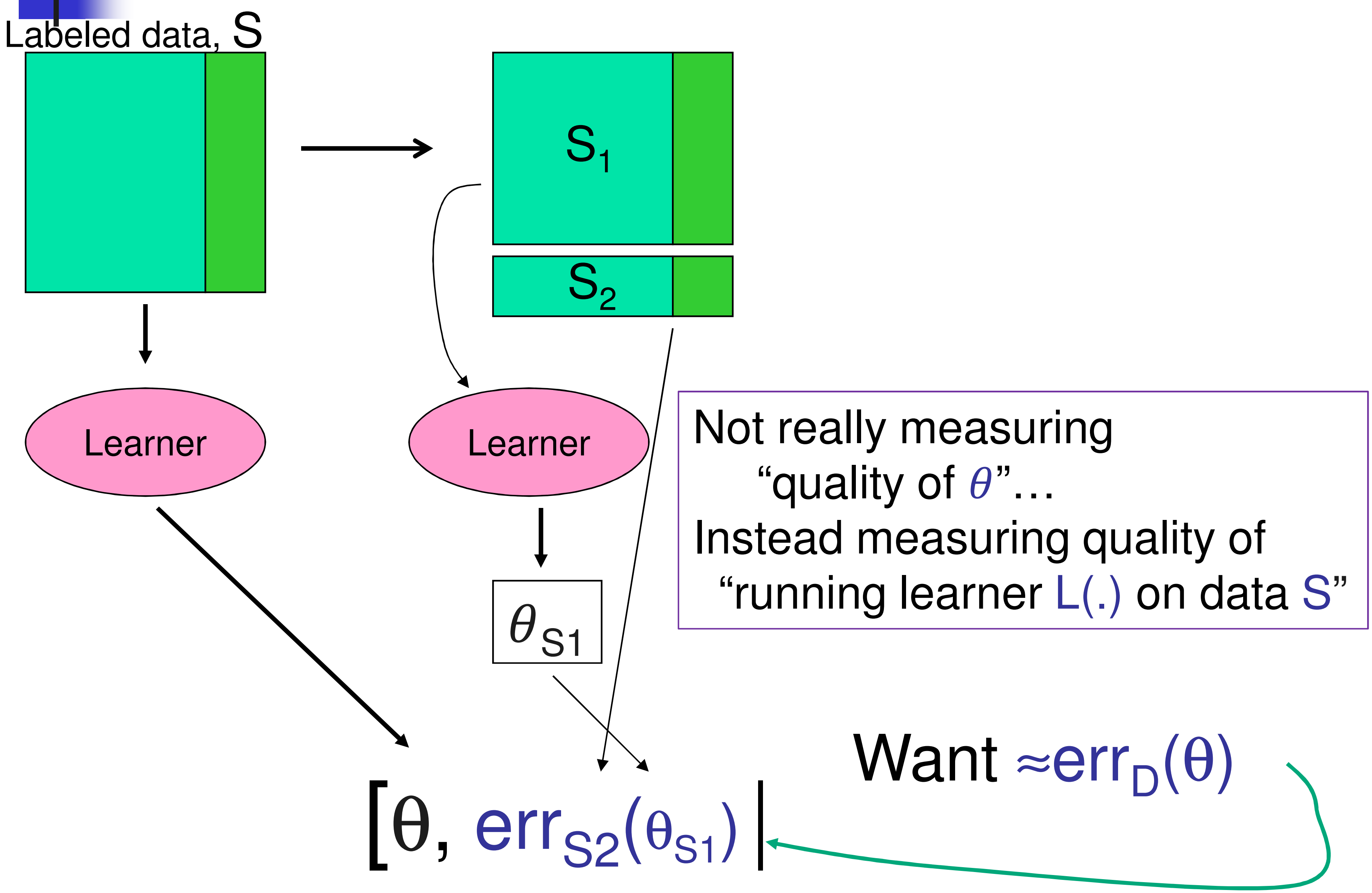
- Linear Regression
- Evaluating Learned Predictors
  - Training set error vs True error
  - Test set error
  - Cross Validation
- Linear Classification
- Overfitting
  - Bias-Variance analysis
  - Regularization, Internal C-V, Bayesian Model

