Machine Learning 10-601

Tom M. Mitchell
Machine Learning Department
Carnegie Mellon University

September 25, 2012

Today:
- Linear regression
- Bias/Variance/Unavoidable errors

Readings:

Required:
- Bishop: Chapt. 1 through 1.2.5
- Bishop: Chapt. 3 through 3.2

Optional:
- Mitchell: Chapt. 6.4

Regression

So far, we've been interested in learning P(Y|X) where Y has discrete values (called 'classification')

What if Y is continuous? (called 'regression')
- predict weight from gender, height, age, …
- predict Google stock price today from Google, Yahoo, MSFT prices yesterday
- predict each pixel intensity in robot's current camera image, from previous image and previous action

Regression

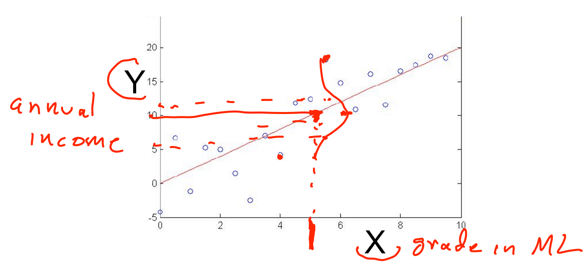
Wish to learn f:X→Y, where Y is real, given {<x^1,y^1>…<x^n,y^n>}

Approach:

1. choose some parameterized form for P(Y|X; θ)
   (θ is the vector of parameters)

2. derive learning algorithm as MLE or MAP estimate for θ

1. Choose parameterized form for P(Y|X; θ)



Assume Y is some deterministic f(X), plus random noise

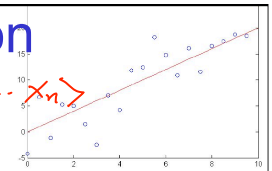
$$y = f(x) + \epsilon \qquad \text{where } \epsilon \sim N(0, \sigma)$$

Therefore Y is a random variable that follows the distribution

$$p(y|x) = N(f(x), \sigma)$$

and the expected value of y for any given x is $E_{p(x,y)}[y]=f(x)$

Consider Linear Regression



$X = \langle x_1, \ldots x_n \rangle$

$$p(y|x) = N(f(x), \sigma)$$

E.g., assume f(x) is linear function of x

$$f(x) = w_0 + \sum w_i x_i$$

$$p(y|x) = N(w_0 + \sum_i w_i x_i, \sigma)$$

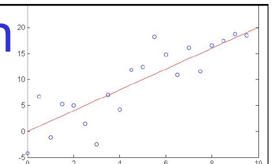
$$E_{p(x,y)}[y|x] = w_0 + \sum_i w_i x_i$$

Notation: to make our parameters explicit, let's write

$$W = < w_0, w_1 \ldots w_n >$$

$$p(y|x; W) = N(w_0 + \sum_i w_i x_i, \sigma)$$

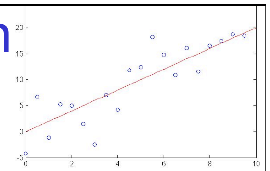
Training Linear Regression



$$p(y|x; W) = N(w_0 + w_1 x, \sigma)$$

How can we learn W from the training data?

Training Linear Regression



$$p(y|x;W) = N(w_0 + w_1 x, \sigma)$$

How can we learn W from the training data?

Learn Maximum Conditional Likelihood Estimate!

$$W_{MCLE} = \arg\max_W \prod_l p(y^l|x^l, W)$$

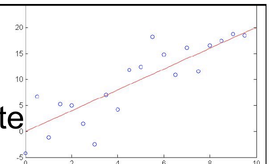
l^{th} training example

$$W_{MCLE} = \arg\max_W \sum_l \ln p(y^l|x^l, W)$$

where

$$p(y|x;W) = \frac{1}{\sqrt{2\pi\sigma^2}} \; e^{-\frac{1}{2}\left(\frac{y-f(x;W)}{\sigma}\right)^2}$$

Training and Regression



Learn Maximum Conditional Likelihood Estimate

$$W_{MCLE} = \arg\max_W \sum_l \ln p(y^l|x^l, W)$$

where $p(y|x;W) = \frac{1}{\sqrt{2\pi\sigma^2}} \; e^{-\frac{1}{2}\left(\frac{y-f(x;W)}{\sigma}\right)^2}$

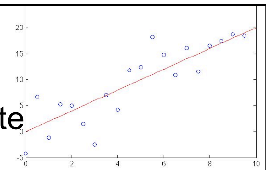
$$\sum_l \ln p(y^l|x^l;W) = \underbrace{\sum_l \ln\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) + \left(-\frac{1}{2}\left(\frac{y-f(x;w)}{\sigma}\right)^2\right)}_{Z}$$

$$W_{MCLE} = \arg\max_w Z$$

$$= \arg\max_w \sum_l -\frac{1}{2\sigma^2}\left(y^l - f(x^l;w)\right)^2$$

$$= \arg\min_w \sum_l \cancel{\frac{1}{2\sigma^2}} \left(y^l - f(x^l;w)\right)^2$$

