

# Regularized Linear Regression

Mahdi Roozbahani  
Georgia Tech

# EVERY GROUP PROJECT




DOES 99%  
OF THE WORK

HAS NO IDEA  
WHAT'S GOING  
ON THE  
WHOLE TIME

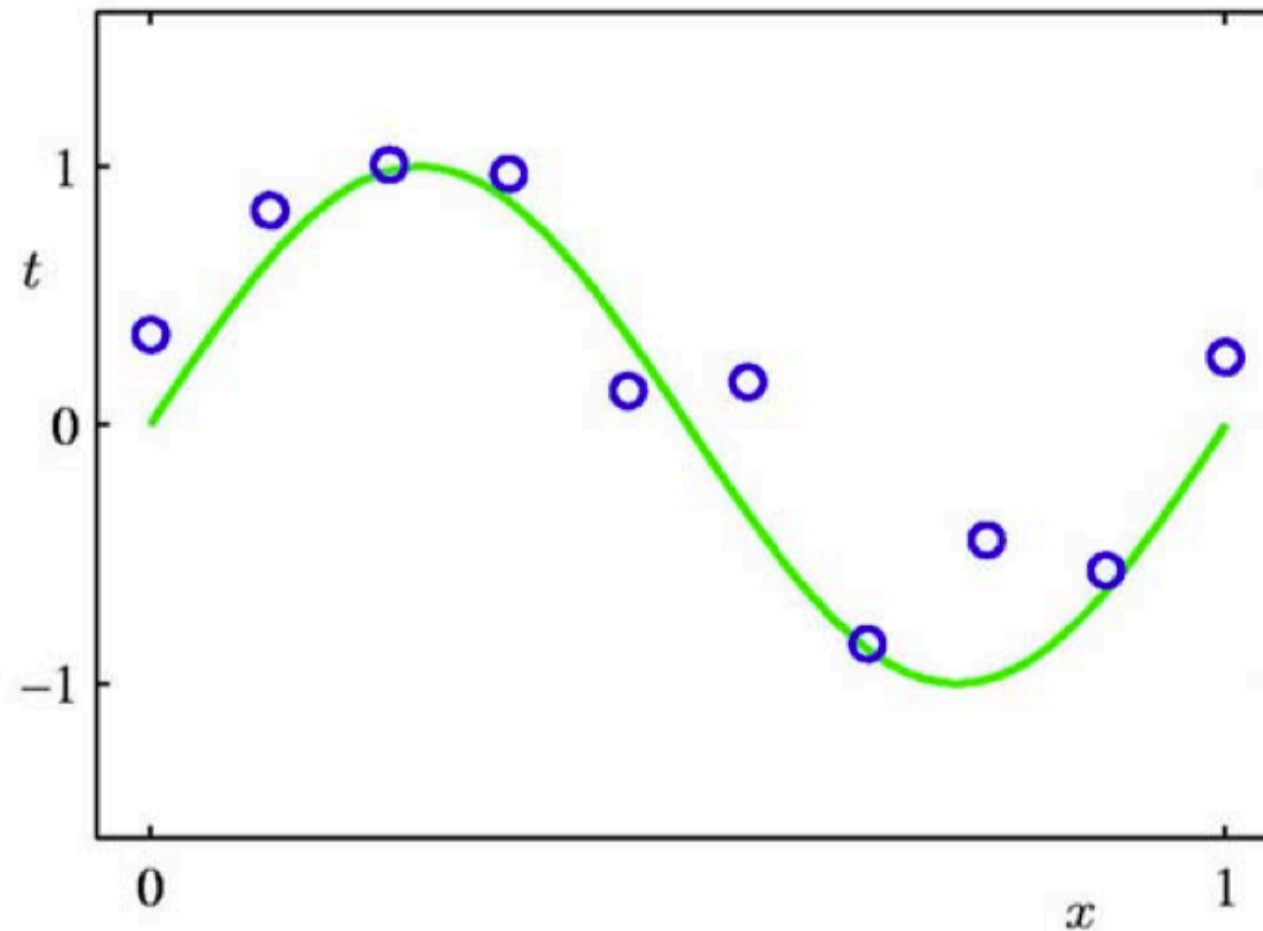
SAYS HE'S  
GOING TO  
HELP  
BUT HE'S  
NOT

DISAPPEAR  
AT THE VERY  
BEGINNING AND  
DOESN'T SHOW  
UP AGAIN TIL  
THE VERY END

# Outline

- Overfitting and regularized learning 
- Ridge regression
- Lasso regression
- Determining regularization strength

# Regression: Recap



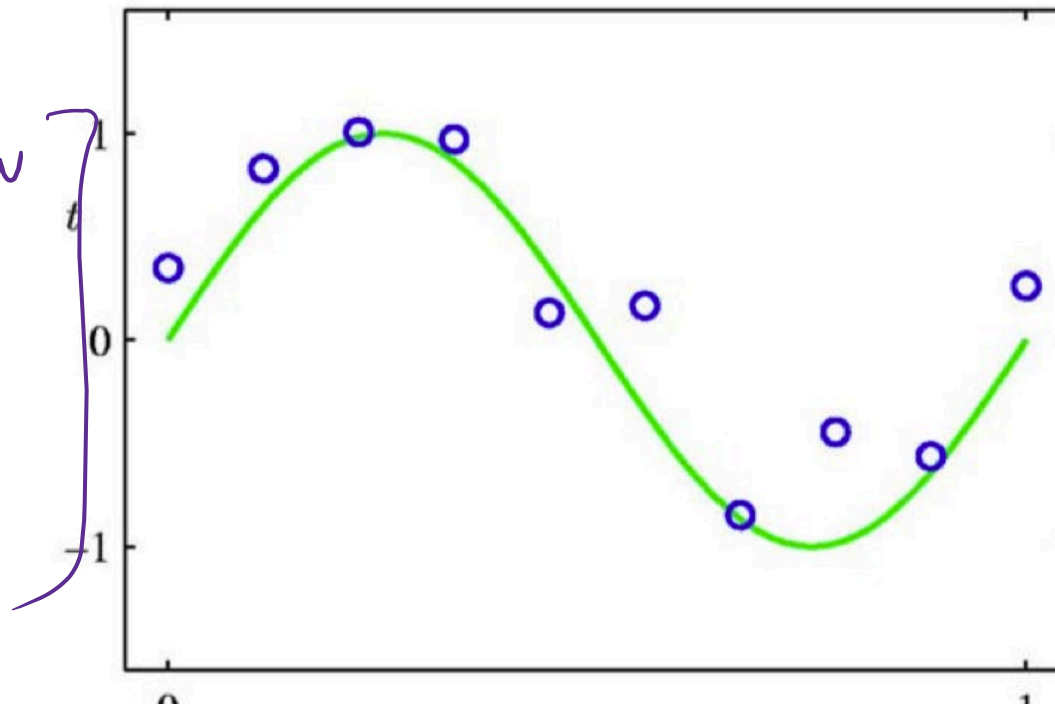
- Suppose we are given a training set of  $N$  observations  $(x_1, \dots, x_N)$  and  $(y_1, \dots, y_N)$
- Regression problem is to estimate  $y(x)$  from this data

# Regression: Recap

$$X = \begin{bmatrix} h & w \end{bmatrix}$$

$$\begin{matrix} h & w \\ \begin{bmatrix} h^2 & hw \\ \cancel{wh} & w^2 \end{bmatrix} \end{matrix}$$

$$Z = \begin{bmatrix} h & w & h^2 & w^2 & hw \end{bmatrix}$$



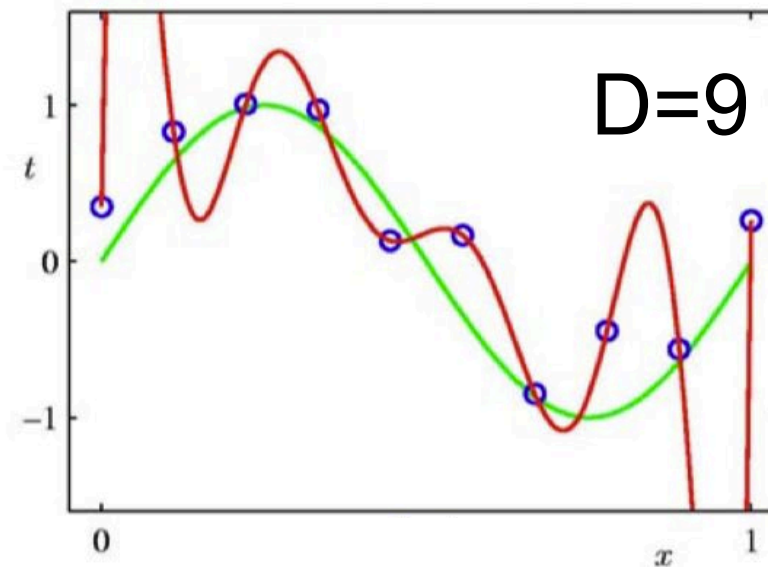
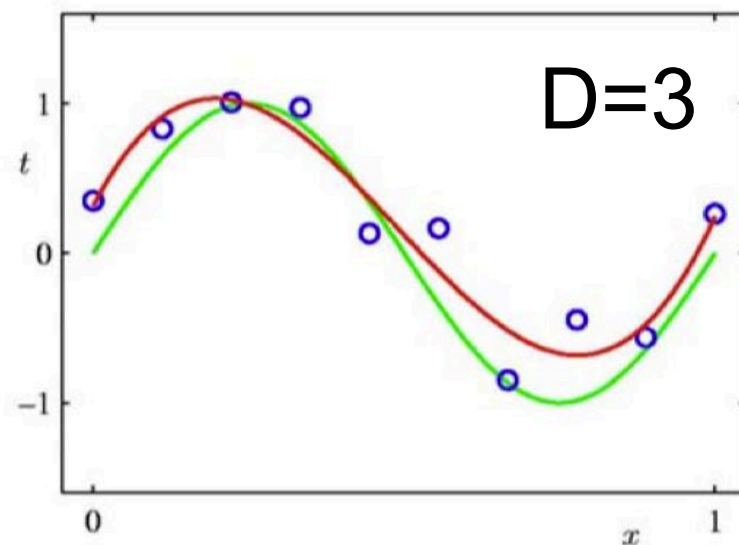
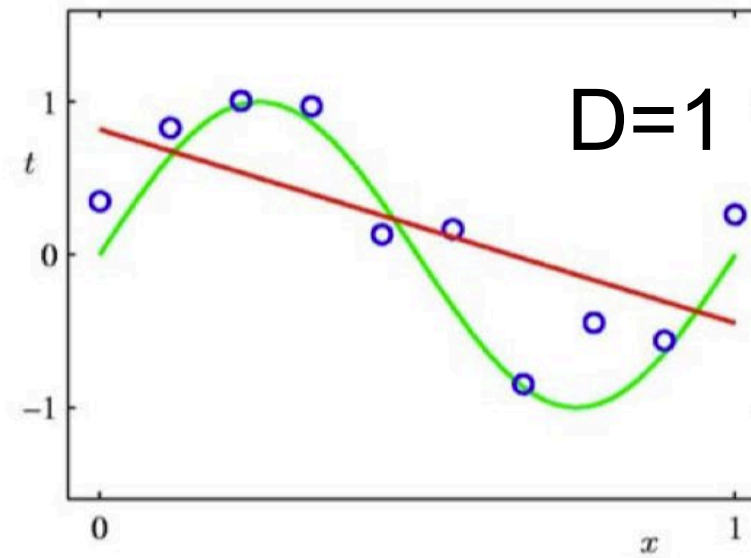
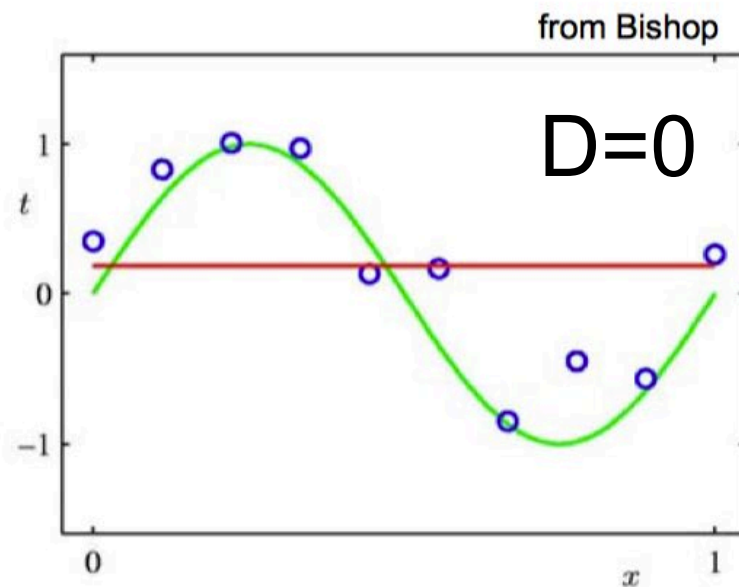
- Want to fit a polynomial regression model

$$y = \theta_0 + \theta_1 x + \theta_2 x^2 + \dots + \theta_d x^d + \epsilon$$

- $z = \{1, x, x^2, \dots, x^d\} \in R^d$  and  $\theta = (\theta_0, \theta_1, \theta_2, \dots, \theta_d)^T$

$$y = z\theta$$

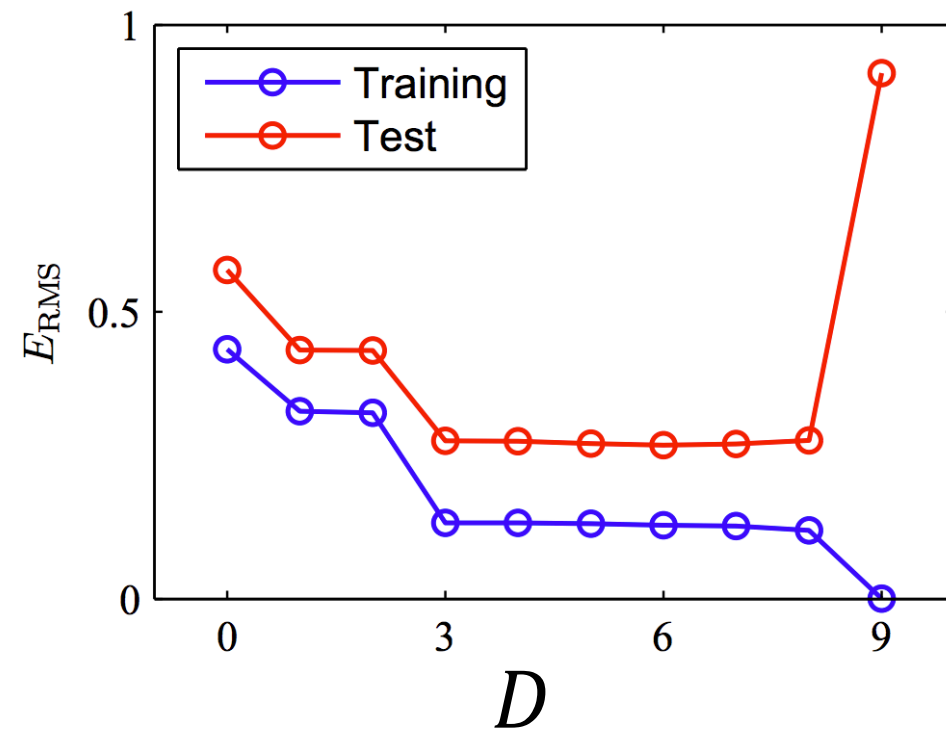
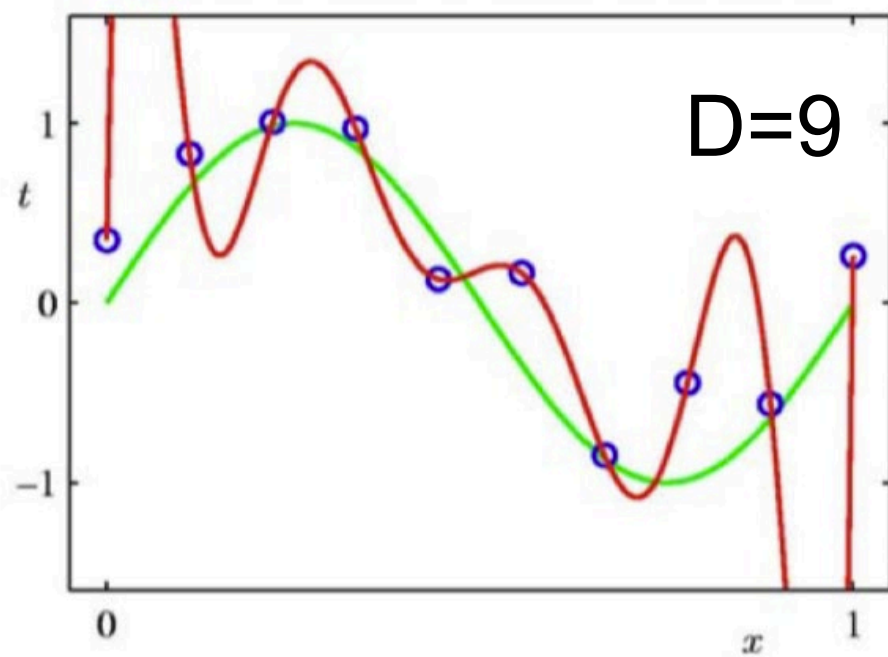
# Which One is Better?



