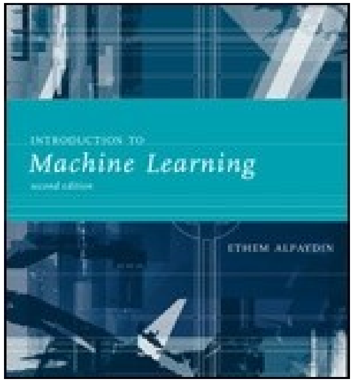
Lecture Slides for

INTRODUCTION TO

Machine Learning

2nd Edition



ETHEM ALPAYDIN, modified by Leonardo Bobadilla
and some parts from
http://www.cs.tau.ac.il/~apartzin/MachineLearning/

alpaydin@boun.edu.tr
http://www.cmpe.boun.edu.tr/~ethem/i2m

CHAPTER 2:

Supervised Learning

Outline

Last Class: Ch 2 Supervised Learning (Sec 2.1-2.4)

- Learning a class from Examples
- VC Dimension
- PAC learning
- Noise

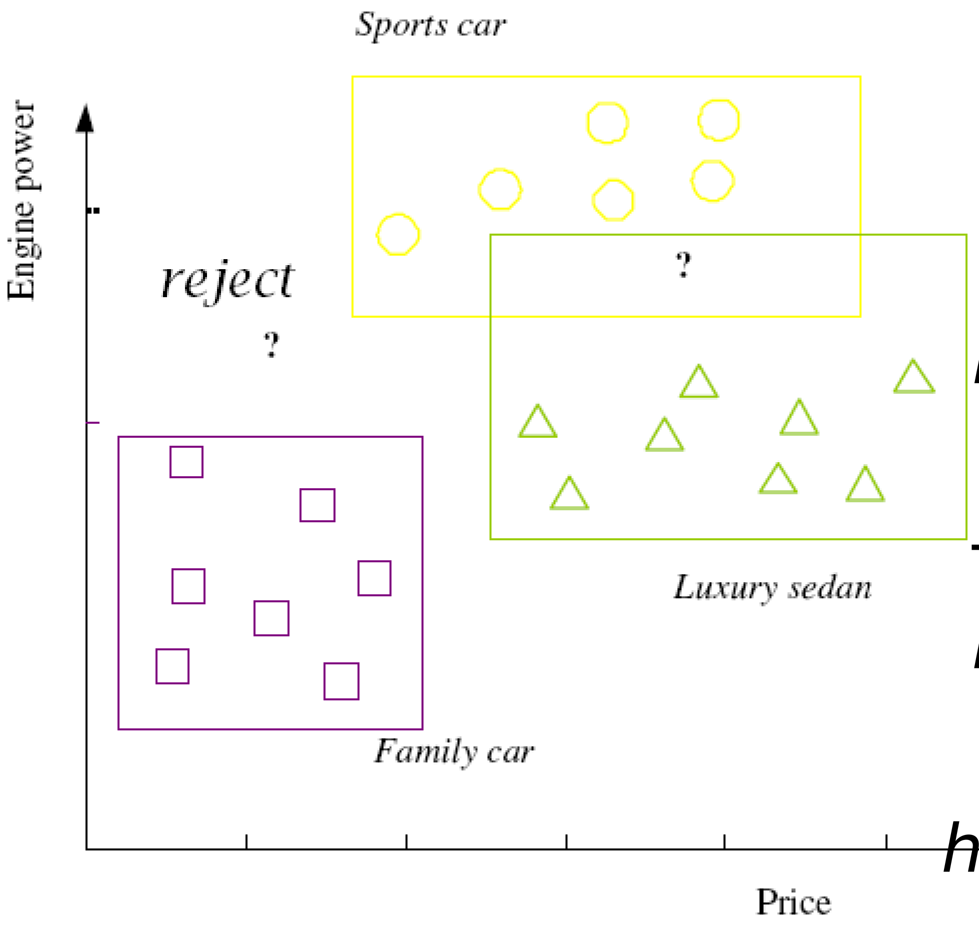
This class:

- Learning Multiple Classes
- Regression
- Model Selection and Generalization
- Dimensions of a Supervised Learning Algorithm

Multiple Classes

- General case K classes
  - Family, Sport , Luxury cars
- Classes can overlap
- Can use different/same hypothesis class
- Fall into two classes? Sometimes worth to reject

