



CIS 678 - Machine Learning

Model Regularization



Plan

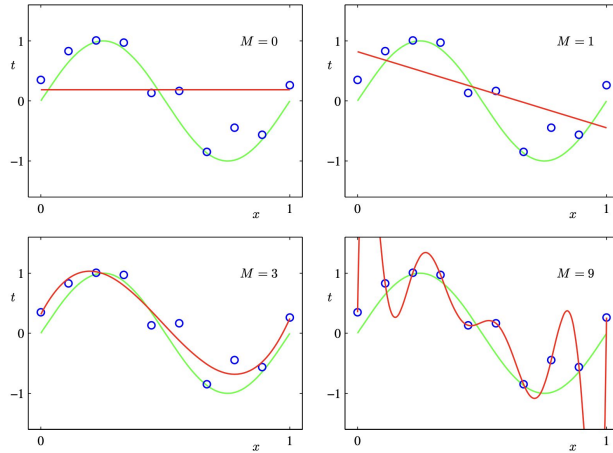
- Project grouping policies
- Regularization (presentation), plus Notebook walkthrough
 - Linear Regression (with and without regularization)
 - Logistic Regression (with and without regularization)
- KNN comparison (notebooks)
- Project topic exercise



Plan

- Project grouping policies
 - Students with similar area preferences are assigned to the same group
 - Experienced programmers are distributed across groups
 - Variance of programming skills across groups is kept to the minimum possible

Linear to Polynomial Regression



$$\hat{y} = \beta_0$$

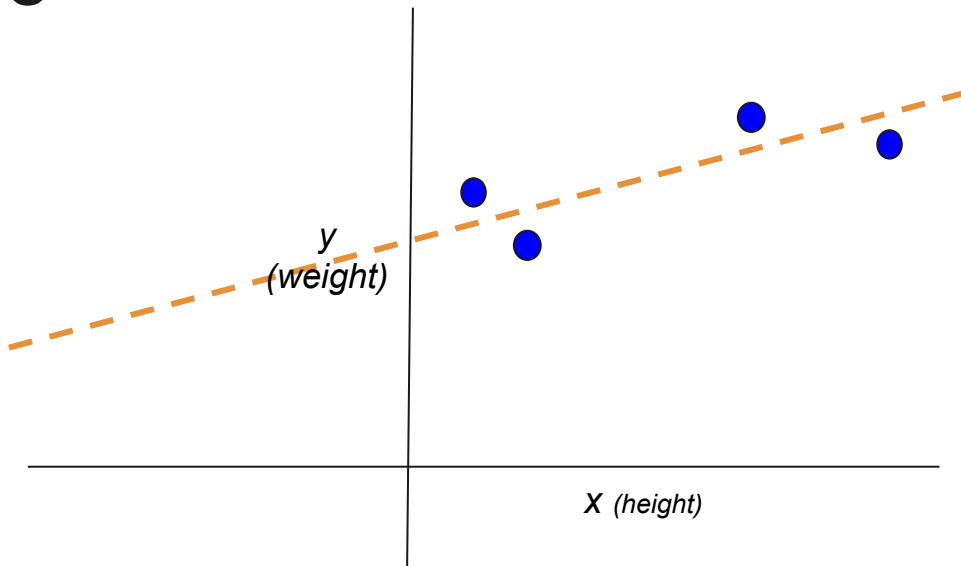
$$\hat{y} = \beta_0 + \beta_1 x$$

$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2$$

$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$$

$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \dots$$

Regularization



So, essentially we are fitting a function; right?

