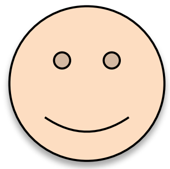
Ridge Regression

Ryota Tomioka
Department of Mathematical Informatics
The University of Tokyo

About this class

- Bad news: This class will be in English.
- Good news: The topic "ridge regression" is probably already familiar to you.



- Even better news: if you ask a question in English during the class, then you don't need to hand in any assignment (no report) for this class.

Of course you can still ask questions in Japanese but you have to hand in your assignment as usual.

Why English?

- Number of speakers? No!
  - Chinese (mandarin) 845 million
  - Spanish 329 million
  - English 328 million
  - …
- Let’s compare “Gamma distribution” in Wikipedia

English for non-native speakers

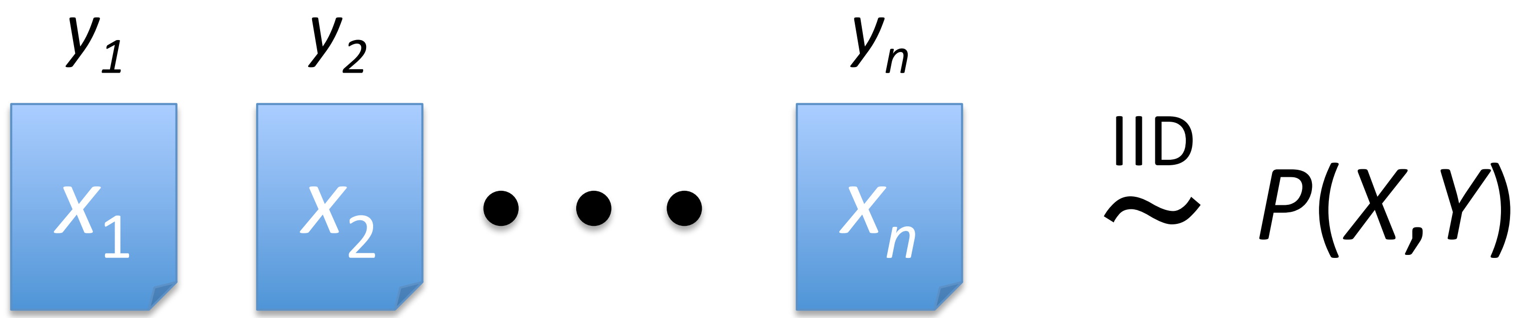
- It is the number of readers.
- Japanese/Spanish/Chinese Wikipedia
  - Read mostly by native speakers
- English Wikipedia
  - Read by many non-native speakers
- English is the best language to express your ideas, inventions, research.
- Nobody speaks (or writes) perfect English
  - The world is full of bad English (but who cares)

Outline

- Ridge Regression (regularized linear regression)
  - Formulation
  - Handling Nonlinearity using basis functions
  - Classification
  - Multi-class classification
- Singularity — the dark side of RR
  - Why does it happen?
  - How can we avoid it?
- Summary

Problem Setting

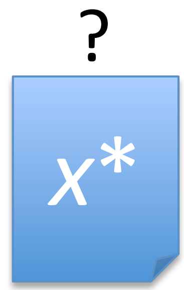
- Training examples: (x_i, y_i) $(i=1,\ldots, n)$, $x_i \in R^p$



- Goal
  - Learn a linear function

$$f(x^*) = w^{\mathsf{T}} x^* \quad (w \in R^p)$$

that predicts the output y^* for a test point



$$(x^*, y^*) \sim P(X,Y)$$

- Note that the test point is not included in the traning examples (We want generalization!)

Ridge Regression

- Solve the minimization problem

$$\underset{\boldsymbol{w}}{\text{minimize}} \quad \underbrace{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|^2}_{\text{Training error}} + \underbrace{\lambda\|\boldsymbol{w}\|^2}_{\text{Regularization (ridge) term}}$$

Training error

Regularization (ridge) term
(λ: regularization const.)

Target output

$$\boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

Design matrix

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1{}^\top \\ \boldsymbol{x}_2{}^\top \\ \vdots \\ \boldsymbol{x}_n{}^\top \end{pmatrix}$$

Note: Can be interpreted as a Maximum A Posteriori (MAP) estimation
– Gaussian likelihood with Gaussian prior.

Designing the design matrix

- Columns of X can be different sources of info
  - e.g., predicting the price of an apartment

$$\boldsymbol{X} = \begin{pmatrix} \text{Size} & \text{\#rooms} & \text{Bathroom} & \text{Sunlight} & \text{Train st.} & \text{Pet OK} \end{pmatrix}$$

- Columns of X can also be derived
  - e.g., polynomial regression

$$\boldsymbol{X} = \begin{pmatrix} x_1^{p-1} & \cdots & x_1^2 & x_1 & 1 \\ x_2^{p-1} & \cdots & x_2^2 & x_2 & 1 \\ \vdots & & & & \vdots \\ x_n^{p-1} & \cdots & x_n^2 & x_n & 1 \end{pmatrix}$$

Solving ridge regression

- Take the gradient, and solve

$$-\boldsymbol{X}^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}) + \lambda \boldsymbol{w} = 0$$

which gives

$$\boldsymbol{w} = \left(\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I}_p\right)^{-1} \boldsymbol{X}^\top \boldsymbol{y}$$

(I_p: p×p identity matrix)

The solution can also be written as (exercise)

$$\boldsymbol{w} = \boldsymbol{X}^\top \left(\boldsymbol{X}\boldsymbol{X}^\top + \lambda \boldsymbol{I}_n\right)^{-1} \boldsymbol{y}$$

Example: polynomial fitting

- Degree (p-1) polynomial model

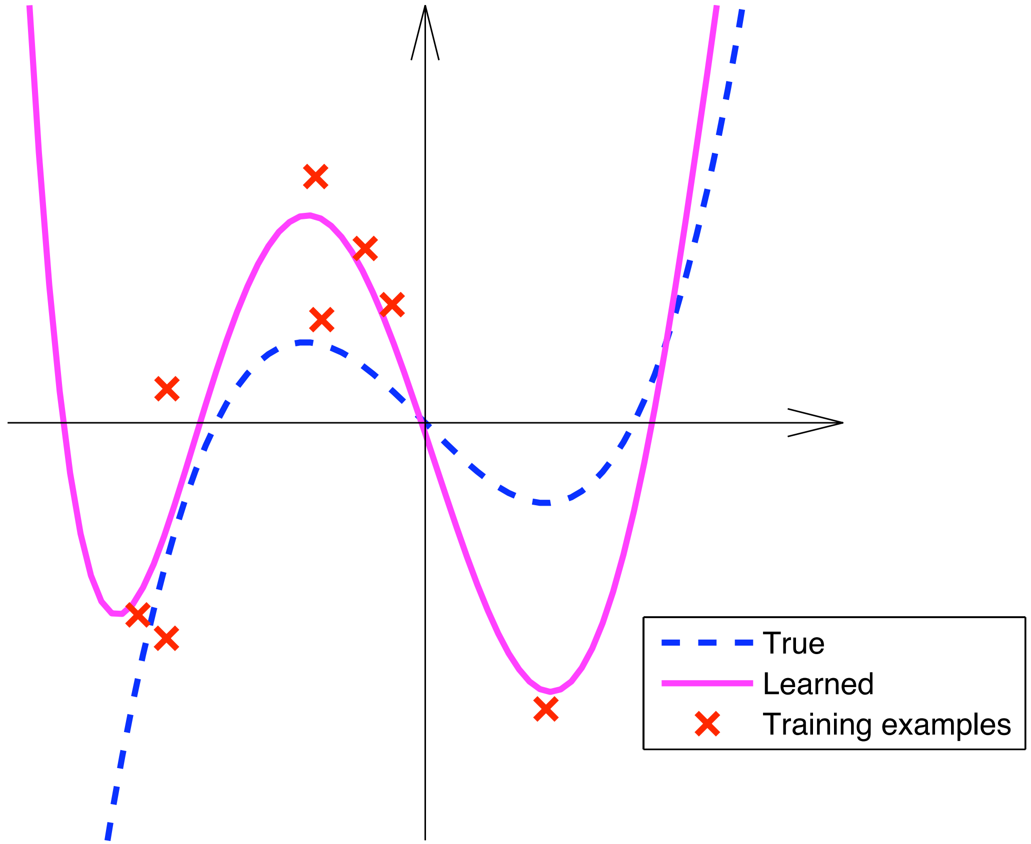
$$y = w_1 x^{p-1} + \cdots + w_{p-1} x + w_p + \text{noise}$$

$$= \begin{pmatrix} x^{p-1} & \cdots & x & 1 \end{pmatrix} \begin{pmatrix} w_1 \\ \vdots \\ w_{p-1} \\ w_p \end{pmatrix} + \text{noise}$$

Design matrix:

$$\boldsymbol{X} = \begin{pmatrix} x_1^{p-1} & \cdots & x_1^2 & x_1 & 1 \\ x_2^{p-1} & \cdots & x_2^2 & x_2 & 1 \\ \vdots & & & & \vdots \\ x_n^{p-1} & \cdots & x_n^2 & x_n & 1 \end{pmatrix}$$

Example: 5th-order polynomial fitting



True

$$\boldsymbol{w}^* = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ -1 \\ 0 \end{pmatrix}$$

Learned

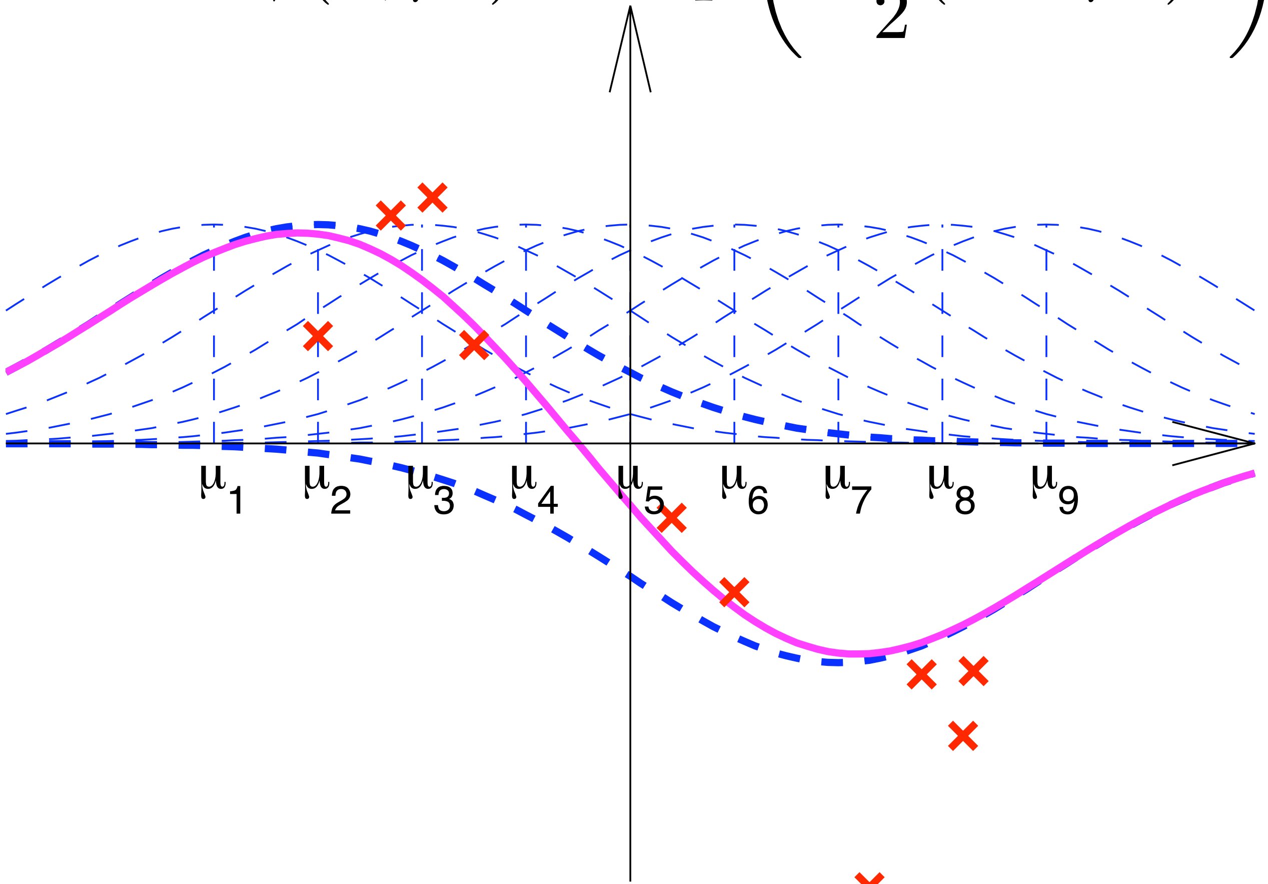
$$\boldsymbol{w} = \begin{pmatrix} -0.63 \\ 0.37 \\ 3.3 \\ -0.41 \\ -3.0 \\ -0.052 \end{pmatrix}$$

$\lambda = 0.001$

Example: RBF fitting

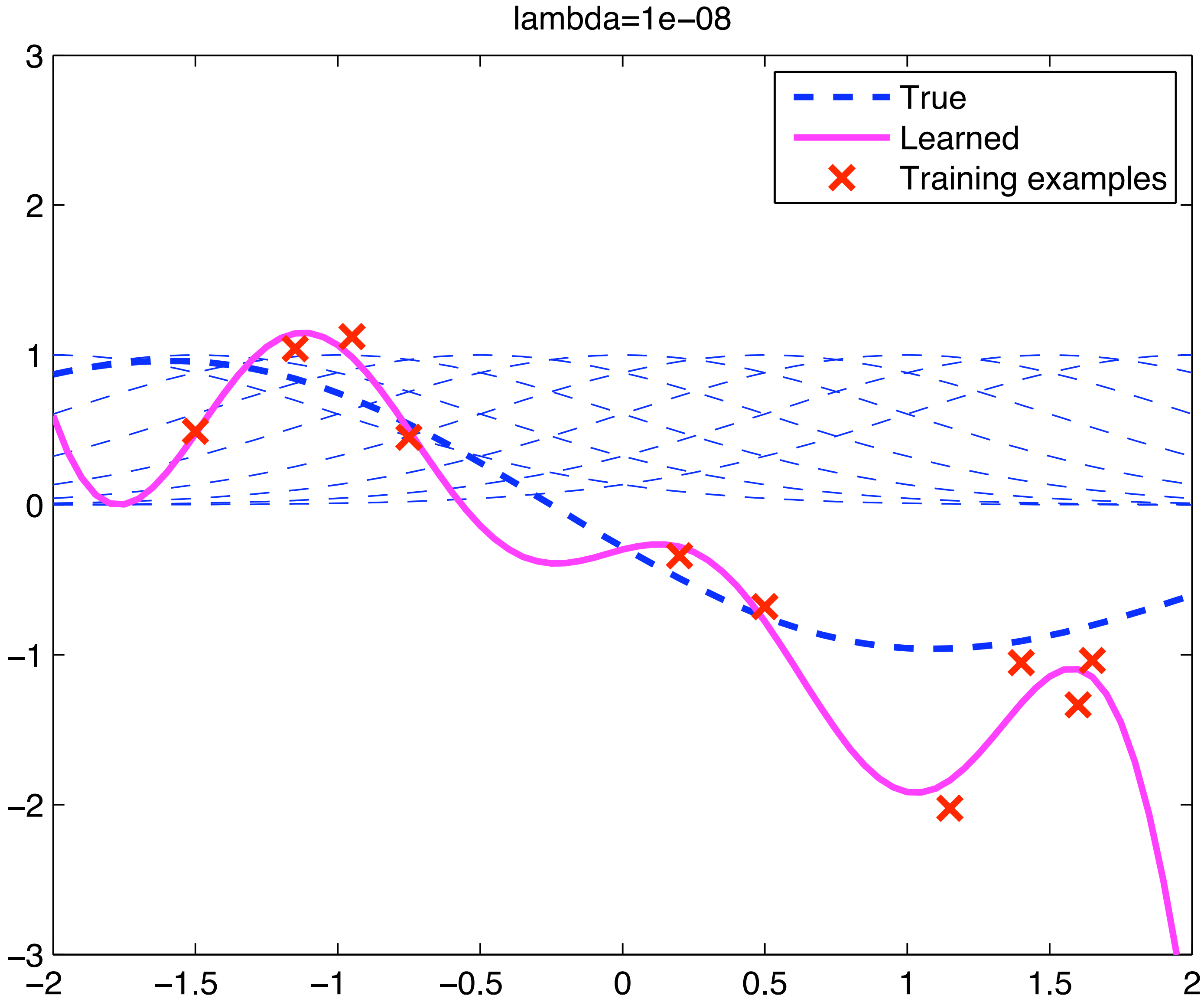
- Gaussian radial basis function (Gaussian-RBF)