Return: [Predictor + Est Quality]

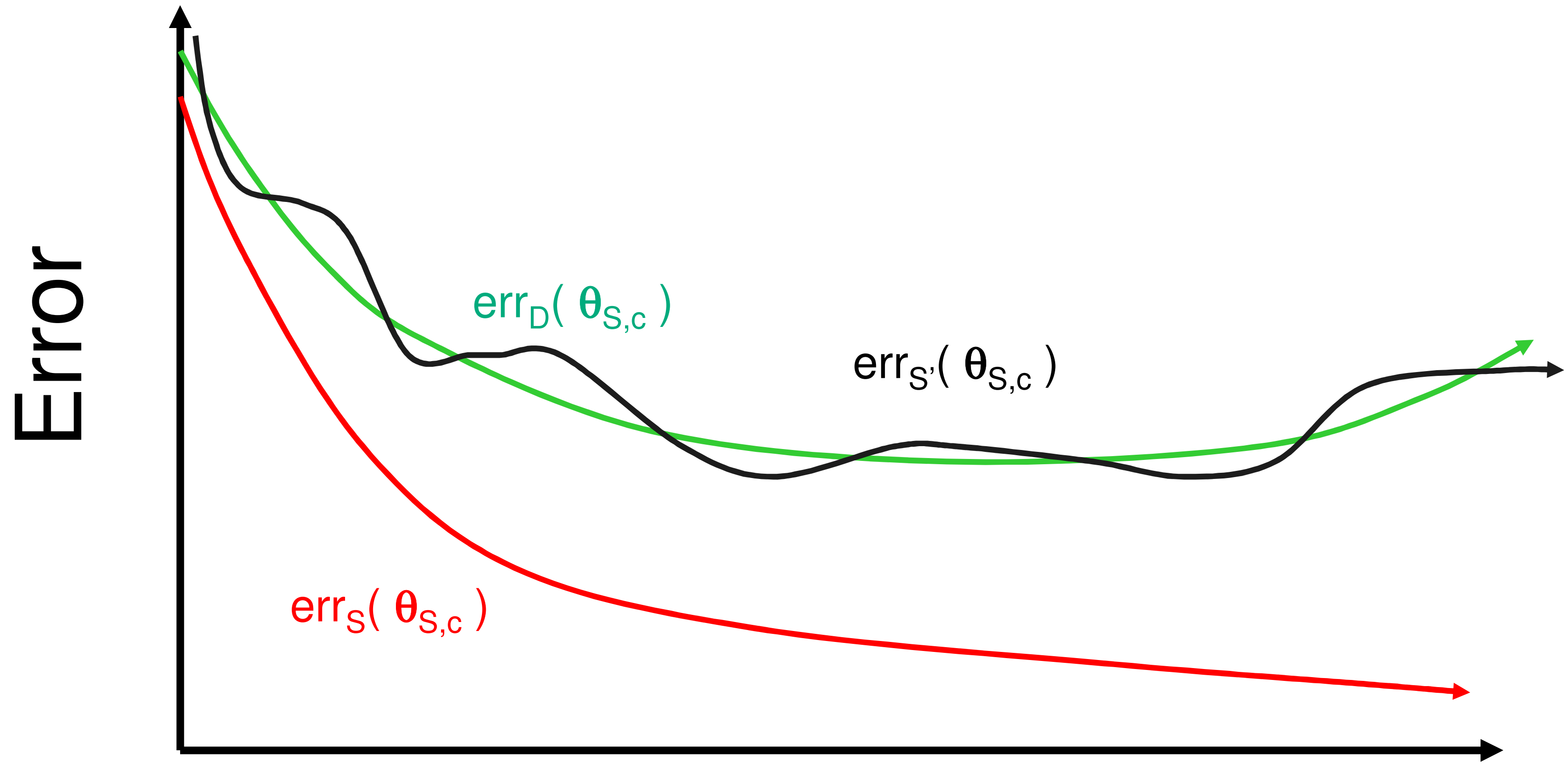


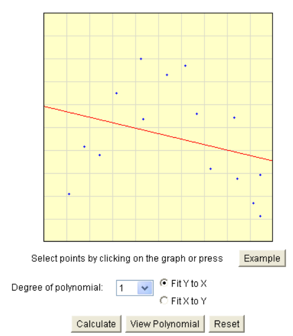
$$\left[\theta_{S1},\ err_{S2}(\theta_{S1})\right]$$

