Artificial Intelligence
CE-417, Group 1
Computer Eng. Department
Sharif University of Technology

Spring 2024

By Mohammad Hossein Rohban, Ph.D.

Courtesy: Most slides are adopted from CSE-573 (Washington U.), original slides for the textbook, and CS-188 (UC. Berkeley).

Regression

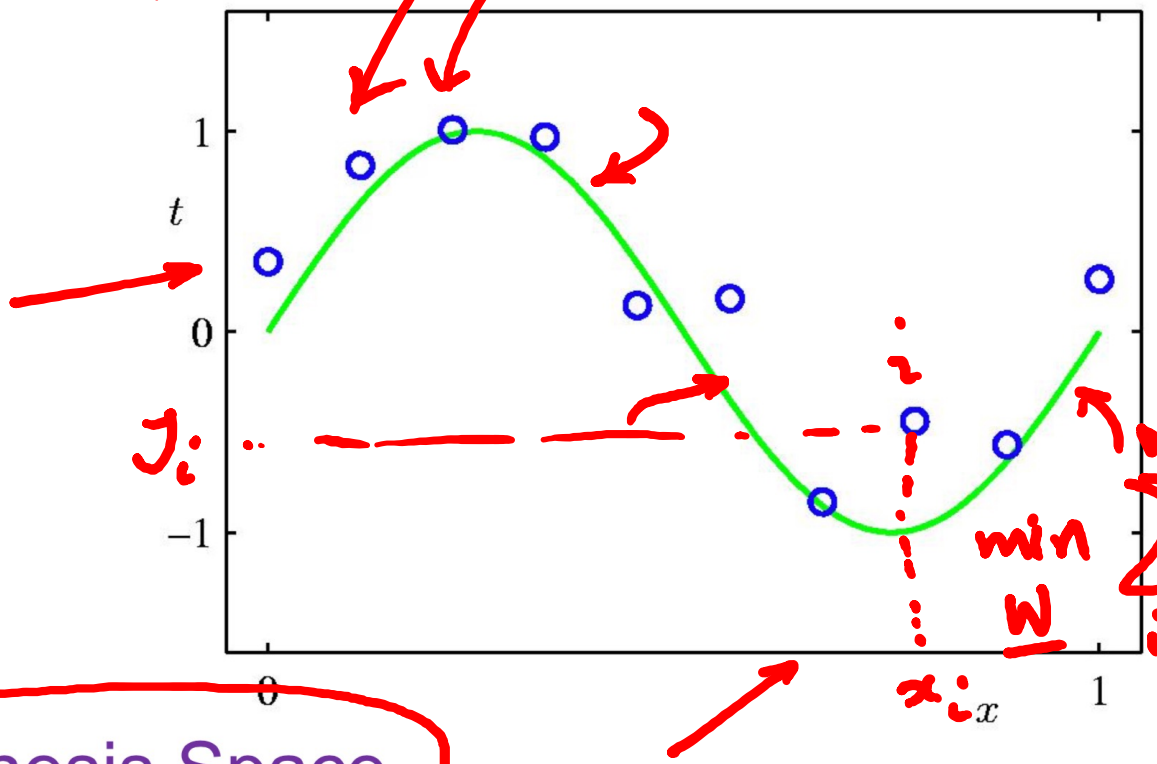
Polynomial Curve Fitting

Inductive Bias

$\underline{W} = (w_0, w_1, \ldots, w_M)$

training set

target noise



loss