- Can we increase the maximal polynomial degree to very large, such that the curve passes through all training points?

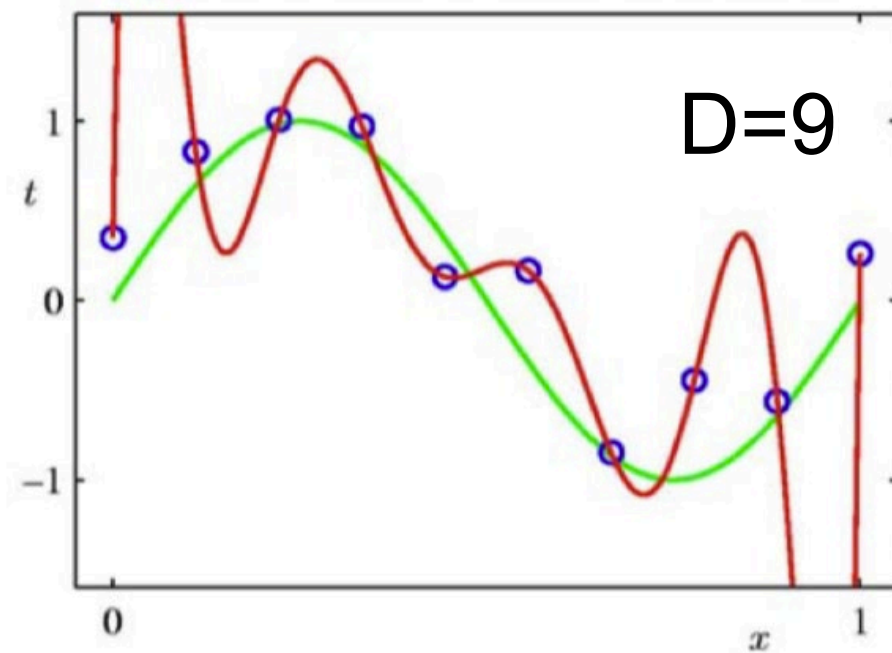
No, this can lead to **overfitting**!

# The Overfitting Problem



- The training error is very low, but the error on test set is large.
- The model captures not only patterns but also noisy nuisances in the training data.

# The Overfitting Problem

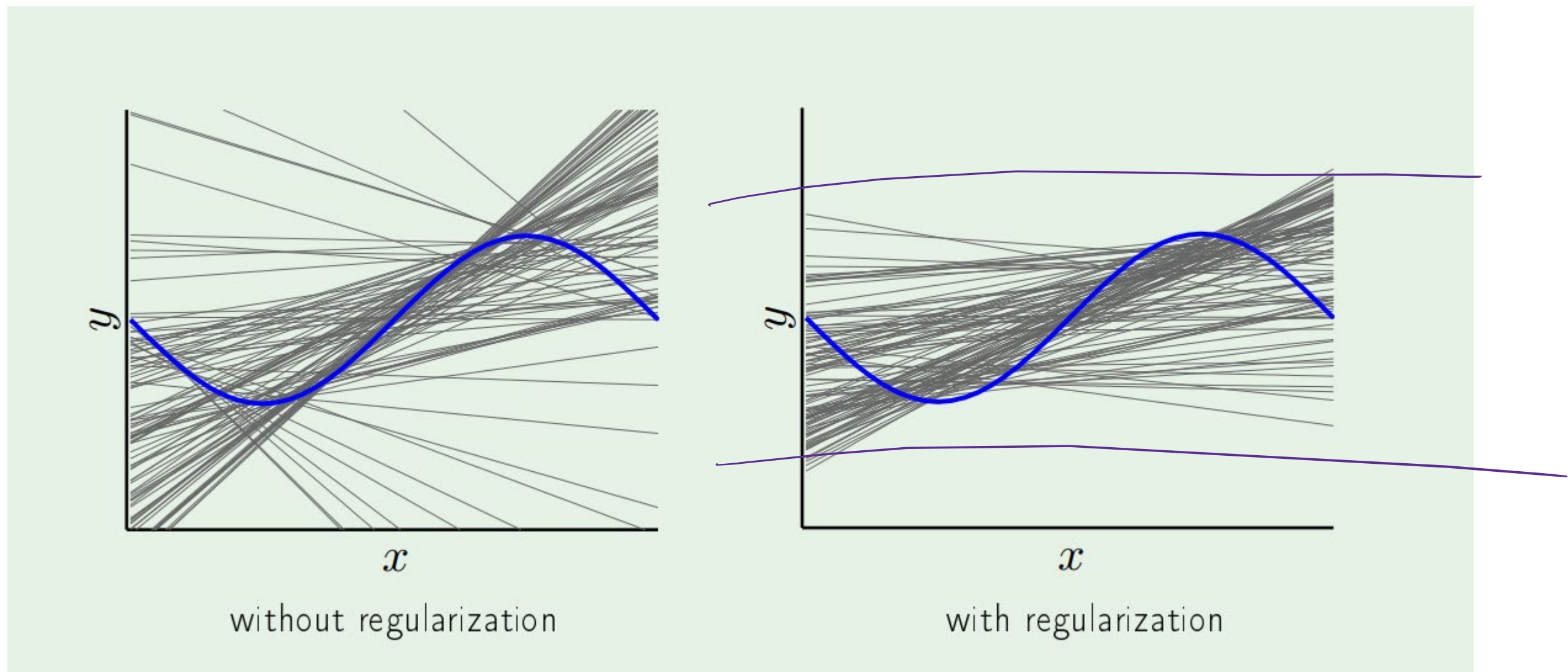


- In regression, overfitting is often associated with large Weights (**severe oscillation**)
- How can we address overfitting?



# Regularization

(smart way to cure overfitting disease )



Put a brake on fitting

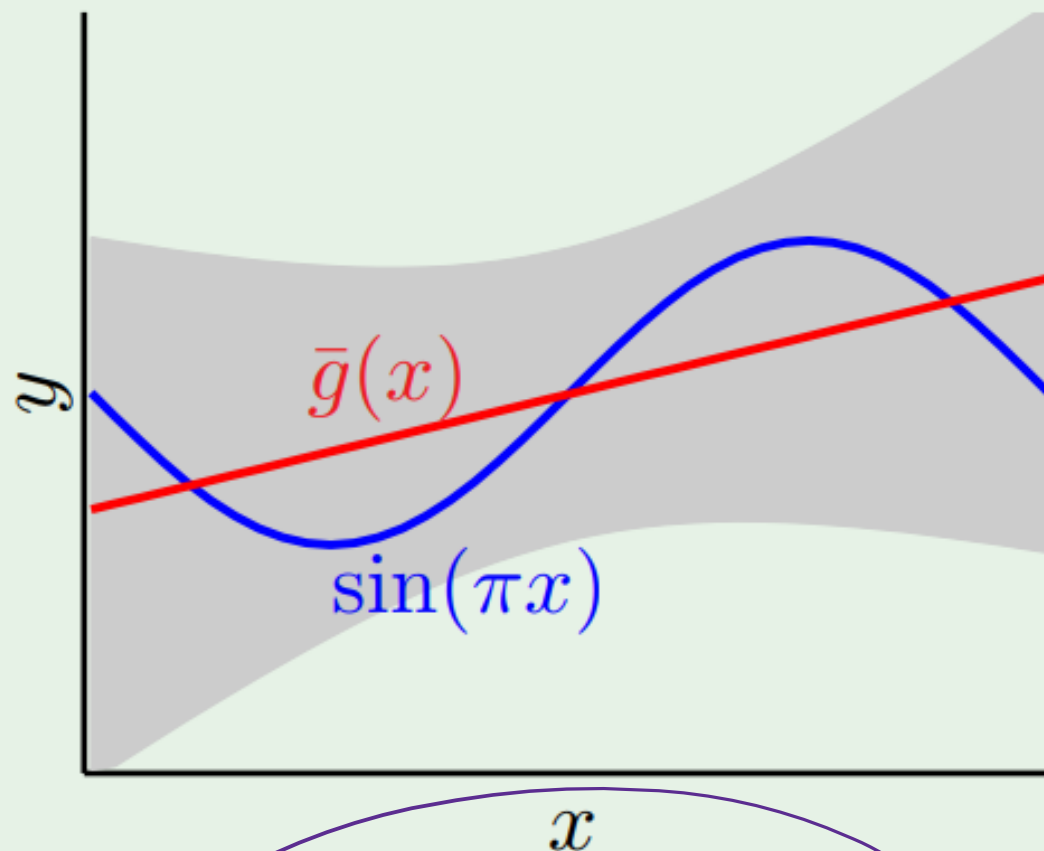
Fit a linear line on sinusoidal with just two data points

# Who is the winner?

$$E(\theta) = \text{bias}^2 + \text{variance}$$

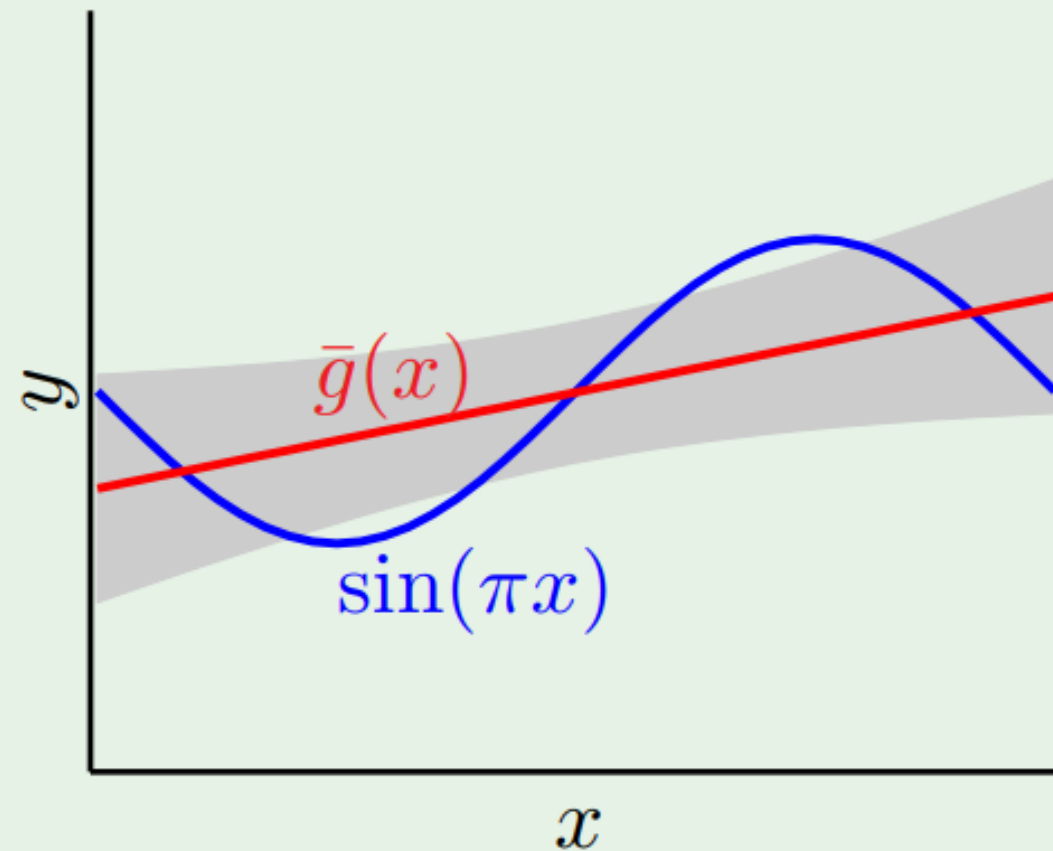
$\bar{g}(x)$ : average over all lines

without regularization



bias=0.21; var=1.69

with regularization



bias=0.23; var=0.33

# Polynomial Model

Want to fit a polynomial regression model

$$y = \theta_0 + \theta_1 x + \theta_2 x^2 + \cdots + \theta_d x^d + \epsilon$$

Let's rewrite it as:

$$y = \theta_0 + \theta_1 z_1 + \theta_2 z_2 + \cdots + \theta_d z_d + \epsilon = \mathbf{z}\boldsymbol{\theta}$$

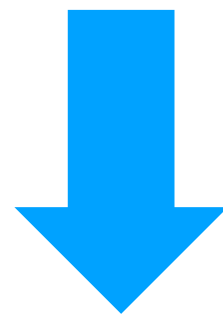
# Regularizing is just constraining the weights ( $\theta$ )

