Classification and regression

Sunita Sarawagi

The setup

Input: $X = \{x_1, x_2, \dots x_d\}$ $\quad x_j \in$ Real or categorical

Output: y — Real (Regression)
Categorical $\{1, 2, \dots k\}$ Classification

k = # of class labels

Goal:
Given training data

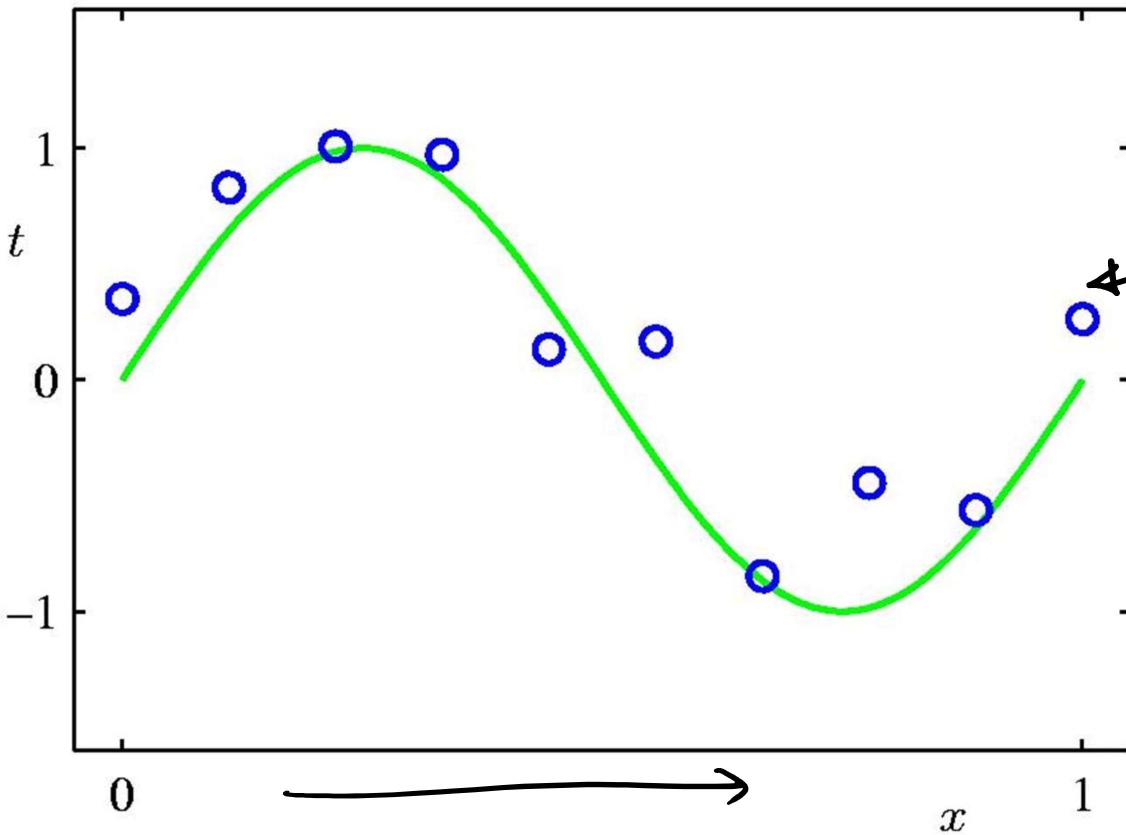
$$D = \{(X^1, y_1), (X^2, y_2) \dots (X^N, y_N)\}$$

Learn: $f_\theta(X) \longrightarrow y$ $\quad \theta$ = parameters

Challenges in classifier training

- Given training data
  - D = {(x1,y1),….(xN,yN)}
- Goal during training: learn function f(x)→y such that
  - Prediction error on unseen instances will be small
- Challenges:
  - Space of possible functions extremely large: need to limit the set from which f is chosen (Hypothesis family)
  - How to ensure f's performance on D will generalize to unseen instances
  - How to achieve computational tractability