i^{th} sample error

$$\min_{W} \sum_{i=1}^{N} \left(y_i - \sum_{j=0}^{M} w_j \, x_i^j \right)^2$$

$\hat{y}_i$

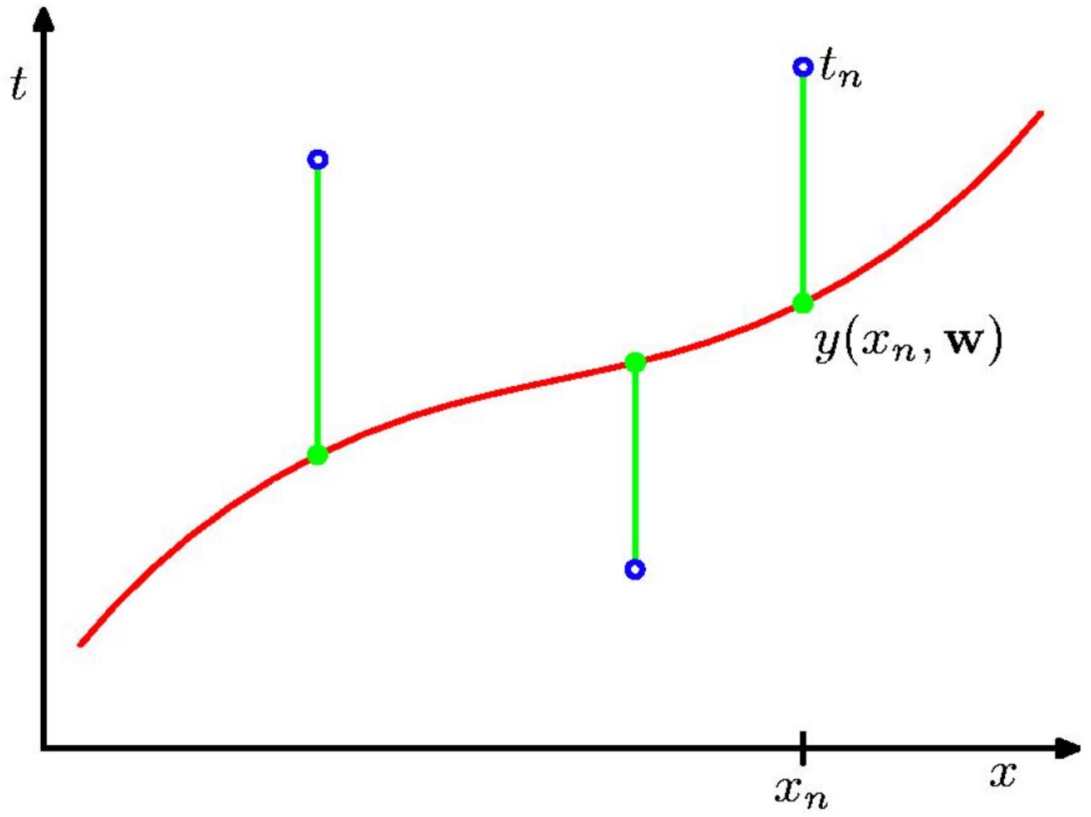
$R^{M+1} \equiv$ Hypothesis Space

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

$$W^{(t)} \leftarrow W^{(t-1)} - \alpha \nabla F$$

$F(W)$

Sum-of-Squares Error Function



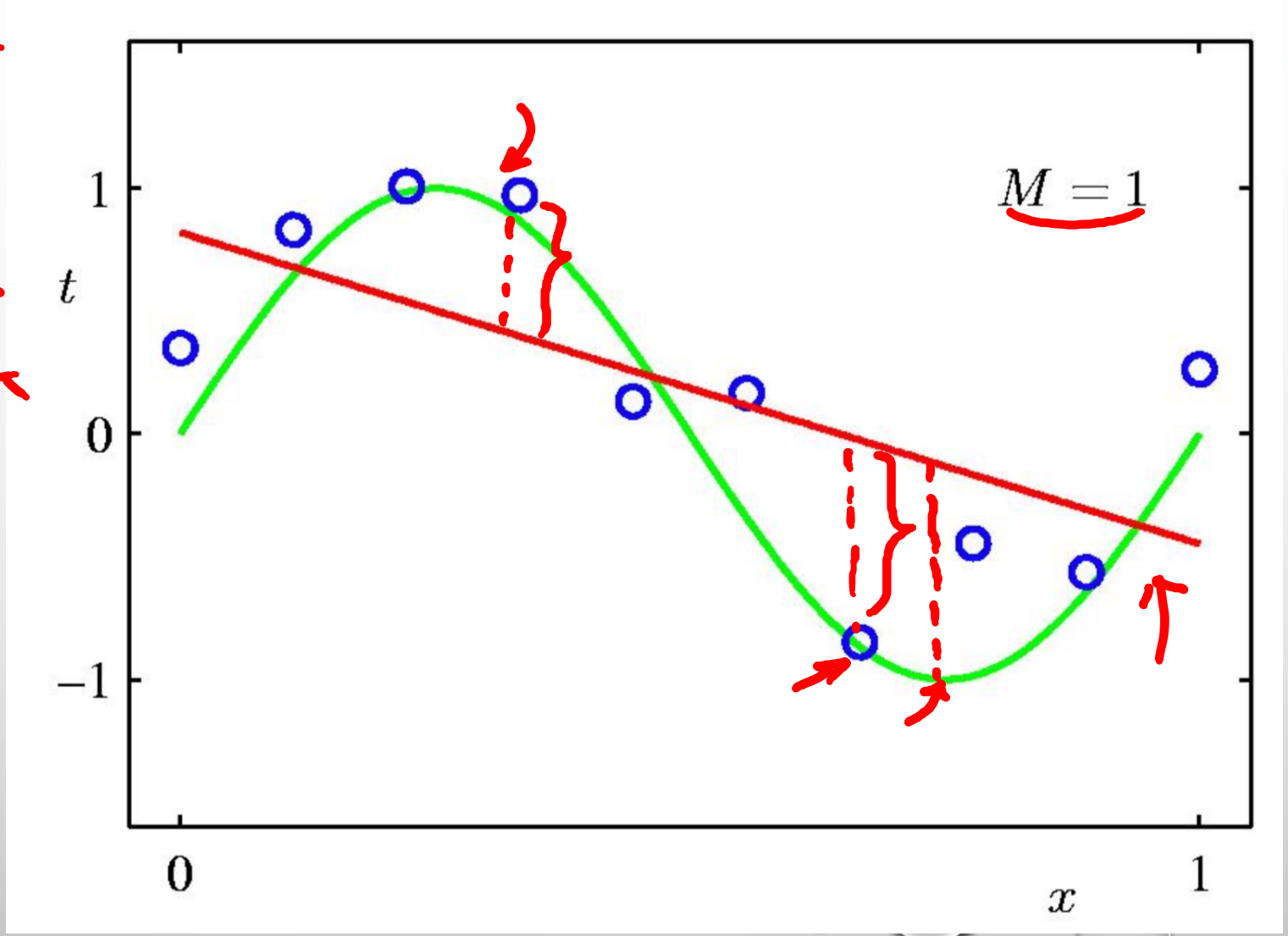
$$E(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}) - t_n\}^2$$