Multiple Classes, Ci i=1,...,K



$$\mathcal{X} = \{\boldsymbol{x}^t, r^t\}_{t=1}^N$$

$$r_i^t = \begin{cases} 1 \text{ if } \boldsymbol{x}^t \in C_i \\ 0 \text{ if } \boldsymbol{x}^t \in C_j, j \neq i \end{cases}$$

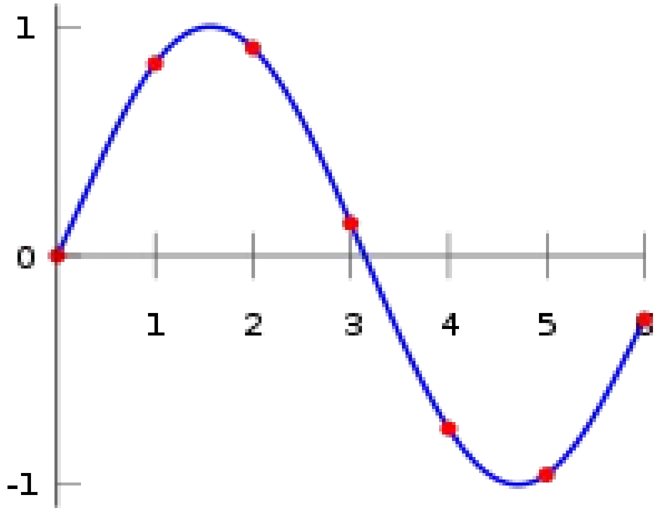
Train hypotheses $h_i(\boldsymbol{x})$, $i = 1,...,K$:

$$h_i\left(\boldsymbol{x}^t\right) = \begin{cases} 1 \text{ if } \boldsymbol{x}^t \in C_i \\ 0 \text{ if } \boldsymbol{x}^t \in C_j, j \neq i \end{cases}$$

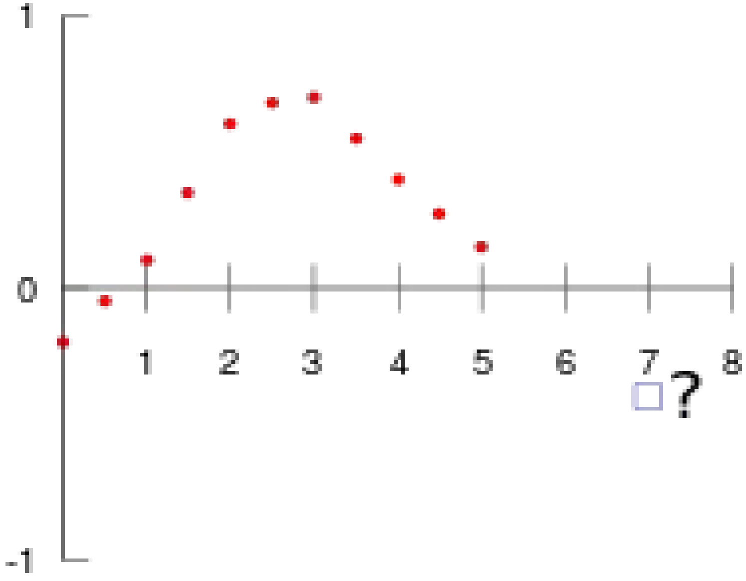
Regression

- Output is not Boolean (yes/no) or label but numeric value
- Training Set of examples $\mathcal{X} = \{\mathbf{x}^t, r^t\}_{t=1}^N$
- Interpolation: fit function (polynomial)
- Extrapolation: predict output for any x
- Regression : added noise $r^t = f(\mathbf{x}^t) + \epsilon$
- Assumption: hidden variables $r^t = f^*(\mathbf{x}^t, \mathbf{z}^t)$
- Approximate output by model: g(x)

