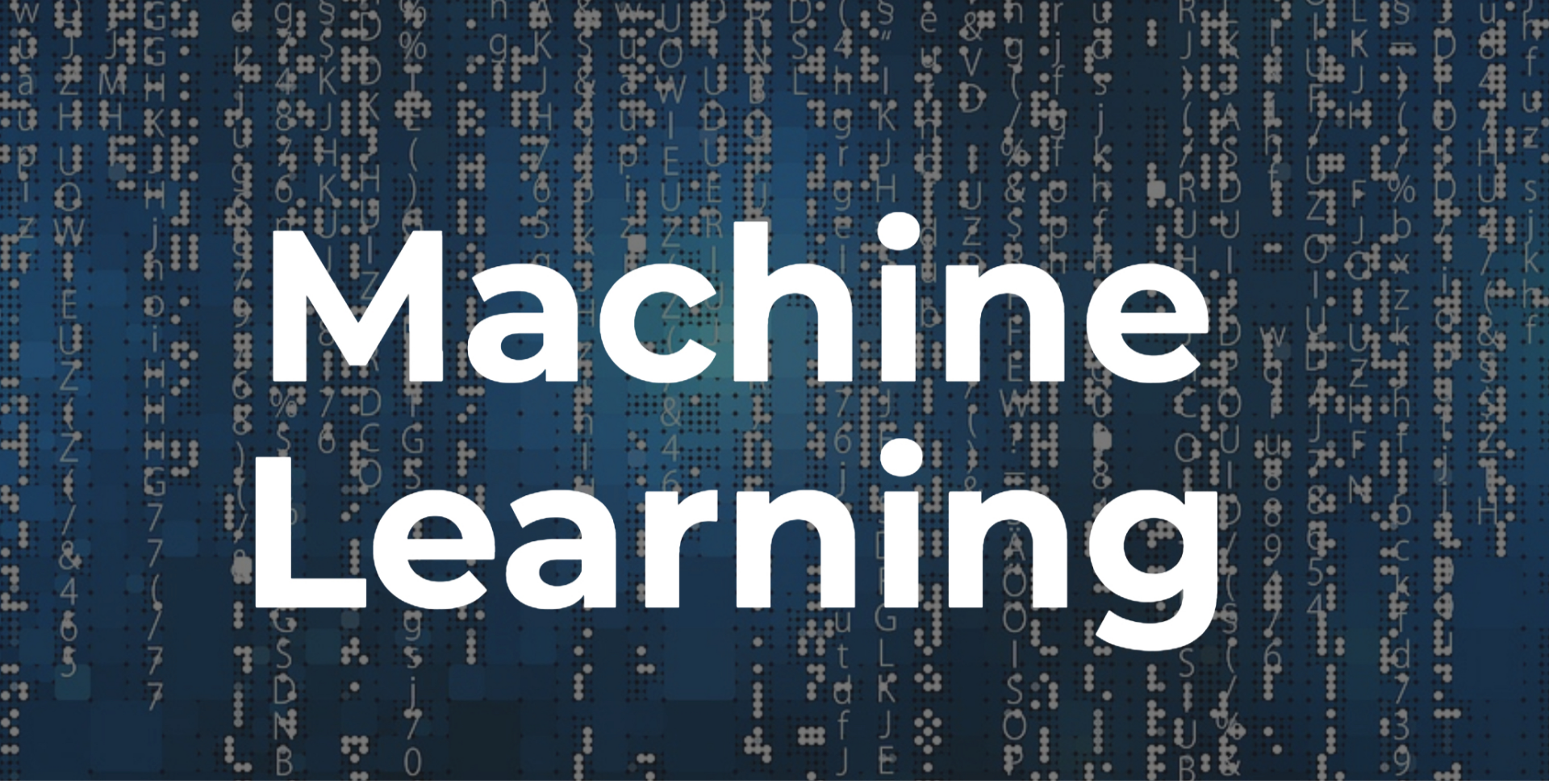


CSCI 5622 Fall 2020

Prof. Claire Monteleoni

Today

- Intro. to Learning Theory
  - Standard probabilistic analysis tools
  - Generalization, Complexity
  - [If time] Structural Risk minimization

Bounding Random Variables

- Markov Inequality
- Chebyshev Inequality

Use the mean, and possibly the variance, to provide bounds on the probabilities of certain events.

Markov Inequality

If X is a random variable that can only take nonnegative values, then:

$$P(X \geq a) \leq \frac{E[X]}{a}$$

Interpretation: If a nonnegative random variable has a small mean, then the probability it takes a large value is small.

Chebyshev Inequality

If X is a random variable with mean μ and var σ^2, then for all $c > 0$:

$$P(|X - \mu| \geq c) \leq \frac{\sigma^2}{c^2}$$

Interpretation: If a random variable has small variance, then the probability it takes a value far from its mean is small.

Probability amplification

- We saw methods to take a statement that holds in expectation and get a statement that holds with high probability: i.e., with probability $\geq 1 - \delta$.
- There are a variety of inequalities for doing so, known as concentration inequalities, e.g.
  - Markov Inequality
  - Chebyshev Inequality
  - Chernoff bounds, etc.
- This is an example of probability amplification. More generally, there are techniques in which the original statement just needs to holds with constant probability.

The Union Bound

Given any probability space, and any events

A_1, …, A_n defined on the space:

$$P(A_1 \cup A_n \cup \cdots \cup A_n) = P\left(\bigcup_{i=1}^{n} A_i\right) \leq \sum_{i=1}^{n} P(A_i)$$

Interpretation: a (possibly loose) upper bound on the probability of **any** of the events, A_i, occurring.

- Often used to bound the probability of any "bad" event occurring.
- The events, A_i, need not be independent!