1st Order Polynomial

Underfit

trn err ↑

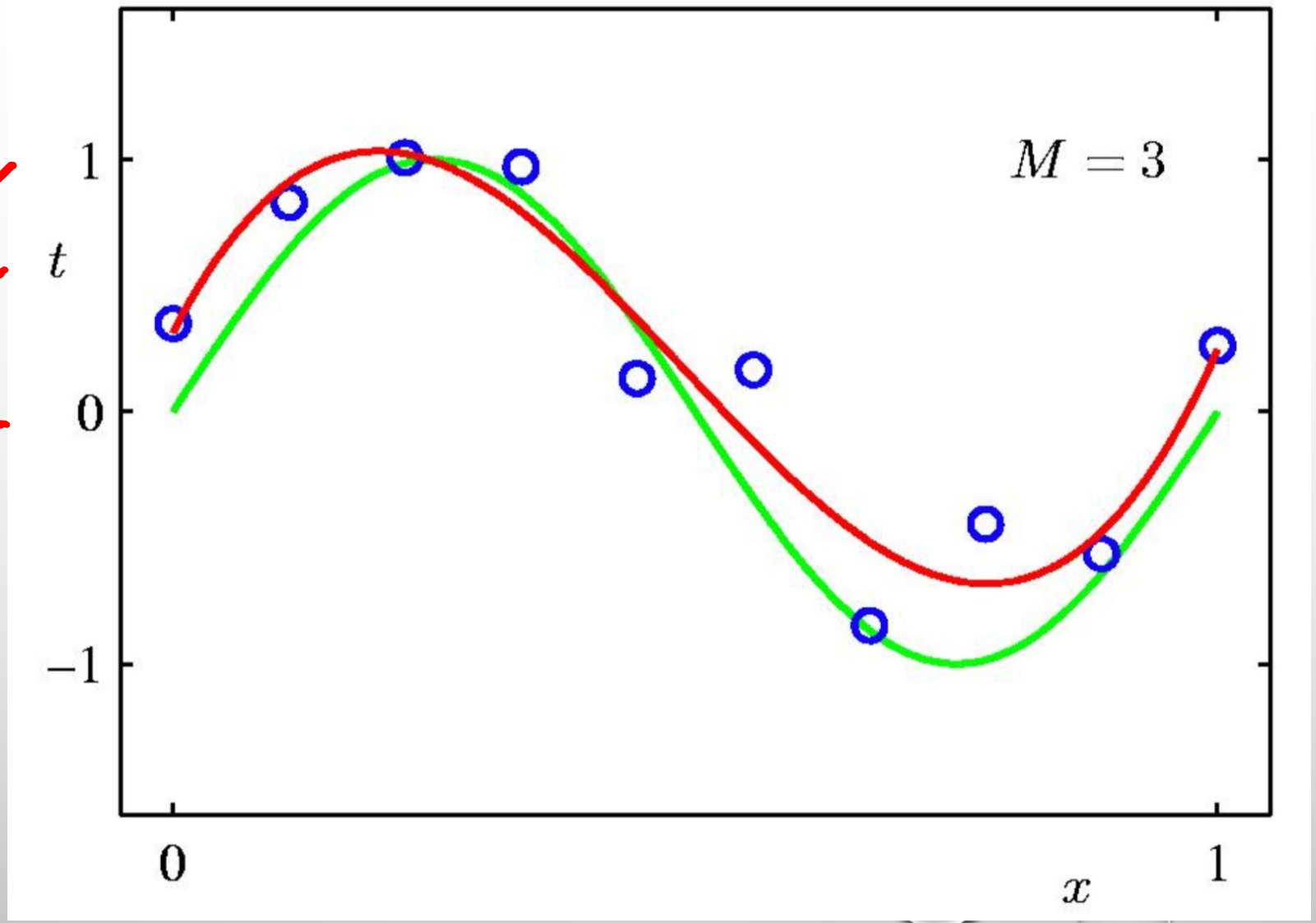
tst err ↑



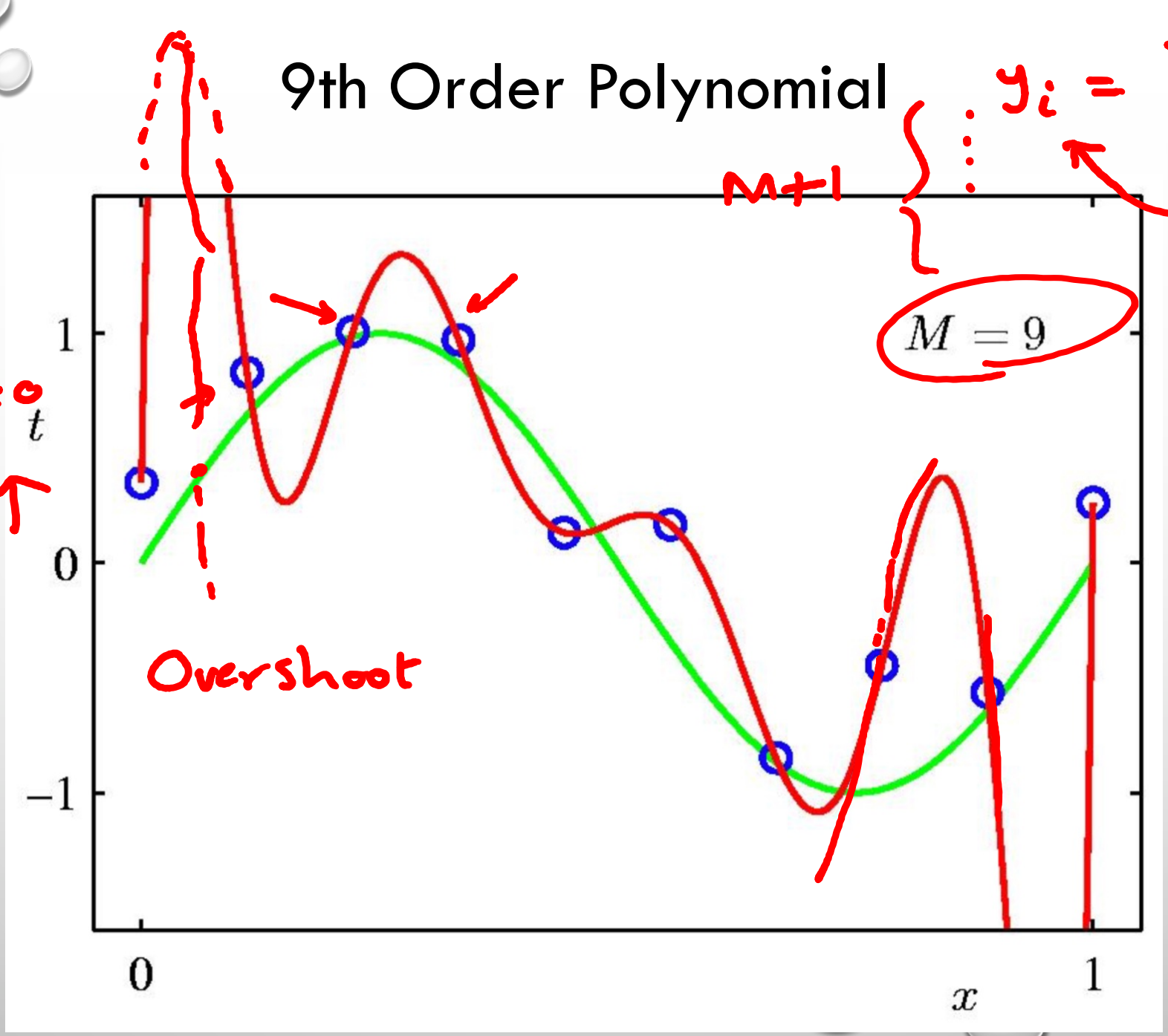