For example: let's do a **hard** constraining

$$y = \theta_0 + \theta_1 z_1 + \theta_2 z_2 + \cdots + \theta_d z_d$$

subject to

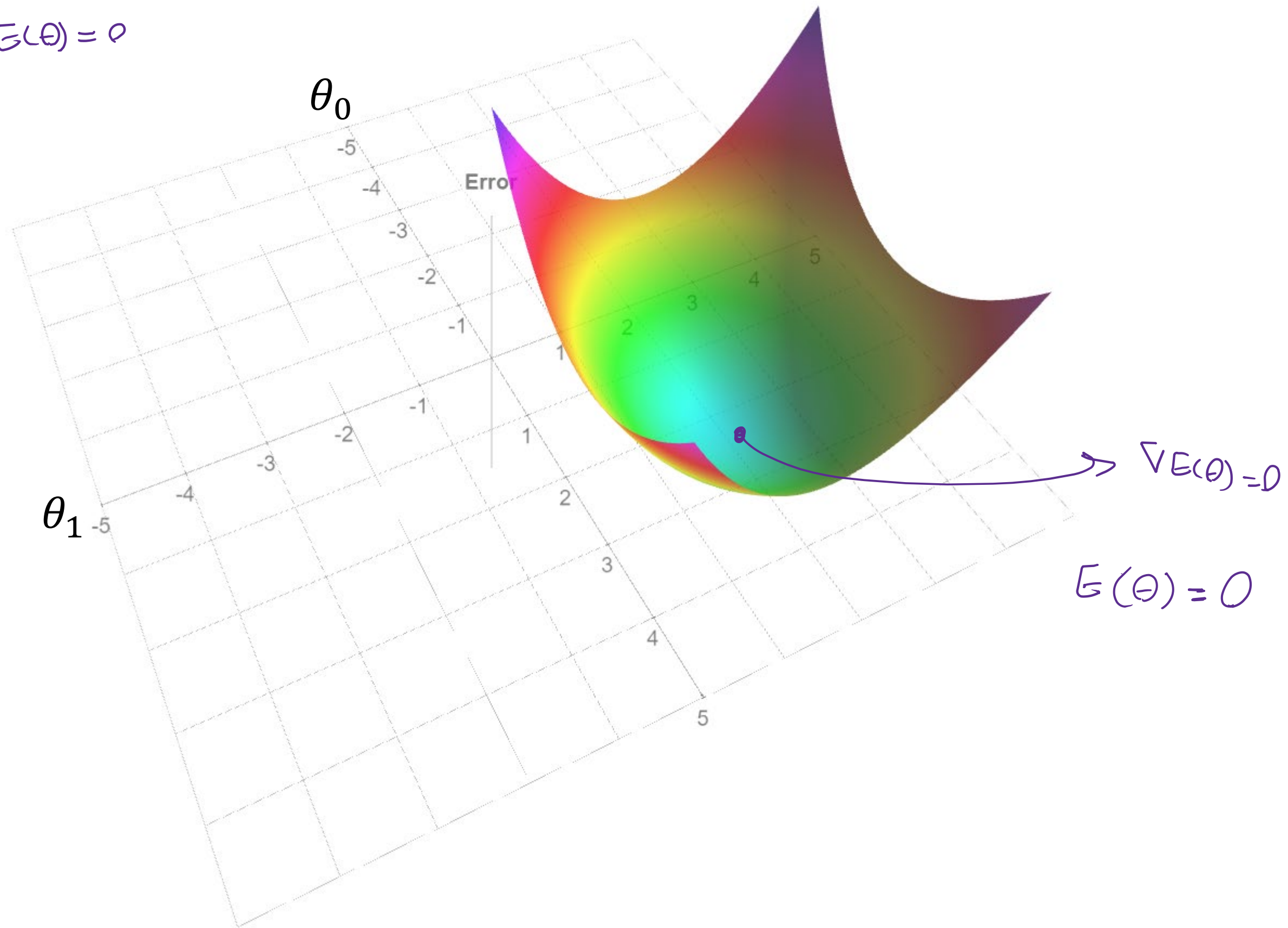
$$\theta_d = 0 \text{ for } d > 2$$



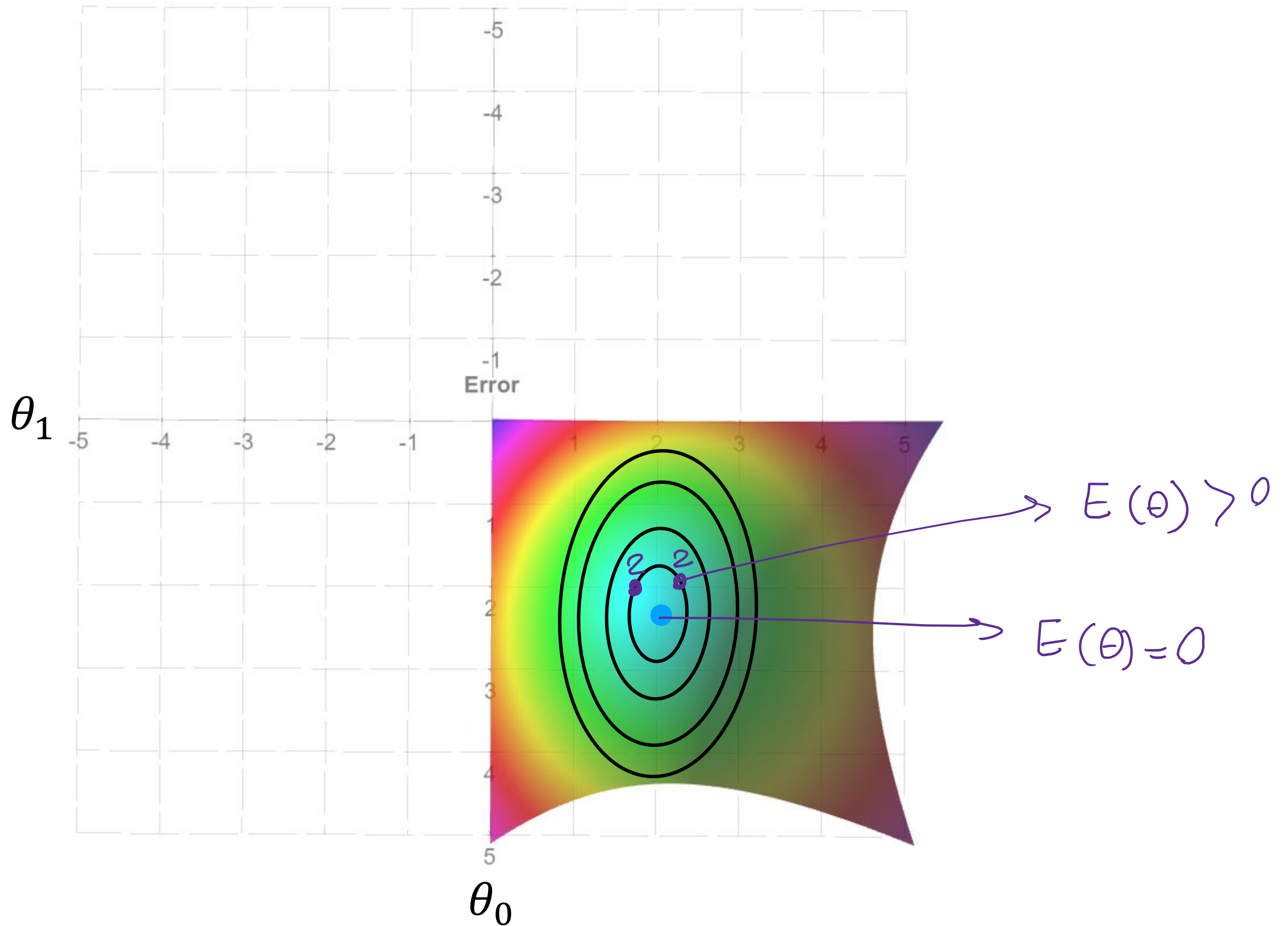
$$y = \theta_0 + \theta_1 z_1 + \theta_2 z_2 + 0 + \cdots + 0$$