Classification problems

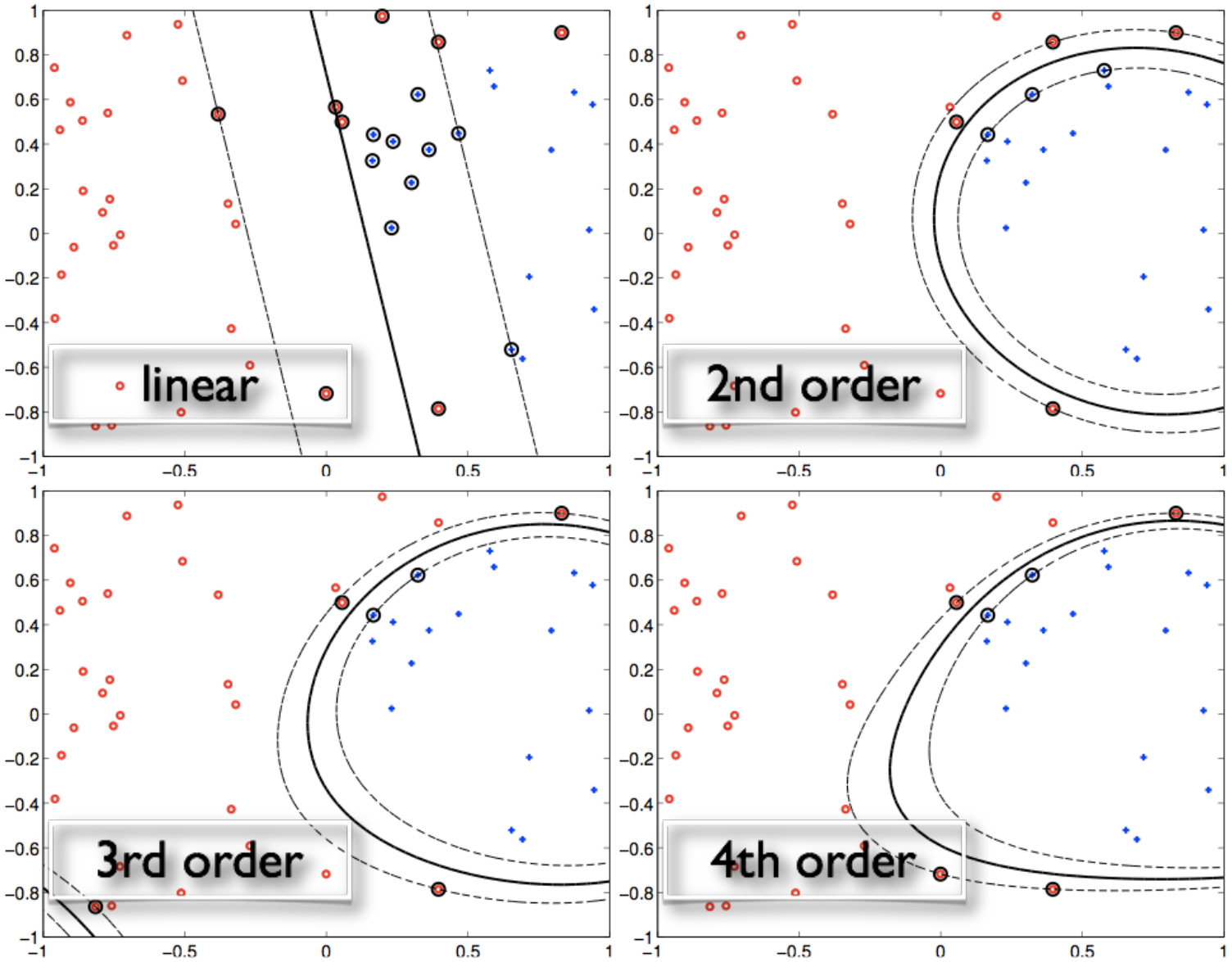
- There are two parts to any classification task

  1) **Estimation:** how to select the best classifier out of a particular set (e.g., linear)

  2) **Model selection:** how to select the best set of classifiers (e.g., degree of polynomial kernel)

- Both of these "selections" have to be made on the basis of the training set of examples and labels.

Which model?



VC-dimension, generalization

- (Vapnik 1979) With probability at least $1-\delta$ over the choice of the training set

$$R(f) \leq R_n(f) + \sqrt{\frac{\log N_{\mathcal{F}}(2n) + \log(4/\delta)}{n}}, \quad \forall f \in \mathcal{F}$$

where $\log N_{\mathcal{F}}(2n) \leq d_{vc}(\log(2n/d_{vc})+1)$ when $n \geq d_{vc}$ and d_{vc} is the VC-dimension of $\mathcal{F}$.

VC-dimension, generalization

- (Vapnik 1979) With probability at least $1-\delta$ over the choice of the training set

$$R(f) \leq R_n(f) + \sqrt{\frac{\log N_{\mathcal{F}}(2n) + \log(4/\delta)}{n}}, \quad \forall f \in \mathcal{F}$$

  where $\log N_{\mathcal{F}}(2n) \leq d_{vc}(\log(2n/d_{vc})+1)$ when $n \geq d_{vc}$ and d_{vc} is the VC-dimension of $\mathcal{F}$.

- If $d_{vc} = \infty$,
  bound on the test error becomes vacuous

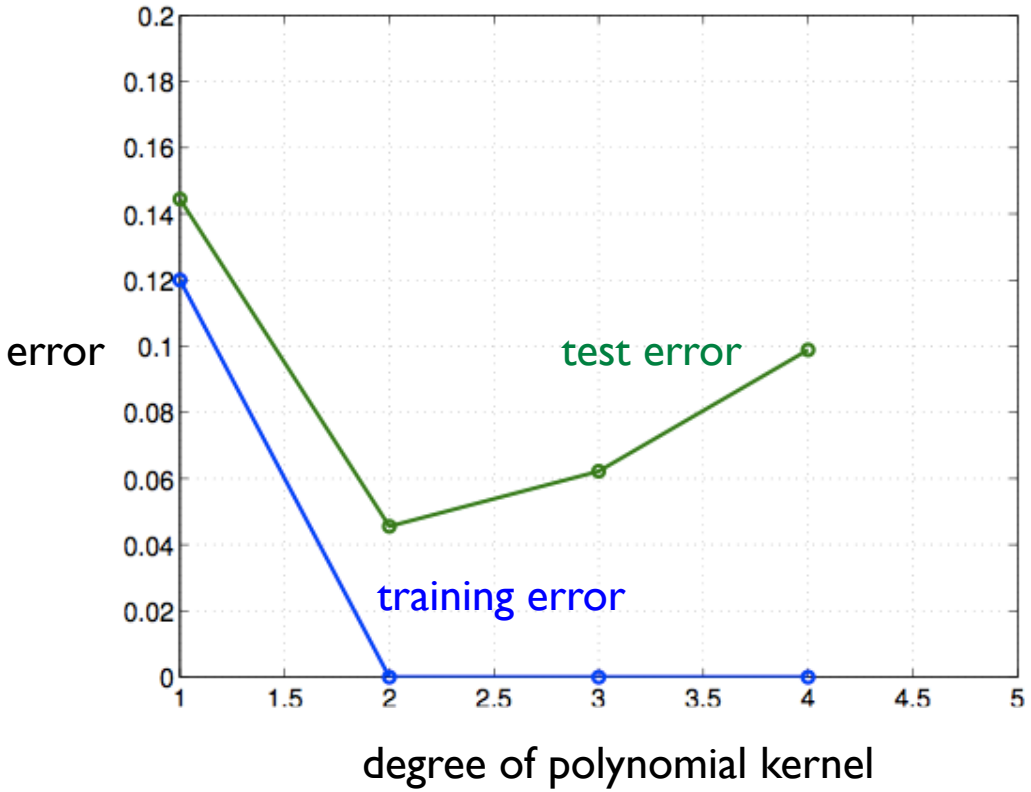
This is called Structural Risk Minimization (SRM)

A note on terminology

- When we mention "training error," here we mean the 0-1 error on the training set (or on the test set, during training).