Model

$$\hat{y} = \beta_0 + \beta_1 x$$

$$\Theta = \{\beta_0, \beta_1\}$$

Fitting Error

$$\epsilon = |\hat{y} - y|$$

Optimization function

$$E_{\Theta} = \frac{1}{2} \sum_{i=1}^N (\hat{y}_i - y_i)^2$$

$$\Theta^* = \operatorname{argmin}_{\Theta} E\{(x_i, y_i)\}_{i=1, \dots, N}$$

Regularization

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

Model

$$\hat{y} = \beta_0 + \beta_1 x$$

$$\Theta = \{\beta_0, \beta_1\}$$

$$\epsilon = |\hat{y} - y|$$

Optimization function

$$E_{\Theta} = \frac{1}{2} \sum_{i=1}^N (\hat{y}_i - y_i)^2$$

$$\Theta^* = \operatorname{argmin}_{\Theta} E\{(x_i, y_i)\}_{i=1, \dots, N}$$

Regularization

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

Essentially, the same formulation

Generally **ML** vs **Math** conventions

Model

$$\hat{y} = \beta_0 + \beta_1 x$$

$$\Theta = \{\beta_0, \beta_1\}$$

$$\epsilon = |\hat{y} - y|$$

Optimization function

$$E_{\Theta} = \frac{1}{2} \sum_{i=1}^N (\hat{y}_i - y_i)^2$$

$$\Theta^* = \operatorname{argmin}_{\Theta} E\{(x_i, y_i)\}_{i=1, \dots, N}$$

Regularization

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

x : scalar
 \mathbf{x}, \mathbf{x} : vector
 \mathbf{X} : Matrix

Model

$$\hat{y} = \beta_0 + \beta_1 x$$

$$\Theta = \{\beta_0, \beta_1\}$$

Essentially, the same formulation

Generally ML vs Math conventions

$$\epsilon = |\hat{y} - y|$$

Optimization function

$$E_{\Theta} = \frac{1}{2} \sum_{i=1}^N (\hat{y}_i - y_i)^2$$

$$\Theta^* = \operatorname{argmin}_{\Theta} E\{(x_i, y_i)\}_{i=1, \dots, N}$$

Regularization

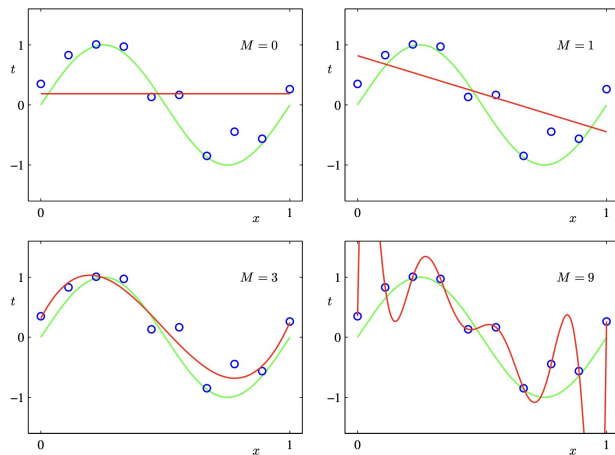


Table 1.1 Table of the coefficients w^* for polynomials of various order. Observe how the typical magnitude of the coefficients increases dramatically as the order of the polynomial increases.

	$M = 0$	$M = 1$	$M = 6$	$M = 9$
w_0^*	0.19	0.82	0.31	0.35
w_1^*		-1.27	7.99	232.37
w_2^*			-25.43	-5321.83
w_3^*			17.37	48568.31
w_4^*				-231639.30
w_5^*				640042.26
w_6^*				-1061800.52
w_7^*				1042400.18
w_8^*				-557682.99
w_9^*				125201.43