Regression: Synthetic data

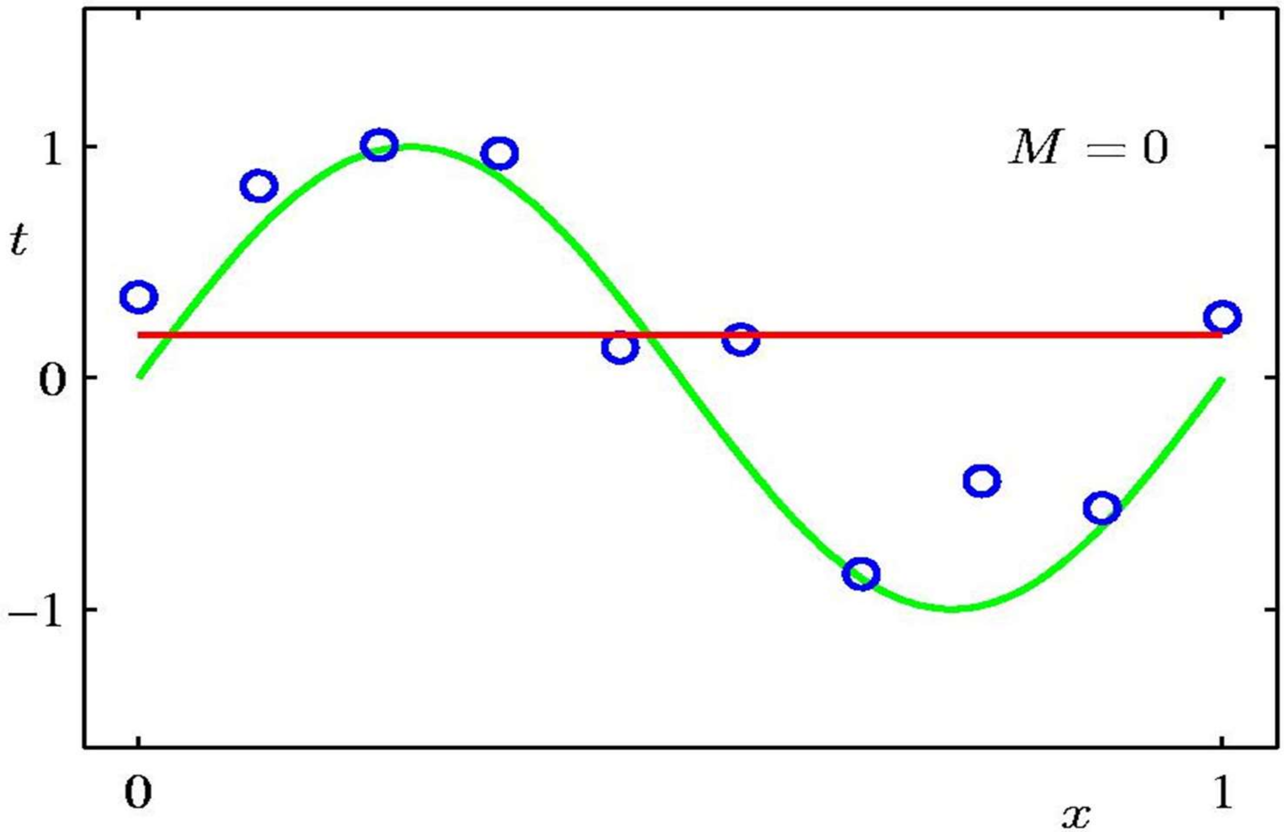


$d = 1$

one training instance

$N = 10$

Fitting with a constant function



$f(x) = \theta_0$

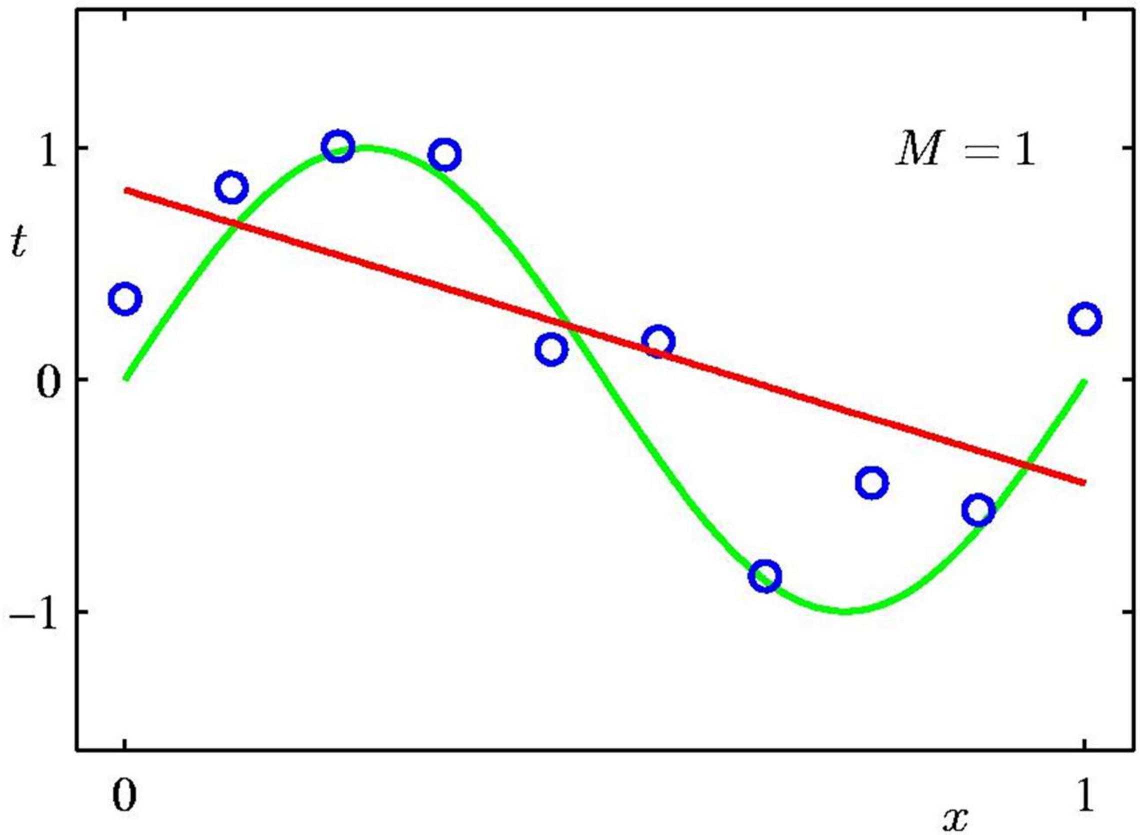
θ_0^D

$\theta_0^D =$ Avg of $y_{i:1 \ldots N}$

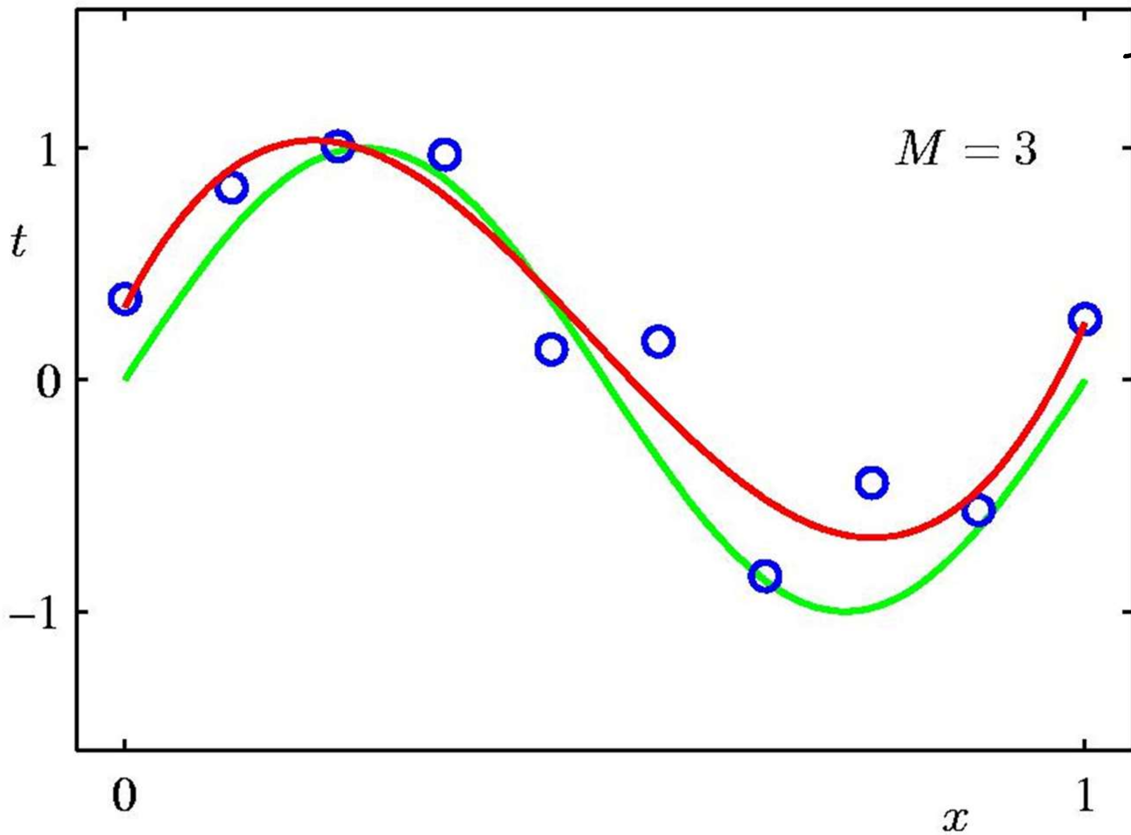
$\frac{\sum y_i}{N}$