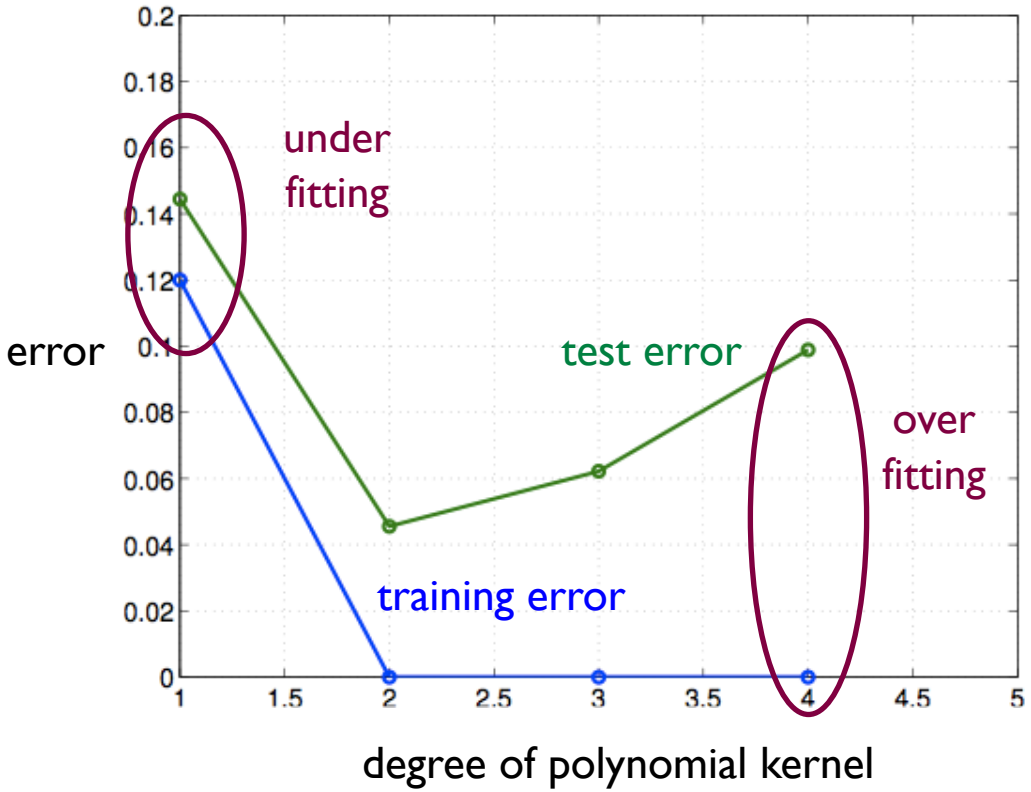
- When we mention "test error," here we mean the 0-1 error on future unseen points drawn i.i.d. from the same distribution as the data we trained on.
  - Other terms for this are "true error," or "generalization error"

Model selection



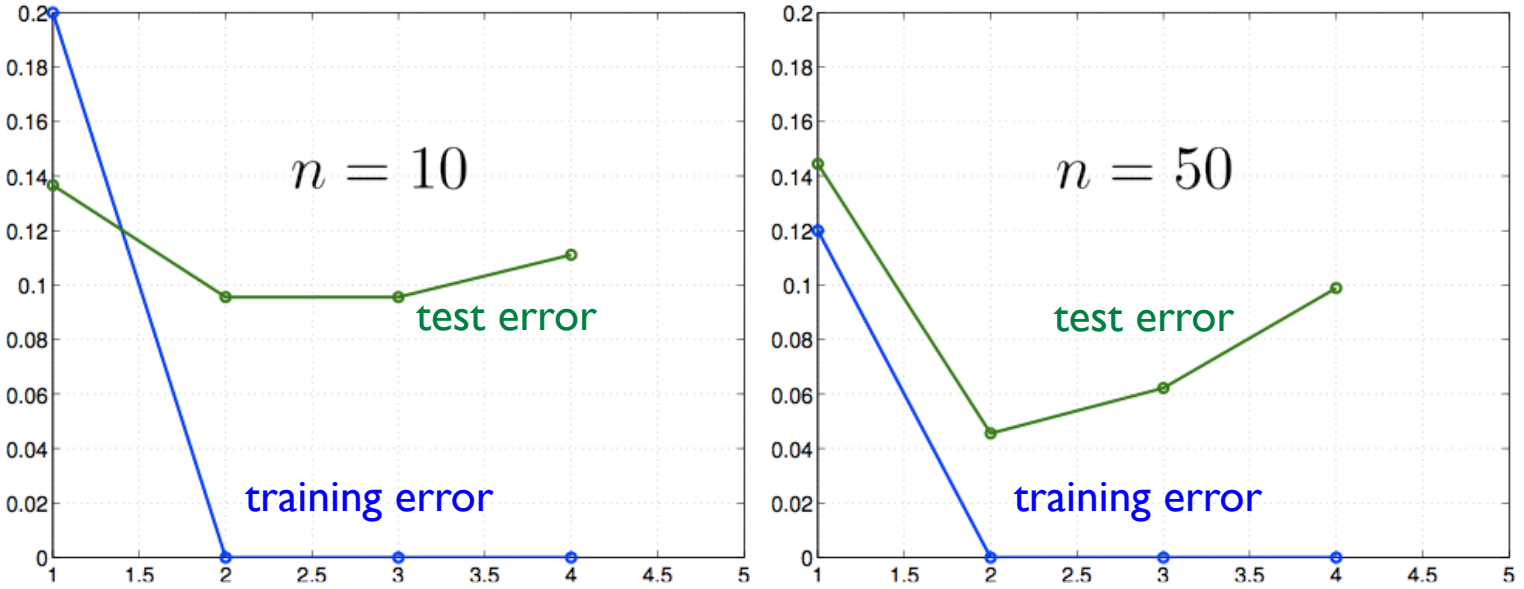
- There would be no model selection problem if we could evaluate the test error of any classifier (i.e., have access to the distribution $P(x, y)$ generating test examples)

Model selection



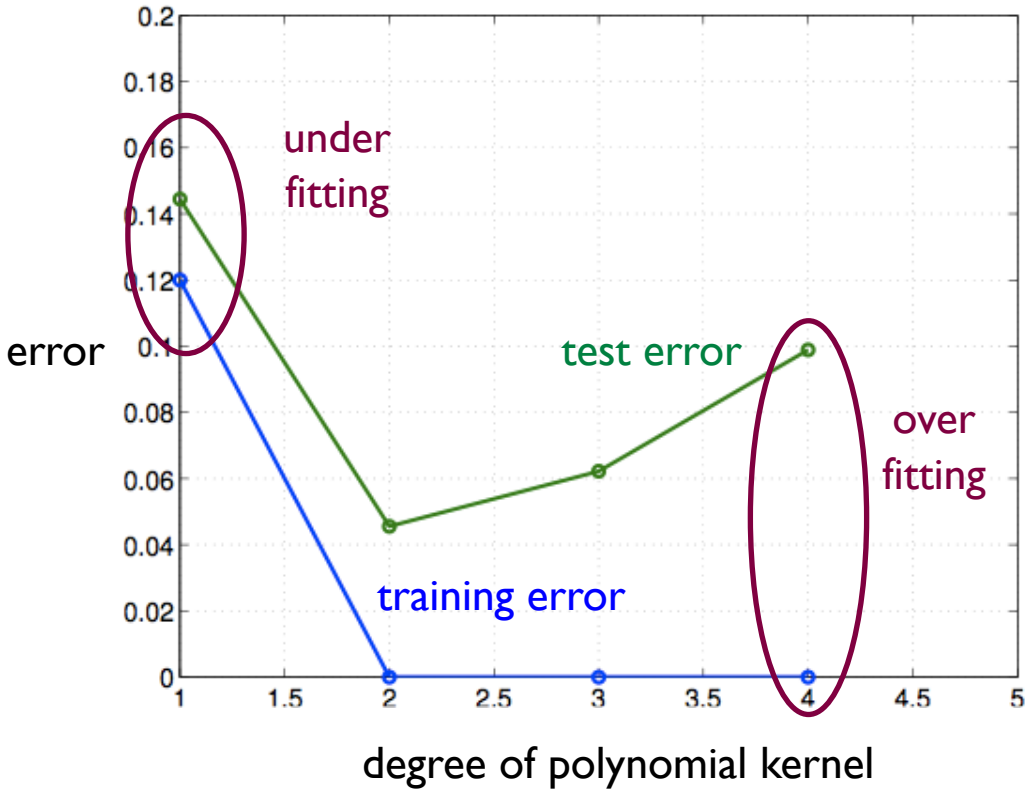
- We need to understand better 1) the classifiers we consider, 2) the errors we compute, 3) the gap between training and test errors