Return: [Predictor + Est Quality]

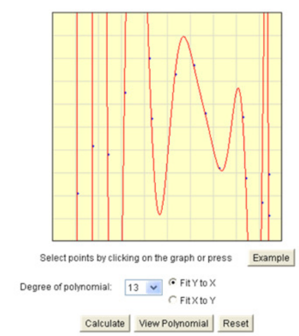


Test Set Error as a function of Model Complexity

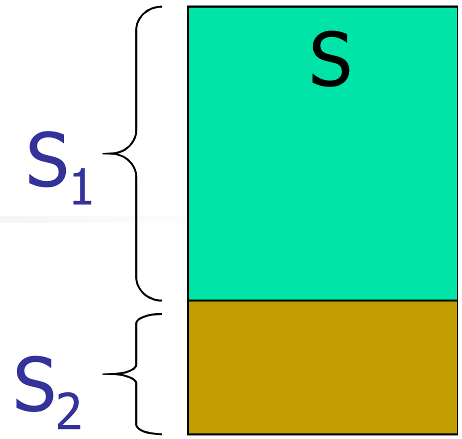




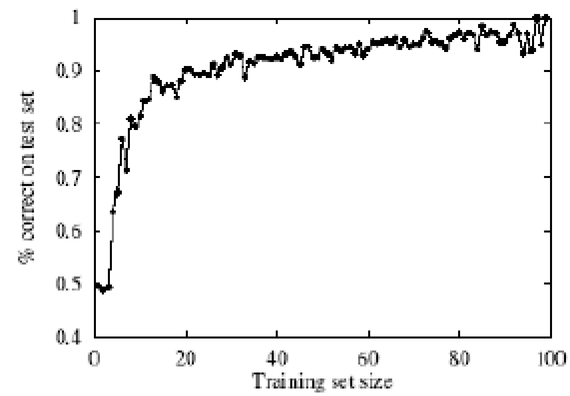
"Model Complexity", c
… eg, #basis functions;
degree of poly, …



Challenge wrt Hold-Out Set



- How to divide S into disjoint S_1, S_2
- As $|S_1| < |S|$,
  $L(S_1)$ not as good as $L(S)$
  Learning curve: $L(S)$'s **quality** as $|S|$ increases
  $\Rightarrow$ want S_1 to be large



- $\widehat{\text{err}}_{S2}(\,\theta_{S_1}\,)$ is estimate of $\text{err}_D(\theta_S)$
  Estimate improves as S_2 gets larger
  $\Rightarrow$ want S_2 to be as large as possible

$$|\,\widehat{\text{err}}_S(\,\theta\,) - \text{err}_D(\,\theta\,)| \;\approx\; \frac{\alpha}{\sqrt{|S|}}$$

- As $S = S_1 \cup S_2$, must trade off
  quality of predictor $\theta_{S_1} = L(S_1)$
  versus
  accuracy of **estimate** $\widehat{\text{err}}_{S2}(\,\theta_{S_1}\,)$