Examples



Interpolation



Extrapolation

From: http://en.wikipedia.org

Regression

- Empirical error on training set

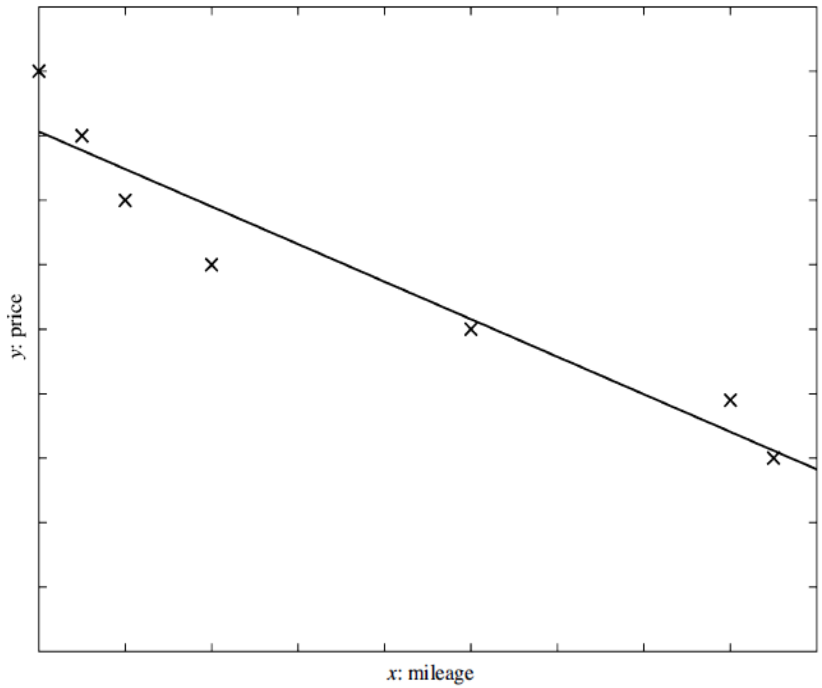
$$E(g|X) = \frac{1}{N}\sum_{t=1}^{N}[r^t - g(\mathbf{x}^t)]^2$$

- Hypothesis space is linear functions

$$g(\mathbf{x}) = w_1x_1 + \cdots + w_dx_d + w_0 = \sum_{j=1}^{d} w_jx_j + w_0$$

- Calculate best parameters to minimize error by taking partial derivatives

Example

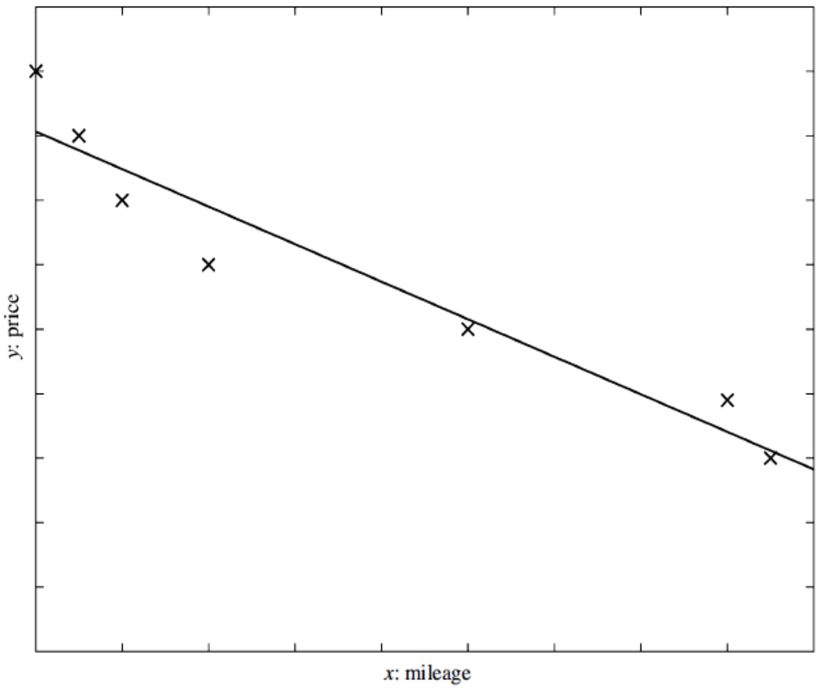
- 

$$g(x) = w_1 x + w_0$$

$$E(w_1, w_0 | \mathcal{X}) = \frac{1}{N} \sum_{t=1}^{N} [r^t - (w_1 x^t + w_0)]^2$$