Model selection (n)



- The training (test) errors are random variables due to the random choice of the training set (and therefore the estimated classifier)
- We can only hope to provide a probabilistic statement about the gap between training and test errors as a function of n, the size of the training set

Model selection



- We need to understand better 1) the classifiers we consider, 2) the errors we compute, 3) the gap between training and test errors as a function of n (probabilistically)

Error of a *single* classifier

Let S be some hold-out validation set, that is NOT the training set (for learning h).

For any classifier h and underlying distribution D on X x Y:

"true error" $\mathrm{err}(h) = \mathbf{P}_{(x,y)\sim D}(h(x) \neq y)$

"error on set S" $\mathrm{err}(h, S) = \frac{1}{|S|} \sum_{(x,y)\in S} \mathbf{1}(h(x) \neq y)$

Suppose S is chosen i.i.d. (independent, identically distributed) from D. Then (over the random choices of S),

$$\mathbf{E}[\mathrm{err}(h, S)] = \mathrm{err}(h)$$

And the standard deviation of err(h,S) is abou$1/\sqrt{|S|}$

Error of a *single* classifier

Fix any h. Suppose S is chosen i.i.d. (independent, identically distributed) from D. Then (over the random choices of S),

$$\mathbf{E}[\mathrm{err}(h, S)] = \mathrm{err}(h)$$

And the standard deviation of err(h,S) is about $1/\sqrt{|S|}$

(i) In this scenario, S is used to assess the accuracy of a single, prespecified classifier h.

Q: Can the same S be used to check many classifiers simultaneously?

A: VC theory gives complexity of a whole model class.

(ii) In particular, if h was created using S as a training set, then the above scenario does not apply. In such situations, err(h,S) might be a very poor estimate of err(h).

What is a "model"?

- Any kernel function such as $K(\underline{x}, \underline{x}') = \underline{\phi}(\underline{x}) \cdot \underline{\phi}(\underline{x}')$ defines a set of linear classifiers in the feature space

$$\mathcal{F} = \left\{ f(\cdot) \ : \ f(\underline{x}) = \text{sign}(\underline{\theta} \cdot \underline{\phi}(\underline{x})) \ \text{ for some } \underline{\theta} \in \mathcal{R}^d \right\}$$

  (we omit the offset parameter here for notational simplicity)

- The "model" corresponding to K is this set of classifiers
- It is often easier to talk about the model as a set of discriminant functions

$$\mathcal{F} = \left\{ f(\cdot) \ : \ f(\underline{x}) = \underline{\theta} \cdot \underline{\phi}(\underline{x}) \ \text{ for some } \underline{\theta} \in \mathcal{R}^d \right\}$$

  (we will refer to a model as a set of discriminant functions unless indicated otherwise)

Training error

- The classification error (empirical risk) of any $f \in \mathcal{F}$ on the training set $S_n = \{(\underline{x}_i, y_i)\}_{i=1,\ldots,n}$ is

$$R_n(f) = \frac{1}{n} \sum_{i=1}^{n} \mathrm{Loss}_{0-1}\big(y_i f(\underline{x}_i)\big)$$

  where $\mathrm{Loss}_{0-1}(z) = 1$ if $z \leq 0$ and 0 otherwise.

Training error

- The classification error (empirical risk) of any $f \in \mathcal{F}$ on the training set $S_n = \{(\underline{x}_i, y_i)\}_{i=1,\ldots,n}$ is

$$R_n(f) = \frac{1}{n} \sum_{i=1}^{n} \mathrm{Loss}_{0-1}\big(y_i f(\underline{x}_i)\big)$$

  where $\mathrm{Loss}_{0-1}(z) = 1$ if $z \leq 0$ and 0 otherwise.

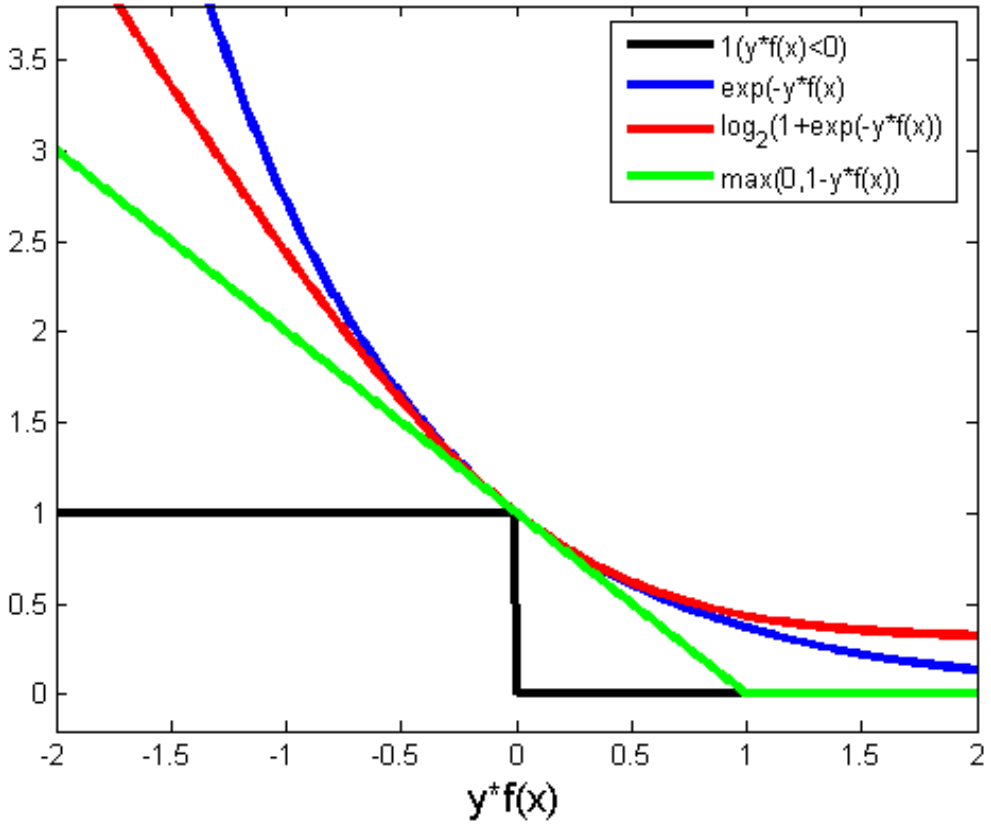
- Note that we do not estimate $\hat{f}$ from the training set by minimizing the empirical risk. Instead, we find $\hat{f}(\underline{x}) = \underline{\hat{\theta}} \cdot \underline{\phi}(\underline{x})$ by minimizing a convex surrogate

$$J(\underline{\theta}) = \frac{1}{n} \sum_{i=1}^{n} \mathrm{Loss}\big(y_i \, \underline{\theta} \cdot \underline{\phi}(\underline{x}_i) \big) + \frac{\lambda}{2} \|\underline{\theta}\|^2$$

  where the loss is, e.g., the Hinge loss (SVM).