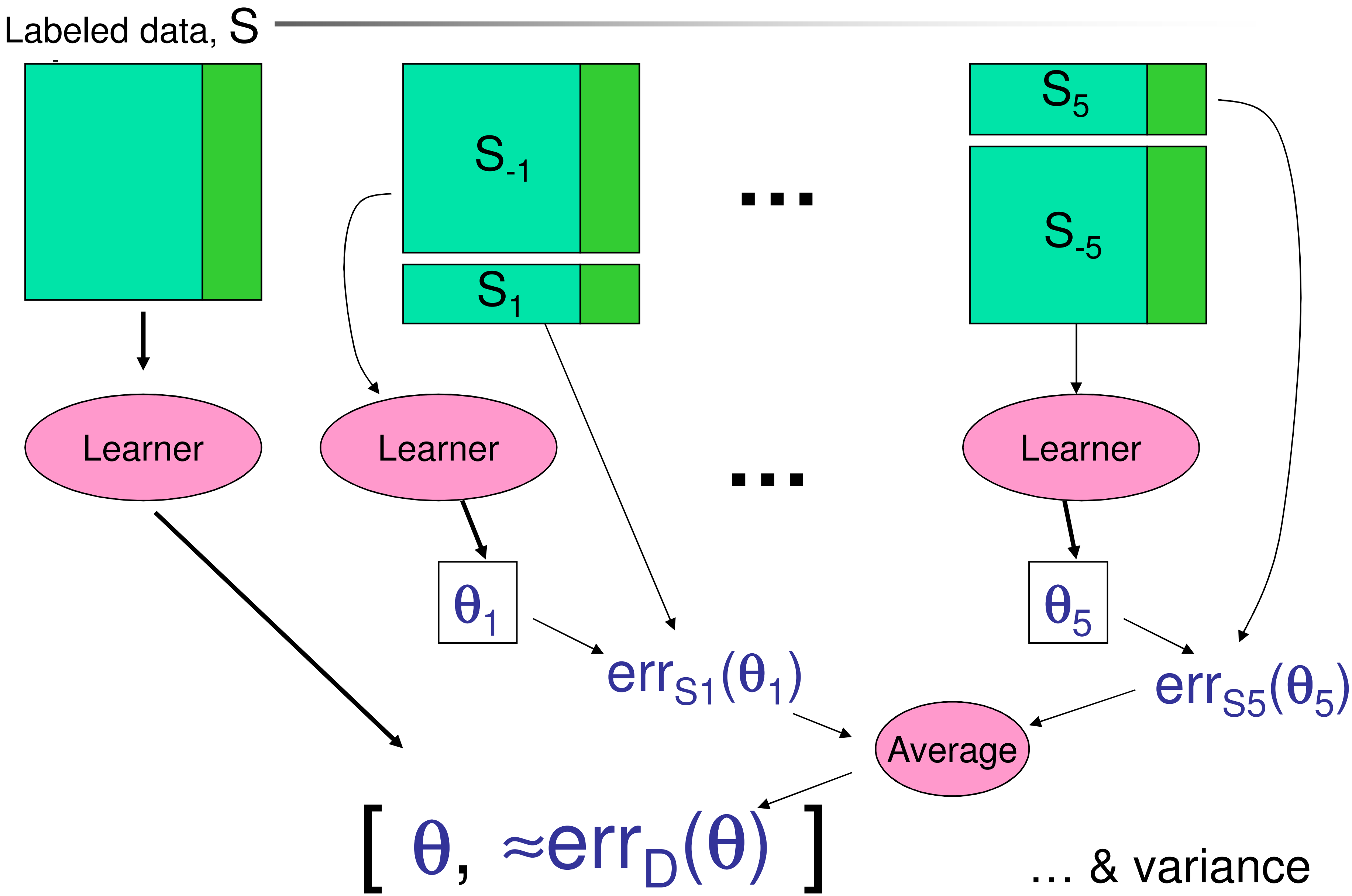
How many points needed for training/testing?

- Very hard question to answer!
  - Too few **training** points, learned θ is bad
  - Too few **test** points, you never know if you reached a good solution
- Bounds [eg, Hoeffding's inequality] can help:

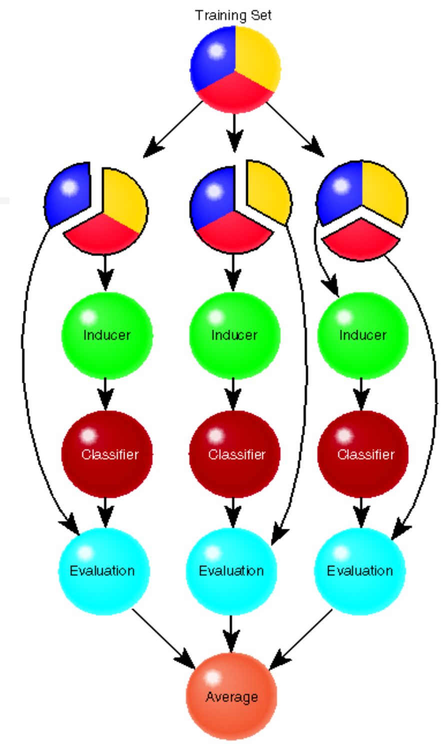
$$P(|\hat{\theta}^N - \theta^*| \geq \epsilon) \leq 2e^{-2N\epsilon^2}$$

  - More on this later …
- Typically:
  - if you have a LARGE amount of data,
    pick test set "large enough" for a "reasonable" estimate of error, and use the rest for learning
  - if you have little data (typical case!),
    then you need other tricks
    - eg, bootstrapping, cross-validation

Cross Validation



Estimating Error: Cross Validation



- "Cross-Validation"

CV(data S, alg L, int k)
  Divide S into k disjoint sets { S_1, S_2, …, S_k }
  For i = 1..k do
    Run L on $S_{-1} = S - S_i$
    obtain $f_i := L(S_{-i})$
    Evaluate f_i on S_i

$$\mathrm{err}_{S_i}(\,\mathrm{f_i}\,) \;=\; \frac{1}{|S_i|}\sum_{(\mathbf{x},y)\in S_i}\left(y - \mathrm{f_i}(\mathbf{x})\right)^2$$

  Return Average $\frac{1}{k}\sum_i \mathrm{err}_{S_i}(f_i)$

⇒ Less Pessimistic

as trained on $\frac{k-1}{k}$ |S| of the data

Comments on Cross-Validation

- Every point used as
  Test 1 time, Training k – 1 times
- Computational cost for *k*-fold Cross-validation … linear in *k*
- Use CV(S, L, k) as ESTIMATE of true error of apply L(·) to S
  … return [L(S), CV(S, L, k)]
- Leave-One-Out-Cross-Validation k = |S| !
  - eg, for Nearest-Neighbor
- Notice different folds are correlated
  as training sets overlap: (k-2)/k unless k=2
- 5 × 2-CV
  - Run 2-fold CV, 5 times…
- Should use "balanced CV"
  If class c_i appears in m_i instances,
  insist each S_k include $\approx \frac{1}{k} \frac{m_i}{|S|}$ such instances

Can use CV to estimate parameters in general!

To Form k *Balanced* Folds

1. Partition the data S based on the class:
   - subset S_+ has all the positive instances
   - subset S_- has all the negative instances

2. Randomly partition each class-subset into k folds:

$$S_+ = \cup \{ S_{+1}, \ldots, S_{+k} \}$$

$$S_- = \cup \{ S_{-1}, \ldots, S_{-k} \}$$

3. $S_j = S_{+j} \cup S_{-j}$ for j=1..k

What is Cross Validation Measuring?

- Let regressor $R_S = L(S)$
  - … the result of running learner $L(\cdot)$ on data S
- Cross-Validation is not really measuring "quality of R_S"
- Instead, it is measuring quality of "running learner $L(\cdot)$ on data S"
  - … even if $L(\cdot)$ is a complex process, that involves finding parameters (perhaps using "internal cross-validation) …

Error Estimators

$$\mathrm{err}_D(\boldsymbol{\theta}_S) = \int_{\mathbf{x},y} \left(y - \sum_i \theta_{S,i}\, h_i(\mathbf{x}) \right)^{\mathbf{2}} D(\mathbf{x}, y)\, d\mathbf{x}\, dy$$

Gold Standard!
Unbiased

$$\cancel{\mathrm{err}_S(\boldsymbol{\theta}_S) = \frac{1}{|S|} \sum_{(\mathbf{x},y) \in S} \left(y - \sum_i \theta_{S,i}\, h_i(x) \right)^2}$$

Uses TRAIN data
… **optimistic**

$$\mathrm{err}_{S'}(\boldsymbol{\theta}_S) = \frac{1}{|S'|} \sum_{(\mathbf{x},y) \in S'} \left(y - \sum_i \theta_{S,i}\, h_i(\mathbf{x}) \right)^2$$

Approx truth…
Unbiased
… if you are careful

Error Estimators

Gold Standard!
Unbiased

Be careful!!!