$$\phi(x;\mu_c) = \exp\left(-\frac{1}{2}(x-\mu_c)^2\right)$$

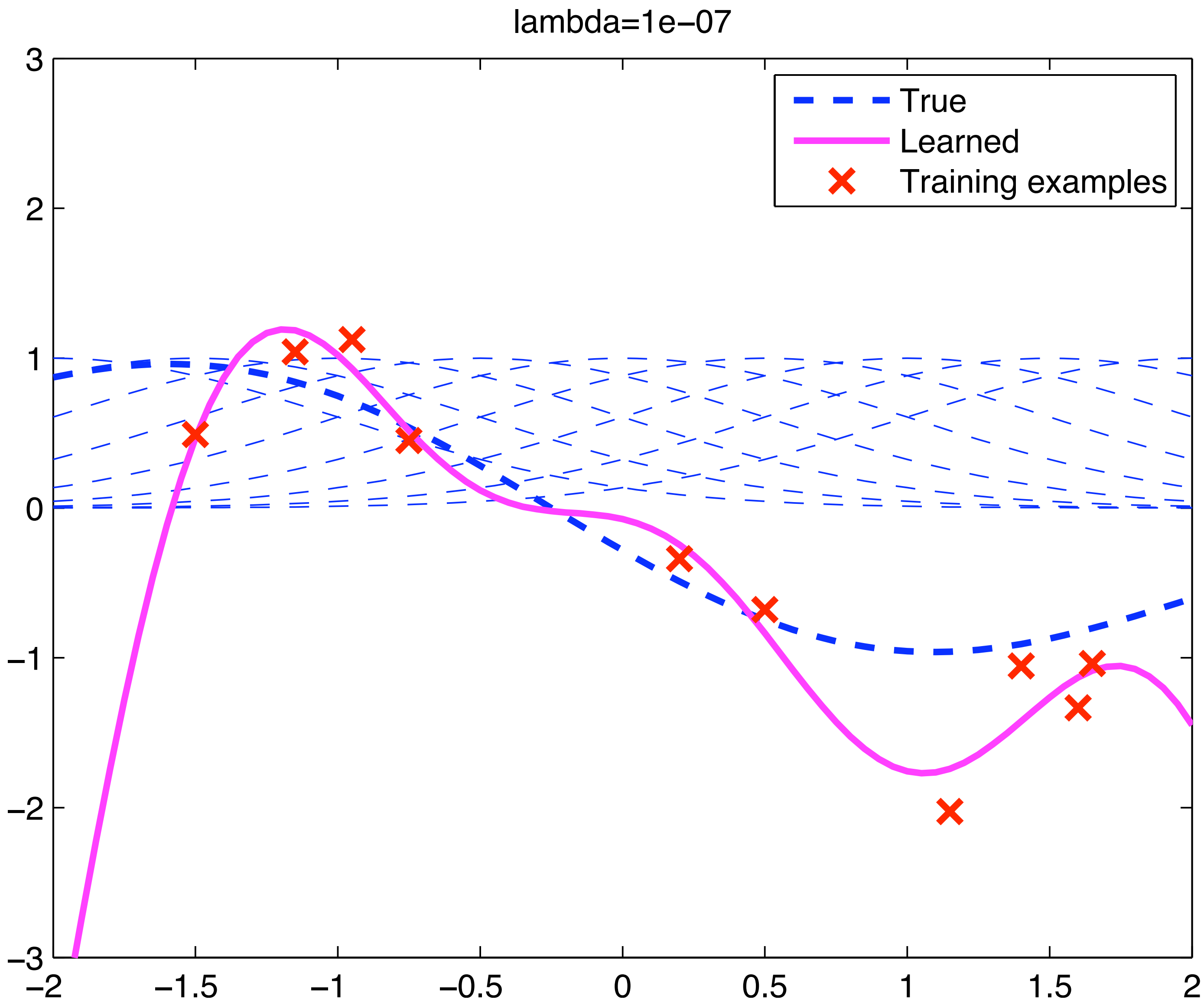


$$\boldsymbol{w}^* = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ 0 \\ 0 \end{pmatrix}$$

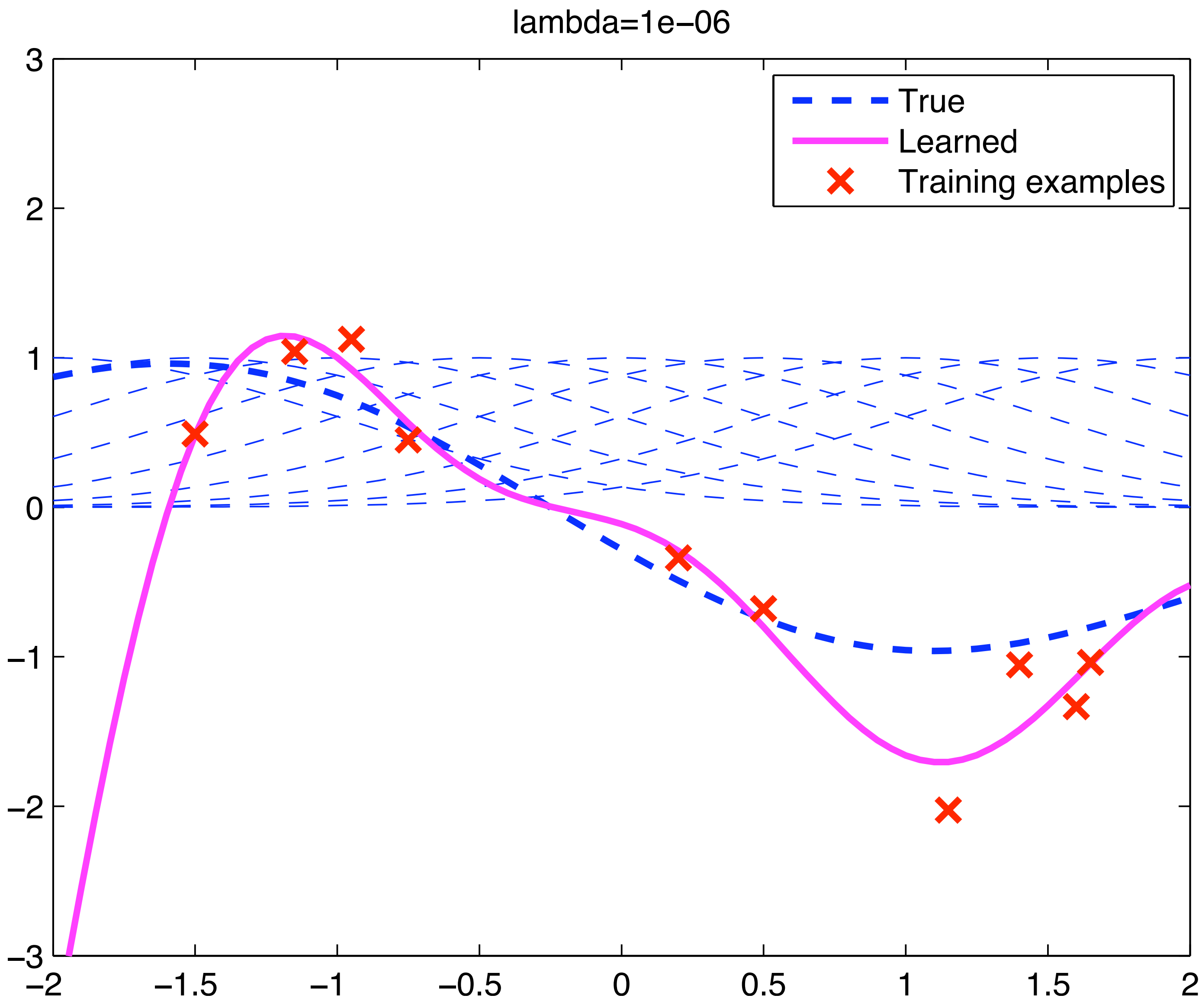
RR-RBF ($\lambda=10^{-8}$)



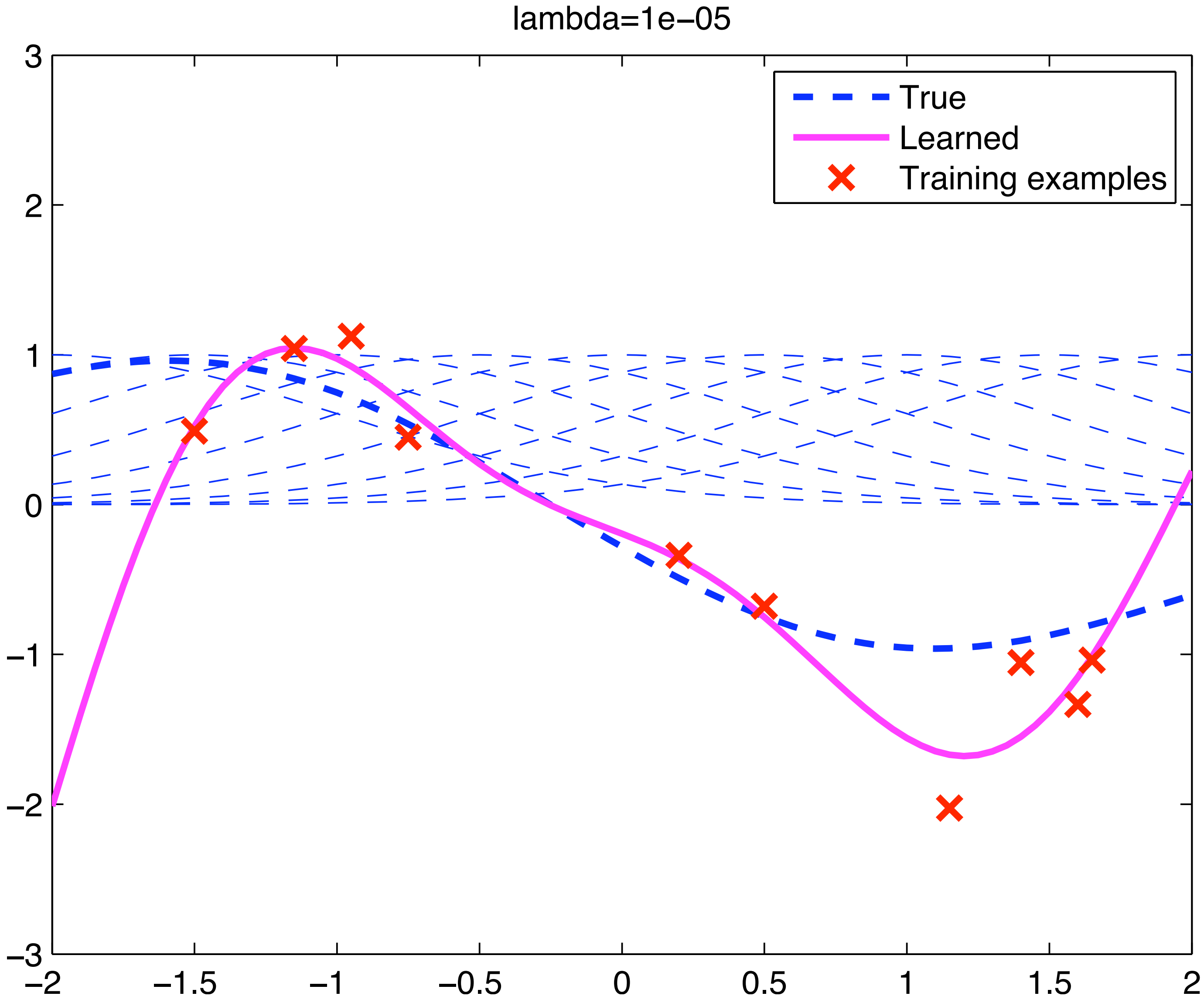
RR-RBF ($\lambda=10^{-7}$)



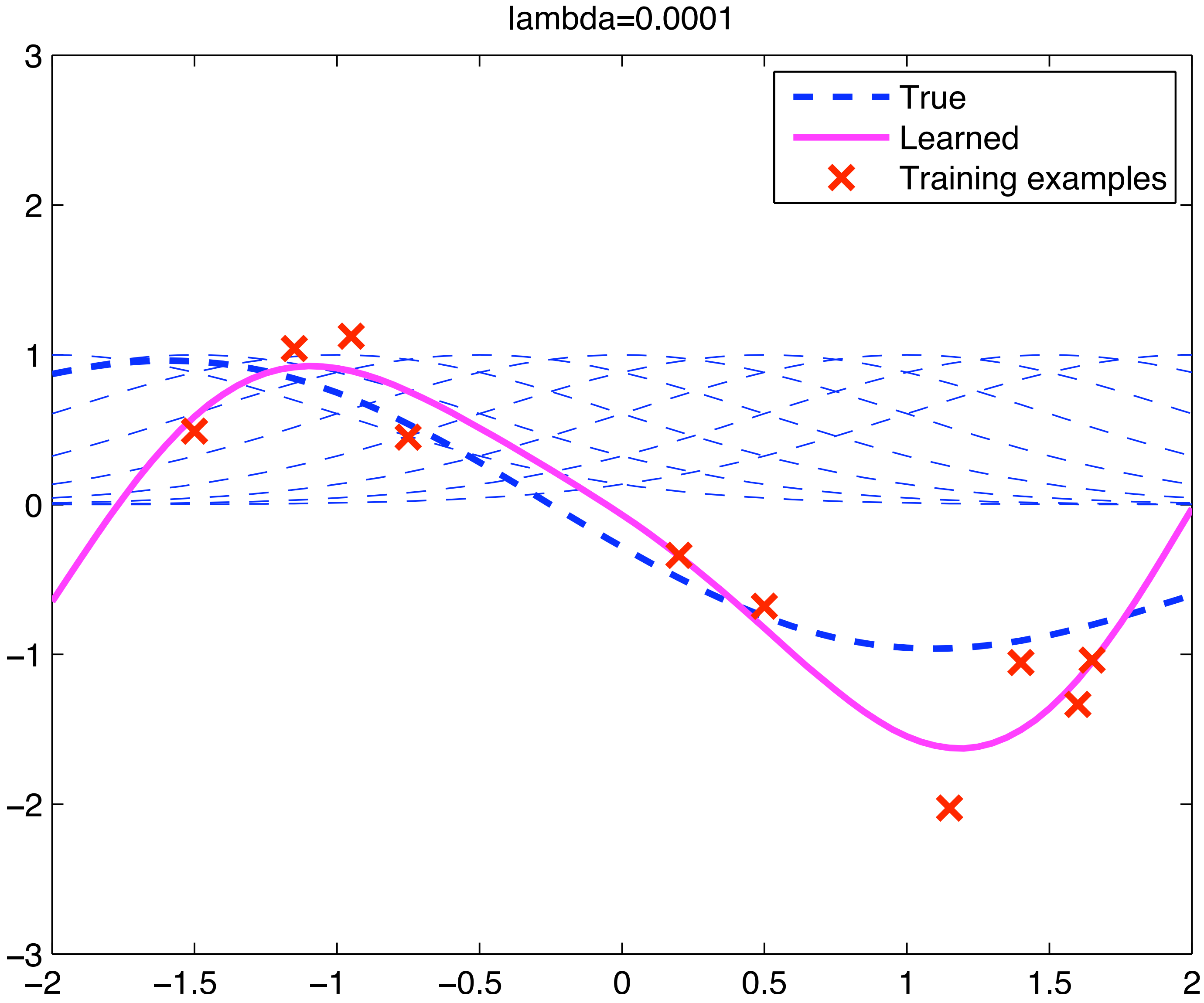
RR-RBF ($\lambda=10^{-6}$)