3rd Order Polynomial

trn err ↓
tst err ↓

Right fit



9th Order Polynomial



Overfit

trn err = 0

tst err ↑

M+1

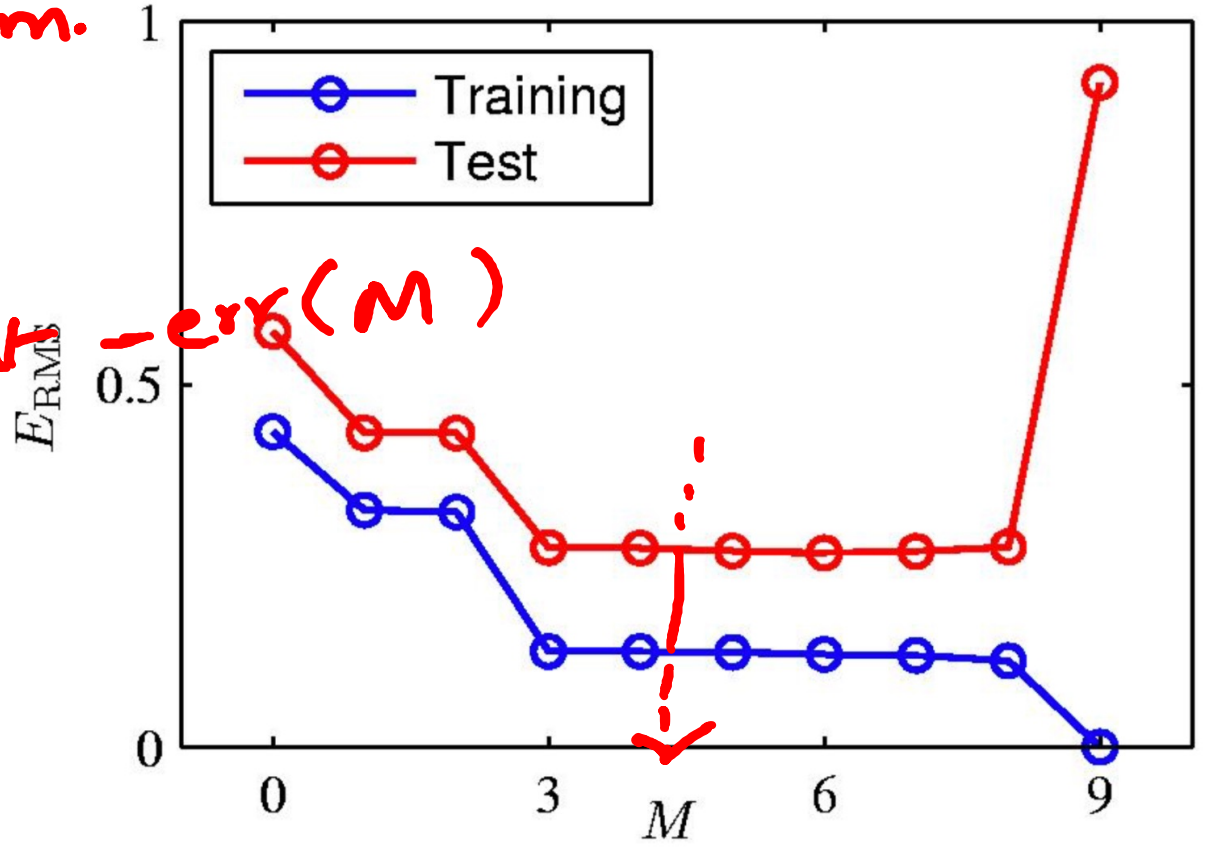