Test set only unbiased if you *never never never never* do *any any any* learning/adjustment/… on the test data

Eg,
if you use the test set to select the degree of the polynomial…
no longer unbiased!!!

Approx truth…
Unbiased
… if you are careful

$$\mathrm{err}_{S'}(\boldsymbol{\theta}_S) = \frac{1}{|S'|} \sum_{(\mathbf{x},y)\in S'} \left(y - \sum_i \theta_{S,i}\, h_i(\mathbf{x}) \right)^2$$

Summary of Estimating Error

- SetUp: Learner L,
  - using labeled training data S
  - produces predictor $r_S = L(S)$
- Want $err_D(\, r_S \,)$

  = r's Generalization Error over distribution D
  - to evaluate predictor r_S
  - to decide among possible predictors
  - to evaluate learner
- But depends on $D(\mathbf{x},y)$: not known!

Estimating $err_D(\, r_S \,)$

~~1. Training Set Error~~
  - ~~Use r_S's empirical error on S~~

    ~~$\underline{err_S}(\, r_S \,)$~~

  ~~$\Rightarrow$ Very Optimistic~~

2. Hold Out Error
   - Divide $S = S_1 \cup S_2$;

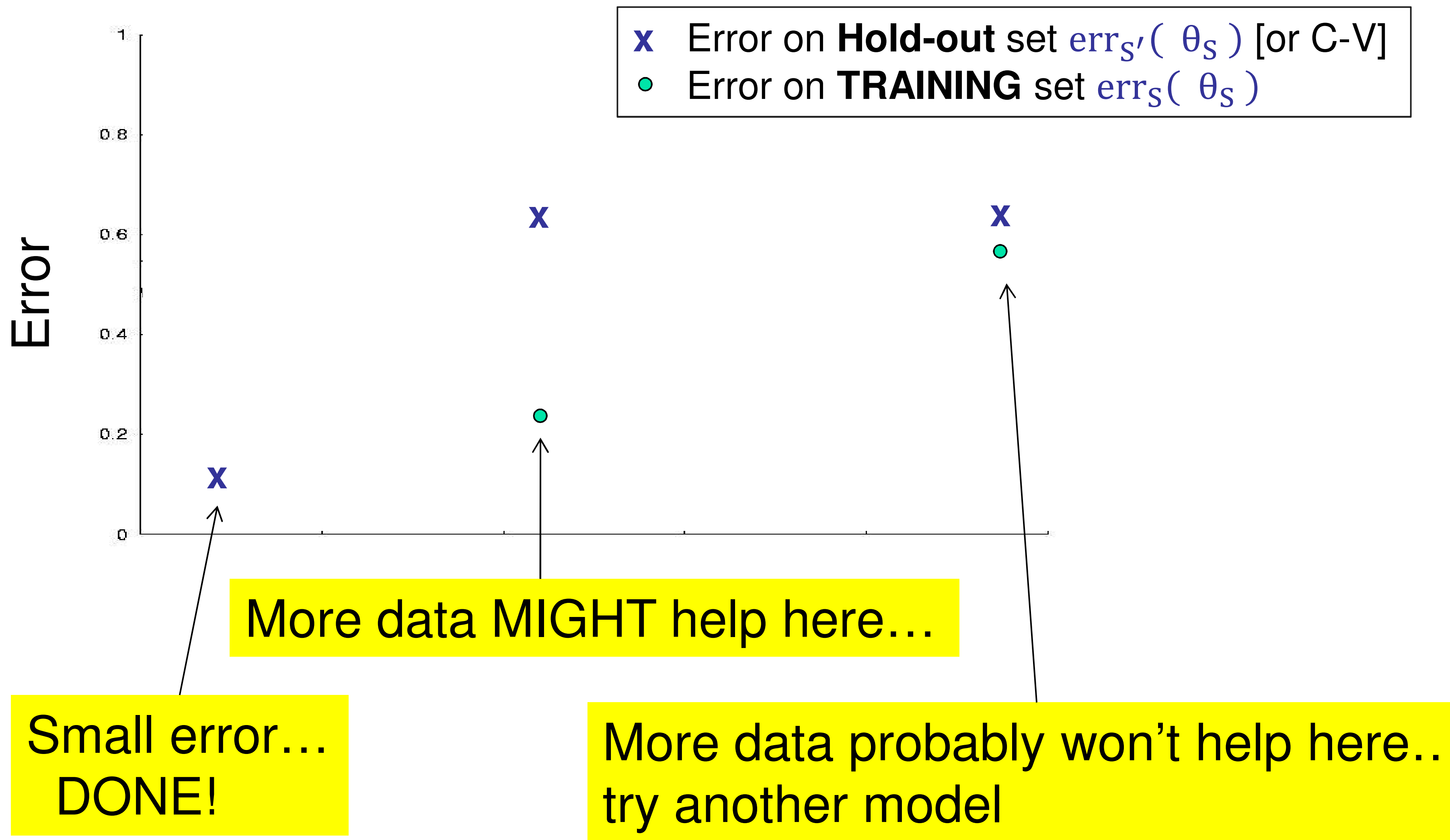
     Return $\underline{err_{S_2}}(r_{S_1})$