$$E(\theta) = \frac{1}{N} \sum_{i=1}^n (y^i - z_i \theta)^2 =$$

$$\nabla E(\theta) = 0$$

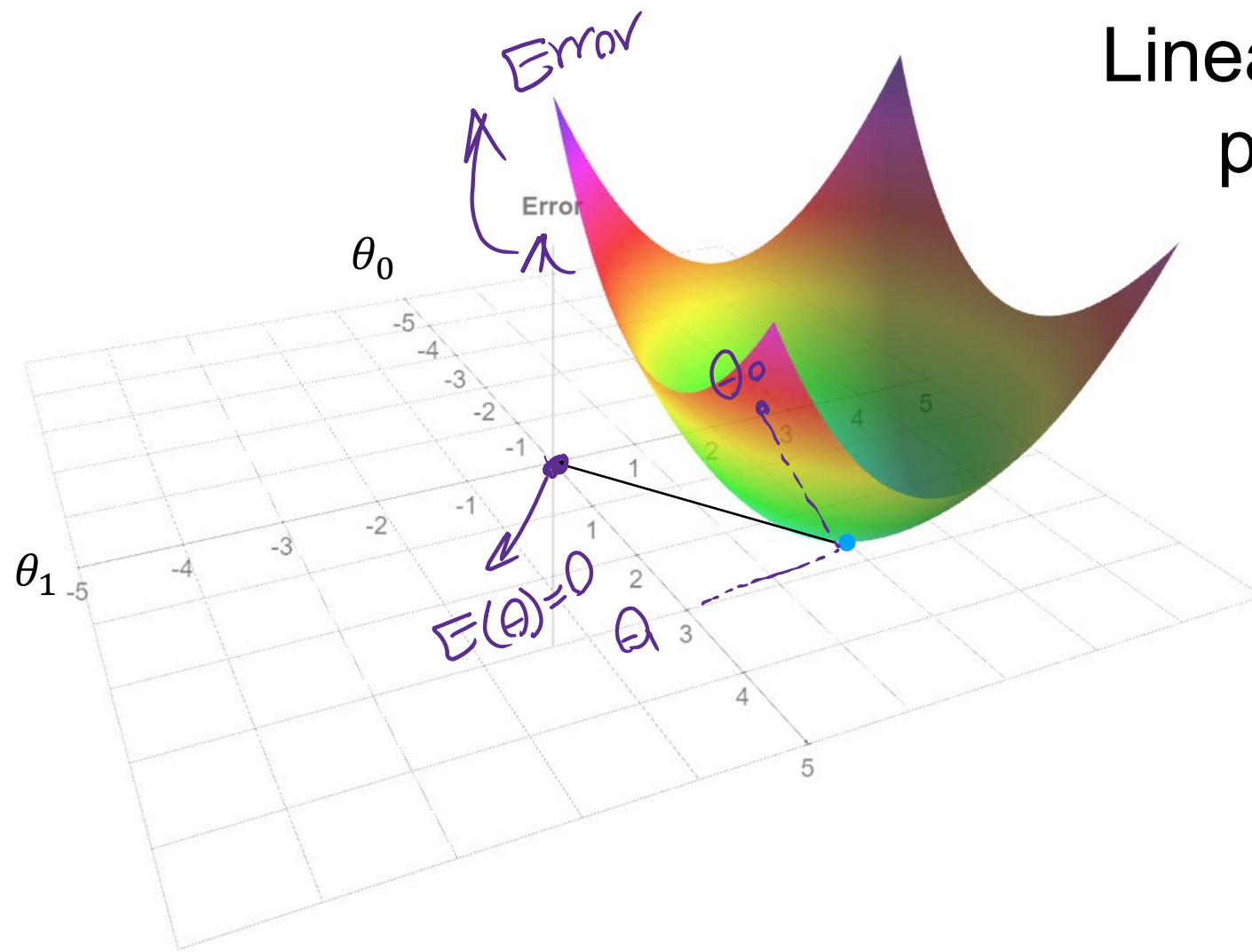


# Project the same graph on x-y using contour plot

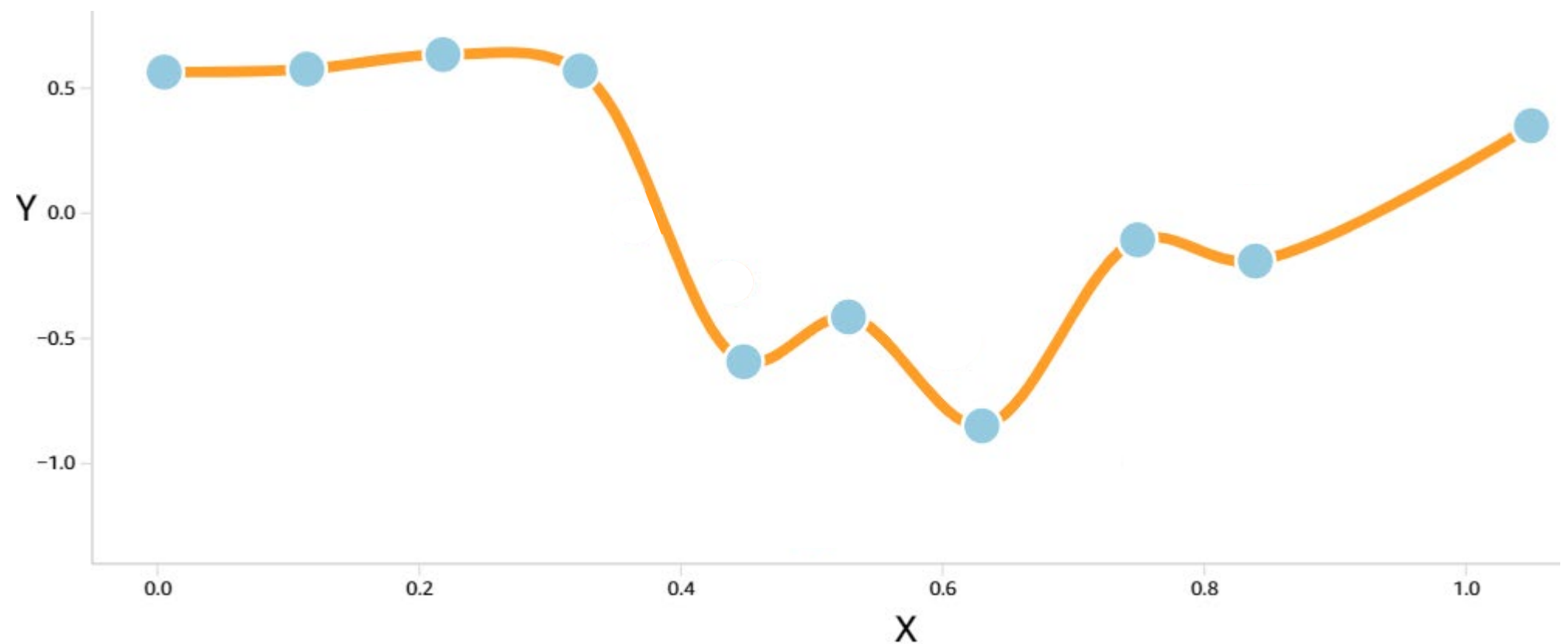


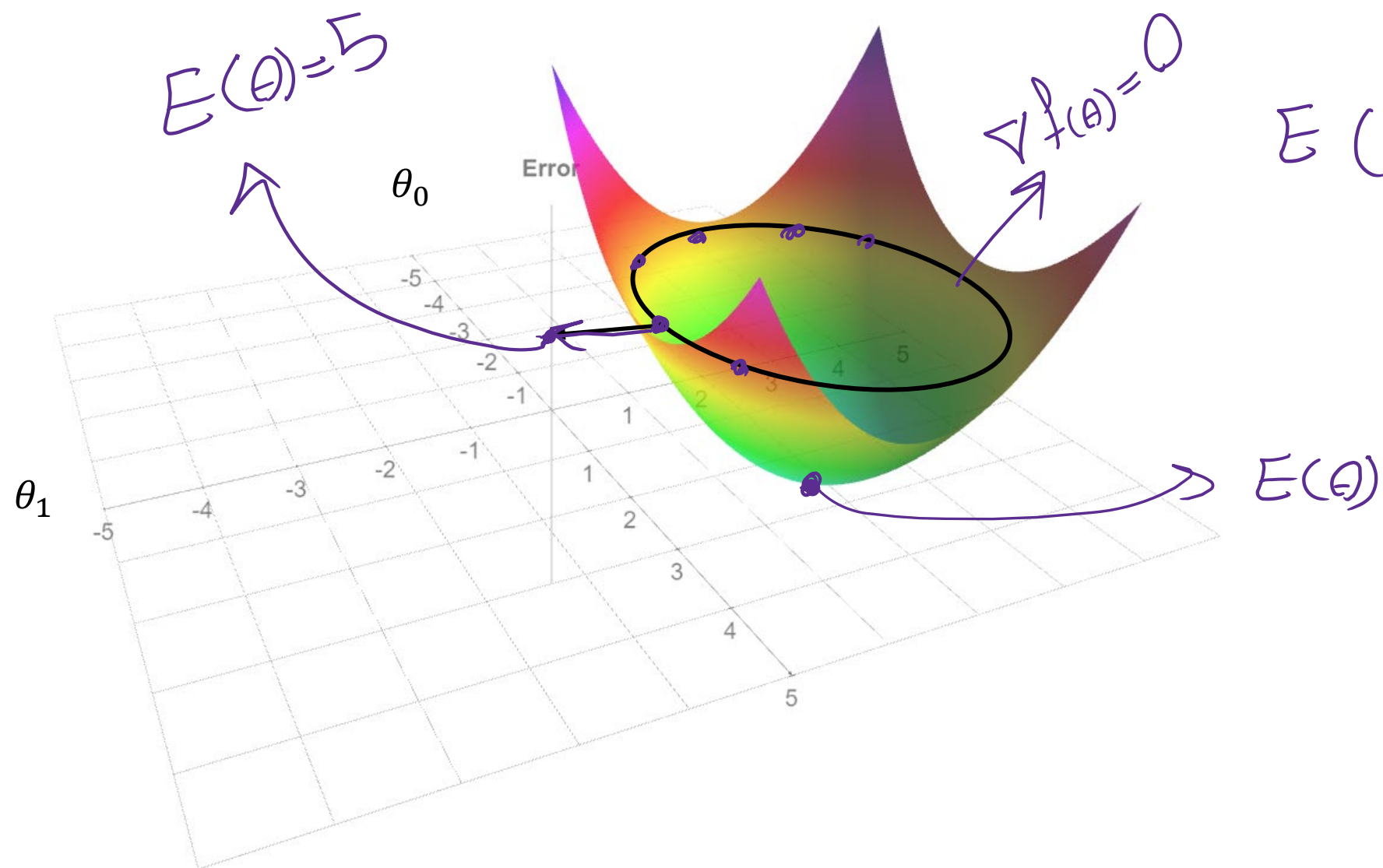


# Linear regression with a very high polynomial degree solution



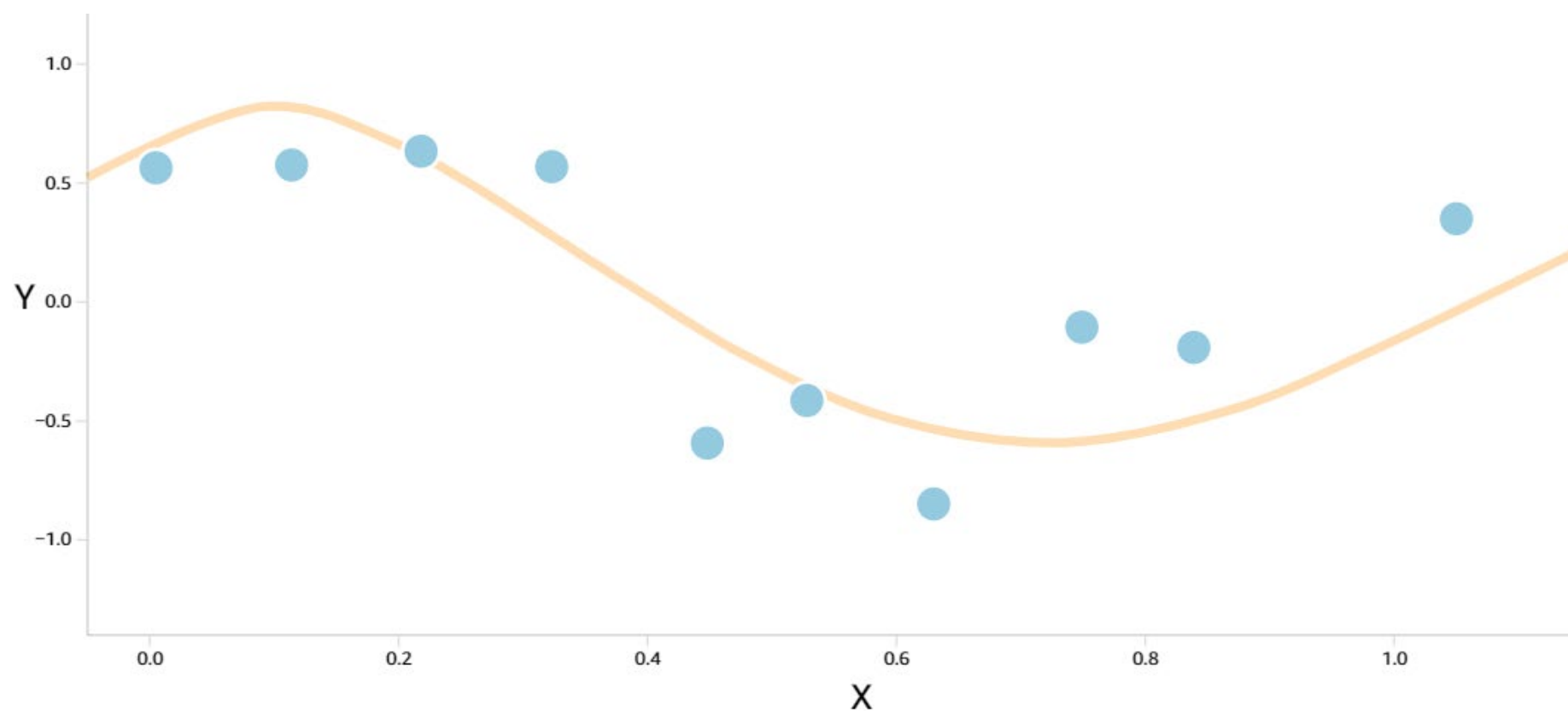
$$y = \theta_0 + \theta_1 z_1$$





$$E(\theta) = \frac{1}{N} \sum (y_a - y_p)^2$$

$$\nabla E(\theta) = 0$$



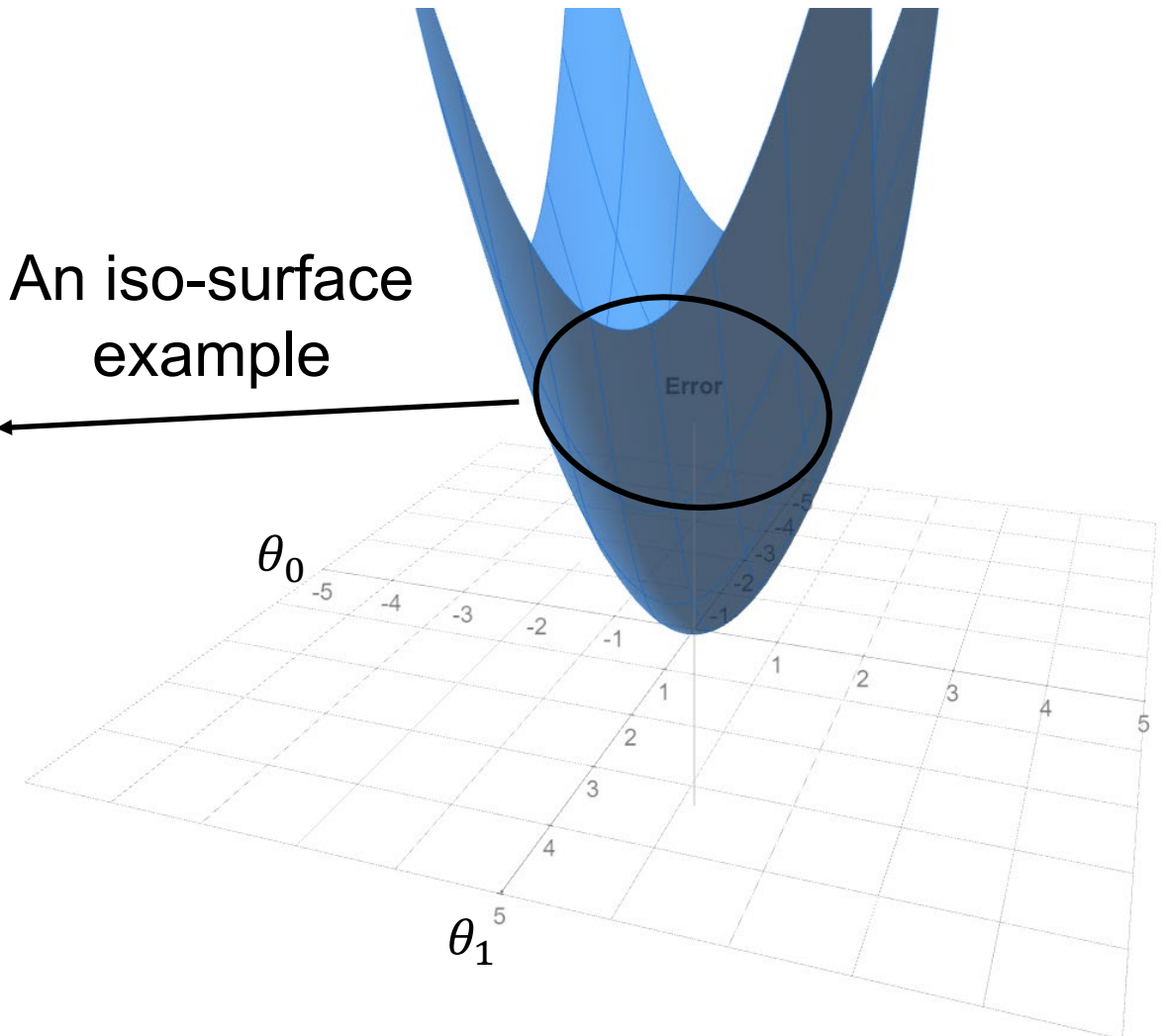


How can we get an optimal solution with a positive error for a model that overfits?

We need to introduce a constraint

$$g(\theta) = \theta_0^2 + \theta_1^2 = \theta^T \theta = C$$

An iso-surface  
example



Error function together with a  
new introduced constraint

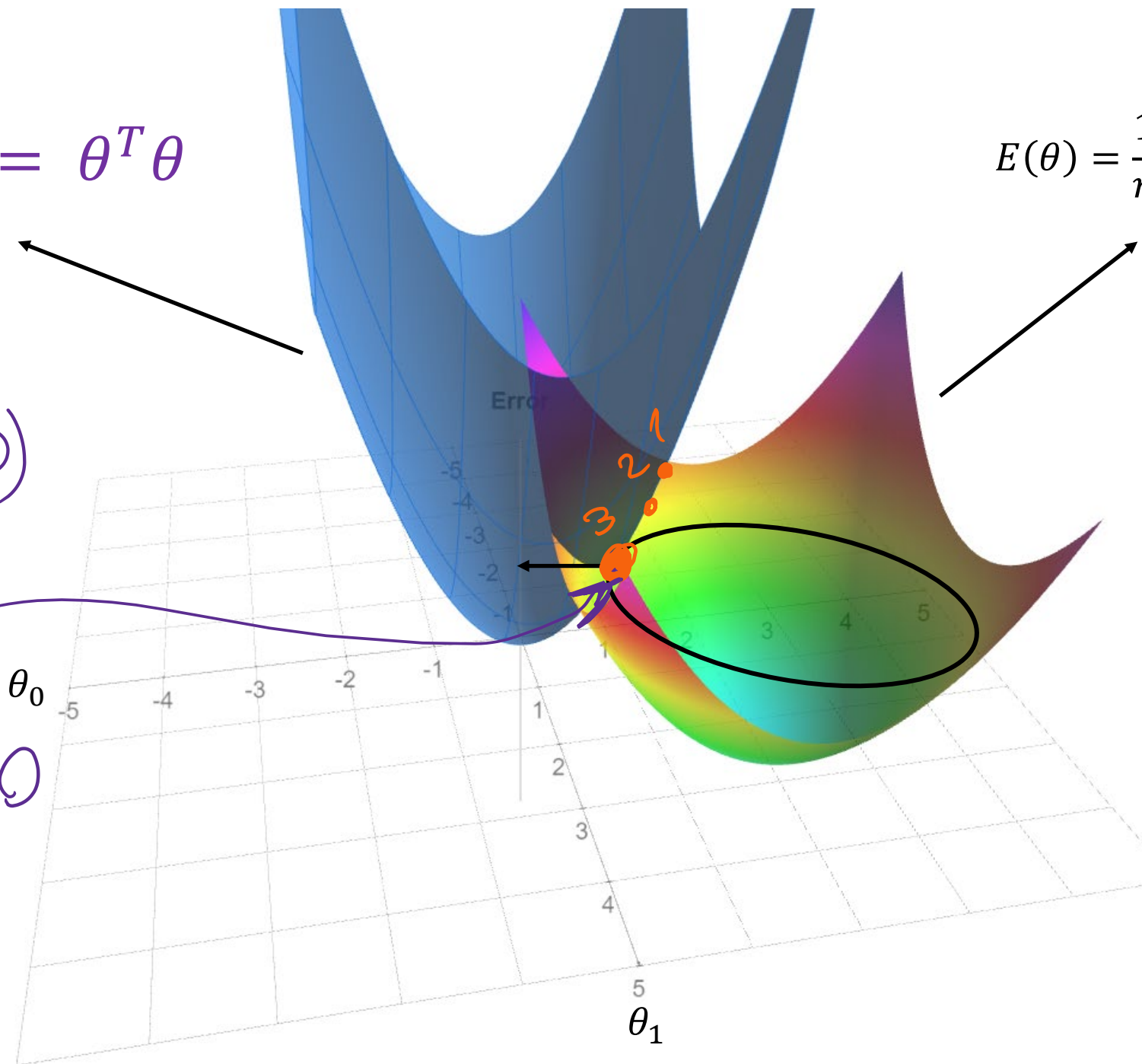
$$\nabla E(\theta) \approx -\nabla g(\theta)$$

$$g(\theta) = \theta_0^2 + \theta_1^2 = \theta^T \theta$$

$$\nabla E(\theta) = -\nabla g(\theta)$$

$$\nabla E(\theta) + \nabla g(\theta) = 0$$

$$E(\theta) = \frac{1}{n} \sum_{i=1}^n (y^i - z_i \theta)^2$$



Let's define the Lagrange function

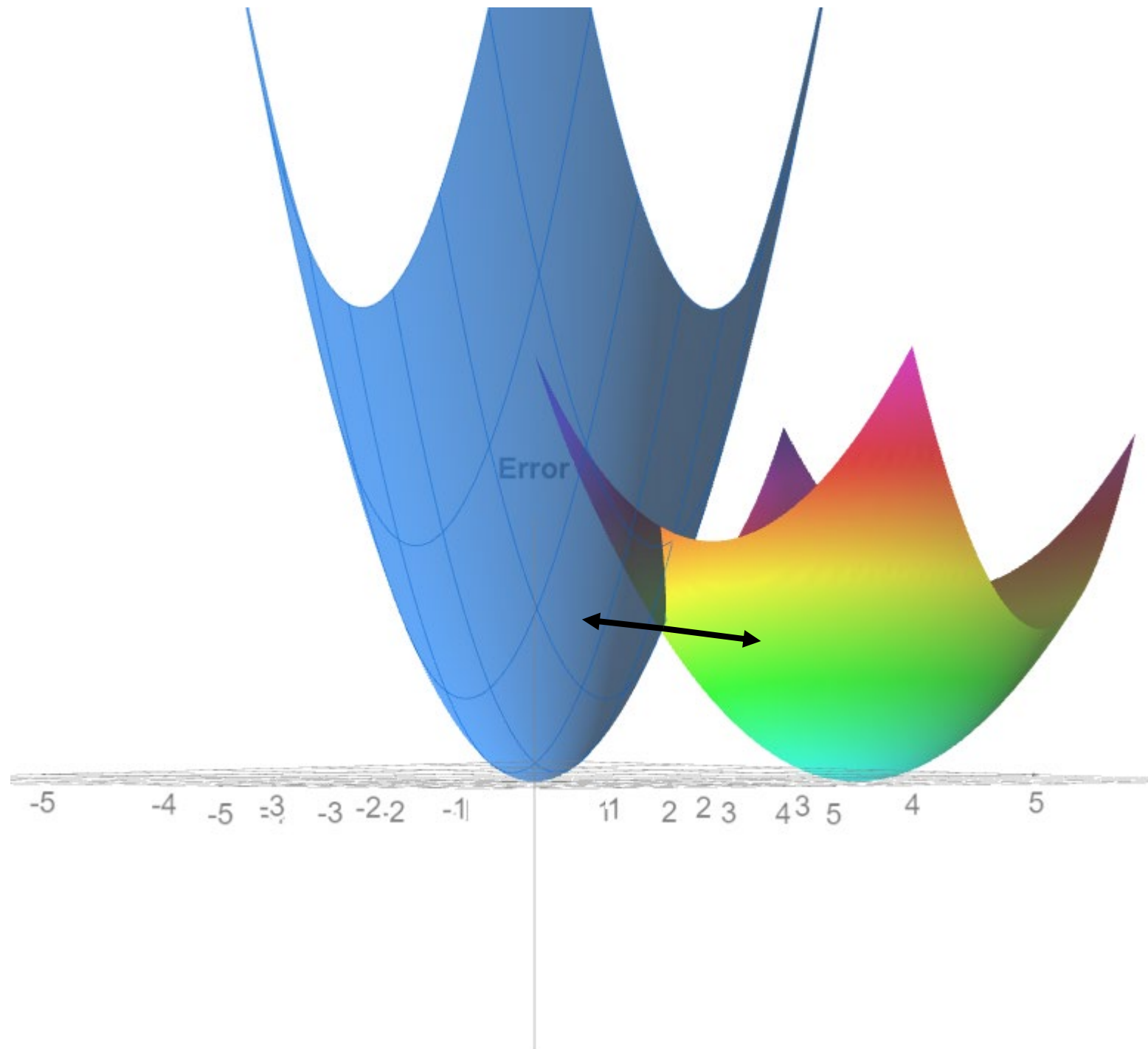
$$L(\theta, \lambda) = E(\theta) + \lambda g(\theta)$$

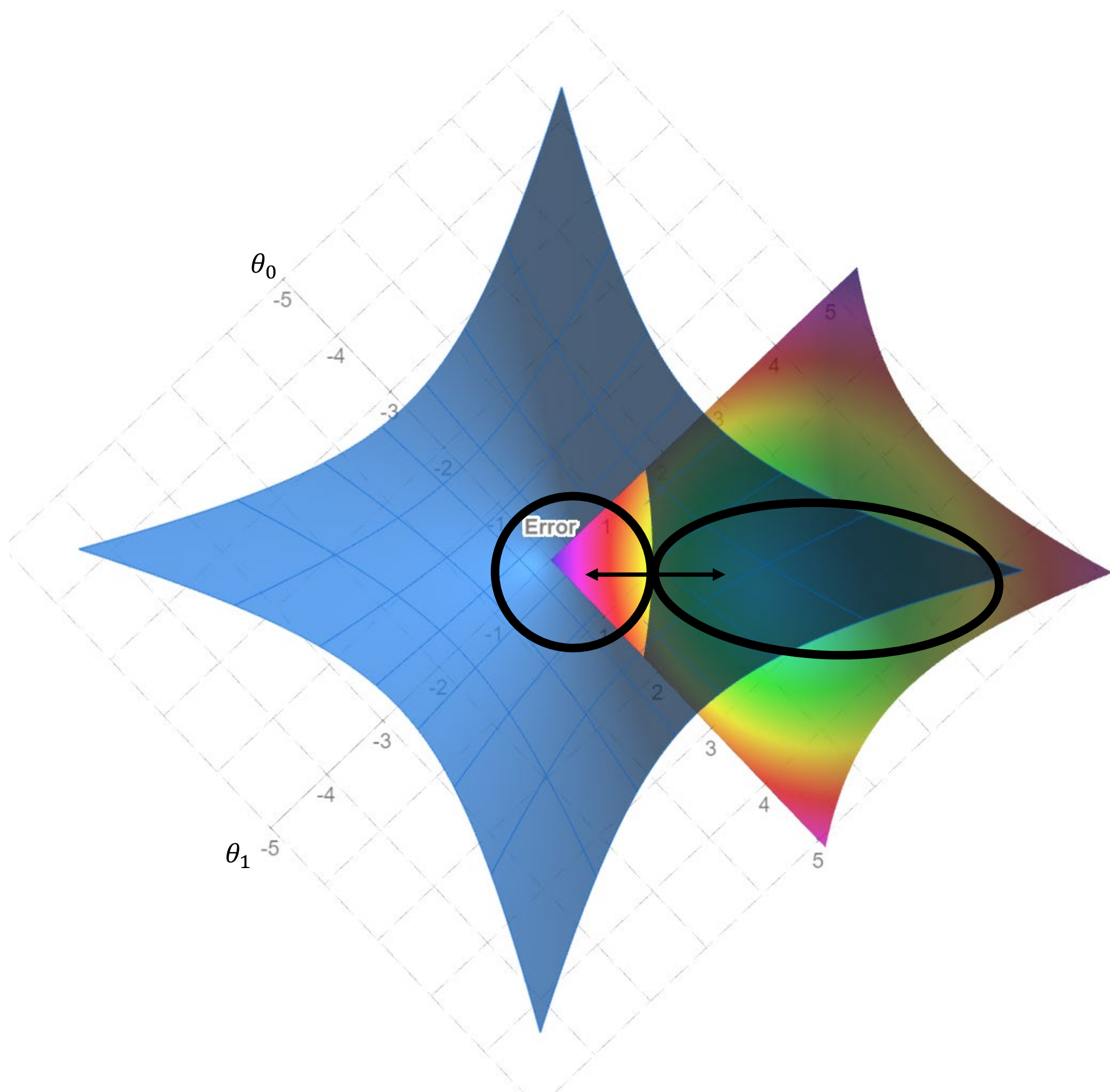
$$L(\theta, \lambda) = E(\theta) + \lambda \theta^T \theta$$

$$\nabla L(\theta, \lambda) = 0 \qquad \nabla [E(\theta) + \lambda \theta^T \theta] = 0$$