$y_i = \sum_{j=1}^{M} w_j x_i^j$

$Aw = b$

Over-fitting

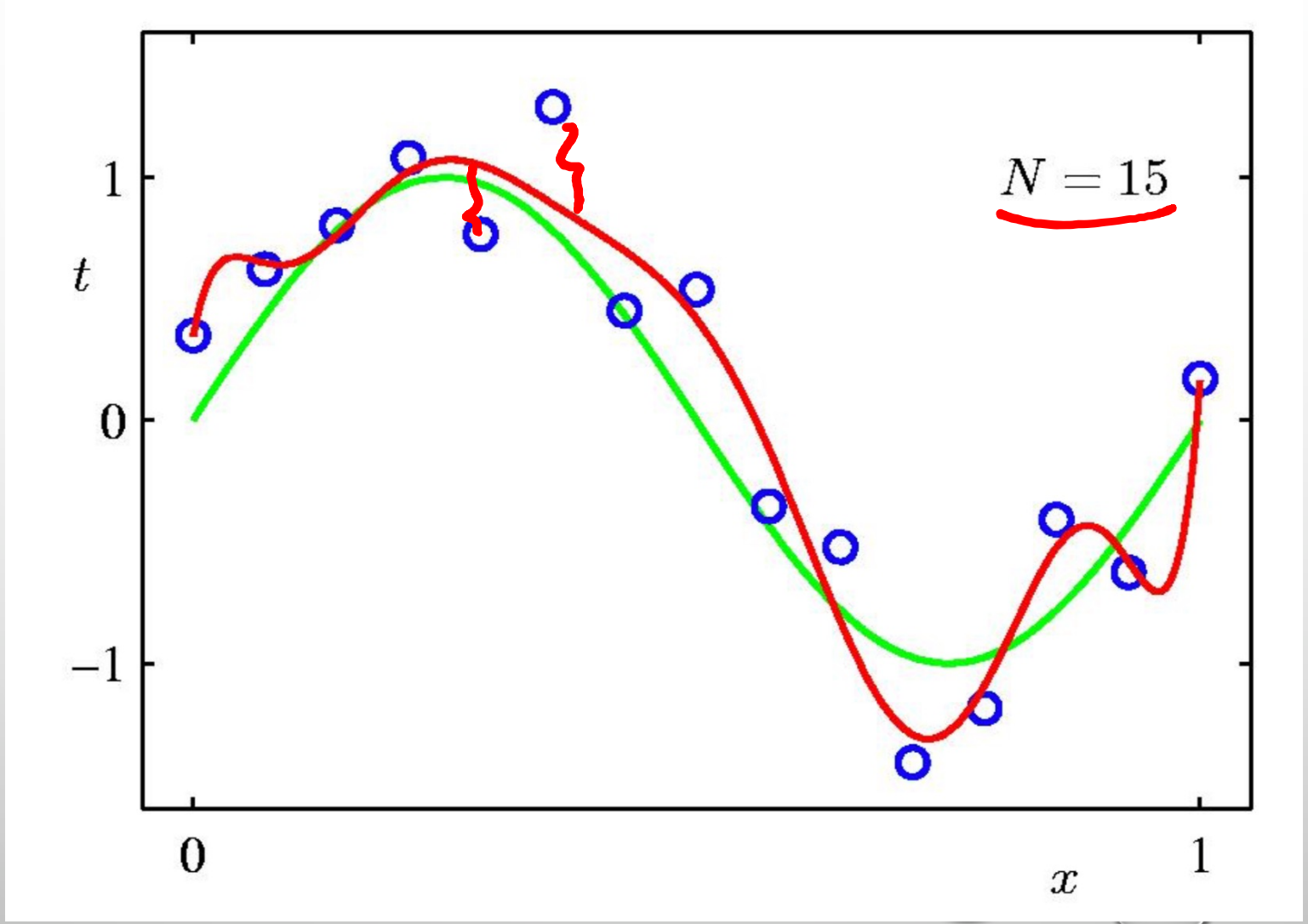
Hyper param. Tuning

min M Held-out-err(M)