Loss functions recap

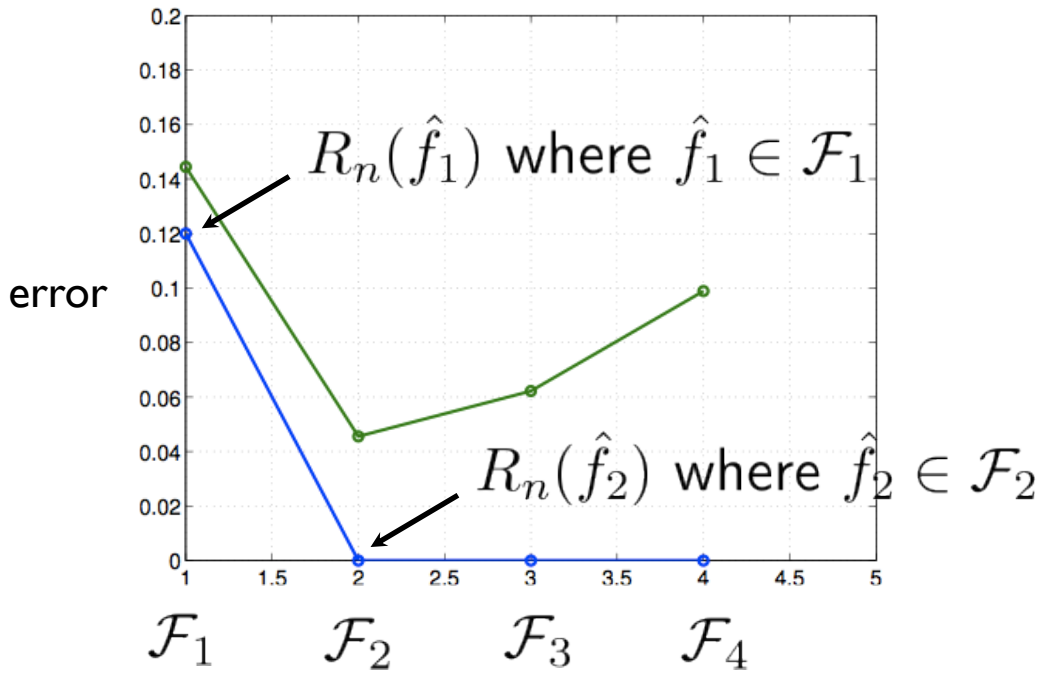
- Loss functions
  - The 0-1 loss is in black
  - Loss functions in color are convex upper bounds on 0-1 loss



- SVM optimizes the Hinge Loss (green)

Training error

training error: $R_n(f) = \frac{1}{n}\sum_{i=1}^{n} \text{Loss}_{0-1}\big(y_i f(\underline{x}_i)\big)$



Key assumption, test error

- The training set is drawn at random from some underlying distribution $P(\underline{x}, y)$ over examples and labels

$$S_n = \{(\underline{x}_i, y_i)\}_{i=1,\ldots,n}, \quad (\underline{x}_i, y_i) \sim P$$

- The test examples are drawn *from the same distribution*

Key assumption, test error

- The training set is drawn at random from some underlying distribution $P(\underline{x}, y)$ over examples and labels

$$S_n = \{(\underline{x}_i, y_i)\}_{i=1,\ldots,n}, \quad (\underline{x}_i, y_i) \sim P$$

- The test examples are drawn *from the same distribution*
- The test error (risk) of any $f \in \mathcal{F}$ is

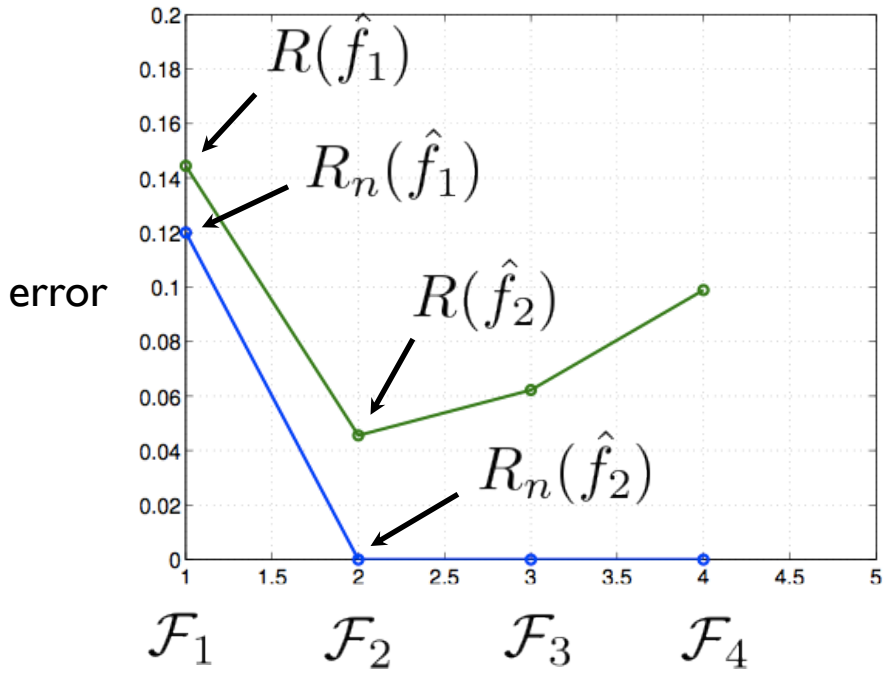
$$R(f) = E_{(\underline{x},y)\sim P}\left\{ \text{Loss}_{0-1}\left(yf(\underline{x}) \right) \right\}$$