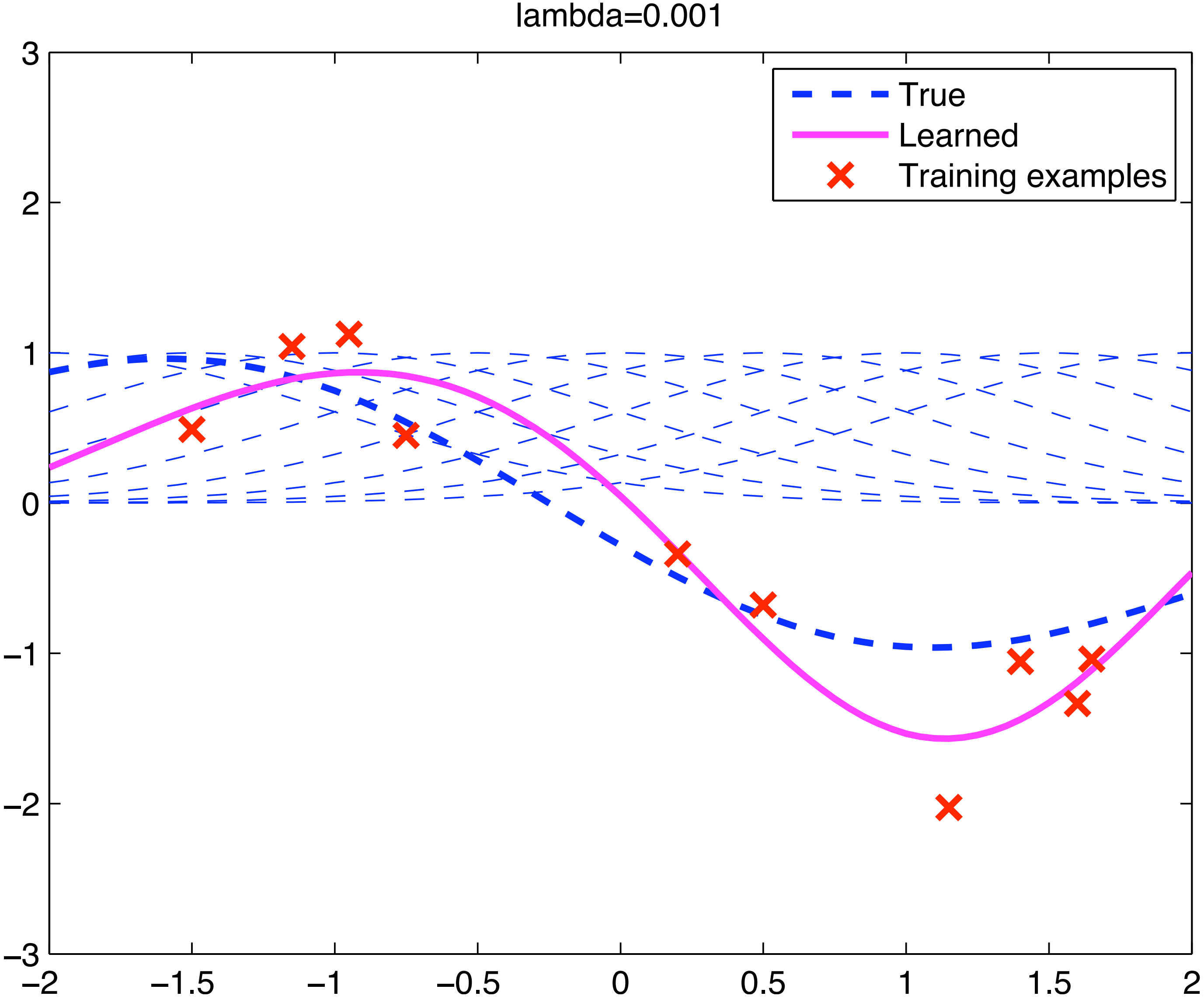
RR-RBF ($\lambda=10^{-5}$)



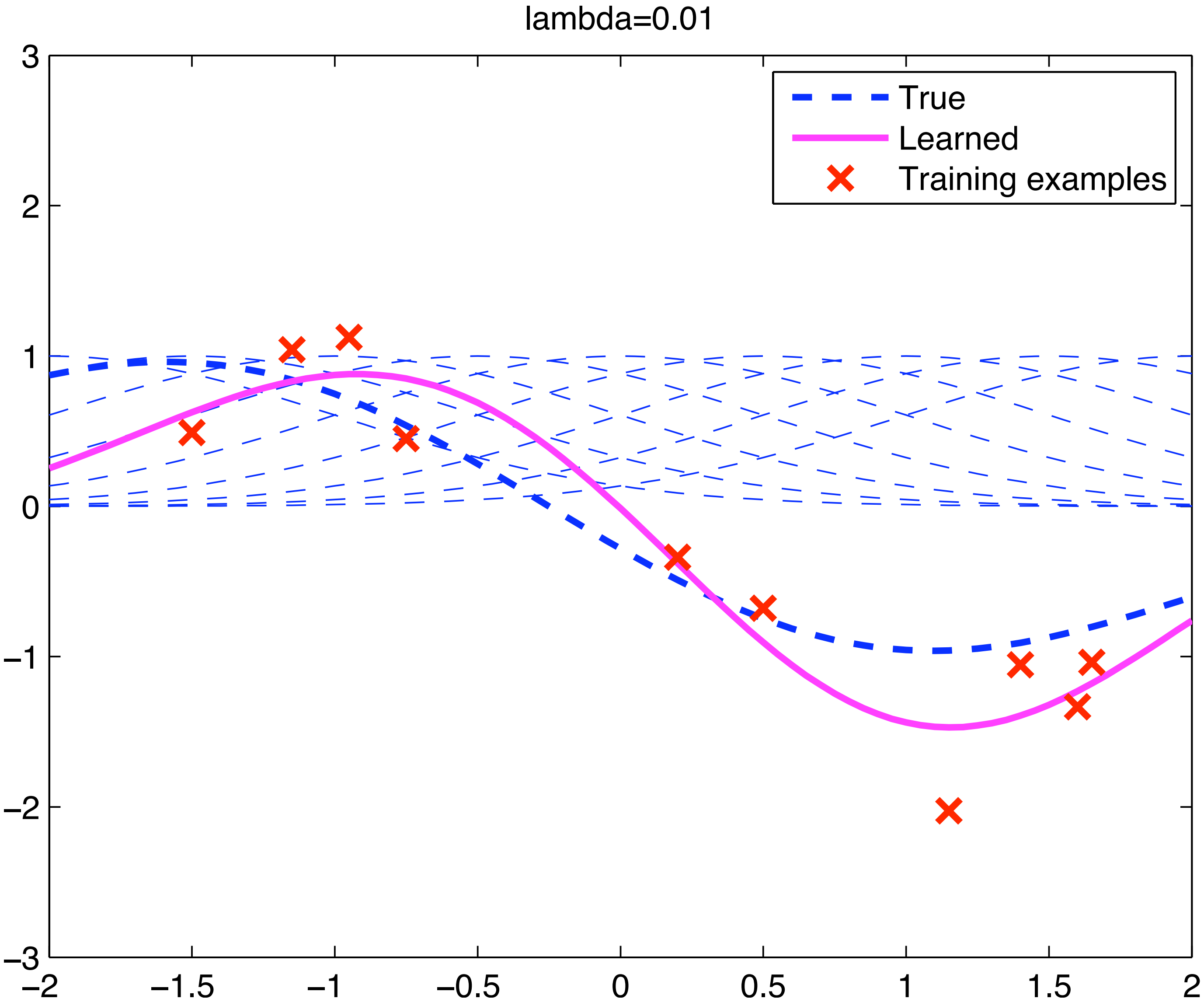
RR-RBF ($\lambda=10^{-4}$)



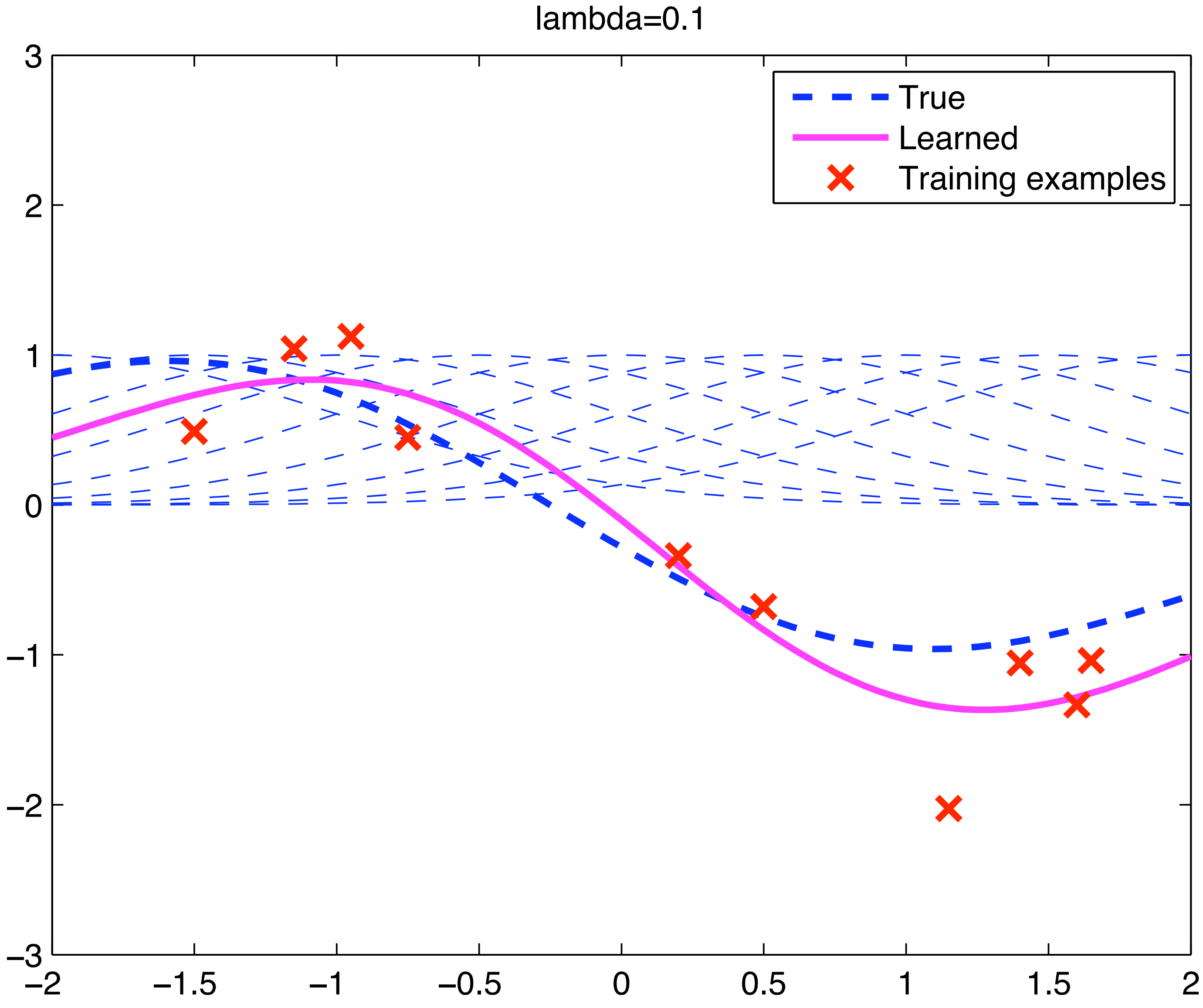
RR-RBF ($\lambda=10^{-3}$)



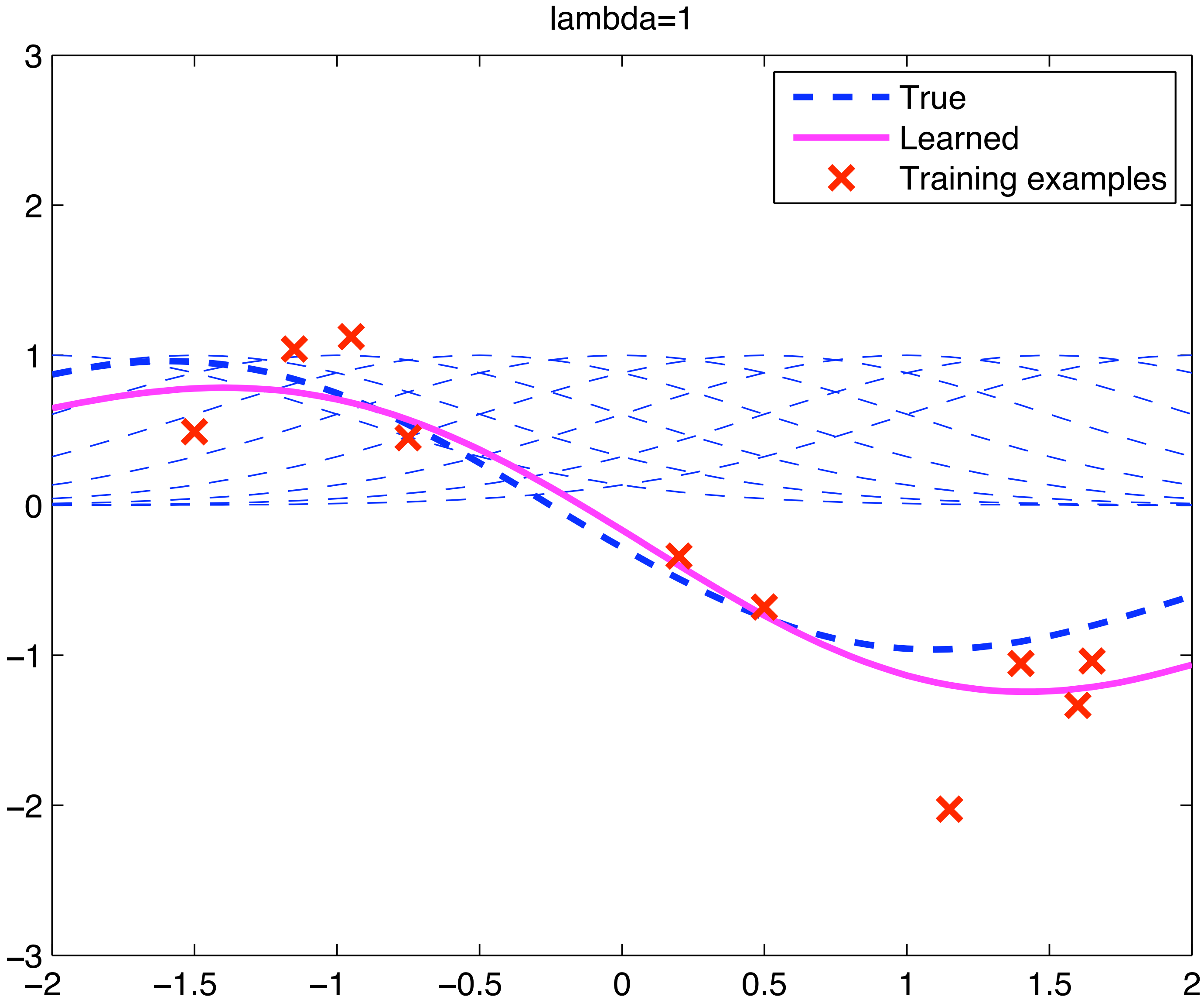
RR-RBF ($\lambda=10^{-2}$)