$$\nabla [E(\theta)] + \lambda \nabla [\theta^T \theta] = 0$$

How to enforce the gradient of Lagrange function to be zero





# Let's calculate the gradients

Gradient of constraint  $g(\theta)$

$$\nabla[\theta^T \theta] = 2\theta$$

$$\nabla[E(\theta)] + \lambda \nabla[\theta^T \theta] = 0$$

$$\nabla[E(\theta)] = -\lambda \nabla[\theta^T \theta]$$

$$\nabla E(\theta) = -2\lambda\theta$$

$\lambda$  is an outsider

$\lambda$  is a hyperparameter

$$\nabla E(\theta) + 2\lambda\theta = 0$$

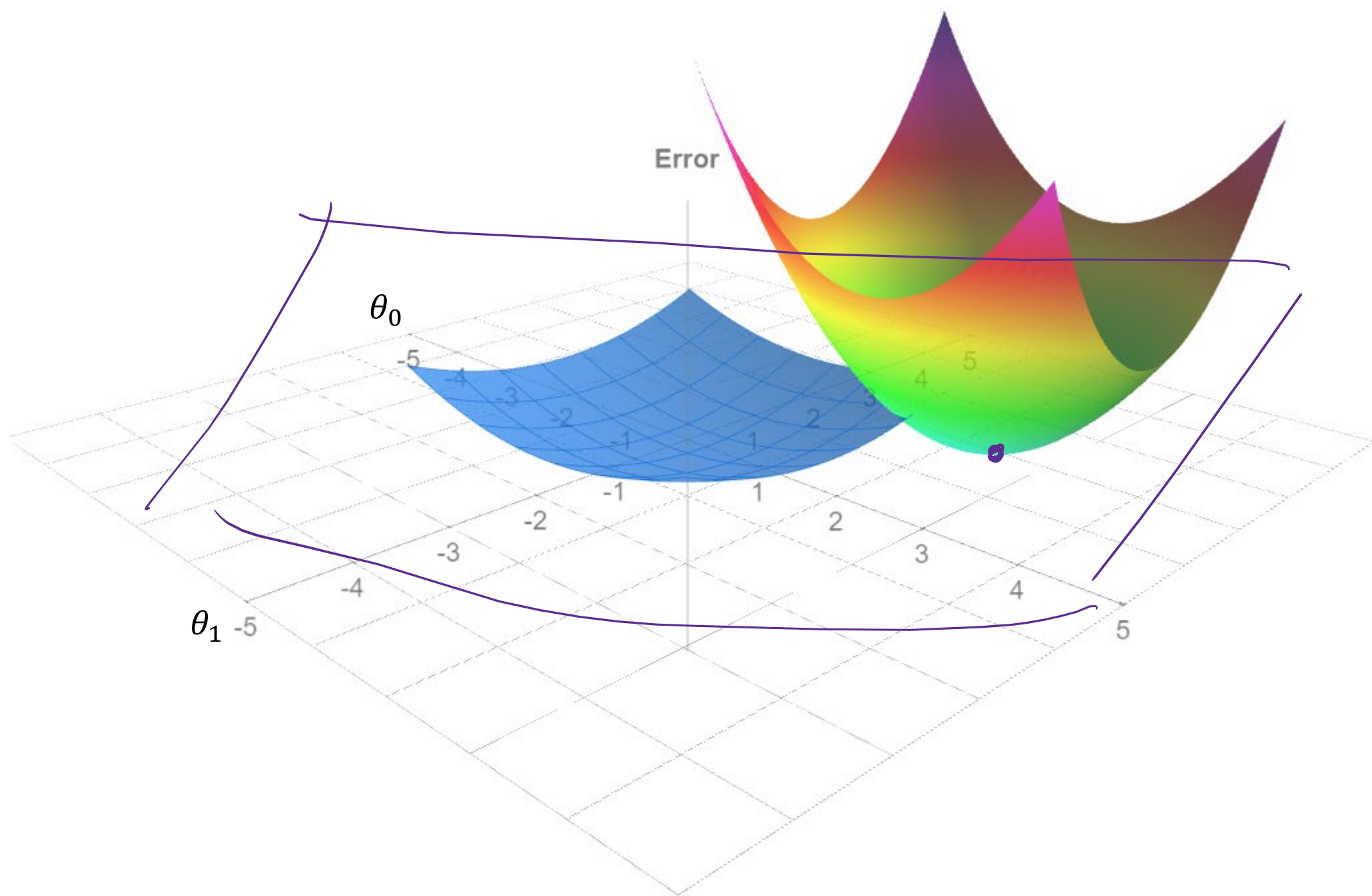
Let's do integration

$$E(\theta) + \lambda\theta^T \theta$$

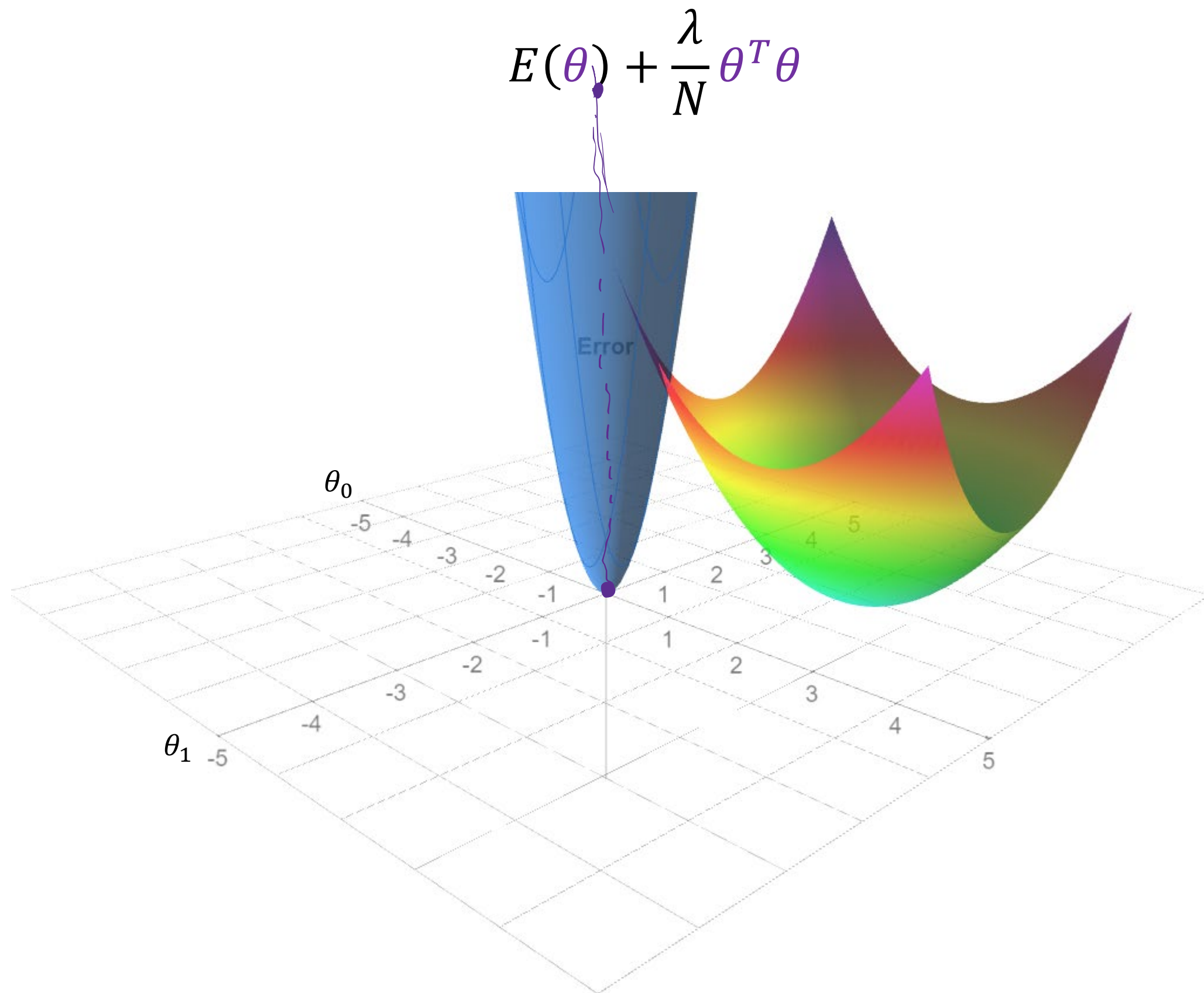


# The effect of low Lambda

$$E(\theta) + \frac{\lambda}{N} \theta^T \theta$$



# The effect of high Lambda





# Regularized Learning

Now we know Why this term leads to the regularization of parameters

Minimize  $E(\theta) + \lambda \theta^T \theta$

Regularized Error

$$\tilde{E}(\theta)$$


$$= \frac{1}{N} \sum_{i=1}^n (y^i - z_i \theta)^2 + \frac{\lambda}{2N} \|\theta\|_2^2$$

$$\|\theta\|_2 = \sqrt{\theta^T \theta}$$

$$\|\theta\|_2^2 = \theta^T \theta$$

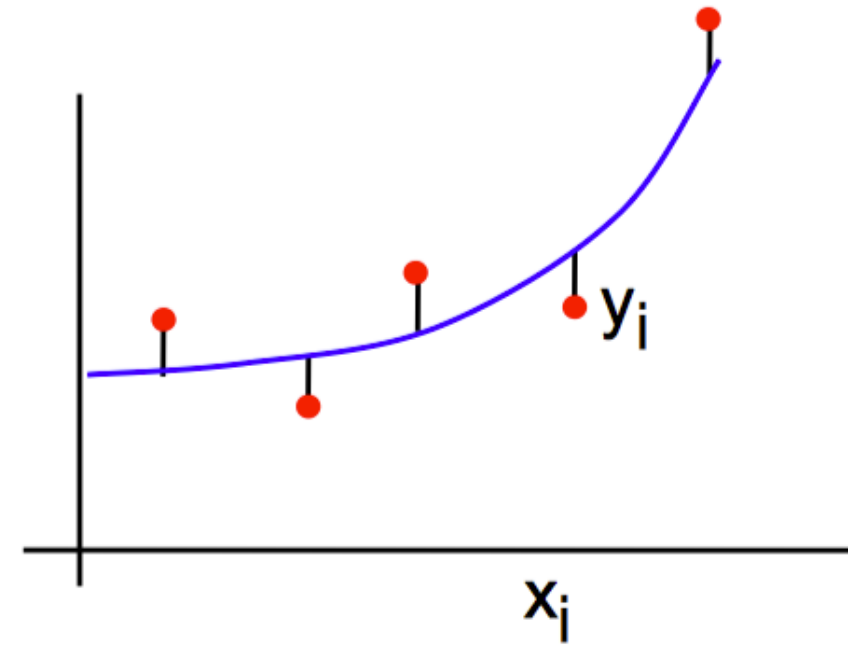
L2 Regularization term

# Outline

- Overfitting and regularized learning
- Ridge regression 
- Lasso regression
- Determining regularization strength

# Ridge Regression

$$\tilde{E}(\theta) = \frac{1}{N} \sum_{i=1}^n (y^i - z_i \theta)^2 + \lambda \|\theta\|_2^2$$



$$\theta_0 + \theta_1 z_1 + \theta_2 z_2 + \cdots + \theta_d z_d + \epsilon = \mathbf{z} \boldsymbol{\theta}$$

General form

$$\tilde{E}(\theta) = \frac{1}{N} \sum_{i=1}^n (y^i - z_i \theta)^2 + \lambda \|\theta\|_2^2$$

Matrix form

$$\tilde{E}(\theta) = \frac{1}{N} (y - z\theta)^T (y - z\theta) + \lambda \|\theta\|_2^2$$

$$\frac{\partial \tilde{E}(\theta)}{\partial \theta} = -z^T (y - z\theta) + \lambda \theta$$

$$\Theta_{\text{overfitted}} = (z^T z)^{-1} z^T y$$

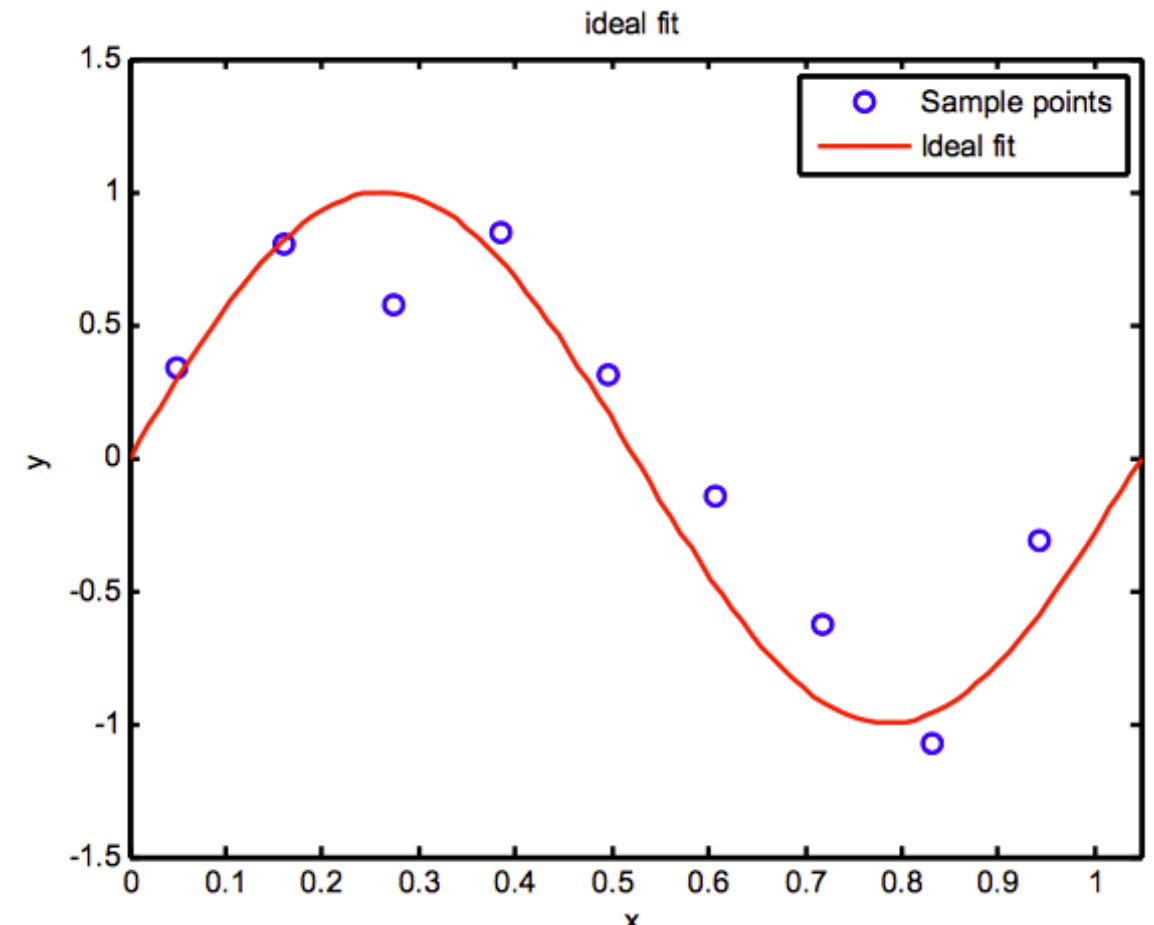
$$(z^T z + \lambda I) \theta = z^T y$$

$$\theta = (z^T z + \lambda I)^{-1} z^T y$$

$$\Theta_{\text{reg}} = \frac{z^T}{(z^T z + \lambda I)} y$$

# Ridge Regression Example

- The red curve is the true function (which is not a polynomial)
- The data points are samples from the curve with added noise in  $y$ .
- There is a choice in both the degree,  $D$ , of the basis functions used, and in the strength of the regularization