The expectation is with respect to the distribution *P*

Training and test errors

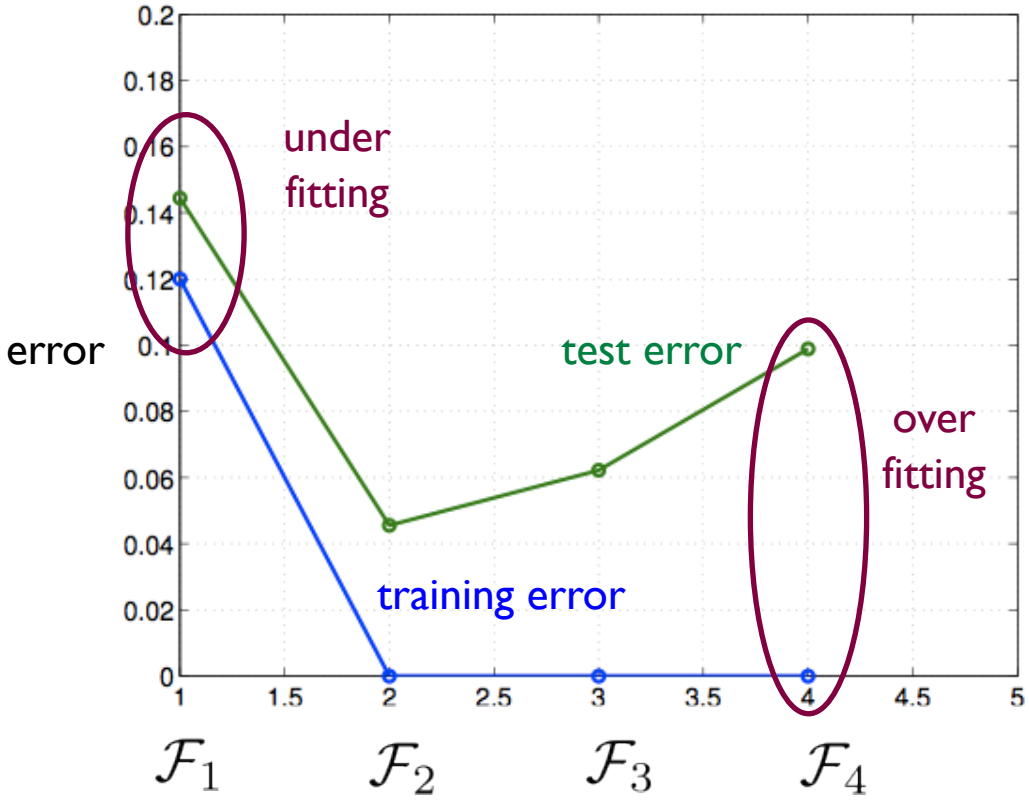
training error: $R_n(f) = \frac{1}{n}\sum_{i=1}^{n} \text{Loss}_{0-1}\big(y_i f(\underline{x}_i)\big)$

test error: $R(f) = E_{(\underline{x},y)\sim P}\left\{ \text{Loss}_{0-1}\big(yf(\underline{x}) \big) \right\}$



- Note that both $R_n(\hat{f}_1)$ and $R(\hat{f}_1)$ are random variables because of the training set [$\hat{f}_1$ is a function of the training set]

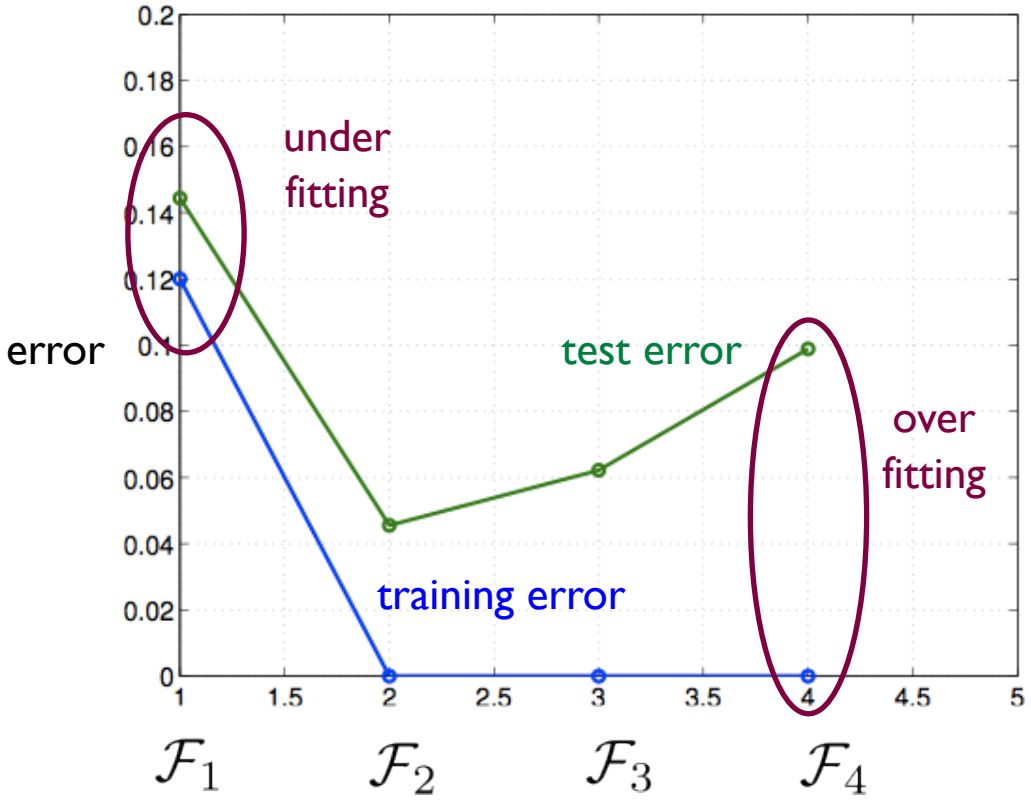
Model selection



- We need to understand how far the training and test errors can be as a function of n for any given model $\mathcal{F}$, i.e.,

we want to upper bound: $\left|R_n(\hat{f}) - R(\hat{f})\right|$

Model selection



- We need to understand how far the training and test errors can be as a function of n for any given model $\mathcal{F}$, i.e.,

$$\left| R_n(\hat{f}) - R(\hat{f}) \right| \leq \sup_{f \in \mathcal{F}} |R_n(f) - R(f)|$$

Model selection, generalization

- We would like to show that for a large enough n, with high probability over the choice of the training set,

$$\sup_{f \in \mathcal{F}} |R_n(f) - R(f)| \leq \epsilon$$

Model selection, generalization

- We would like to show that for a large enough n, with high probability over the choice of the training set,

$$\sup_{f \in \mathcal{F}} |R_n(f) - R(f)| \leq \epsilon$$

- Such a result ties together

  $n =$ size of the training set,

  $\epsilon =$ gap between training and test errors,

  $|\mathcal{F}| =$ "size" of the model $\mathcal{F}$

  $1 - \delta =$ the probability that the result holds

- We will start with the case where $\mathcal{F}$ is finite (there are only a finite number of classifiers in the set)

Model selection, generalization

- For any finite $\mathcal{F}$, $|\mathcal{F}| < \infty$, with probability at least $1 - \delta$ over the choice of the training set,

$$\sup_{f \in \mathcal{F}} |R_n(f) - R(f)| \leq \sqrt{\frac{\log |\mathcal{F}| + \log(2/\delta)}{2n}} = \epsilon(n, \delta, |\mathcal{F}|)$$

Model selection, generalization

- For any finite $\mathcal{F}$, $|\mathcal{F}| < \infty$, with probability at least $1 - \delta$ over the choice of the training set,

$$\sup_{f \in \mathcal{F}} |R_n(f) - R(f)| \leq \sqrt{\frac{\log |\mathcal{F}| + \log(2/\delta)}{2n}} = \epsilon(n, \delta, |\mathcal{F}|)$$