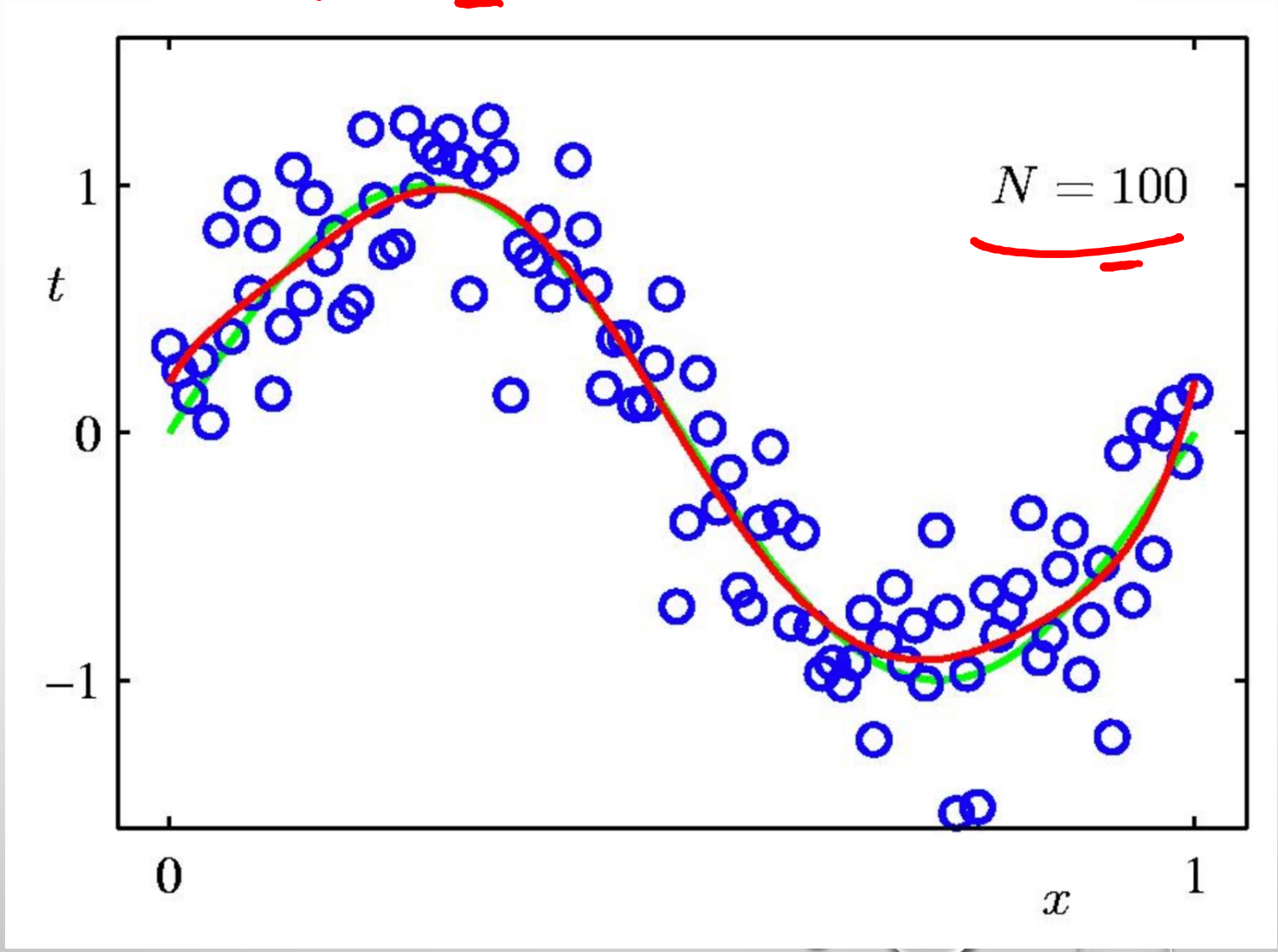


Root-Mean-Square (RMS) Error: $E_{\mathrm{RMS}} = \sqrt{2E(\mathbf{w}^\star)/N}$

Data Set Size: 9th Order Polynomial



Data Set Size: 9th Order Polynomial



Polynomial Coefficients

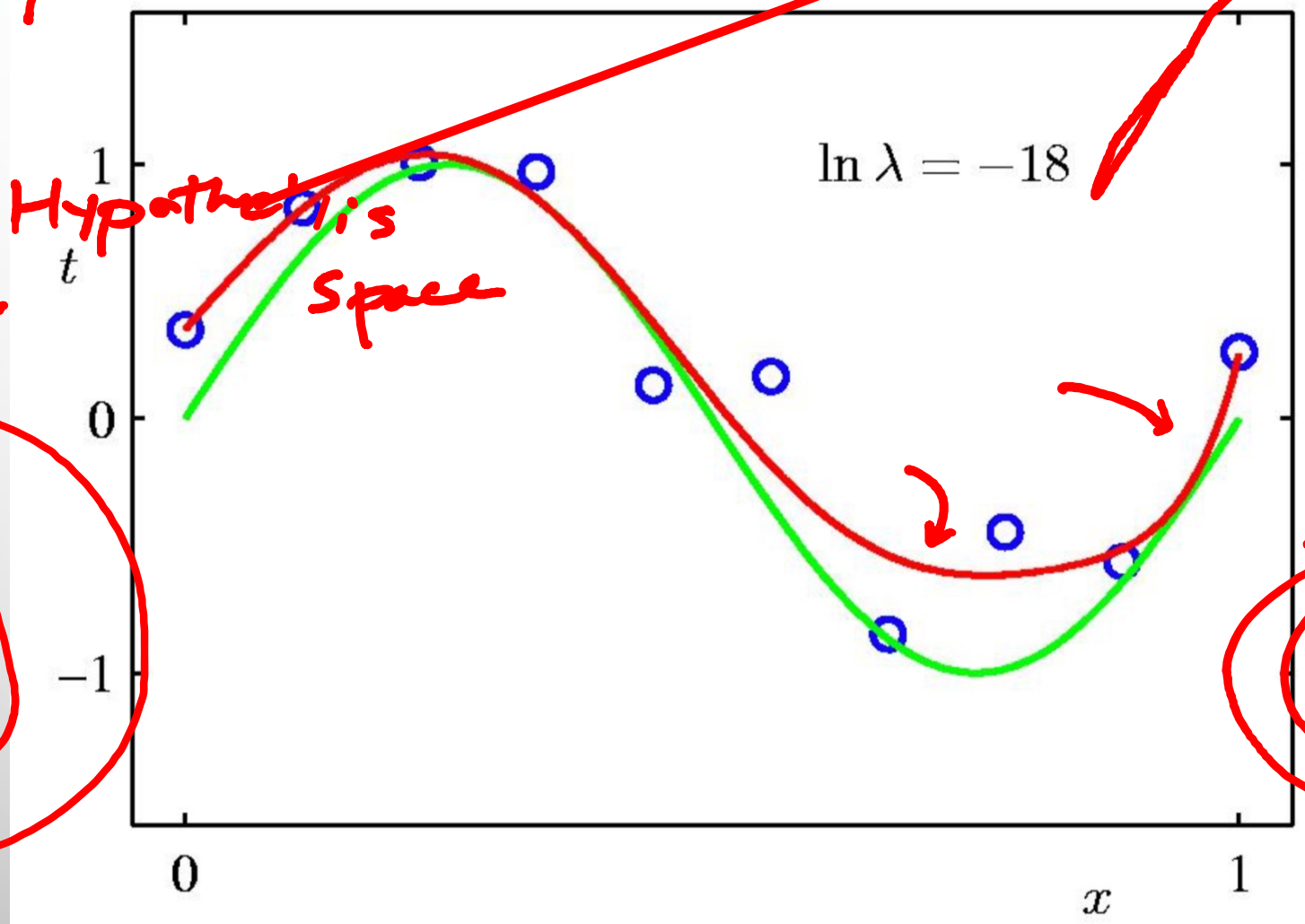
	$M=0$	$M=1$	$M=3$	$M=9$
$w_0^\star$	0.19	0.82	0.31	0.35
$w_1^\star$		-1.27	7.99	232.37
$w_2^\star$			-25.43	-5321.83
$w_3^\star$			17.37	48568.31
$w_4^\star$				-231639.30
$w_5^\star$				640042.26
$w_6^\star$				-1061800.52
$w_7^\star$				1042400.18
$w_8^\star$				-557682.99
$w_9^\star$				125201.43