   $\Rightarrow$ Slightly Pessimistic
3. Cross Validation
   - $\frac{1}{k} \sum_i err_{S_i}(\, L(S_{-i})$

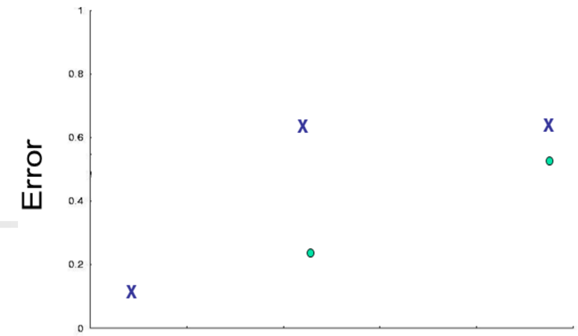
   $\Rightarrow$ Slightly Less Pessimistic

For evaluating GENERAL PREDICTORS
- classifiers, regressors
- … best values for parameters

Will more data help?



3 Cases …



- If $\widehat{\mathrm{err}}_{S'}(\ \theta_S\ ) \approx 0$:
  All is good! We have a good predictor!
- If $\widehat{\mathrm{err}}_{S'}(\ \theta_S\ ) \gg 0$ but $\widehat{\mathrm{err}}_{S}(\ \theta_S\ ) \approx 0$
  then High Variance [Overfitting]

  $\Rightarrow$ Try more data
- If $\widehat{\mathrm{err}}_{S'}(\ \theta_S\ ) \gg 0$ and $\widehat{\mathrm{err}}_{S}(\ \theta_S\ ) \gg 0$

  then High Bias

$\Rightarrow$ Try new model
(Just more data probably won't help)

See Bias/Variance lecture …

Summary

- Want error on NOVEL instances

$$\mathrm{err}_D(\boldsymbol{\theta}_S) = \int_{\mathbf{x},y} \left(y - \sum_i \theta_{S,i}\, h_i(\mathbf{x})\right)^2 D(\mathbf{x}, y)\, d\mathbf{x}\, dy$$

- NOT error on training

$$\mathrm{err}_S(\boldsymbol{\theta}_S) = \frac{1}{|S|} \sum_{(\mathbf{x},y)\in S} \left(y - \sum_i \theta_{S,i}\, h_i(\mathrm{x})\right)^2$$

- Should use
  - "Test set error"
  - Cross Validation
- Comparing
  TrainingSetError with HoldOutError
  suggests whether more data might be useful