- So, in particular, for any $\hat{f} \in F$ selected on the basis of the training set

$$R(\hat{f}) \leq R_n(\hat{f}) + \epsilon(n, \delta, |\mathcal{F}|)$$

Model selection, generalization

- For any finite $\mathcal{F}$, $|\mathcal{F}| < \infty$, with probability at least $1 - \delta$ over the choice of the training set,

$$\sup_{f \in \mathcal{F}} |R_n(f) - R(f)| \leq \sqrt{\frac{\log |\mathcal{F}| + \log(2/\delta)}{2n}} = \epsilon(n, \delta, |\mathcal{F}|)$$

- So, in particular, for any $\hat{f} \in F$ selected on the basis of the training set

$$R(\hat{f}) \leq R_n(\hat{f}) + \epsilon(n, \delta, |\mathcal{F}|)$$

- We can use this generalization guarantee as a basis for model selection: select the model with the best guarantee of generalization (structural risk minimization)

Finite case

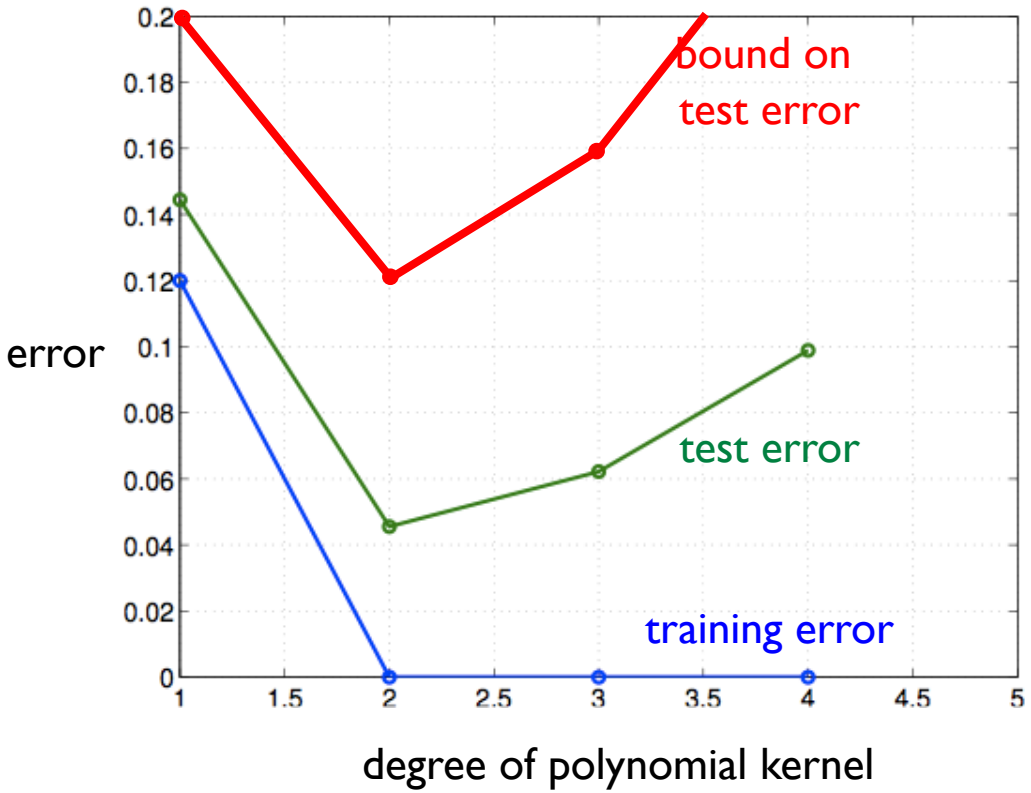
- For any finite $\mathcal{F}$, $|\mathcal{F}| < \infty$, with probability at least $1 - \delta$ over the choice of the training set,

$$R(f) \leq R_n(f) + \sqrt{\frac{\log |\mathcal{F}| + \log(2/\delta)}{2n}}, \quad \forall f \in \mathcal{F}$$

test error | training error | gap between training and test errors (complexity penalty)

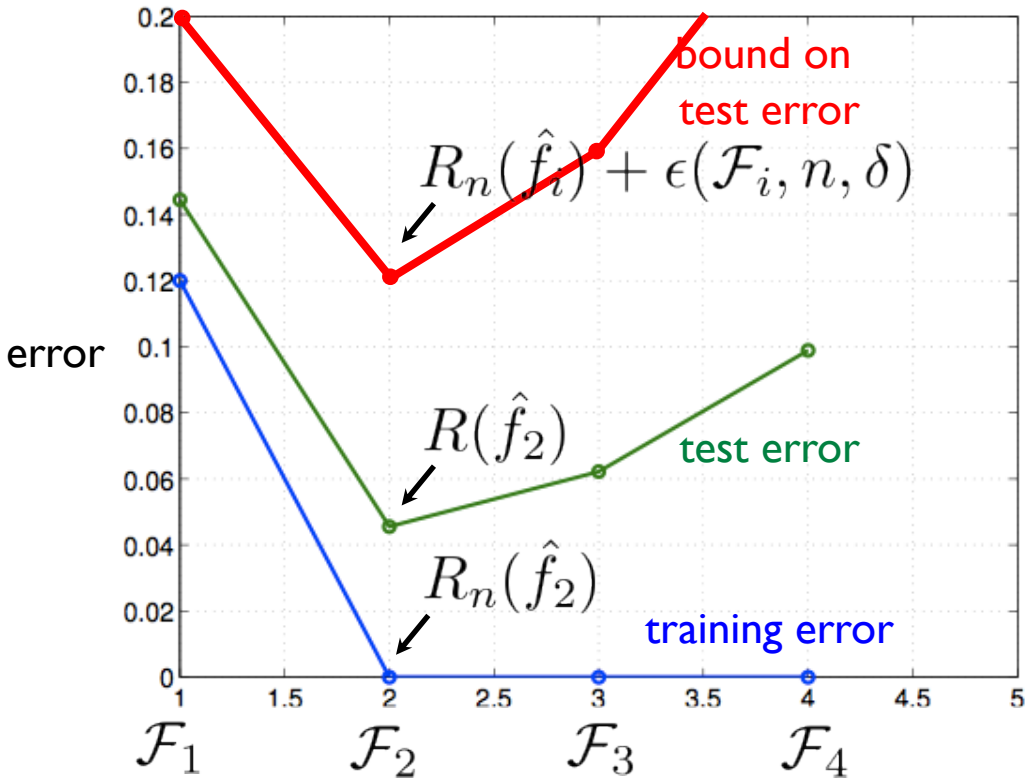
- The result holds, in particular, for $\hat{f} \in \mathcal{F}$ selected on the basis of the training set
- We can use these generalization results to select the model $\mathcal{F}$ with the best guarantee of generalization

Model selection



- Each polynomial degree kernel corresponds to a set of discriminant functions $\mathcal{F}_i$ (model)
- We try to bound the test (generalization) error of each estimated classifier $\hat{f}_i \in \mathcal{F}_i$

Model selection



- We select the model for which we can give the best guarantee of test (generalization) error (with high probability)

$$R(\hat{f}_i) \leq R_n(\hat{f}_i) + \epsilon(\mathcal{F}_i, n, \delta)$$

test error — training error — gap between training and test errors (complexity penalty)