Fitting with a linear function

$f(x) = \theta_1 x_1 + \theta_0$

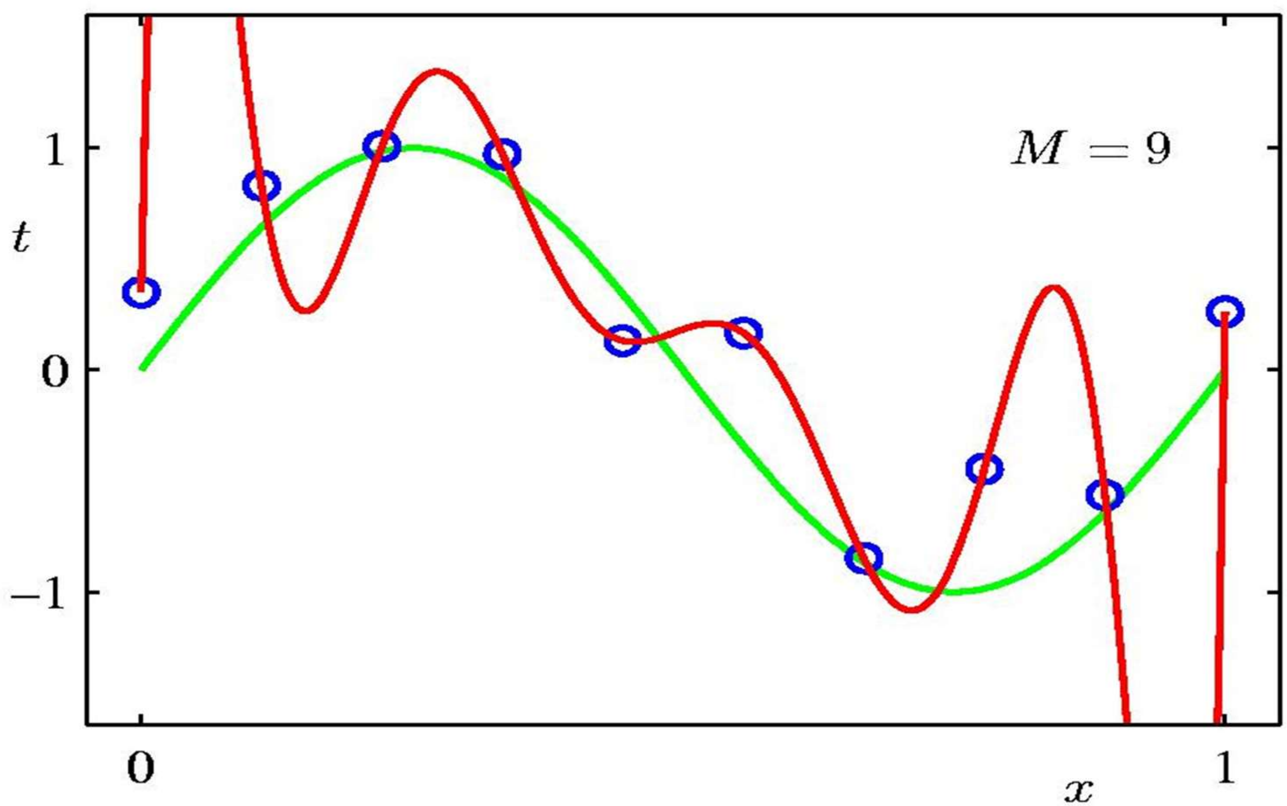


Fitting with a cubic function

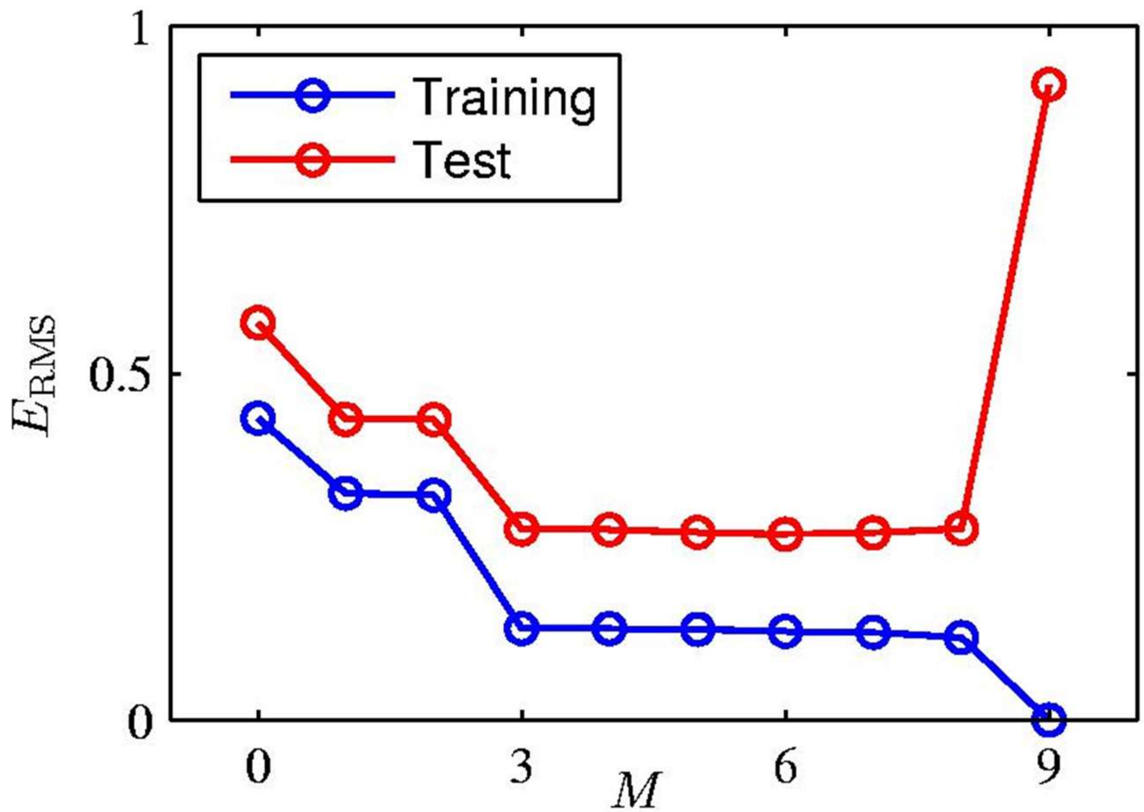


$$f(x) = \theta_4 x_1^3 + \theta_3 x_1^2 + \theta_2 x_1 + \theta_1$$

Overfitting with a 9th degree polynomial



How to detect overfitting



How to avoid overfitting

- **Regularization:** Assess a (large) penalty for a (highly) complicated model
  - Balance model penalty with data fit
- Being a **Bayesian:** Imagine the model is itself a random object drawn from a *prior* distribution
  - Then model generates observed data
  - Given data and prior, find (properties of) model
- When labeled data is abundant
  - **Cross-validation** with unseen data

Types of classifiers

- Probabilistic
  - Generative
  - Conditional
- Discriminative
  - Decision trees
  - Neural Networks
  - Support Vector Machines
- Distance-based
  - Nearest neighbor classifier