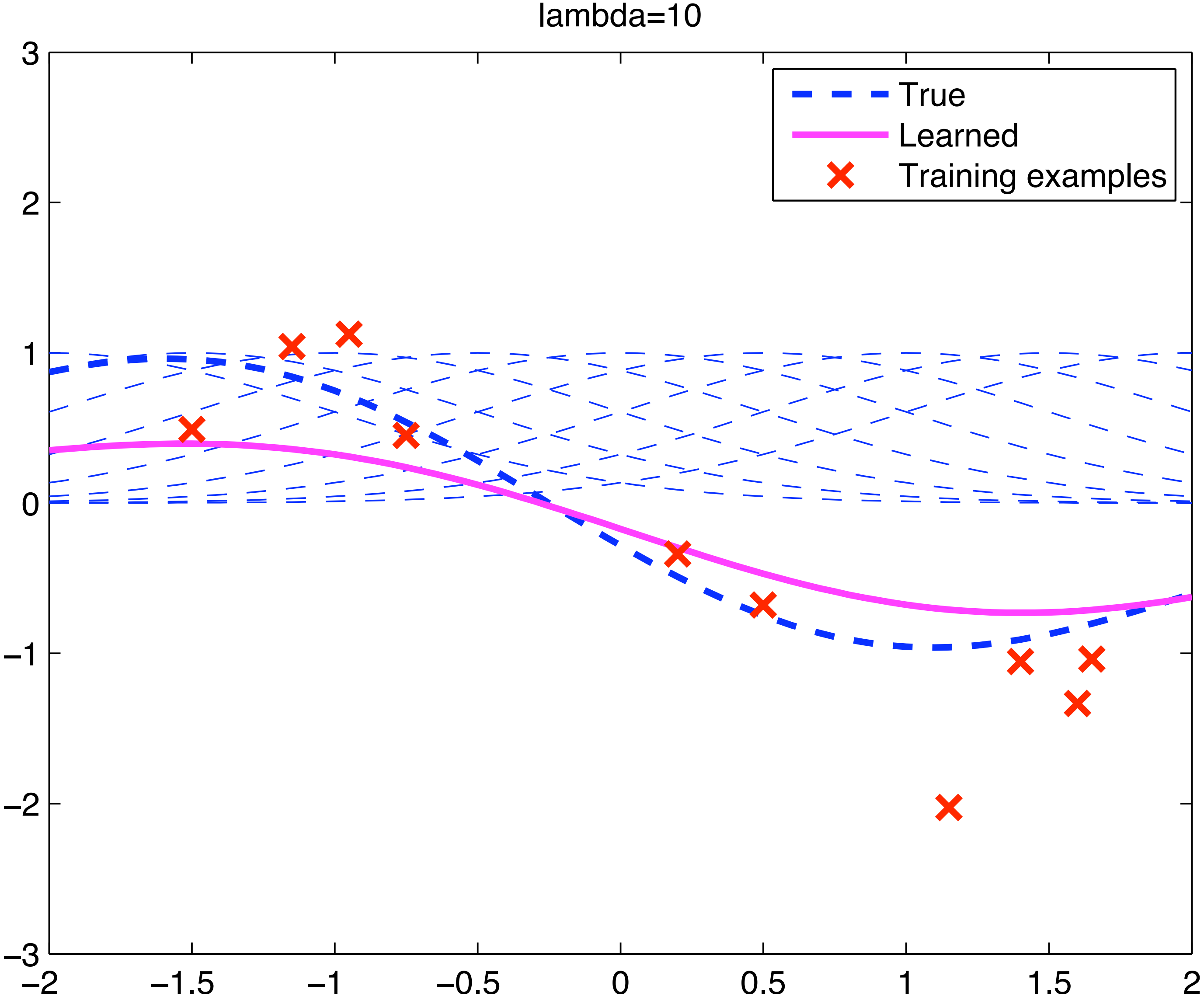
RR-RBF ($\lambda=10^{-1}$)



RR-RBF (λ=1)


lambda=1
True
Learned
Training examples
3
2
1
0
−1
−2
−3
−2
−1.5
−1
−0.5
0
0.5
1
1.5
2

RR-RBF (λ=10)



lambda=10
True
Learned
Training examples
3
2
1
0
−1
−2
−3
−2
−1.5
−1
−0.5
0
0.5
1
1.5
2

Binary classification

- Target y is +1 or -1.

Outputs
to be
predicted $\boldsymbol{y} = \begin{pmatrix} 1 \\ -1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$

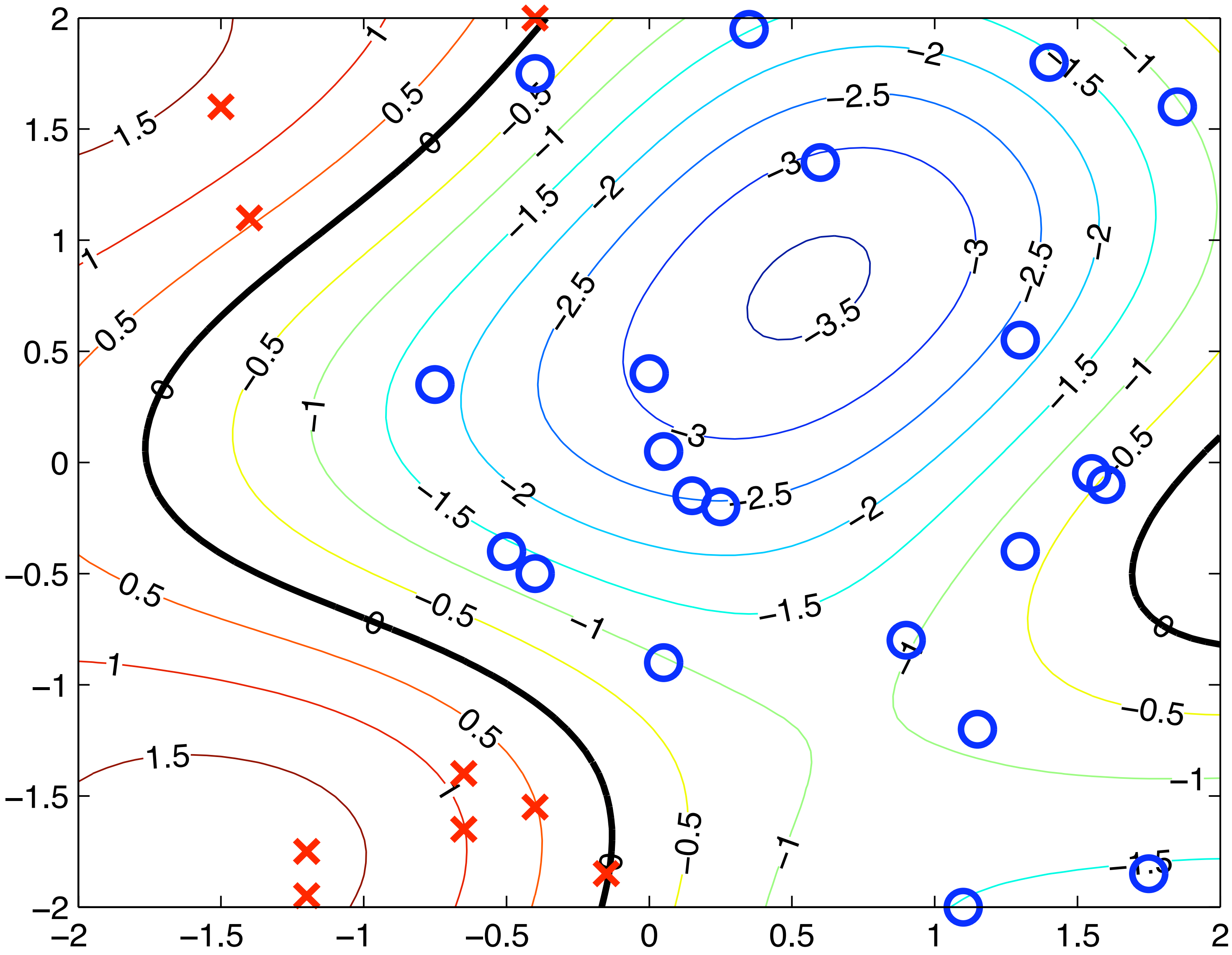
Orange (+1)
or lemon (-1)

- Just apply ridge regression with +1/-1 targets (forget about the Gaussian noise assumption!)
- We again use Gaussian RBF:

$$\phi(x;\mu_c) = \exp\left(-\frac{1}{2}\|\boldsymbol{x} - \boldsymbol{\mu}_c\|^2\right)$$