Beyond the finite case

- Even the simple set of linear classifiers corresponds to an uncountable set of discriminant functions ($|\mathcal{F}| = \infty$)
- We need to quantify how "powerful" the set of classifiers is that we are considering

  1) How many ways we can label any set of n points using classifiers in the set (leads to VC-dimension)

  2) How much we deviate from a prior measure over functions (PAC-Bayesian analysis)

Classifiers and labelings

- Consider a set of n training examples. We can label these examples in different ways by applying discriminant functions $f \in \mathcal{F}$ in our model

$\underline{x}_1$	$\underline{x}_2$	$\dots$	$\underline{x}_n$
$f(\underline{x}_1)$	$f(\underline{x}_2)$	$\dots$	$f(\underline{x}_n)$
$f'(\underline{x}_1)$	$f'(\underline{x}_2)$	$\dots$	$f'(\underline{x}_n)$
$\dots$	$\dots$	$\dots$	$\dots$

$\Rightarrow$

$\underline{x}_1$	$\underline{x}_2$	$\dots$	$\underline{x}_n$
$+$	$-$	$\dots$	$+$
$+$	$-$	$\dots$	$-$
$\dots$	$\dots$	$\dots$	$\dots$

Classifiers and labelings

- Consider a set of n training examples. We can label these examples in different ways by applying discriminant functions $f \in \mathcal{F}$ in our model

$\underline{x}_1$	$\underline{x}_2$	$\dots$	$\underline{x}_n$
$f(\underline{x}_1)$	$f(\underline{x}_2)$	$\dots$	$f(\underline{x}_n)$
$f'(\underline{x}_1)$	$f'(\underline{x}_2)$	$\dots$	$f'(\underline{x}_n)$
$\dots$	$\dots$	$\dots$	$\dots$

$\Rightarrow$

$\underline{x}_1$	$\underline{x}_2$	$\dots$	$\underline{x}_n$	
$+$	$-$	$\dots$	$+$	
$+$	$-$	$\dots$	$-$	a labeling
$\dots$	$\dots$	$\dots$	$\dots$	

Classifiers and labelings

- Consider a set of n training examples. We can label these examples in different ways by applying discriminant functions $f \in \mathcal{F}$ in our model

$\underline{x}_1$	$\underline{x}_2$	$\cdots$	$\underline{x}_n$
$f(\underline{x}_1)$	$f(\underline{x}_2)$	$\cdots$	$f(\underline{x}_n)$
$f'(\underline{x}_1)$	$f'(\underline{x}_2)$	$\cdots$	$f'(\underline{x}_n)$
$\cdots$	$\cdots$	$\cdots$	$\cdots$

$\Rightarrow$

$\underline{x}_1$	$\underline{x}_2$	$\cdots$	$\underline{x}_n$	
$+$	$-$	$\cdots$	$+$	
$+$	$-$	$\cdots$	$-$	a labeling
$\cdots$	$\cdots$	$\cdots$	$\cdots$	

- Depending on $\mathcal{F}$, we may be able to generate a large number of distinct labelings in this way (but never more than 2^n)

$$N_{\mathcal{F}}(\underline{x}_1, \ldots, \underline{x}_n) = \# \text{ of distinct labelings}$$

Classifiers and labelings

- Consider a set of n training examples. We can label these examples in different ways by applying discriminant functions $f \in \mathcal{F}$ in our model

$\underline{x}_1$	$\underline{x}_2$	$\cdots$	$\underline{x}_n$
$f(\underline{x}_1)$	$f(\underline{x}_2)$	$\cdots$	$f(\underline{x}_n)$
$f'(\underline{x}_1)$	$f'(\underline{x}_2)$	$\cdots$	$f'(\underline{x}_n)$
$\cdots$	$\cdots$	$\cdots$	$\cdots$

$\Rightarrow$

$\underline{x}_1$	$\underline{x}_2$	$\cdots$	$\underline{x}_n$
$+$	$-$	$\cdots$	$+$
$+$	$-$	$\cdots$	$-$
$\cdots$	$\cdots$	$\cdots$	$\cdots$

a labeling

- Depending on $\mathcal{F}$, we may be able to generate a large number of distinct labelings in this way (but never more than 2^n)

$$N_{\mathcal{F}}(\underline{x}_1, \ldots, \underline{x}_n) = \# \text{ of distinct labelings}$$

- The largest number of distinct labelings we can obtain for any set of n examples is known as the growth function of $\mathcal{F}$

$$N_{\mathcal{F}}(n) = \max_{\underline{x}_1, \ldots, \underline{x}_n} N_{\mathcal{F}}(\underline{x}_1, \ldots, \underline{x}_n)$$

Growth function, generalization

- (Vapnik 1979) With probability at least $1-\delta$ over the choice of the training set

$$R(f) \leq R_n(f) + \sqrt{\frac{\log N_{\mathcal{F}}(2n) + \log(4/\delta)}{n}}, \quad \forall f \in \mathcal{F}$$

- So, essentially, $\log |\mathcal{F}|$ in the finite case is replaced by the logarithm of the growth function (number of labelings that the classifiers can produce)

Growth function, VC-dimension

- The number of data points that can be classified in all possible ways (shattered) by $\mathcal{F}$ is called the Vapnik-Chervonenkis (VC) dimension of the set $\mathcal{F}$

$$d_{vc} = \max \left\{ n \ : \ N_F(n) = 2^n \right\}$$

Growth function, VC-dimension

- The number of data points that can be classified in all possible ways (shattered) by $\mathcal{F}$ is called the Vapnik-Chervonenkis (VC) dimension of the set $\mathcal{F}$: d_{vc}

Recall that by Sauer's Lemma:

- Classifiers undergo a sharp transition from an exponentially increasing number of labelings to a polynomially increasing number as n increases past the VC-dimension , d_{vc}

$$N_F(n) = \begin{cases} 2^n, & n \leq d_{vc} \\ \leq \left(\frac{ne}{d_{vc}}\right)^{d_{vc}}, & \text{if } n > d_{vc} \end{cases}$$

VC-dimension, generalization

- (Vapnik 1979) With probability at least $1-\delta$ over the choice of the training set

$$R(f) \leq R_n(f) + \sqrt{\frac{\log N_{\mathcal{F}}(2n) + \log(4/\delta)}{n}}, \quad \forall f \in \mathcal{F}$$

where $\log N_{\mathcal{F}}(2n) \leq d_{vc}(\log(2n/d_{vc})+1)$ when $n \geq d_{vc}$ and d_{vc} is the VC-dimension of $\mathcal{F}$.

VC-dimension, generalization

- (Vapnik 1979) With probability at least $1-\delta$ over the choice of the training set

$$R(f) \leq R_n(f) + \sqrt{\frac{\log N_{\mathcal{F}}(2n) + \log(4/\delta)}{n}}, \quad \forall f \in \mathcal{F}$$

  where $\log N_{\mathcal{F}}(2n) \leq d_{vc}(\log(2n/d_{vc})+1)$ when $n \geq d_{vc}$ and d_{vc} is the VC-dimension of $\mathcal{F}$.

- If $d_{vc} = \infty$,
  bound on the test error becomes vacuous