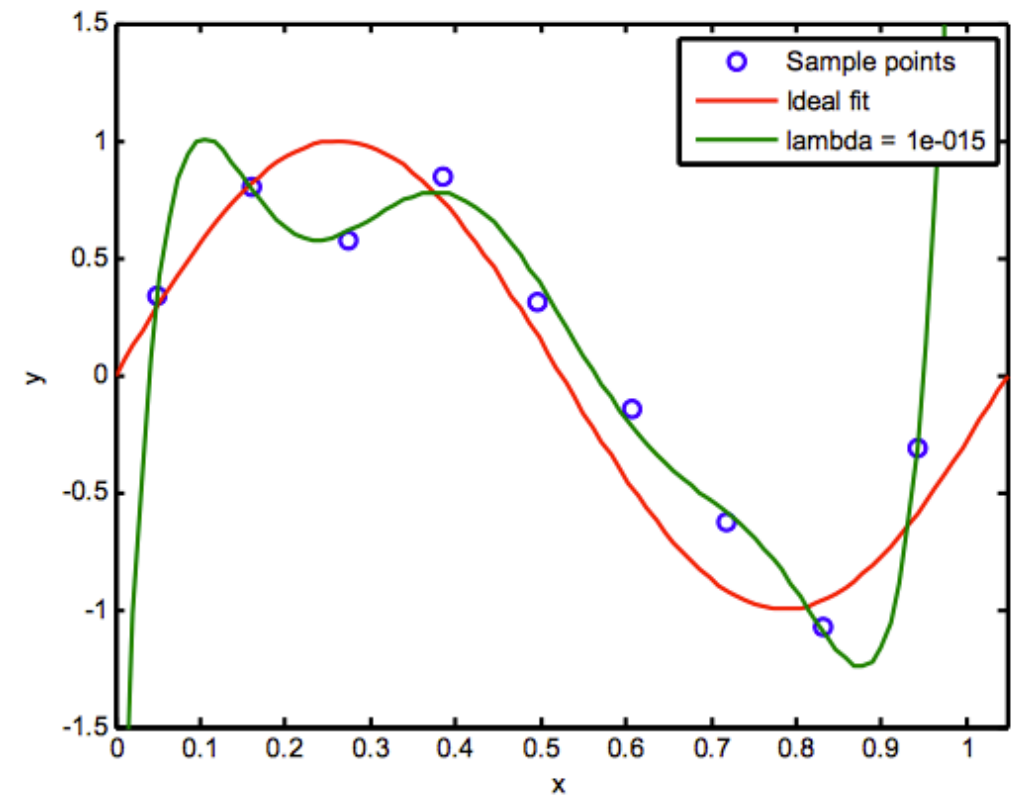
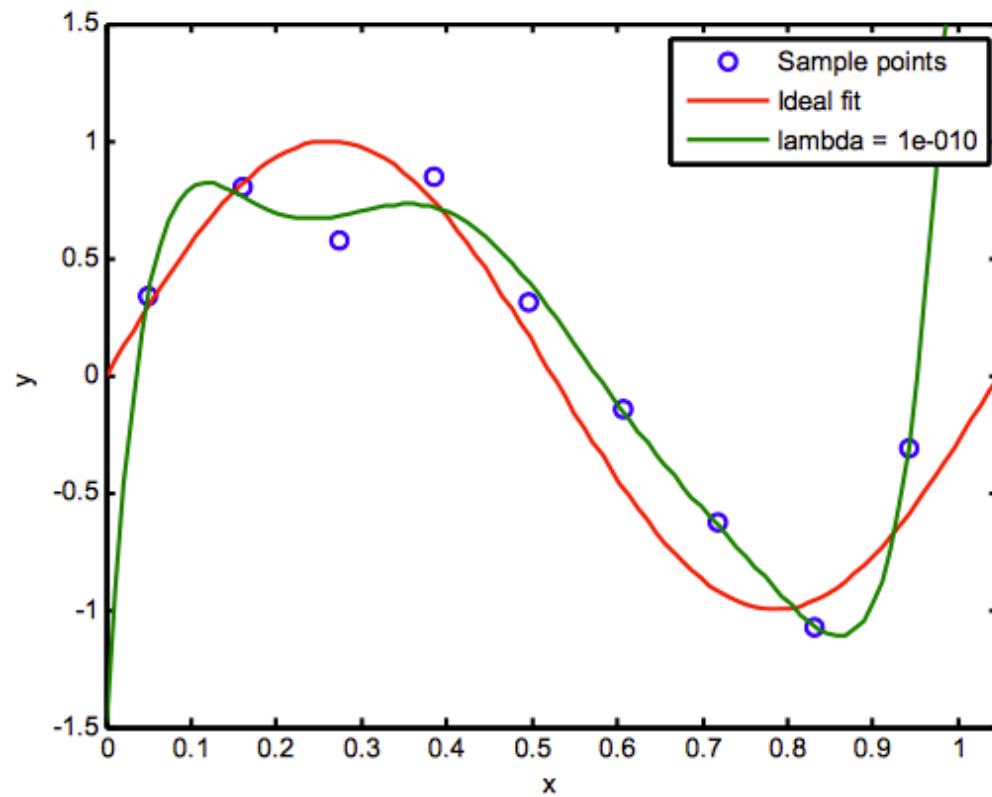
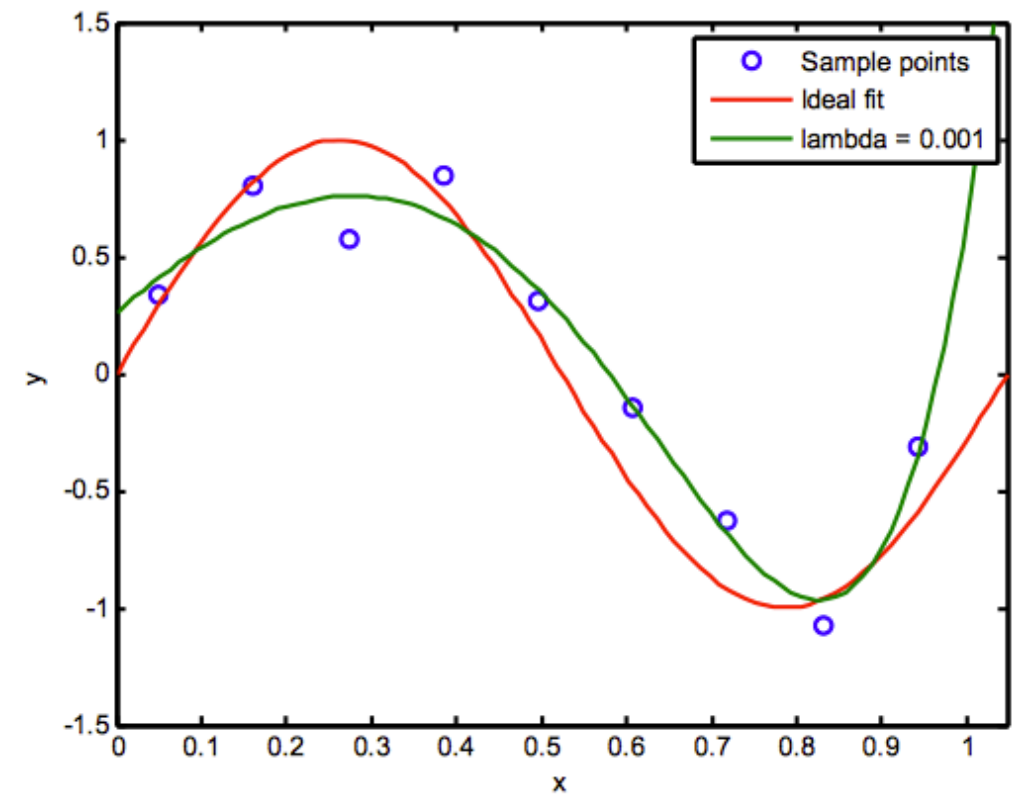
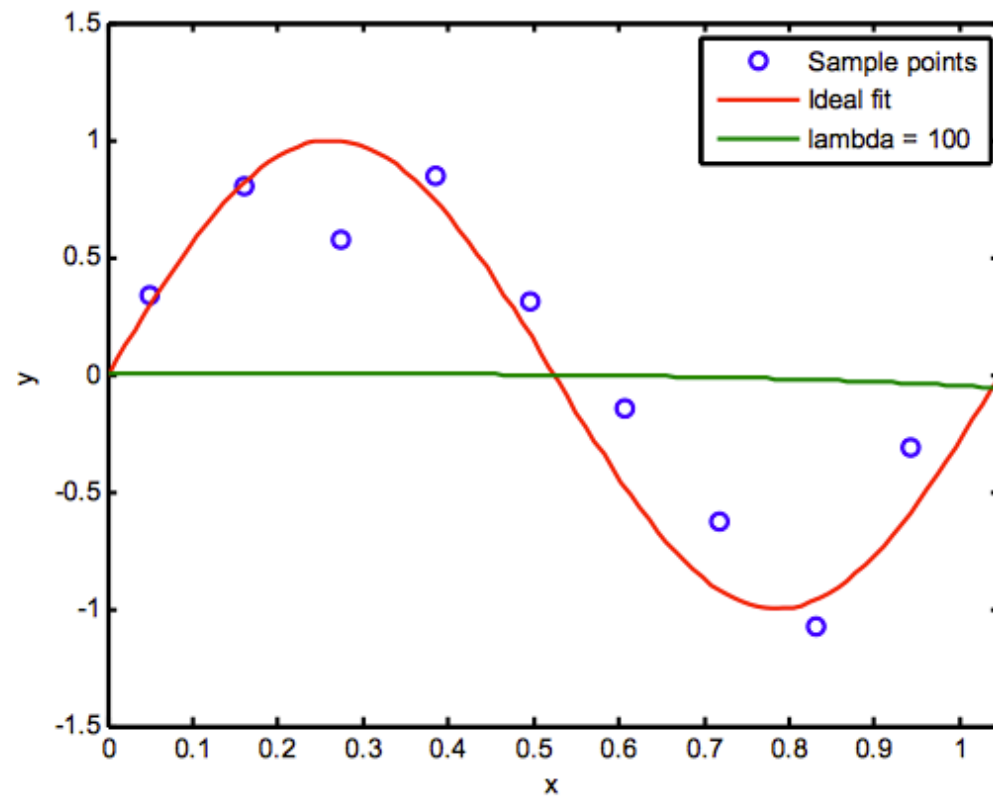


$$f(x, \theta) = z\theta$$

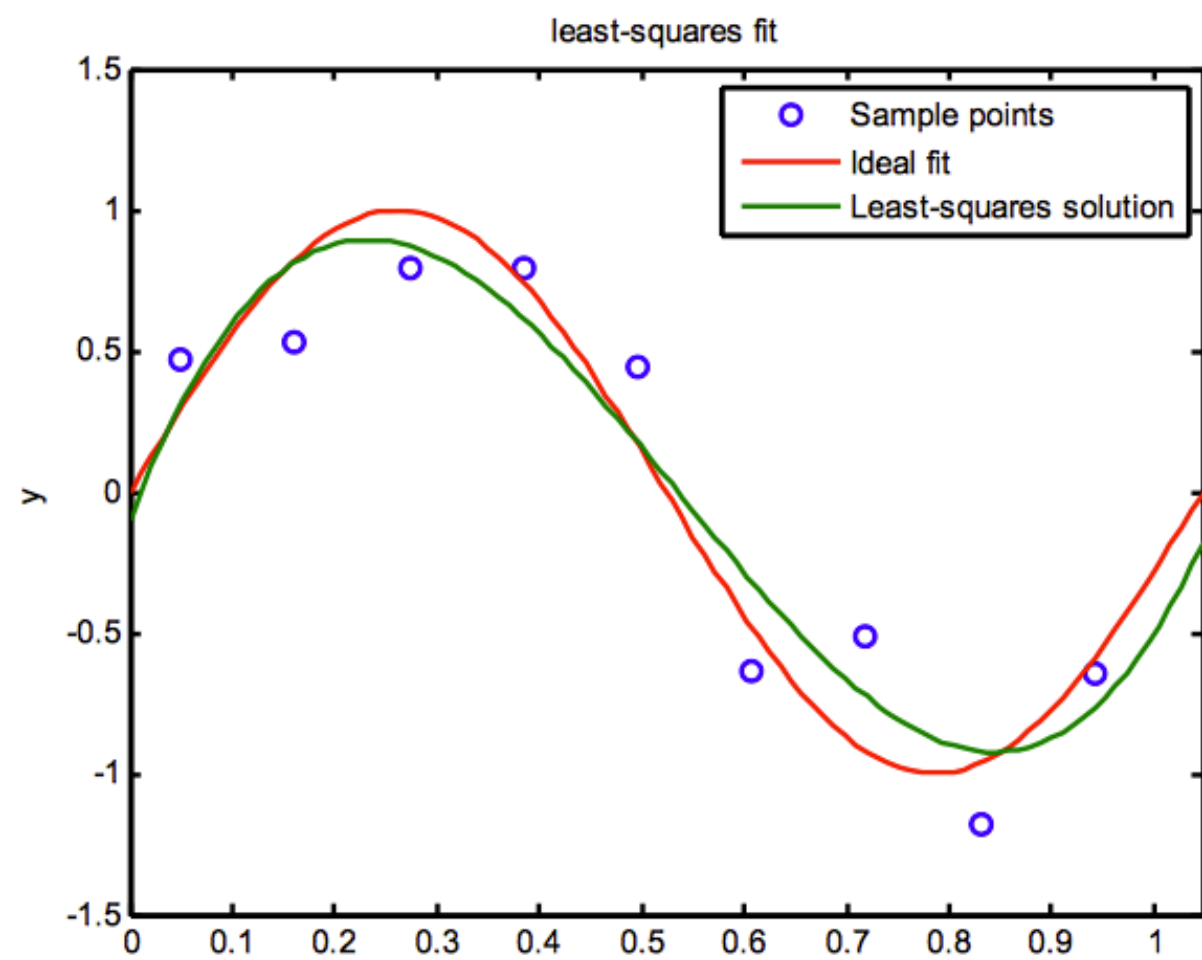
$$z: x \rightarrow z$$

$$\tilde{E}(\theta) = \frac{1}{N} \sum_{i=1}^n (y^i - z_i \theta)^2 + \lambda \|\theta\|_2^2 \quad \theta \in \mathbb{R}^{D+1}$$

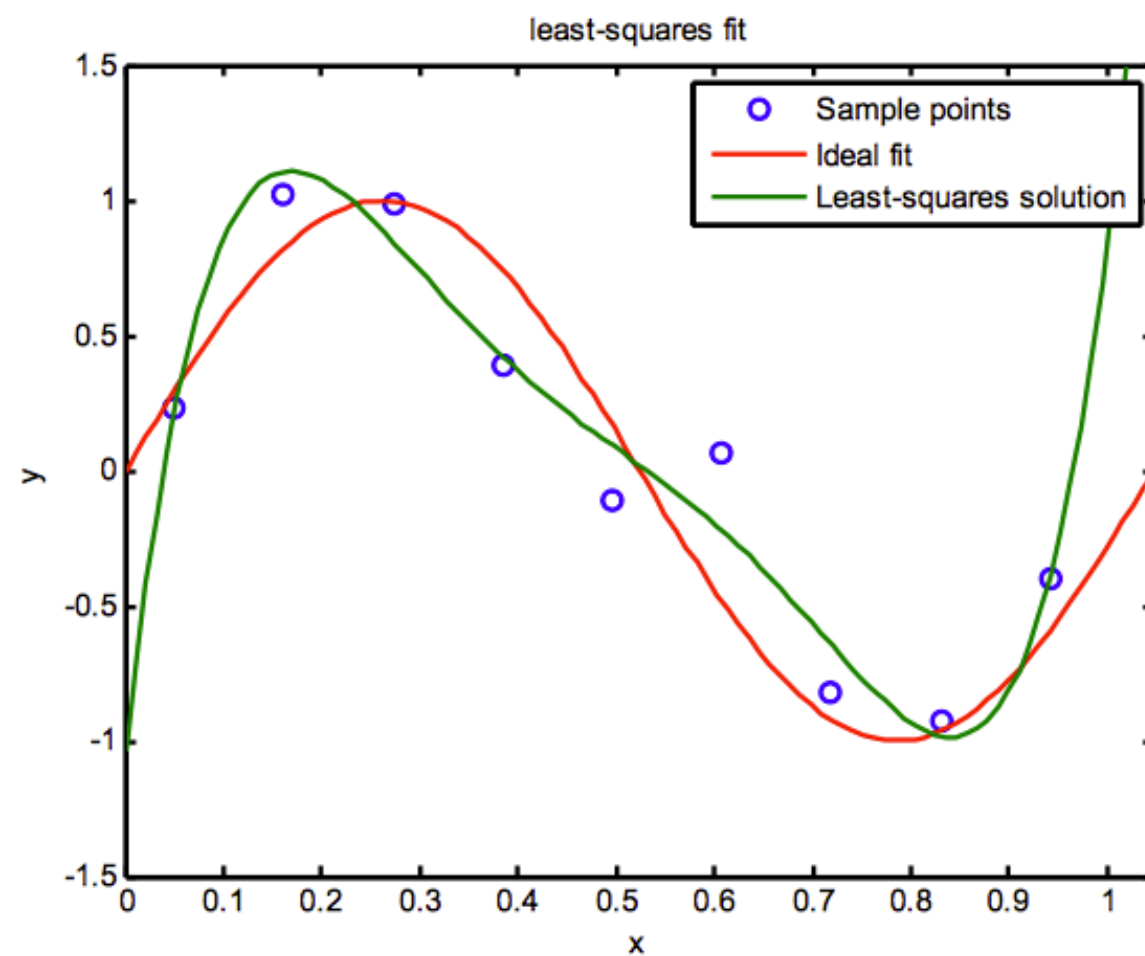
$N = 9$  samples,  $D = 7$




$D = 3$



$D = 5$



# Outline

- Overfitting and regularized learning
- Ridge regression
- Lasso regression 
- Determining regularization strength



# Regularized Regression

$$\tilde{E}(\theta) = \frac{1}{N} \sum_{i=1}^n (y^i - z_i \theta)^2 + \lambda \|\theta\|_2^2$$

Squared loss\Error

$$\frac{1}{N} \sum_{i=1}^n (y^i - z_i \theta)^2$$

L2 Regularizer

$$\lambda \|\theta\|_2^2$$

Now let's look at another regularization choice.