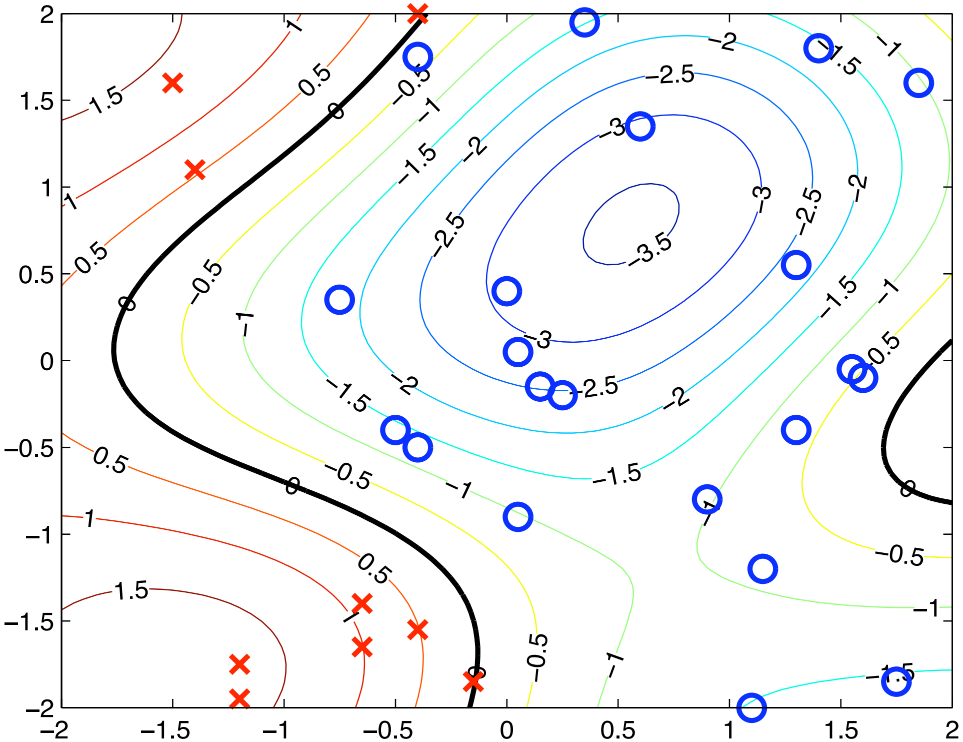
Vector

Classification: Truth

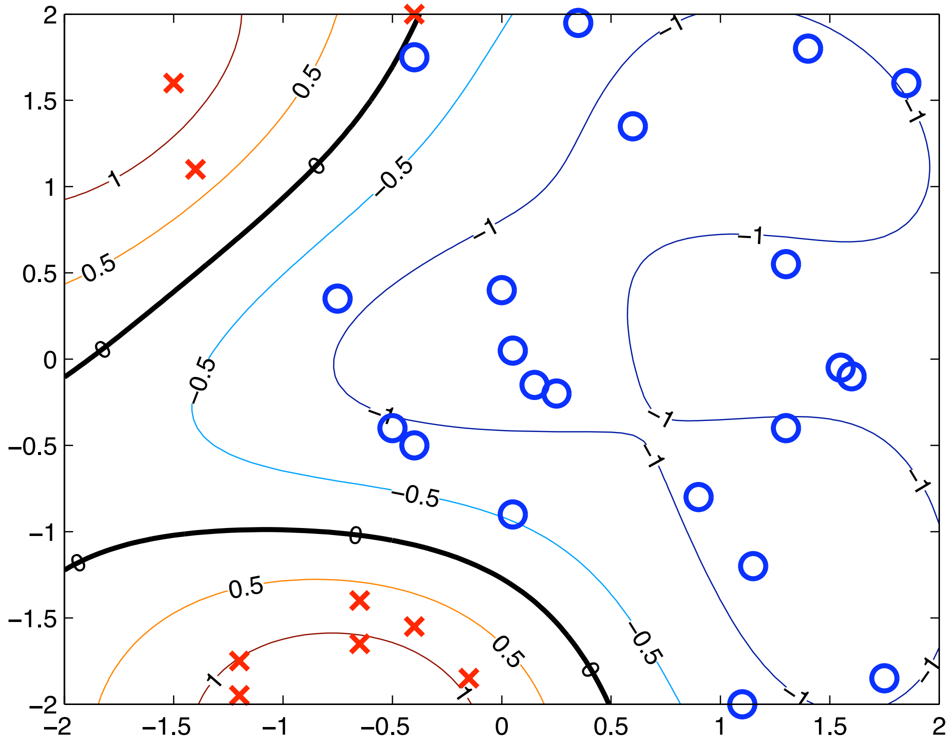


Classification with RR, λ=0.1

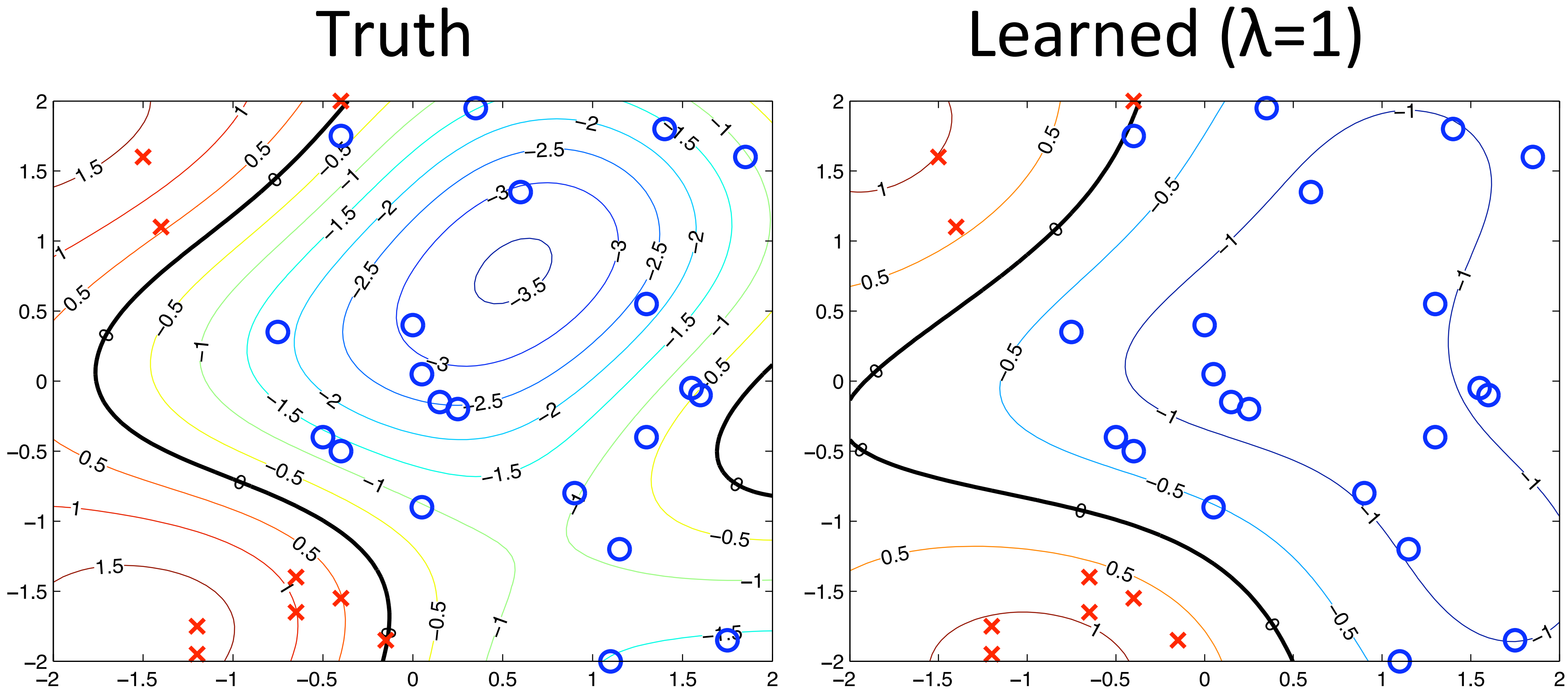
Truth



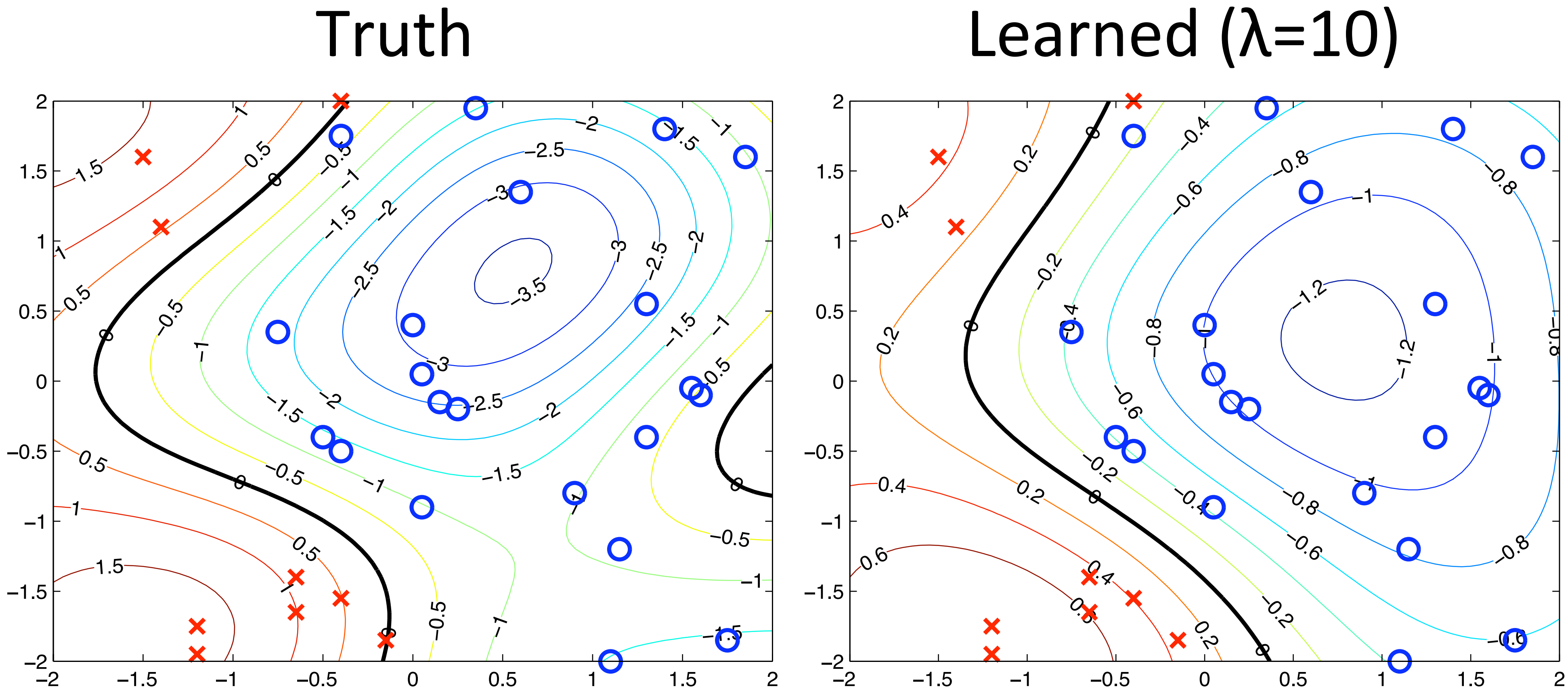
Learned (λ=0.1)



Classification with RR, λ=1

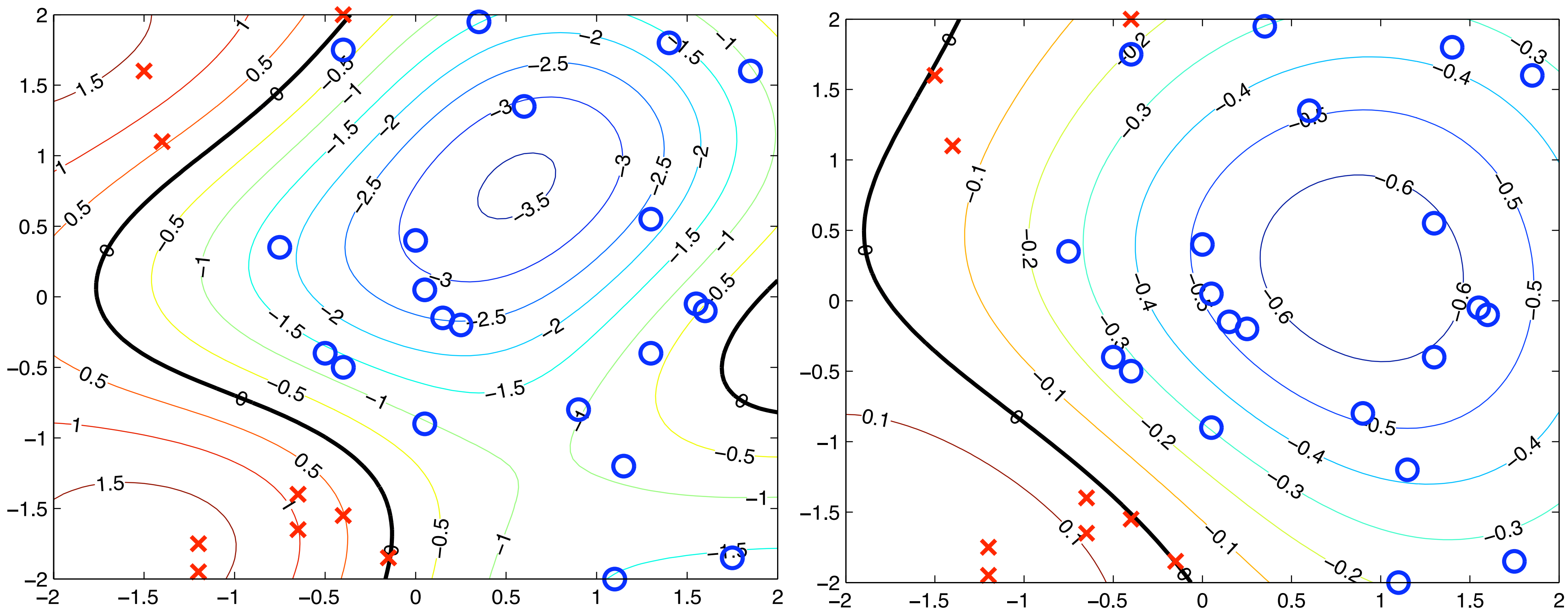


Classification with RR, λ=10



Classification with RR, λ=100

Truth　　　　Learned (λ=100)

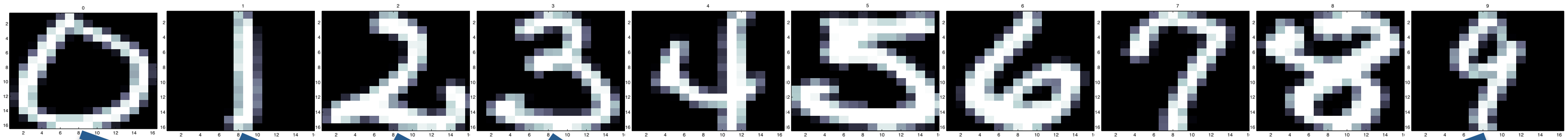


Multi-class classification

USPS digits dataset

7291 training samples,
2007 test samples

http://www-stat-class.stanford.edu/~tibs/ElemStatLearn/datasets/zip.info

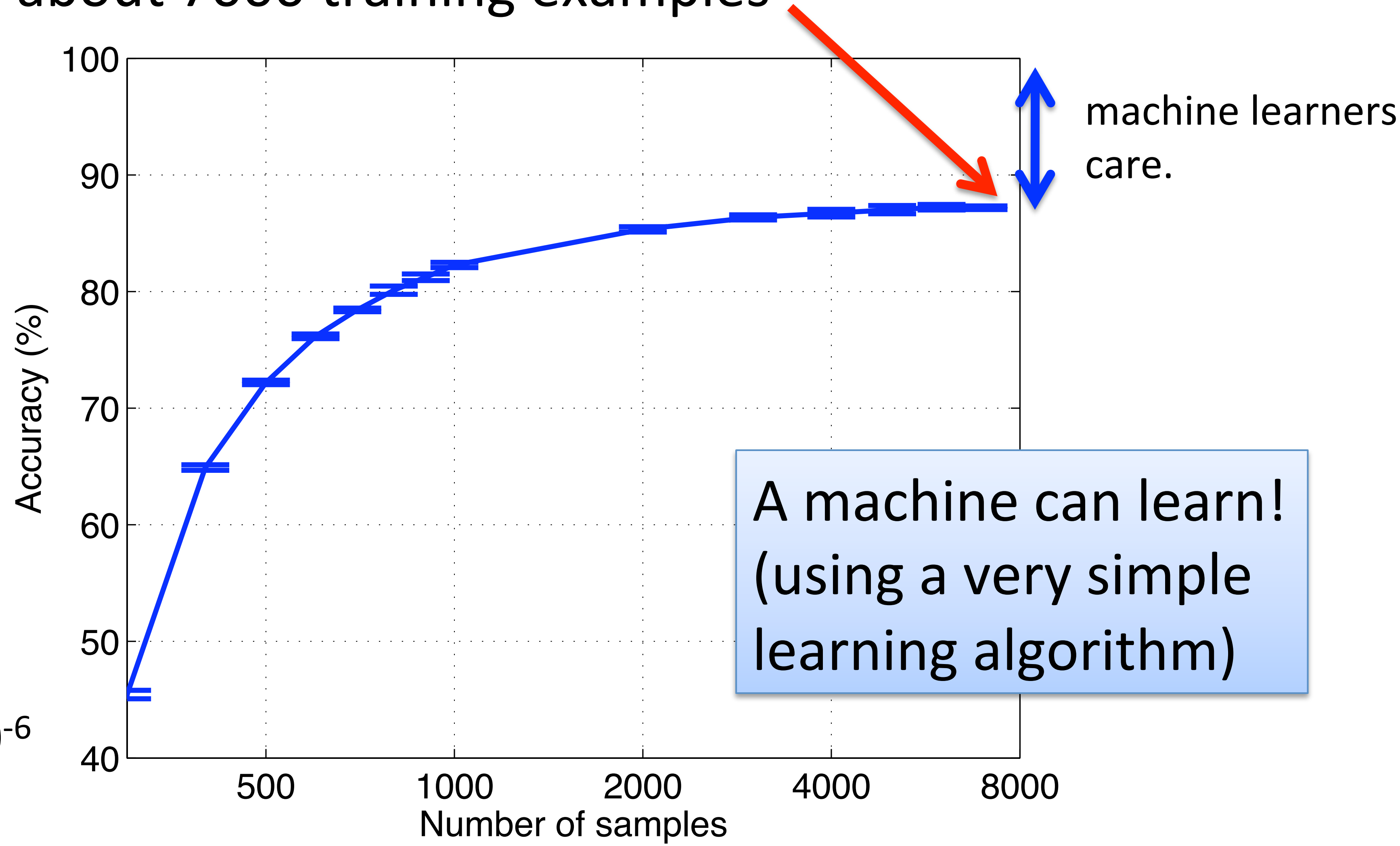


$$\boldsymbol{y} = \begin{pmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & & & \vdots \\ 0 & 1 & 0 & 0 & \cdots & 0 \end{pmatrix}$$

Number of samples

USPS dataset

We can obtain 88% accuracy on a held-out test-set using about 7000 training examples



$\lambda=10^{-6}$

Summary (so far)

- Ridge regression (RR) is very simple.
- RR can be coded in one line:

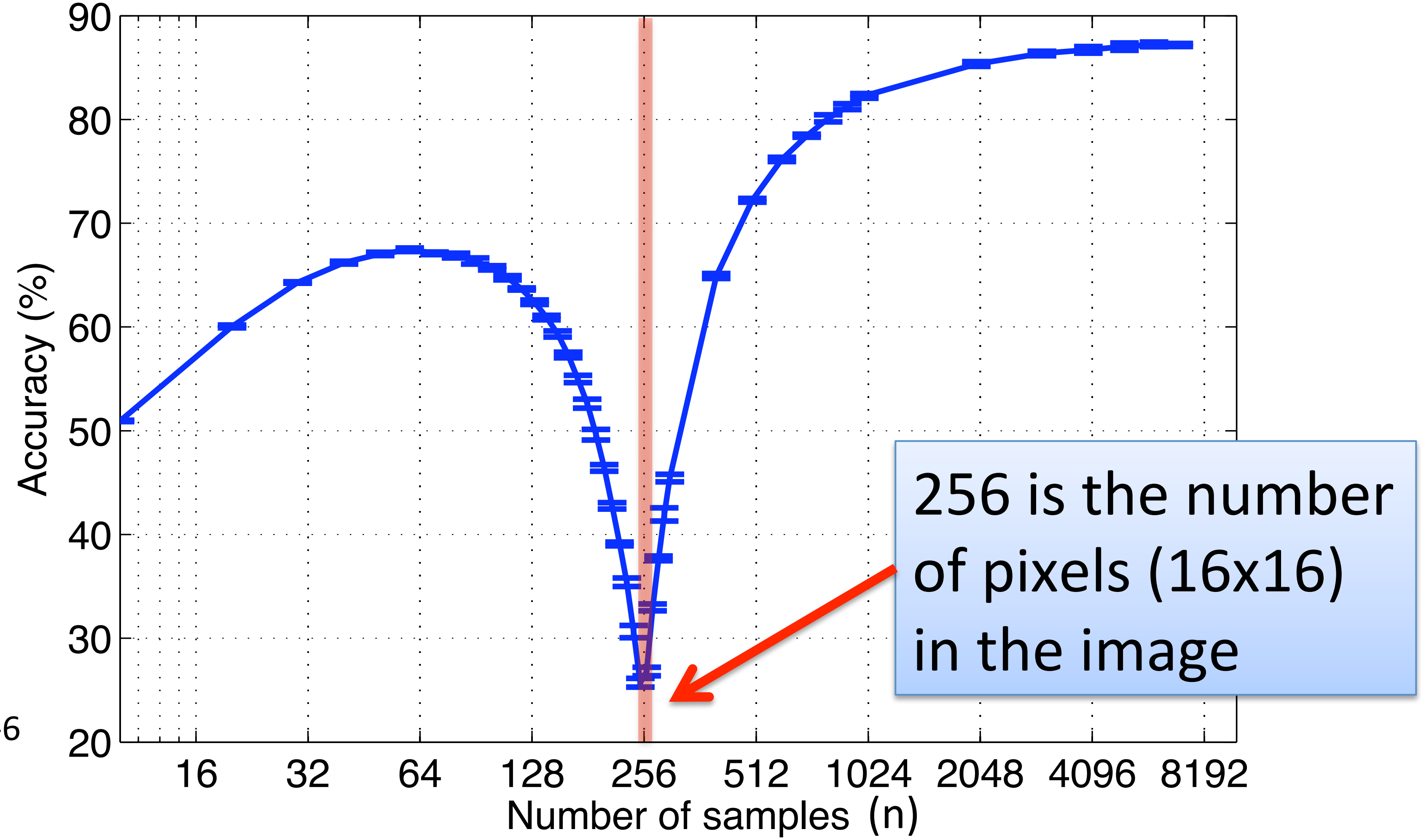
```
W=(X'*X+lambda*eye(n))\(X'*Y);
```

- RR can prevent over-fitting by regularization.
- Classification problem can also be solved by properly defining the output Y.
- Nonlinearities can be handled by using basis functions (polynomial, Gaussian RBF, etc.).