Example

-



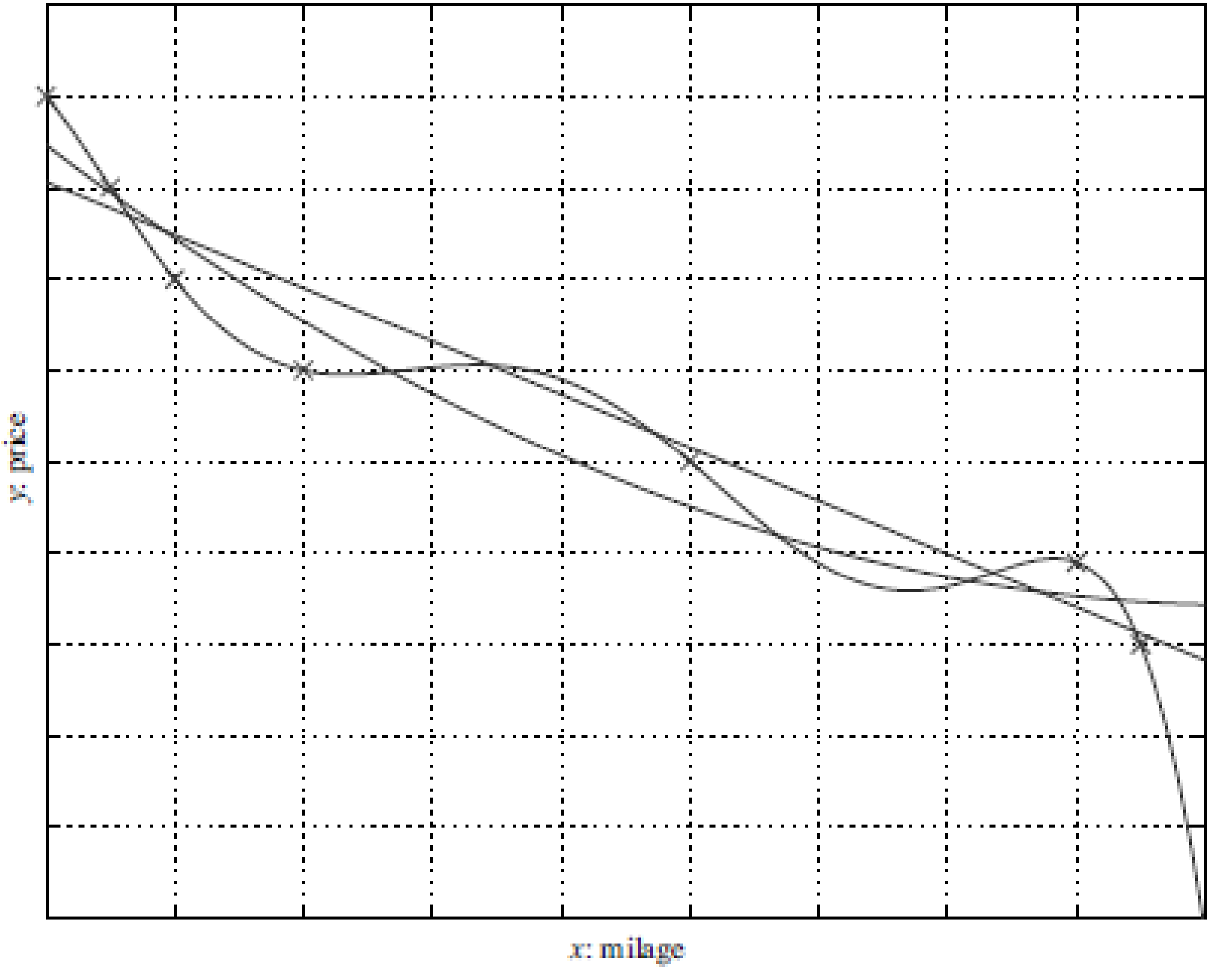
$$w_1 = \frac{\sum_t x^t r^t - \overline{x}\overline{r}N}{\sum_t (x^t)^2 - N\overline{x}^2}$$

$$w_0 = \overline{r} - w_1\overline{x}$$

$$g(x) = w_2 x^2 + w_1 x + w_0$$

A more complex model

Higher-order polynomials



Model Selection & Generalization

x_1	x_2	h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8	h_9	h_{10}	h_{11}	h_{12}	h_{13}	h_{14}	h_{15}	h_{16}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