# The Lasso Regularization (L1 norm) and sparsity

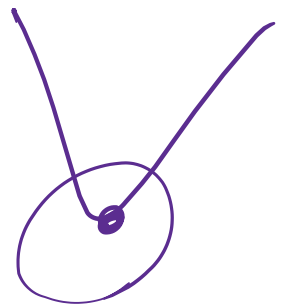
Lasso = **L**east **A**bsolute **S**hrinkage and **S**election **O**perator

$$\tilde{E}(\theta) = \frac{1}{N} \sum_{i=1}^n (y^i - z_i \theta)^2 + \lambda \|\theta\|_1$$

$\|\theta\|_2^2$



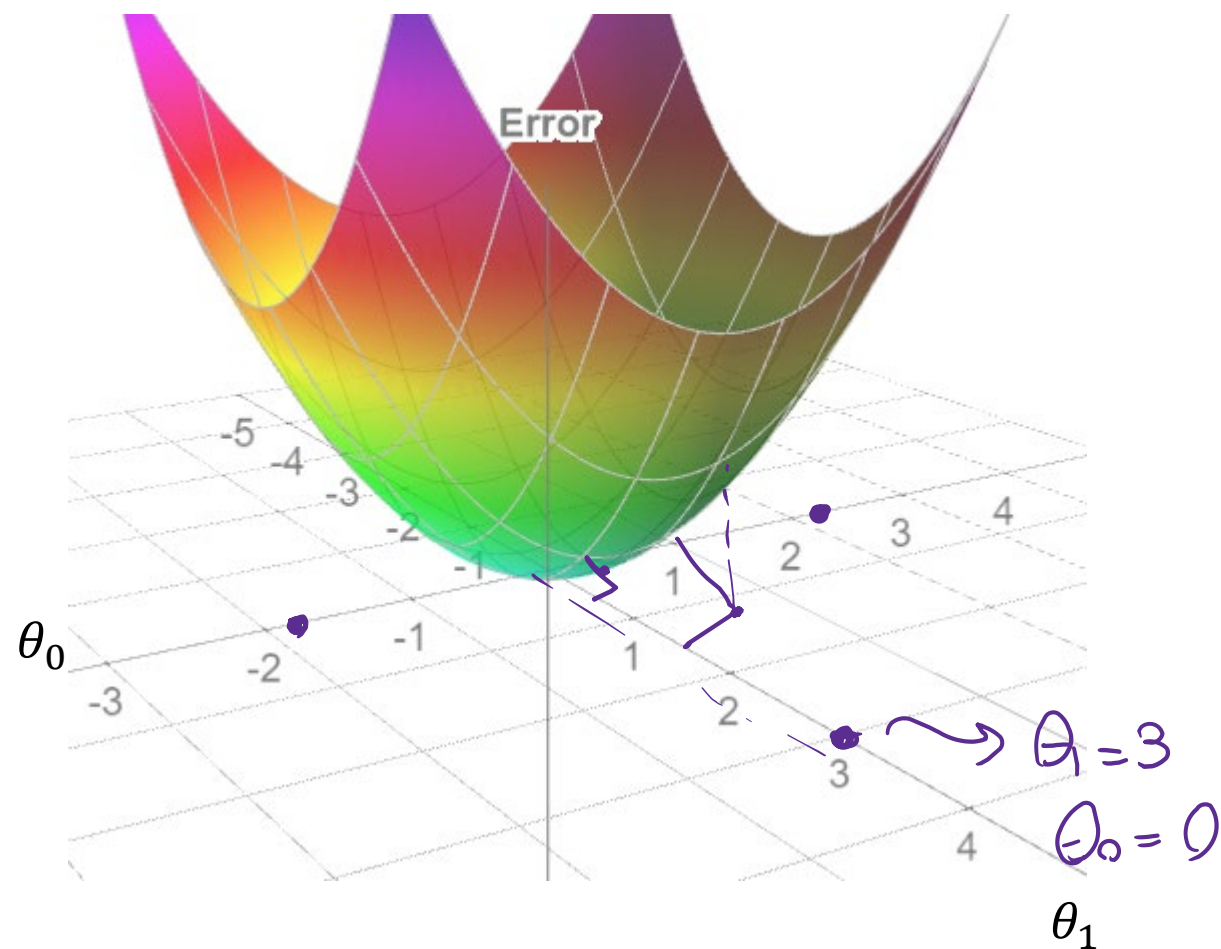
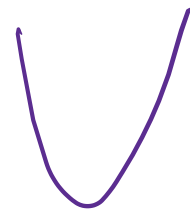
$\|\theta\|_1$



L1 norm induces sparsity. This means that some of the weights become zero, and the feature contribution will be completely removed. L1 Regularizer could be used for feature selection

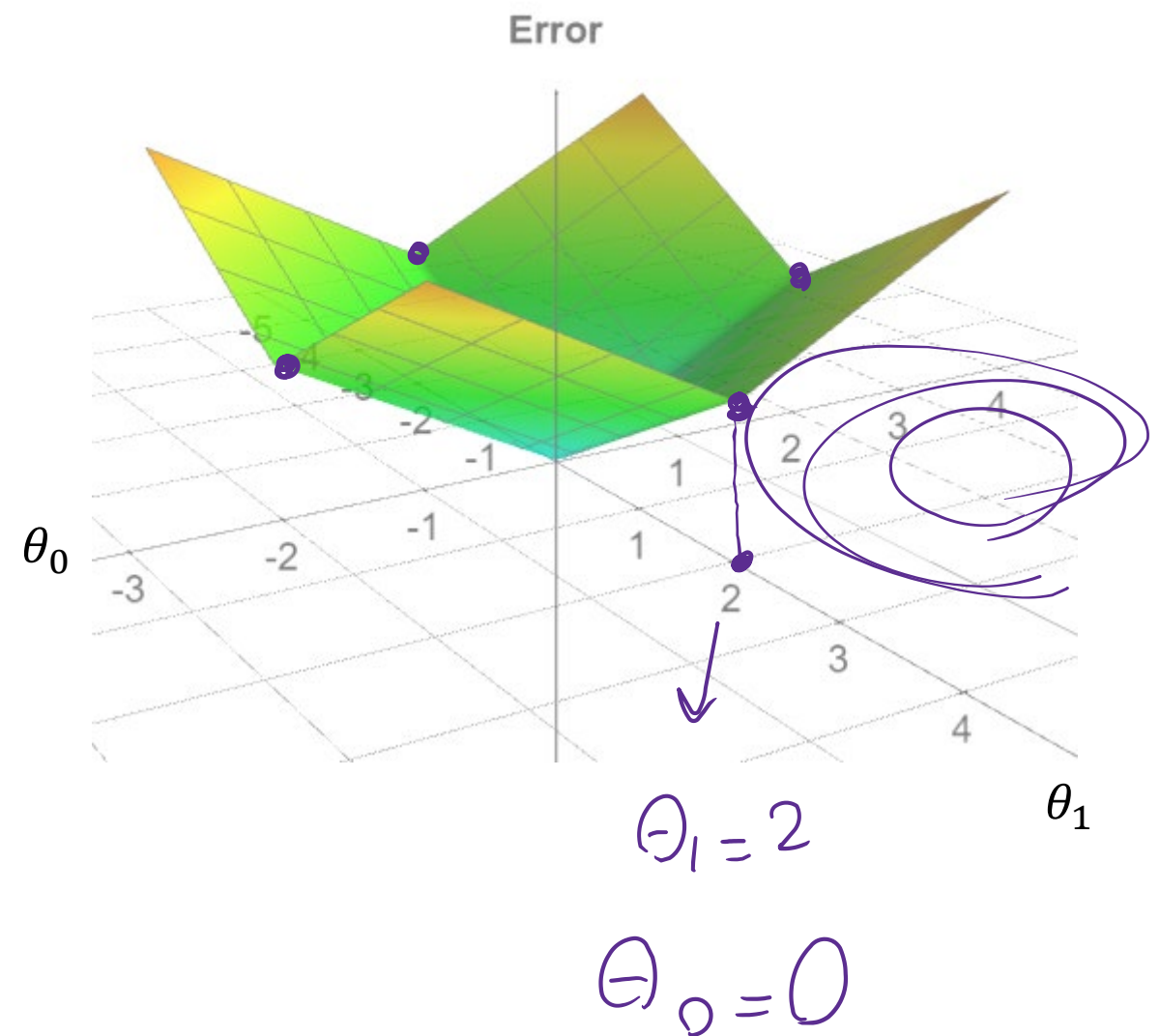
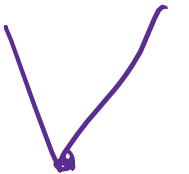
## Ridge Regularizer

$$g(\theta) = \theta_0^2 + \theta_1^2 = \theta^T \theta$$



## Lasso Regularizer

$$g(\theta) = \theta_0 + \theta_1 = \theta$$

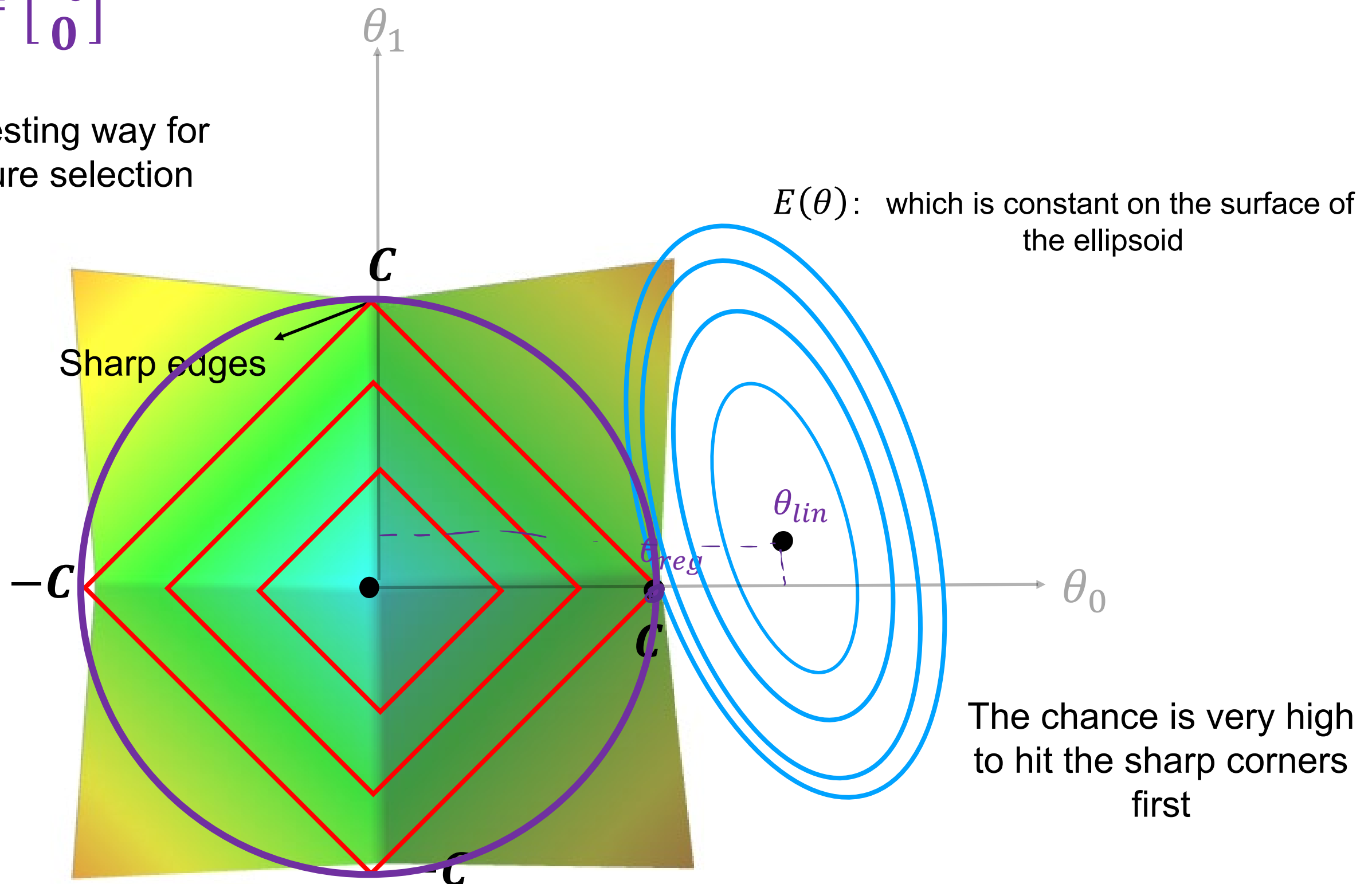


Let's say we have two parameters ( $\theta_0$  and  $\theta_1$ )

$$\text{Min } E(\theta) = \frac{1}{N} (\mathbf{z}\mathbf{w} - y)^T (\mathbf{z}\theta - y) + \lambda \|\theta\|_1$$

$$\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}$$

Interesting way for  
feature selection



# Ridge versus Lasso

## Ridge

$$\tilde{E}(\theta) = \frac{1}{N} (y - z\theta)^T (y - z\theta) + \lambda \|\theta\|_2^2$$

It is a convex model

Both mean squared error  
and L2 regularizer are  
differentiable.

We can get a closed form  
solution

## Lasso

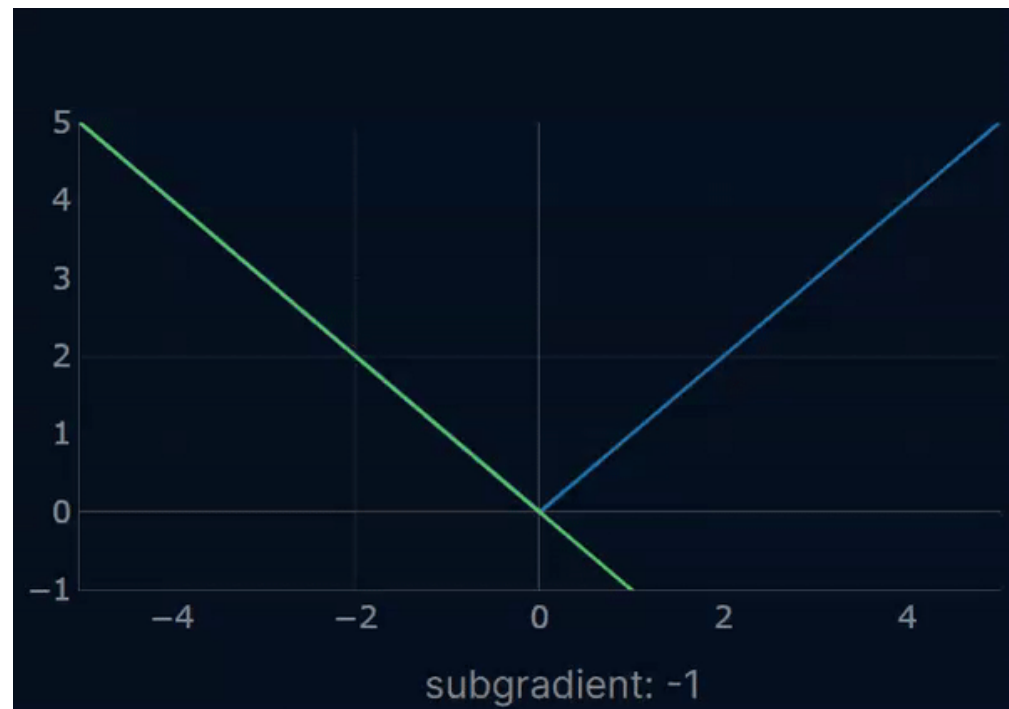
$$\tilde{E}(\theta) = \frac{1}{N} (y - z\theta)^T (y - z\theta) + \lambda \|\theta\|_1$$

It is a convex model

L1 regularizer is NOT  
differentiable.