Regularization

$$\widetilde{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2$$

- Penalize large coefficient values

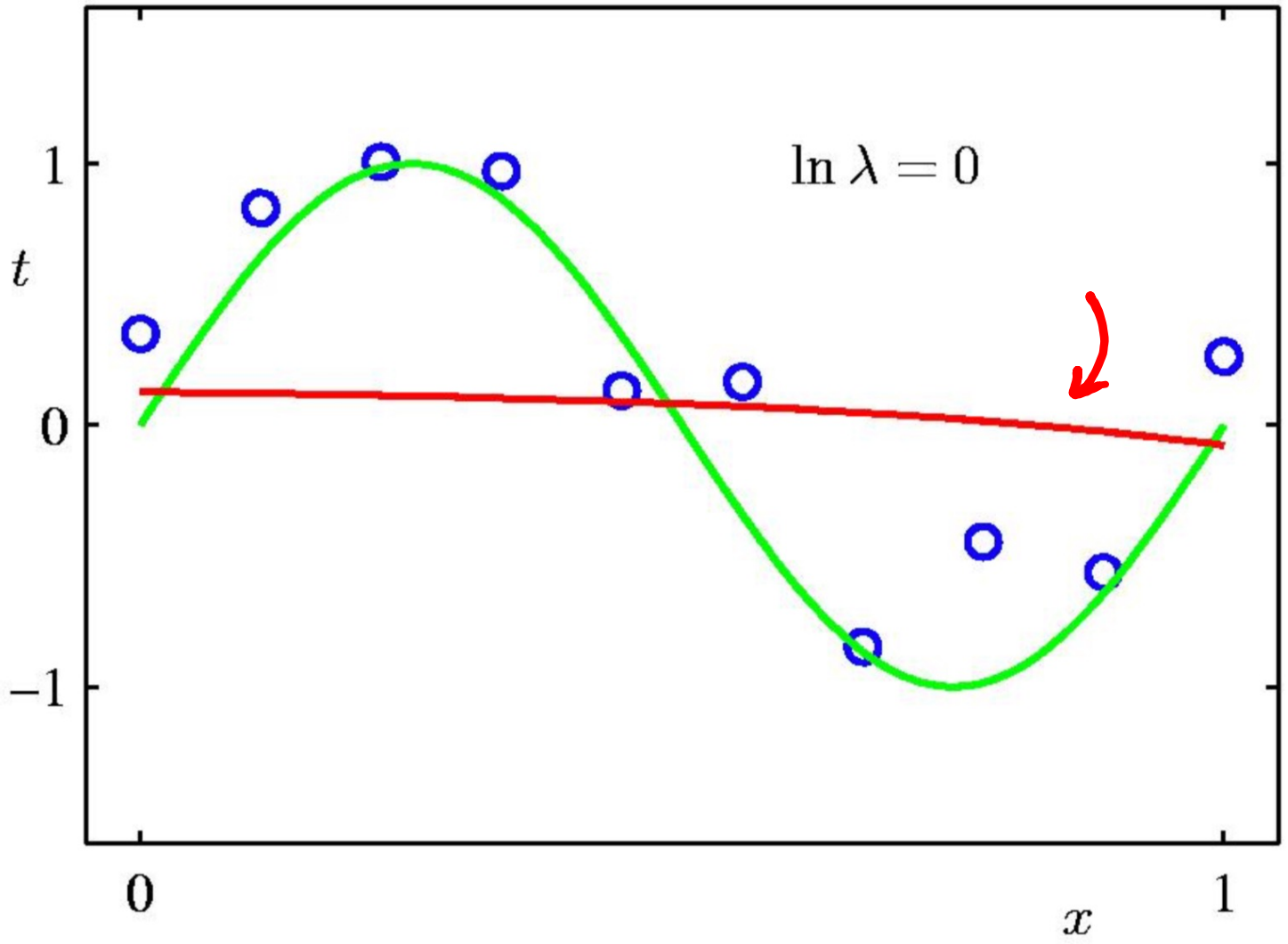
Regularization: ln λ = -18



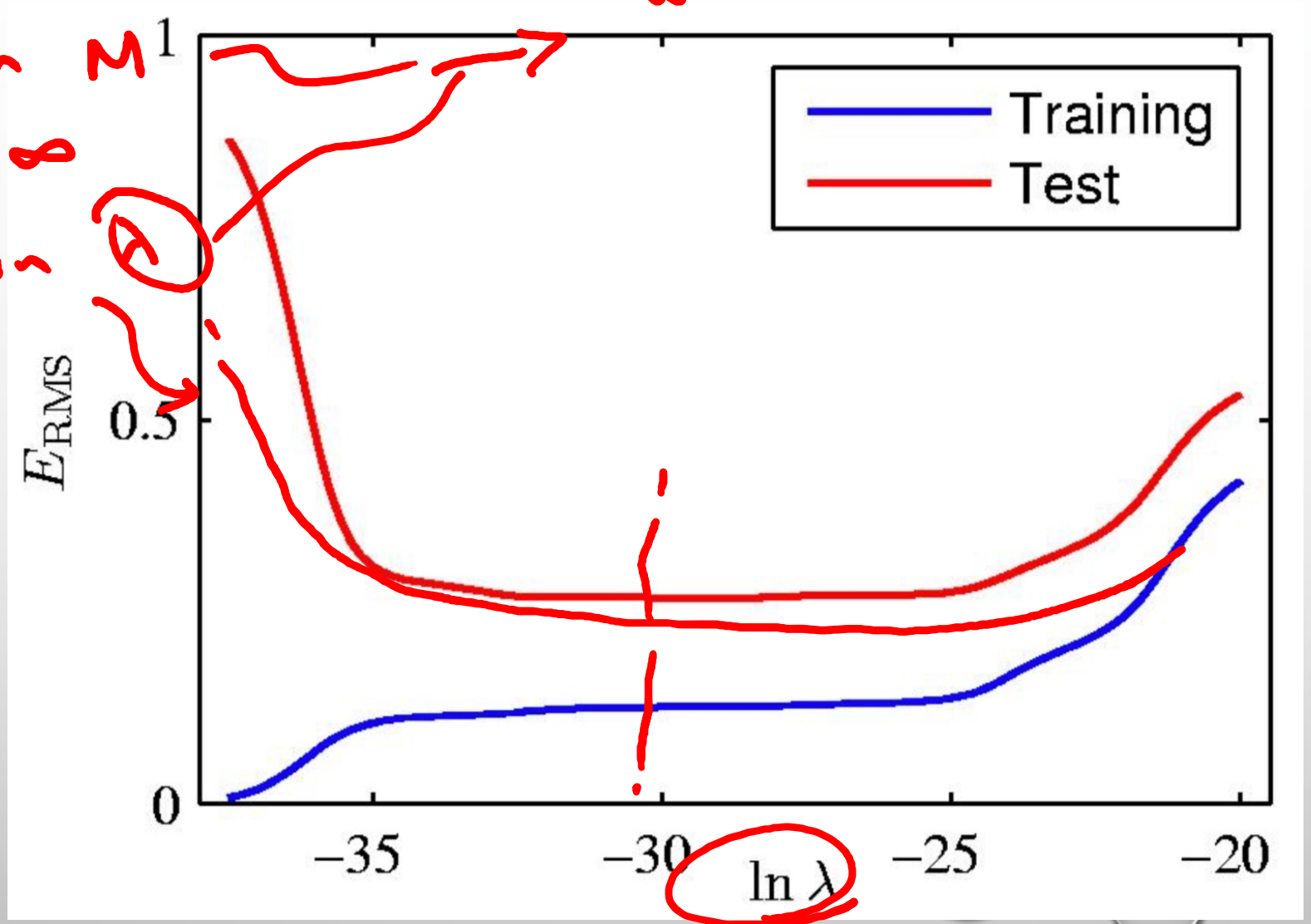
Regularization:
$\ln \lambda = 0$

$\lambda = 1$

Underfit



Regularization : E_{RMS} vs. ln λ



Polynomial Coefficients

	$\ln \lambda = -\infty$	$\ln \lambda = -18$	$\ln \lambda = 0$
$w_0^\star$	0.35	0.35	0.13
$w_1^\star$	232.37	4.74	-0.05
$w_2^\star$	-5321.83	-0.77	-0.06
$w_3^\star$	48568.31	-31.97	-0.05
$w_4^\star$	-231639.30	-3.89	-0.03
$w_5^\star$	640042.26	55.28	-0.02
$w_6^\star$	-1061800.52	41.32	-0.01
$w_7^\star$	1042400.18	-45.95	-0.00
$w_8^\star$	-557682.99	-91.53	0.00
$w_9^\star$	125201.43	72.68	0.01