- Consider learning boolean functions
- If *d* inputs, 2^d examples at most

  Each example can be labeled *0* or *1*
- Therefore 2^{2^d} possible functions of d variables

Model Selection & Generalization

x_1	x_2	h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8	h_9	h_{10}	h_{11}	h_{12}	h_{13}	h_{14}	h_{15}	h_{16}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

- Each training example removes half the hypothesis
- Learning as a way to remove hypothesis inconsistent with data
- But we need to see 2^d examples to learn

Model Selection & Generalization

- Learning is an ill-posed problem; data is not sufficient to find a unique solution
  - Each sample remove irrelevant hypothesis
- The need for inductive bias, assumptions about H
  - E.g. rectangles in our example
- But each hypothesis can only learn some functions

Model Selection & Generalization

- Learning needs an inductive bias
- Model selection: How to choose the right bias?
  - Each sample remove irrelevant hypothesis
- Want the model to be able to generalize
  - Predict new data even more than fitting the training dataset
- Generalization: How well a model performs on new data

Model Selection & Generalization

- Best generalization requires mathing the complexity of the hypothesis with the complexity of the function underlying the data
- Overfitting: H more complex than *C* or *f*
  - *e.g* Fitting two rectangles to data sampled from one rectangle
  - *e.g* Fitting a sixth-order polynomal to noisy data from a third-order polynomial
- Underfitting: H less complex than *C* or *f*
  - *e.g* Fit a line to data sample from a third-order polynomial

Triple Trade-Off

- There is a trade-off between three factors (Dietterich, 2003):
  1. Complexity of H, c (H),
  2. Training set size, N,
  3. Generalization error, E, on new data
- As $N\uparrow$, $E\downarrow$
- As c (H)$\uparrow$, first $E\downarrow$ and then $E\uparrow$ why?
-

Cross-Validation

- To estimate generalization error, we need data unseen during training. We split the data as
  - Training set (50%)
    - To train a model
  - Validation set (25%)
    - To select a model (e.g. degree of polynomials)
  - Test (publication) set (25%)
    - Estimate the error, evaluate performance
- Resampling when there is few data

Dimensions of a Supervised Learner

- Let us now recapitulate and generalize. We have a sample $X = \{x^t, r^t\}_{t=1}^N$
-
- The sample is independent and identically distributed (i.i.d) from the same joint distribution $p(x, r)$
  - r^t İs 0/1 for classification
    - K binary vector for multiclass classification
    - real value in regression
    - Goal: Build a good and useful approximation to r^t using the model $g(x^t|\theta)$

Dimensions of a Supervised Learner

We must make three decisions:

1. Model: $g(x|\theta)$

1. $g(\cdot)$ *model* x *input* θ *parameters*

$g(\cdot)$ Defines the hypothesis class H and θ defines $h \in$ H

-E.g. In classification ?

2. In regression ,

Dimensions of a Supervised Learner

We must make three decisions:

1. Model: $g(x|\theta)$

1. $g(\cdot)$ *model* x *input* θ *parameters*

$g(\cdot)$ Defines the hypothesis class H and θ defines $h \in$ H

-E.g. In classification rectangle is the model and the paramentes are the four coordinates

2. In regression , model is a linear function of the input, slope and intersect are the parameters

Dimensions of a Supervised Learner

2. Loss function: L()

Difference between desire outpot and approximation given the parameters

$$E(\theta \mid \mathcal{X}) = \sum_t L\left(r^t, g\left(\boldsymbol{x}^t \mid \theta\right)\right)$$

Class: learning 0/1

Regression: numerical value

Dimensions of a Supervised Learner

3. Optimization procedure: Find

$$\theta^* = \arg\min_\theta E(\theta \mid X)$$

the value of the parameters that minimize the total error.

Can be found analytically as in regression or through more complex optimization methods for more complicated models

Dimensions of a Supervised Learner

3. Optimization procedure: Find

the value of the parameters that minimize the total error.

Can be found analytically as in regression or through more complex optimization methods for more complicated models

Dimensions of a Supervised Learner

The following conditions should be satisfied:

- 1) Hypothesis class g() must be big enough
- 2) Enough training data to find the best hypothesis
- 3) Good optimization procedure

Different machine learning differ either in model, loss function or optimization procedure