Singularity
- The dark side of RR

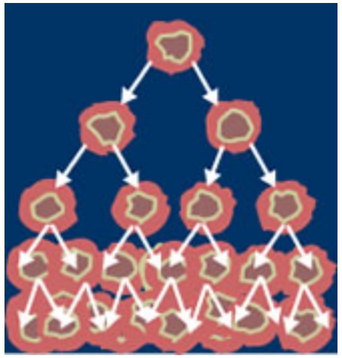
USPS dataset (p=256)
(What I have been hiding)

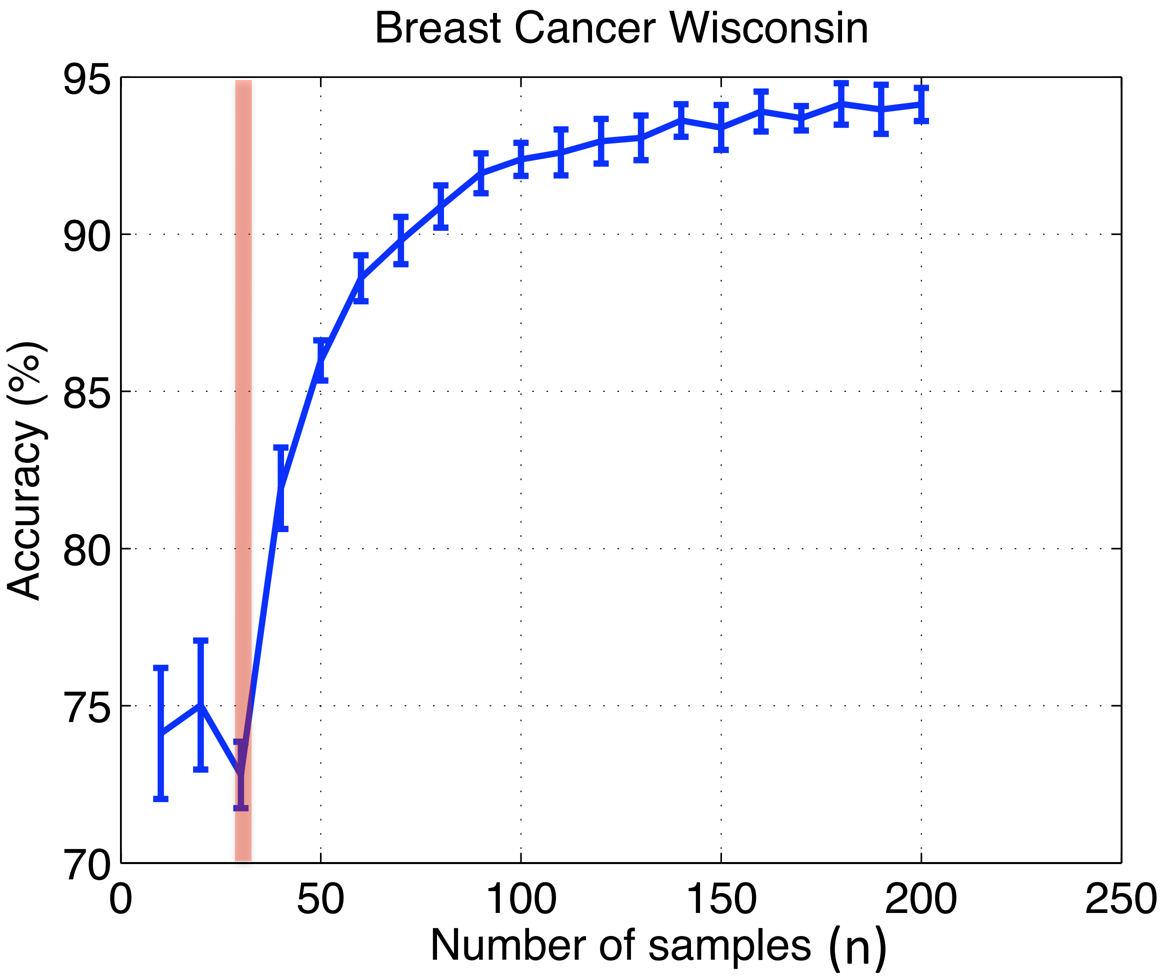
- The more data the less accurate??



$\lambda=10^{-6}$

Breast Cancer Wisconsin (diagnostic) dataset (p=30)



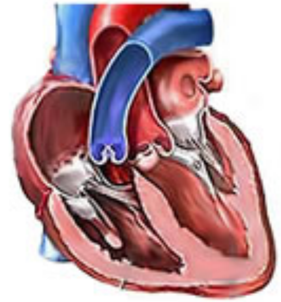


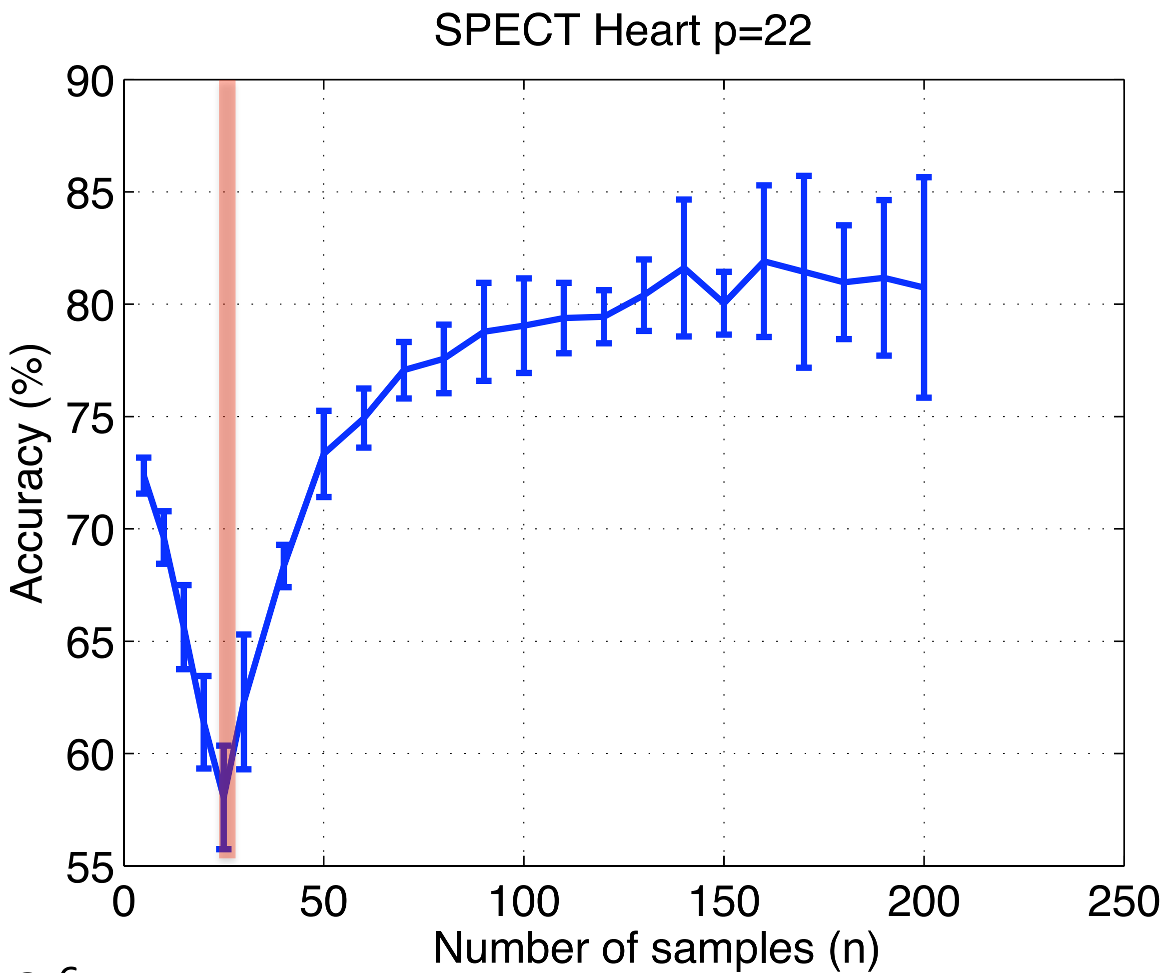
30 real-valued features

- radius
- texture
- perimeter
- area, etc.

$\lambda=10^{-6}$

SPECT Heart dataset (p=22)

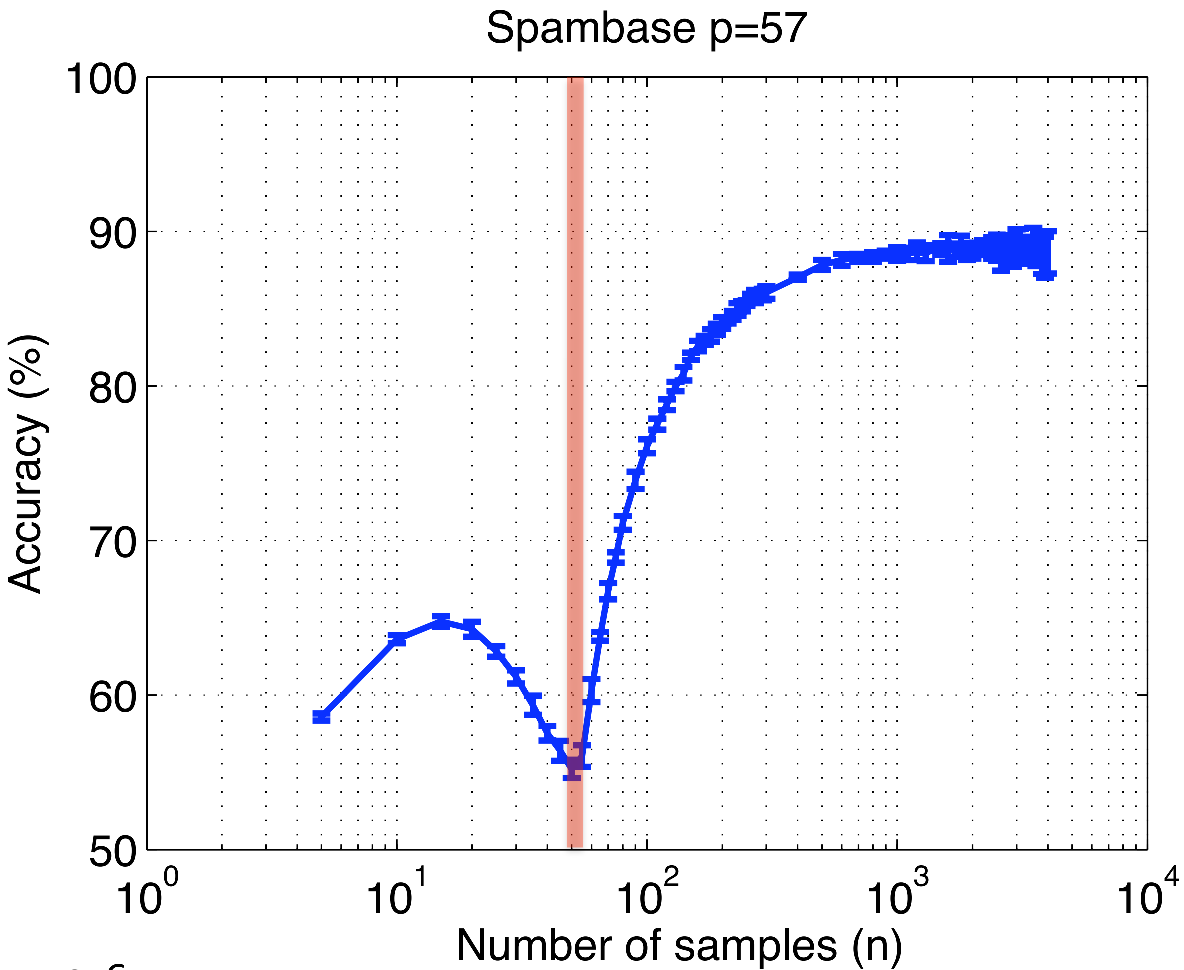




22 binary features

$\lambda=10^{-6}$

Spambase dataset (p=57)