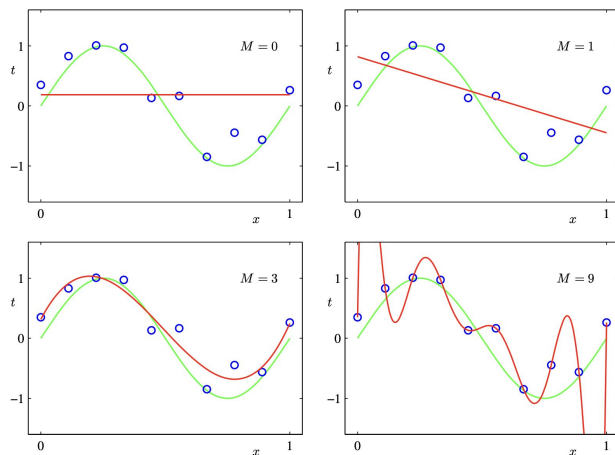
Regularization

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

Regularizer

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

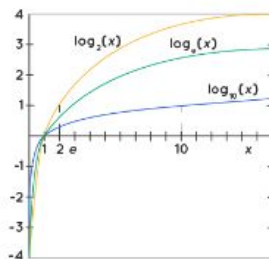
Regularization



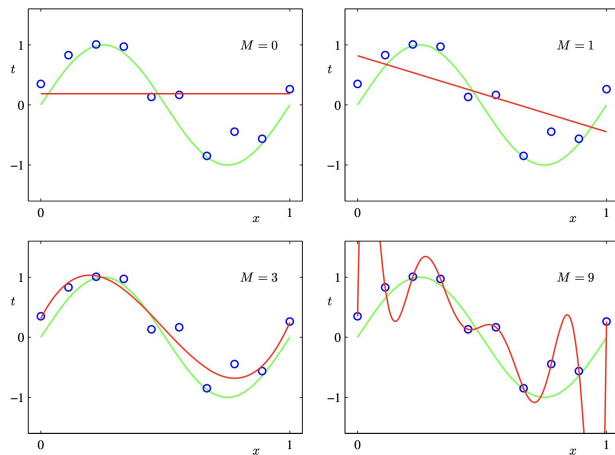
$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

Table 1.2 Table of the coefficients w^* for $M = 9$ polynomials with various values for the regularization parameter λ . Note that $\ln \lambda = -\infty$ corresponds to a model with no regularization, i.e., to the graph at the bottom right in Figure 1.4. We see that, as the value of λ increases, the typical magnitude of the coefficients gets smaller.

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



Linear to Polynomial Regression + Regularization

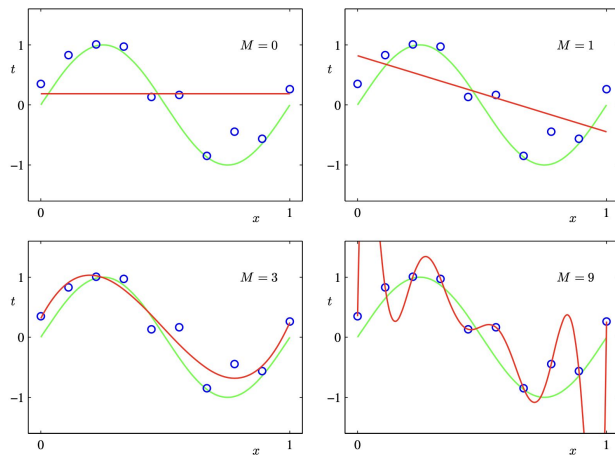


Learned function is nonlinear

$$\begin{aligned}\hat{y} &= \beta_0 \\ \hat{y} &= \beta_0 + \beta_1 x \\ \hat{y} &= \beta_0 + \beta_1 x + \beta_2 x^2 \\ \hat{y} &= \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 \\ \hat{y} &= \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \dots\end{aligned}$$

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

Linear to Polynomial Regression + Regularization



Learned function is **nonlinear**

$$\begin{aligned}\hat{y} &= \beta_0 \\ \hat{y} &= \beta_0 + \beta_1 x \\ \hat{y} &= \beta_0 + \beta_1 x + \beta_2 x^2 \\ \hat{y} &= \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 \\ \hat{y} &= \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \dots\end{aligned}$$

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

Model (still) **linear**

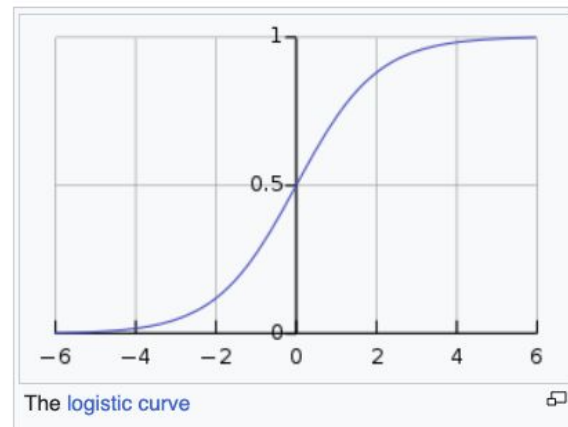


Classification

- General Idea (two steps process)
 - LR (Bias Only)
 - LR (general)

Classification

- General Idea (two steps process)
 - LR (Bias Only)
 - LR (general)
- **Logistic Regression (one single step, but with probability distribution)**
 - The famous sigmoid function
 - In NN they call it activation function



$$p(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}}$$



Notebook presentation

- Without regularizer
- With regularizer

Predictive modeling: [Regression \(diabetes\)](#)

Predictive modeling: [Classification](#)