Training and Regression



Learn Maximum Conditional Likelihood Estimate

$$W_{MCLE} = \arg\max_W \sum_l \ln p(y^l|x^l, W)$$

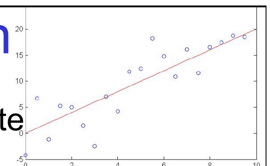
where $$p(y|x;W) = \frac{1}{\sqrt{2\pi\sigma^2}}\; e^{-\frac{1}{2}\left(\frac{y-f(x;W)}{\sigma}\right)^2}$$

$$\sum_l \ln p(y^l|x^l;W) =$$

so: $$W_{MCLE} = \arg\max_W \sum_l -(y^l - f(x^l;W))^2$$

$$= \arg\min_W \sum_l (y^l - f(x^l;W))^2$$

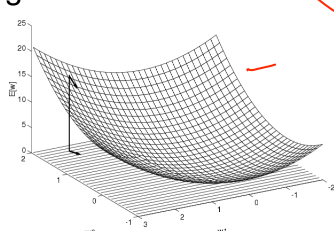
Training Linear Regression



Learn Maximum Conditional Likelihood Estimate

$$W_{MCLE} = \arg\min_W \underbrace{\sum_l (y - f(x;W))^2}_{E[w]}$$

Can we derive gradient descent rule for training?



Gradient

$$\nabla E[\vec{w}] \equiv \left[\frac{\partial E}{\partial w_0}, \frac{\partial E}{\partial w_1}, \cdots \frac{\partial E}{\partial w_n}\right]$$

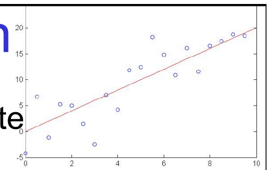
Training rule:

$$\Delta \vec{w} = -\eta \nabla E[\vec{w}]$$

i.e.,

$$\Delta w_i = -\eta \frac{\partial E}{\partial w_i}$$

Training Linear Regression



Learn Maximum Conditional Likelihood Estimate

$$W_{MCLE} = \arg\min_W \sum_l (y - f(x;W))^2$$

Can we derive gradient descent rule for training?

$$\frac{\partial \sum_l (y - f(x;W))^2}{\partial w_i} = 2(y - f(x;w)) \cdot \underbrace{\frac{\partial (y - f(x;w))}{\partial w_i}}_{-\frac{\partial f(x;w)}{\partial w_i}}$$

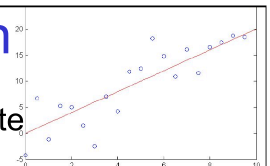
$$f(x) = w_0 + \sum_i w_i x_i$$

$$\frac{\partial (y - f(x))^2}{\partial w_i} = \sum_l 2(y^l - f(x))(-x_i^l)$$

And if $f(x) = w_0 + \sum_i w_i x_i$

$$w_i \leftarrow w_i + \eta \sum_l (y^l - f(x^l;W))\ x_i^l$$

Training Linear Regression



Learn Maximum Conditional Likelihood Estimate

$$W_{MCLE} = \arg\min_W \sum_l (y - f(x;W))^2$$

Can we derive gradient descent rule for training?

$$\frac{\partial \sum_l (y - f(x;W))^2}{\partial w_i} = \sum_l 2(y - f(x;W)) \frac{\partial (y - f(x;W))}{\partial w_i}$$

$$= \sum_l -2(y - f(x;W)) \frac{\partial f(x;W)}{\partial w_i}$$

And if $f(x) = w_0 + \sum_i w_i x_i$ ...

Gradient descent rule: $\boxed{w_i \leftarrow w_i + \eta \sum_l (y^l - f(x^l;W))\ x_i^l}$

How about MAP instead of MLE estimate?

$w_{MAP} \propto \boxed{P(Y|X;w)P(w)}$

Let's assume Gaussian prior: each $w_i \sim N(0, \sigma)$

$$\boxed{p(w_i) = \frac{1}{Z}\exp\left(-\frac{(w_i - 0)^2}{2\sigma^2}\right)}$$

w_i

Then MAP estimate is

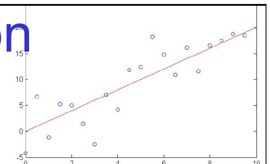
$$W = \arg\max_W \underbrace{-\frac{1}{2\sigma^2}\sum_{w_i \in W} w_i^2}_{\ln P(w)} + \sum_{l \in training\ data} \ln P(Y^l|X^l;W)$$

$$= \arg\min_W \frac{1}{2\sigma^2}\sum_{w_i \in W} w_i^2 + \sum_{l \in training\ data} (y^l - f(x^l;W))^2$$