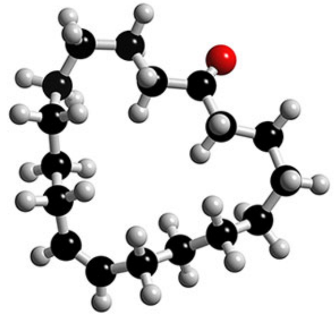


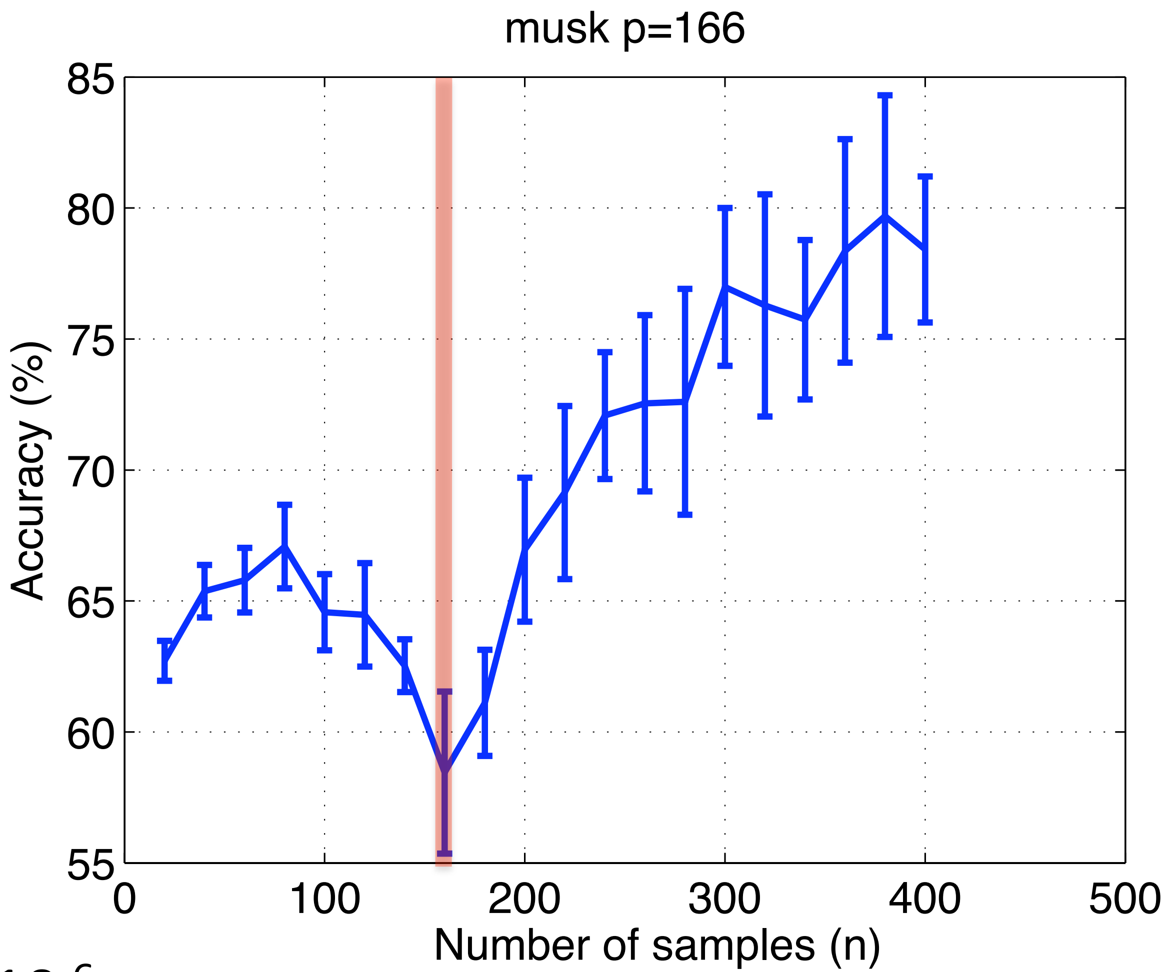
55 real-valued features
- word frequency
- character frequency

2 integer-valued feats
- run-length

$\lambda = 10^{-6}$

Musk dataset (p=166)





166 real-valued features

$\lambda=10^{-6}$

Singularity

Why does it happen?

How can we avoid it?

Why does it happen?
Let's analyze the simplest case: regression.

- Model
  - Design matrix *X* is fixed (*X* is *not* a random var.).
  - Output

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{w}^* + \boldsymbol{\xi}, \quad \boldsymbol{\xi} \sim \underbrace{\mathcal{N}(0, \sigma^2 \boldsymbol{I}_n)}_{\text{Gaussian noise}}$$

- Estimator

$$\hat{\boldsymbol{w}} = \left(\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I}_p\right)^{-1} \boldsymbol{X}^\top \boldsymbol{y}$$

- Generalization Error

$$\mathbb{E}_{\boldsymbol{\xi}} \|\hat{\boldsymbol{w}} - \boldsymbol{w}^*\|^2$$

Expectation wrt ξ   Estimated   Truth

Analysis Strategy

(1) Bias-variance decomposition

$$\mathbb{E}_{\boldsymbol{\xi}}\|\hat{\boldsymbol{w}} - \boldsymbol{w}^*\|^2 = \underbrace{\mathbb{E}_{\boldsymbol{\xi}}\|\hat{\boldsymbol{w}} - \bar{\boldsymbol{w}}\|^2}_{\text{Variance}} + \underbrace{\|\bar{\boldsymbol{w}} - \boldsymbol{w}^*\|^2}_{\text{Bias (squared)}}$$

where $\bar{\boldsymbol{w}}$ is the mean estimator $\bar{\boldsymbol{w}} = \mathbb{E}_{\boldsymbol{\xi}}\hat{\boldsymbol{w}}$

(2) Analyze the variance

$$\mathbb{E}_{\boldsymbol{\xi}}\|\hat{\boldsymbol{w}} - \bar{\boldsymbol{w}}\|^2 = ?$$

(3) Analyze the bias

$$\|\bar{\boldsymbol{w}} - \boldsymbol{w}^*\|^2 = ?$$

Analyze the variance (sketch)

1. Show that

Variance

$$\mathbb{E}_{\boldsymbol{\xi}}\|\hat{\boldsymbol{w}} - \bar{\boldsymbol{w}}\|^2 = \sigma^2 \mathrm{Tr}\left((\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I}_p)^{-2} \boldsymbol{X}^\top \boldsymbol{X}\right)$$

2. Let $s_1>0,\ldots,s_m>0$ be the positive singular values of X ($m=\min(n,p)$). Show that

$$\mathbb{E}_{\boldsymbol{\xi}}\|\hat{\boldsymbol{w}} - \bar{\boldsymbol{w}}\|^2 = \sigma^2 \sum_{i=1}^{m} \frac{s_i^2}{(s_i^2 + \lambda)^2} \xrightarrow{\lambda \to 0} \sigma^2 \sum_{i=1}^{m} s_i^{-2}$$

Variance can be large if the min. singular-value is close to zero!

Analyze the bias (sketch)

1. Show that

$$\|\bar{\boldsymbol{w}} - \boldsymbol{w}^*\|^2 = \lambda^2 \|(\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I}_p)^{-1} \boldsymbol{w}^*\|^2$$

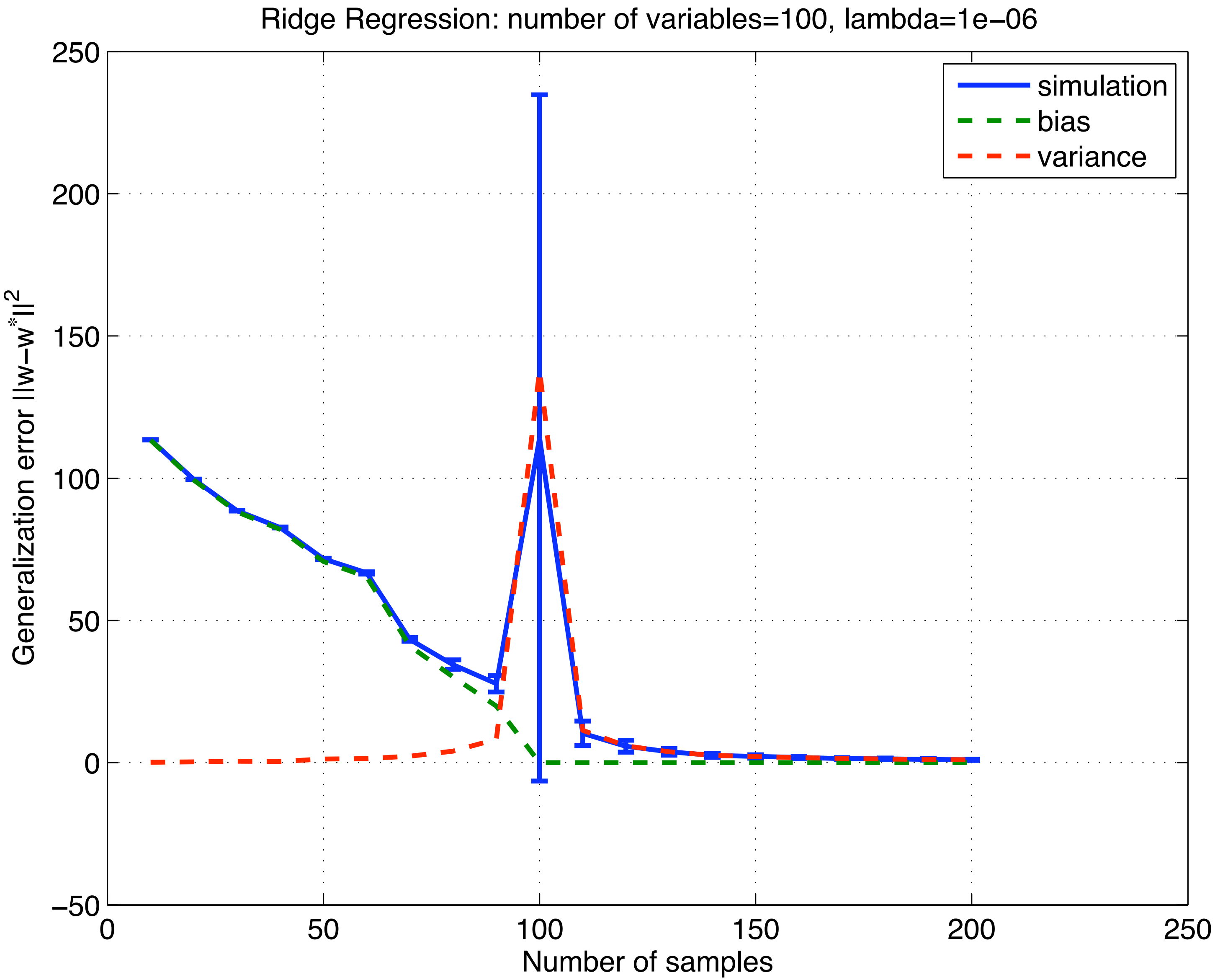
2. Show that

$$\|\bar{\boldsymbol{w}} - \boldsymbol{w}^*\|^2 = \sum_{i=1}^{p} \left(\frac{\lambda {\boldsymbol{v}_i}^\top \boldsymbol{w}^*}{s_i^2 + \lambda} \right)^2$$

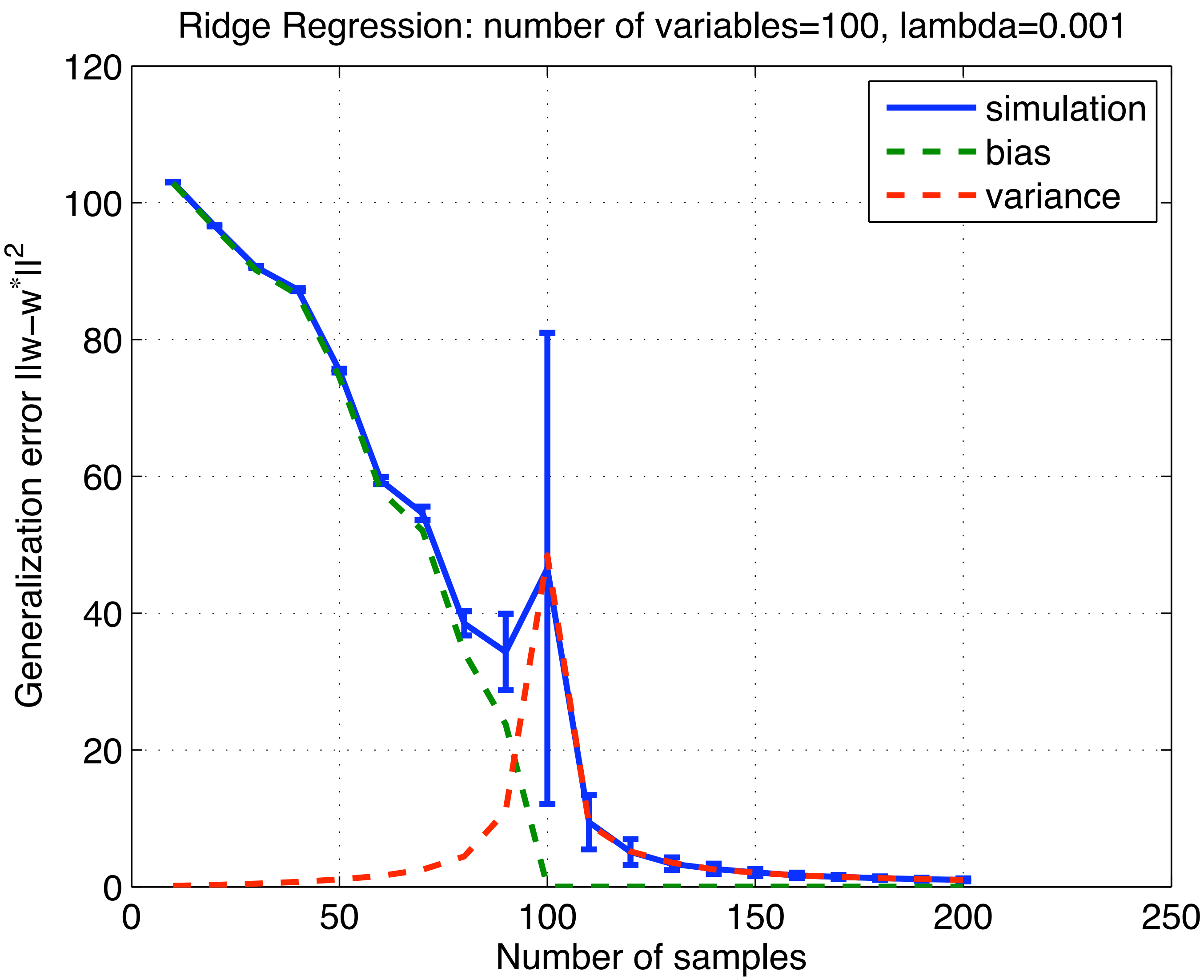
where $s_i = 0$ (if $i>m$),
v_i is the ith right singular vector of X

$$\|\bar{\boldsymbol{w}} - \boldsymbol{w}^*\|^2 \xrightarrow{\lambda \to 0} \begin{cases} \sum_{i=n+1}^{p} \left({\boldsymbol{v}_i}^\top \boldsymbol{w}^* \right)^2 & (n < p), \\ 0 & (\text{otherwise}). \end{cases}$$