Logistic Regression

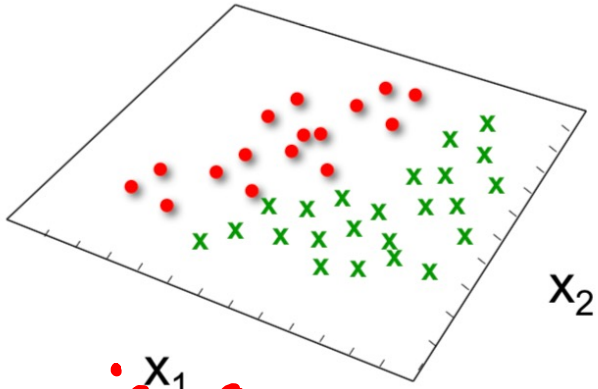
tool → Classification

Confidence Value

$$P(Y = 1 | X = < X_1, \ldots X_n >) = \frac{1}{1 + exp(w_0 + \sum_i w_i X_i)}$$

implies

$\langle \underline{w}, \underline{x} \rangle$



$$\ln \frac{P(Y = 0|X)}{P(Y = 1|X)} = w_0 + \sum_i w_i X_i$$

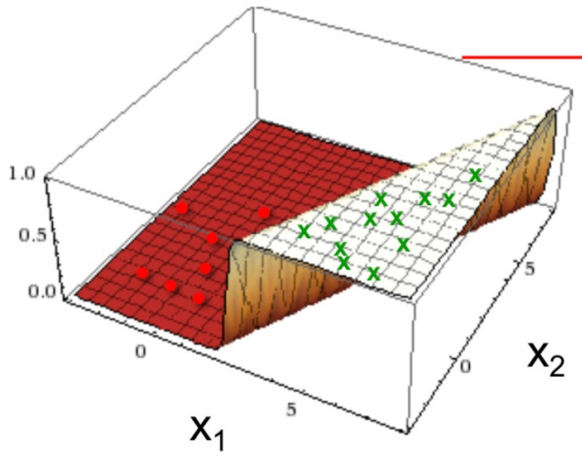
$\underline{w}$ is a signature of class 0

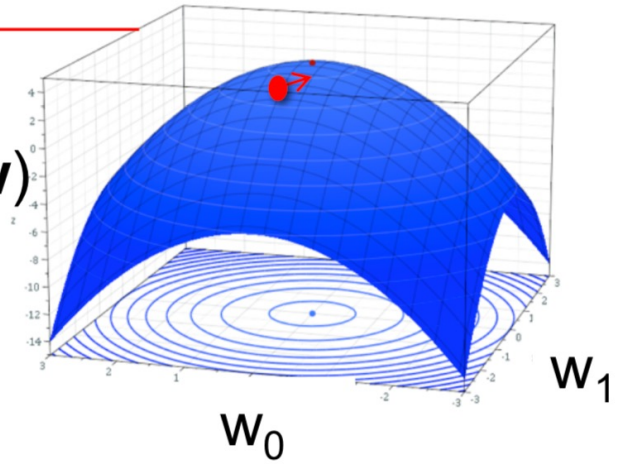
linear classification rule!

Gradient Ascent

w_0=40 w_1= -10 w_2 =5

Maximize $l(\mathbf{w}) = \ln P(D_Y \mid D_x, H_w)$





Update rule:

$$\Delta \mathbf{w} = \eta \nabla_{\mathbf{w}} l(\mathbf{w})$$

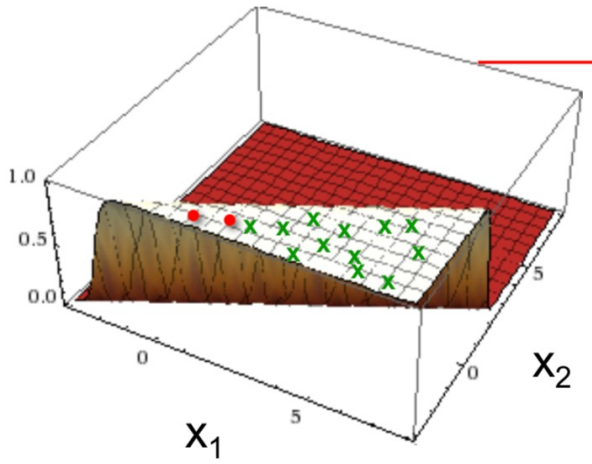
$$w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \frac{\partial l(\mathbf{w})}{\partial w_i}$$

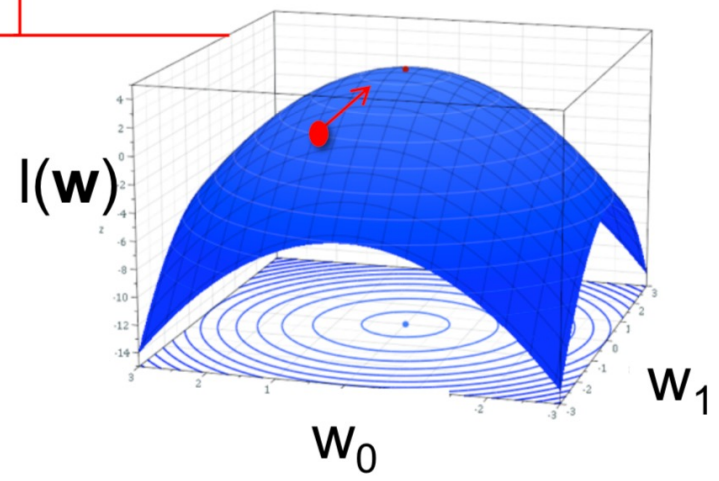
Logistic w/ Initial Weights

w_0=20 w_1= -5 w_2 =10

Loss(H_w) = Error(H_w, data)

Minimize Error → Maximize $l(\mathbf{w}) = \ln P(D_Y \mid D_X, H_w)$





Update rule:

$$\Delta \mathbf{w} = \eta \nabla_{\mathbf{w}} l(\mathbf{w})$$

$$w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \frac{\partial l(\mathbf{w})}{\partial w_i}$$

Step size

$H = \sum \alpha_i x_i x_i^T \quad \alpha_i \geq 0$

How to learn W from the training data $D \overset{iid}{=} \{(x_1, y_1), \cdots, (x_N, y_N)\}$?

$$P(Y=y \mid x, W) = \frac{1}{1+\exp\{y W^T x\}}$$

$$P(Y=-1 \mid x) = \frac{\exp(W^T x)}{1+\exp(W^T x)}$$

Max. Likelihood

$$\max_W \; P(Y^{(1)}=y_1 \cdots Y^{(N)}=y_N \mid x_1, \cdots x_N) = \prod_{i=1}^{N} P(Y^{(i)}=y_i \mid x_i)$$

GD on W

$$= \left\{ \prod_{i=1}^{N} \frac{1}{1+\exp(y_i W x_i)} \right\} \equiv \min_W \left\{ \sum_{i=1}^{N} \log(1+e^{y_i W^T x_i}) \right\}$$