We can **NOT** get a closed  
form solution

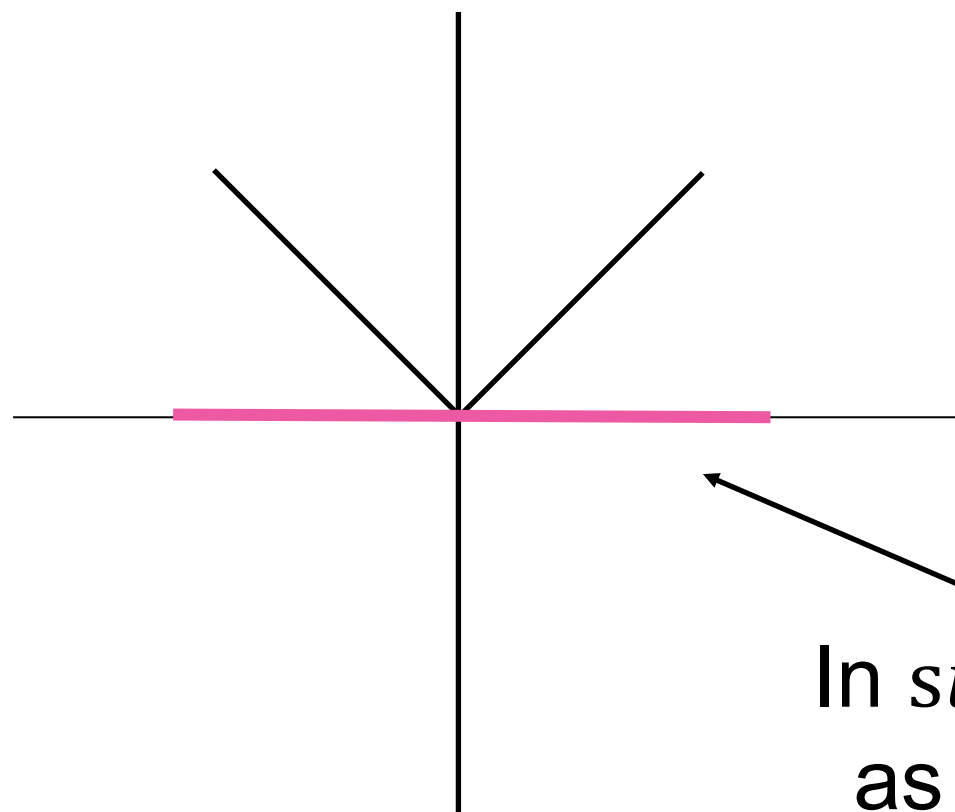
# Sub-gradient Descend in Lasso



$$\tilde{E}(\theta) = \frac{1}{N} (y - z\theta)^T (y - z\theta) + \lambda \|\theta\|_1$$

$$\frac{\partial \tilde{E}(\theta)}{\partial \theta} = -z^T (y - z\theta) + \frac{\partial (\lambda \|\theta\|_1)}{\partial \theta}$$


Using Sub-gradient



$$\frac{\partial \tilde{E}(\theta)}{\partial \theta} = -z^T (y - z\theta) + \lambda \text{sign}(\theta)$$

In *sign* function, we use this sub-gradient line as our under-estimator (below our function)

# Outline

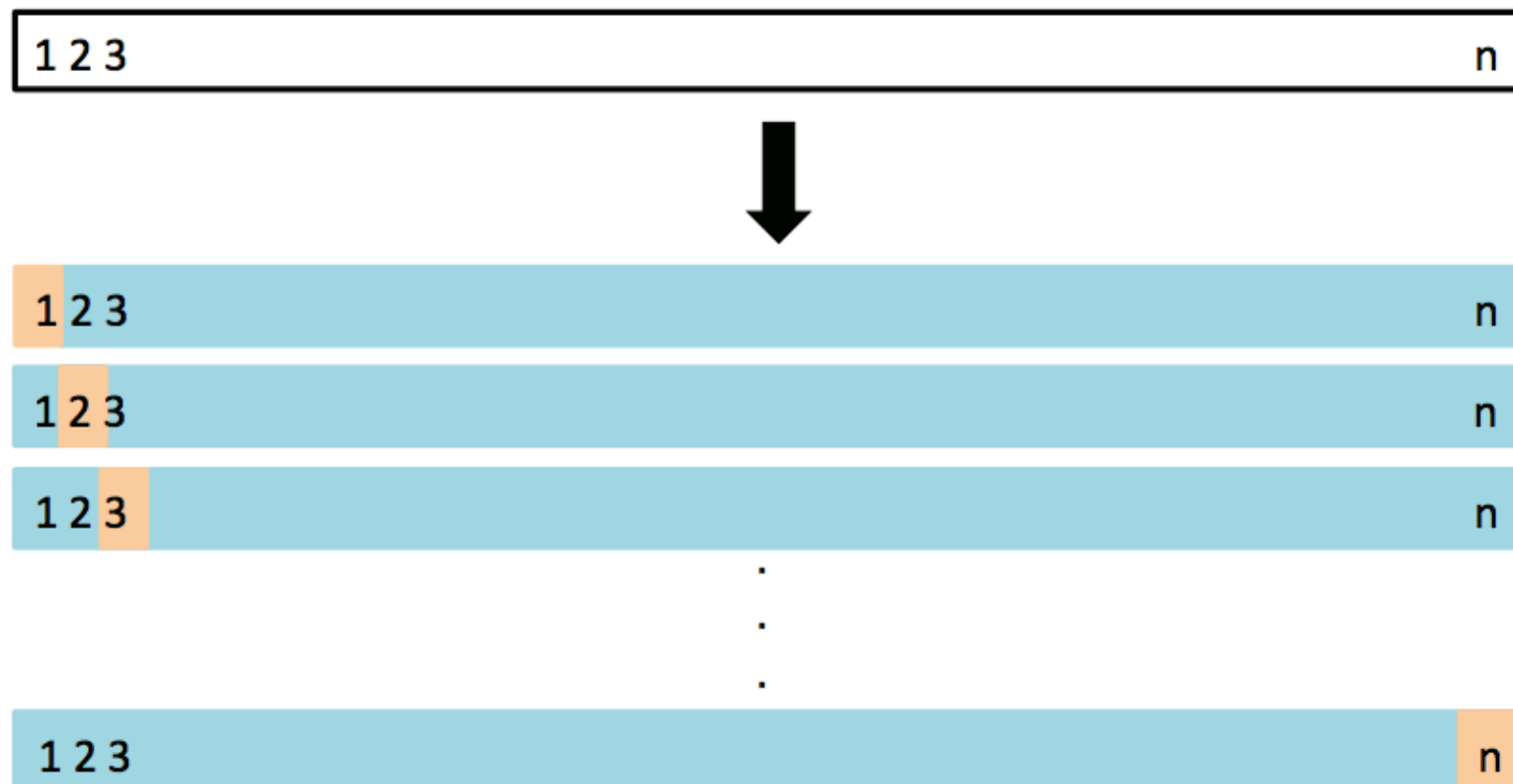
- Overfitting and regularized learning
- Ridge regression
- Lasso regression
- Determining regularization strength 

# Leave-One-Out Cross Validation

For every  $i = 1, \dots, n$ :

- ▶ train the model on every point except  $i$ ,
- ▶ compute the test error on the held out point.

Average the test errors.  $CV_{(n)} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i^{(-i)})^2$





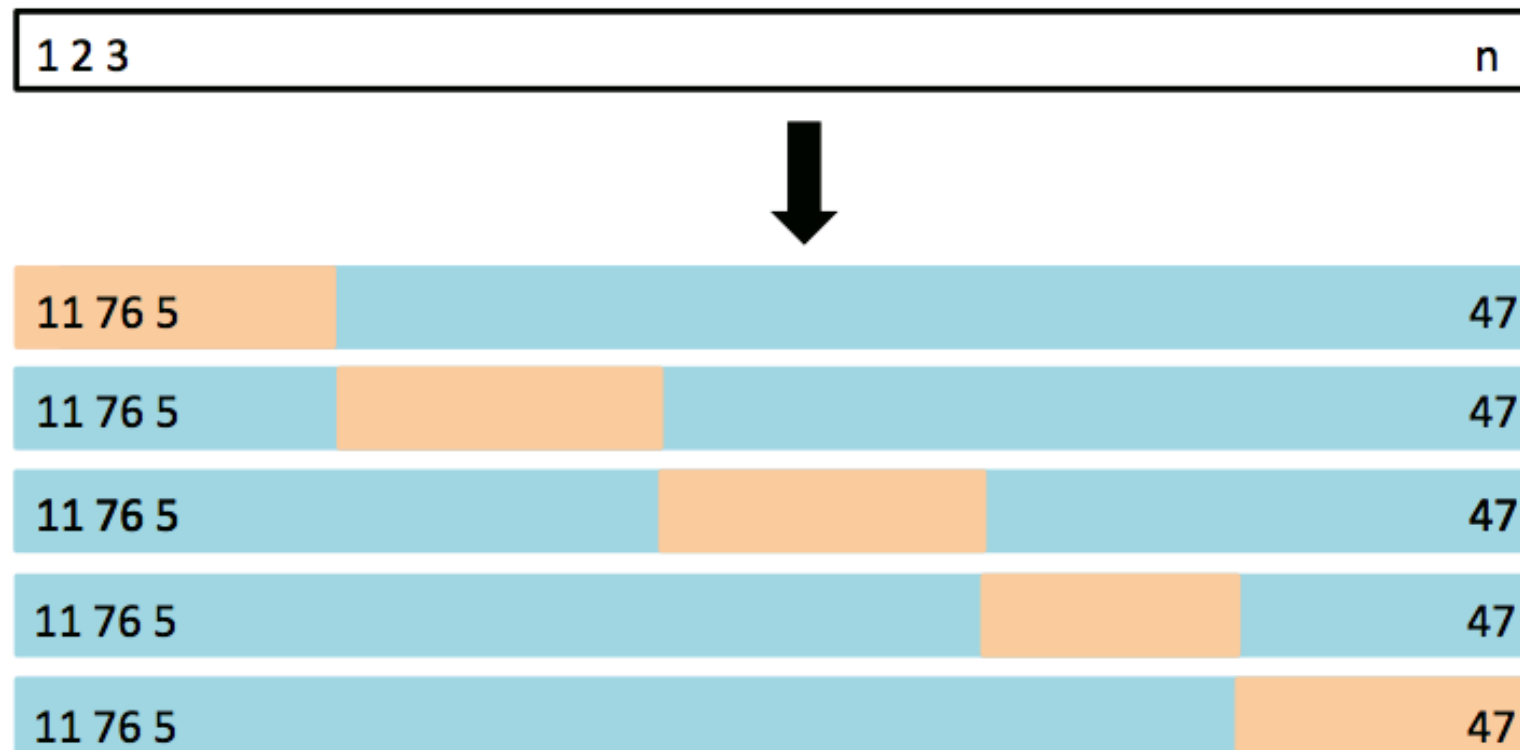
# K-Fold Cross Validation

Split the data into  $k$  subsets or *folds*.

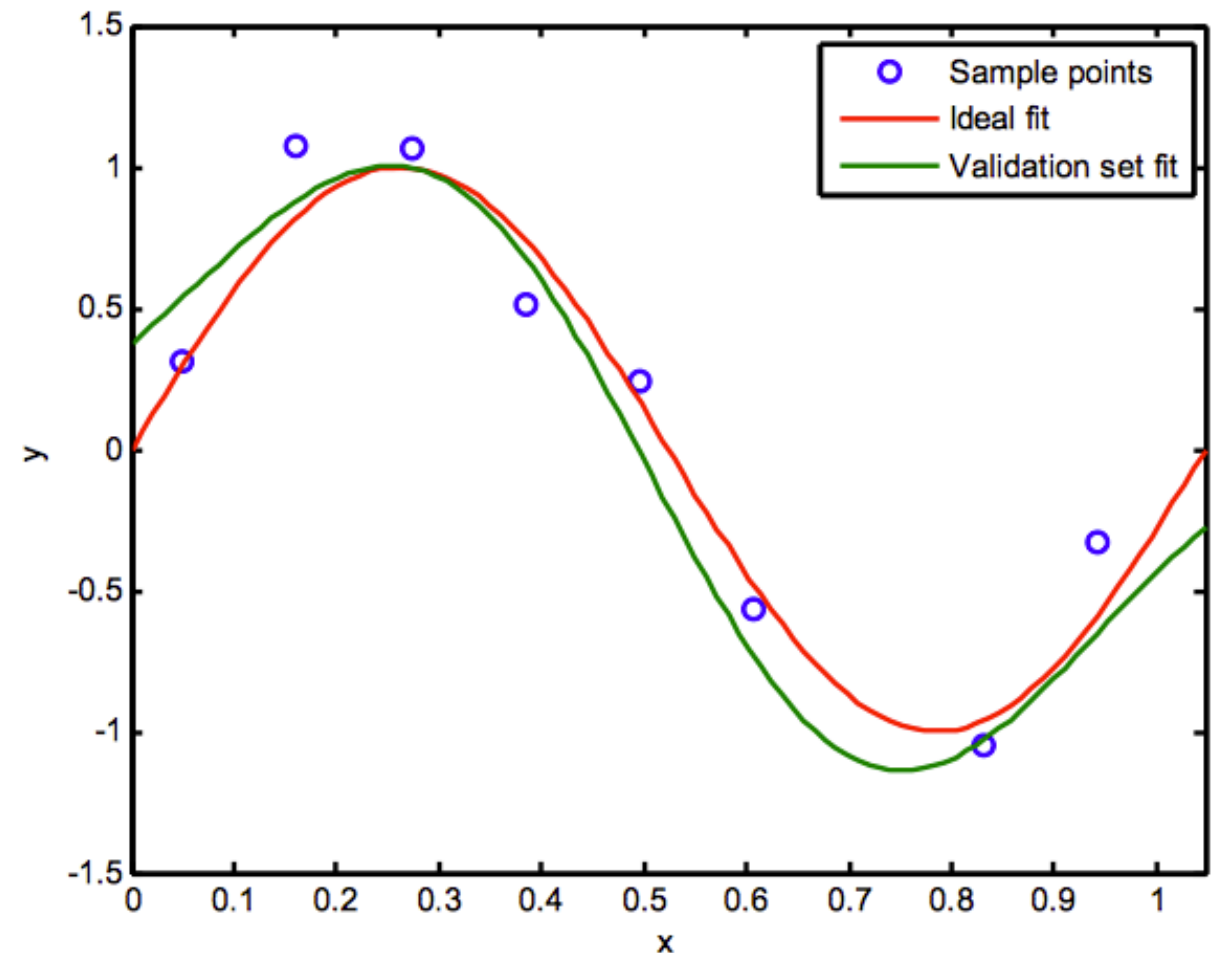
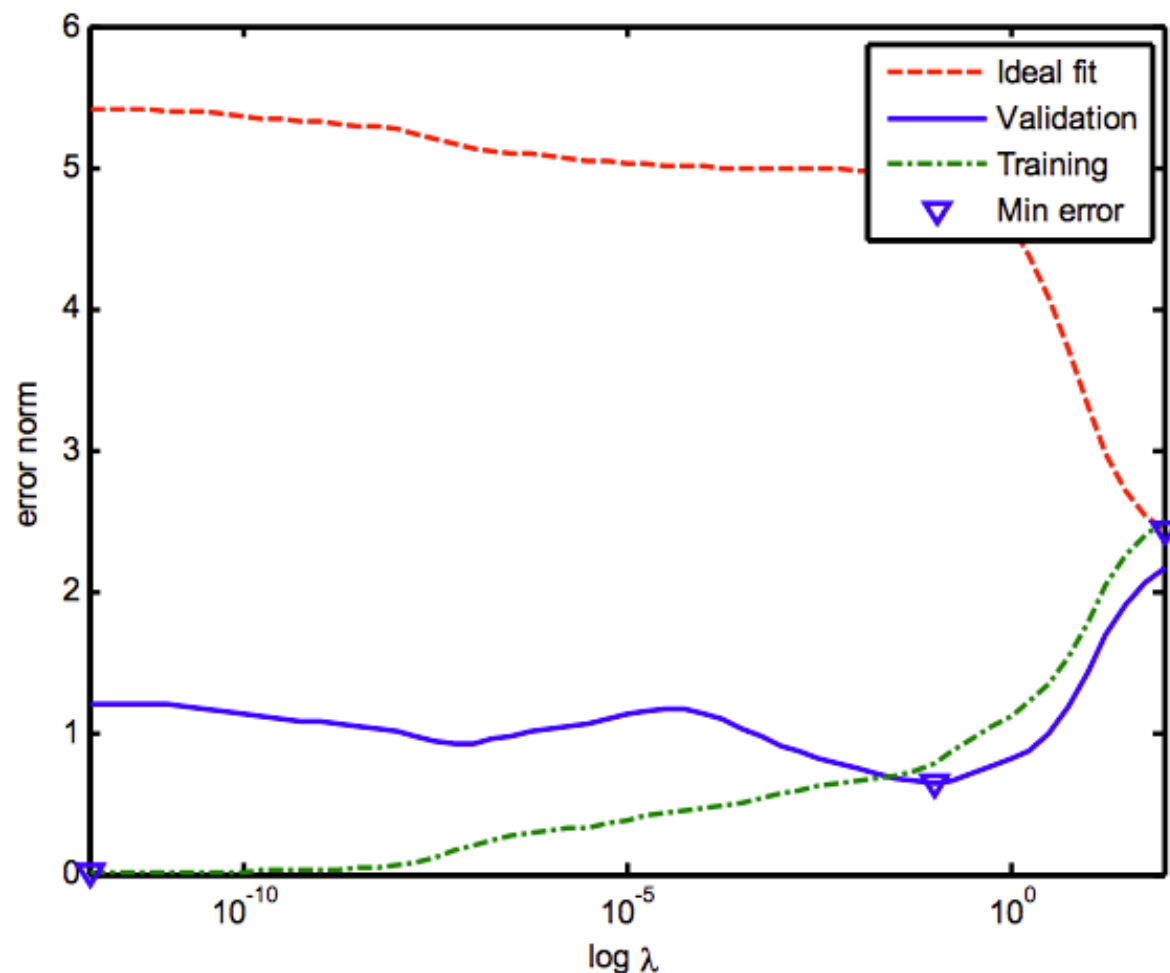
For every  $i = 1, \dots, k$ :

- ▶ train the model on every fold except the  $i$ th fold,
- ▶ compute the test error on the  $i$ th fold.

Average the test errors.



# Choosing $\lambda$ Using Validation Dataset



Pick up the lambda with the lowest  
mean value of rmse calculated by  
Cross Validation approach

# Take-Home Messages

- What is overfitting
- What is regularization
- How does Ridge regression work
- Sparsity properties of Lasso regression
- How to choose the regularization coefficient  $\lambda$