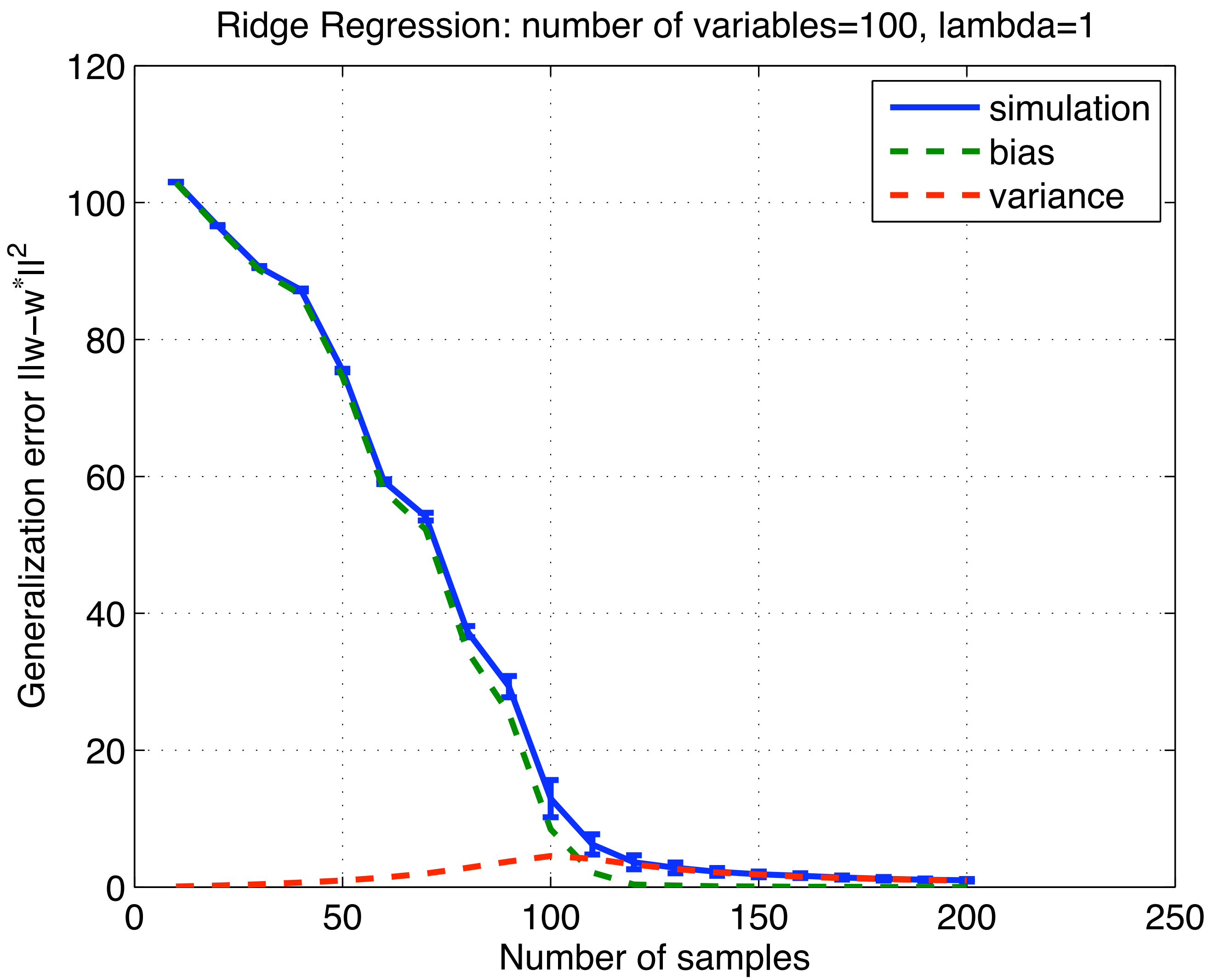
Result (λ=10^{-6})



Result (λ=0.001)



Result (λ=1)



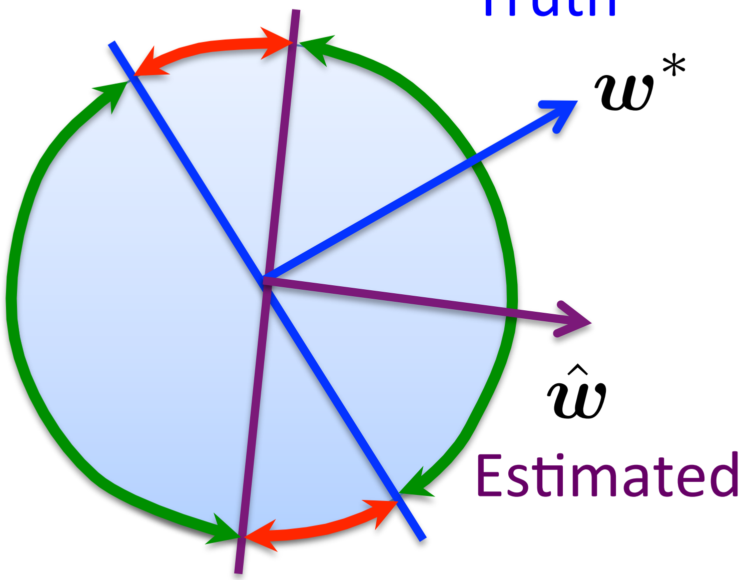
Ridge Regression: number of variables=100, lambda=1
Generalization error ||w−w*||²
Number of samples
simulation
bias
variance
0
20
40
60
80
100
120
0
50
100
150
200
250

How about classification?

- Model
  - Input vector x_i is sampled from standard Gaussian distribution (x_i is a random variable):
$$x_i \sim \mathcal{N}(0, \boldsymbol{I}_p) \quad (i = 1, \ldots, n)$$
  - The true classifier is also a normal random variable:
$$\boldsymbol{w}^* \sim \mathcal{N}(0, \boldsymbol{I}_p)$$
  - Output
$$\boldsymbol{y} = \mathrm{sign}(\boldsymbol{X}\boldsymbol{w}^*)$$

(Not a Gaussian noise!)

- Generalization Error
$$\epsilon = \frac{1}{\pi} \arccos\left(\frac{\hat{\boldsymbol{w}}^\top \boldsymbol{w}^*}{\|\hat{\boldsymbol{w}}\|\|\boldsymbol{w}^*\|}\right)$$



Analyzing classification

- Let α = n/p and assume that

Number of samples	Number of features	Regularization constant
$n \to \infty,$	$p \to \infty,$	$\lambda \to 0$

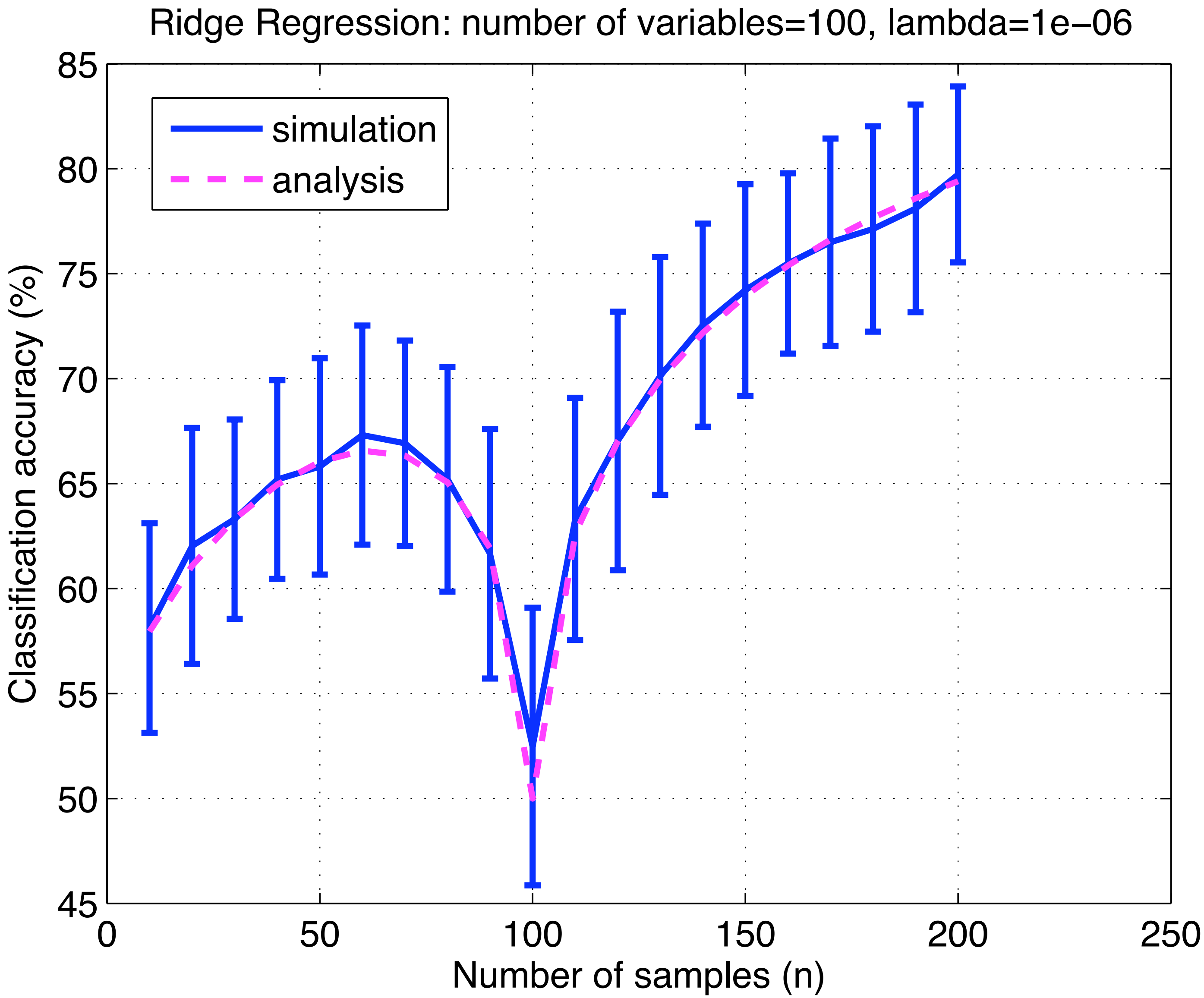
- Analyze the inner product

$$\mathbb{E}\hat{\boldsymbol{w}}^\top \boldsymbol{w}^* = \begin{cases} \sqrt{p}\sqrt{\frac{2}{\pi}}\alpha & (\alpha < 1), \\ \sqrt{p}\sqrt{\frac{2}{\pi}} & (\alpha > 1). \end{cases}$$

- Analyze the norm

$$\mathbb{E}\|\hat{\boldsymbol{w}}\|^2 = \begin{cases} \frac{\alpha(1-\frac{2}{\pi}\alpha)}{1-\alpha} & (\alpha < 1), \\ \frac{\frac{2}{\pi}(\alpha-1)+1-\frac{2}{\pi}}{\alpha-1} & (\alpha > 1). \end{cases} \qquad \mathbb{E}\|w^*\|^2 = p.$$

Analyzing classification (result)

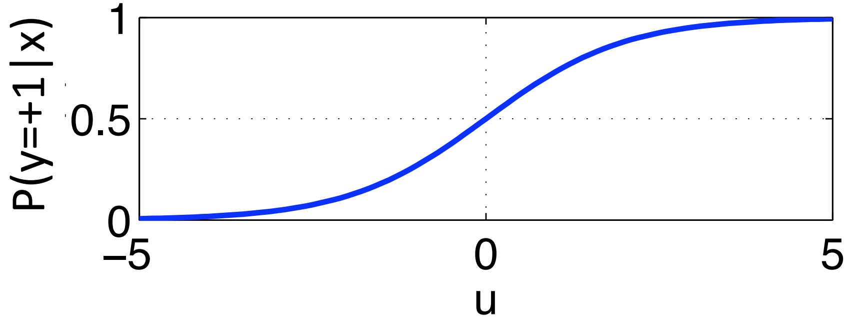


How can we avoid the singularity?

✓Regularization

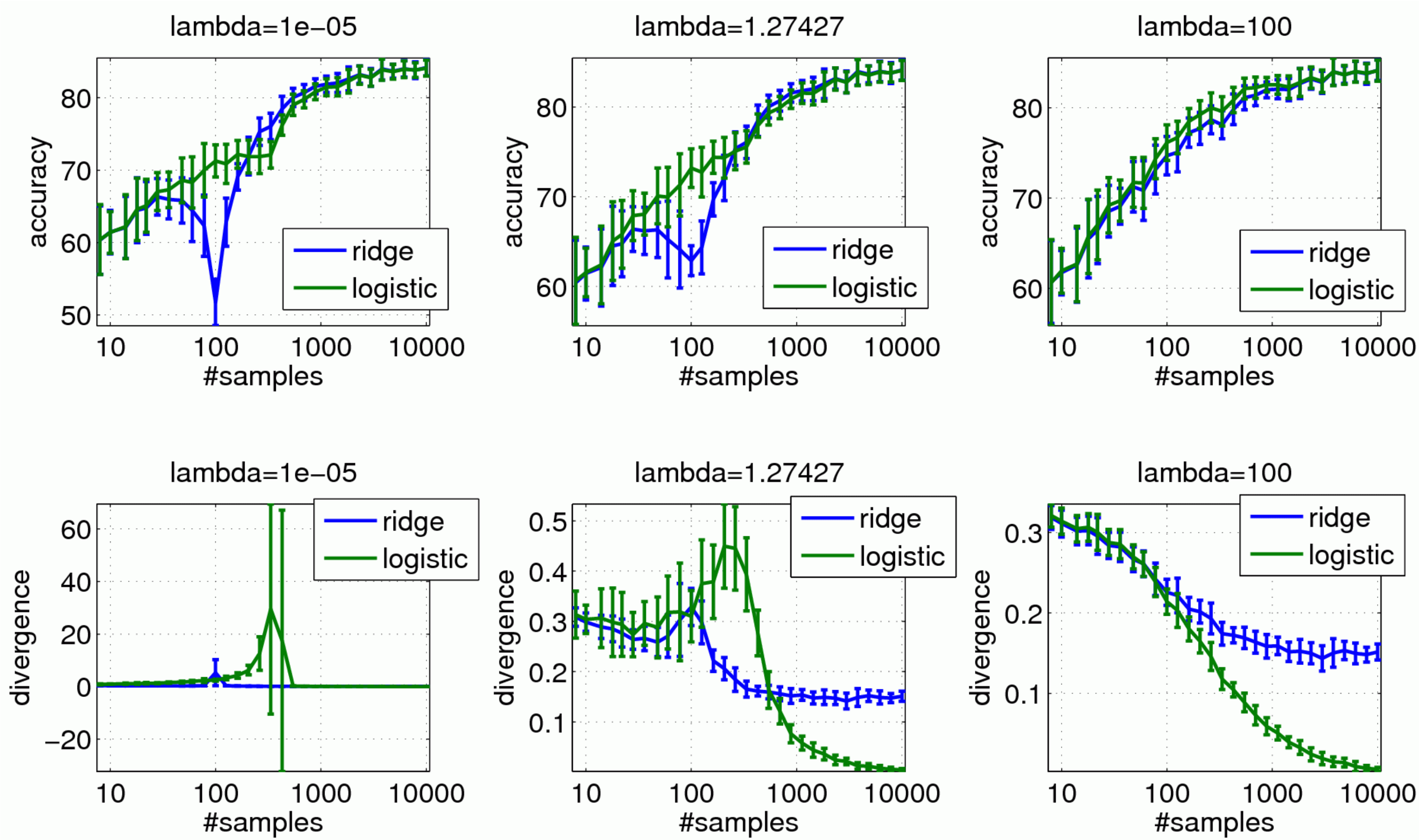
✓Logistic regression

$$\log \frac{P(y=+1|\boldsymbol{x})}{P(y=-1|\boldsymbol{x})} = \boldsymbol{w}^\top \boldsymbol{x}$$



$$\underset{\boldsymbol{w}}{\text{minimize}} \quad \underbrace{\sum_{i=1}^{n} \log(1+\exp(-y_i \boldsymbol{w}^\top \boldsymbol{x}_i))}_{\text{Training error}} + \underbrace{\frac{\lambda}{2}\|\boldsymbol{w}\|^2}_{\substack{\text{Regularization term} \\ (\lambda\text{: regularization const.})}}$$

How can we avoid singularity?



Summary

- Ridge regression (RR) is very simple and easy to implement.
- RR has wide application, e.g., classification, multi-class classification
- Be careful about the singularity. Adding data does not always help improve performance.
- Analyzing the singularity: predicts the simulated performance quantitatively.
  - Regression setting: variance goes to inifity at n=p.
  - Classification setting: norm $\|\hat{w}\|^2$ goes to inifinity at n=p.

Further readings

- Elements of Statistical Learning (Hastie, Tibshirani, Friedman) 2009 (2nd edition)
  - Ridge regression (Sec. 3.4)
  - Bias & variance (Sec. 7.3)
  - Cross validation (Sec. 7.10)
- Statistical Mechanics of Generalization (Opper and Kinzel) in *Models of neural networks III: Association, generalization, and representation*, 1995.
  - Analysis of perceptron
  - Singularity