Gradient descent:

$$\boxed{w_i \leftarrow w_i - \lambda w_i + \eta \sum_l (y^l - f(x^l;W))\ x_i^l}$$

Consider Linear Regression



$$p(y|x) = N(f(x), \sigma)$$

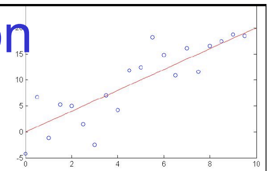
E.g., assume f(x) is linear function of x

$$f(x) = w_0 + \sum_i w_i x_i$$

$$p(y|x) = N(w_0 + \sum_i w_i x_i, \sigma)$$

$$\boxed{w_i \leftarrow w_i - \lambda w_i + \eta \sum_l (y^l - f(x^l;W))\ x_i^l}$$

Consider Linear Regression



$$p(y|x) = N(f(x), \sigma)$$

$X = \langle x_1, \dots, x_n \rangle$

E.g., assume f(x) is linear function of $\boxed{\phi_i(x)}$

$w_1 \phi_1(x) = x_1^2 x_3^5 + w_2 \phi_2(x) = (x_5 - x_7)^2$

$$f(x) = \sum_i w_i \phi_i(x)$$

$=$

$$p(y|x) = N\left(\sum_i w_i \phi_i(x),\ \sigma\right)$$

$f(x) = w_1 w_2 x_1$

$$\boxed{w_i \leftarrow w_i - \lambda w_i \ + \eta \sum_l (y^l - f(x^l; W))\ \phi_i(x^l)}$$

Regression – What you should know

Under general assumption $p(y|x; W) = N(f(x; W), \sigma)$

1. MLE corresponds to minimizing Sum of Squared prediction Errors
2. MAP estimate minimizes SSE plus sum of squared weights
3. Again, learning is an optimization problem once we choose our objective function
   - maximize data likelihood
   - maximize posterior probability, P(W | data)
4. Again, we can use gradient descent as a general learning algorithm
   - as long as our objective fn is differentiable wrt W
5. Nothing we said here required that f(x) be linear in x -- just linear in W
6. Gradient descent is just one algorithm – linear algebra solutions too

Decomposition of Error in Learned Hypothesis

1. Bias

2. Variance

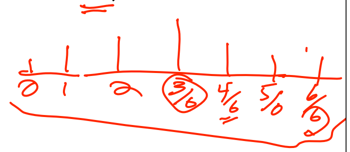
3. Unavoidable error

Bias and Variance

$E[A] = \sum_i P(A=i)\, i$

given some estimator A for some parameter θ, we define

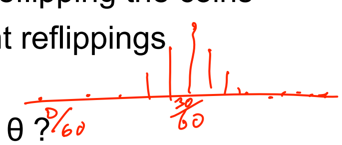
$$bias(A) = E[A] - \theta$$
$$var(A) = E[(A - E[A])^2]$$



e.g., θ is probability of heads for a coin, A is the MLE estimate for θ, based on *n* independent coin flips

A is a random variable, sampled by reflipping the coins

Expected value is taken over different reflippings



is A biased or unbiased estimator for θ ?

variance decreases as sqrt(1/*n*)

Bias – Variance decomposition of error

Reading: Bishop chapter 3.2 (different notation)

- Consider simple regression problem f:X→Y

$y = f(x) + \varepsilon$

deterministic: f(x)

noise $N(0,\sigma)$: ε

What is expected error of a hypothesis learned (estimated) from randomly drawn training data D? ← n training examples

$$E_D\left[\int_y \int_x (h_D(x) - y)^2 \underbrace{p(y|x)p(x)}_{p(x,y)}\, dydx\right]$$

learned estimate of f(x), from training data D: $h_D(x)$

Sources of error

= $

- What if we have perfect learner, infinite data?
  - Our learned h(x) satisfies h(x)=f(x)
  - Still have remaining, *unavoidable error* because of ε

    $y = f(x) + \boxed{\varepsilon,}$ $\varepsilon \sim N(0,\sigma)$

$$[h(x) = f(x)] \rightarrow [(h(x) - y)^2 = \sigma^2]$$

Sources of error

- What if we have only n training examples?
- What is our expected error
  - Taken over random training sets of size n, drawn from distribution $D=p(x,y)$?

Decomposition of error: $y = f(x) + \varepsilon;\quad \varepsilon \sim N(0,\sigma)$

learned estimate of f(x), from training data D

$$E_D\left[\int_y \int_x (h_D(x) - y)^2 p(y|x) p(x)\ dy dx\right]$$

$$= unavoidableError + bias^2 + variance$$

$$unavoidableError = \sigma^2$$

$$bias^2 = \int_x (E_D[h_D(x)] - f(x))^2\ p(x) dx$$

esp when hyp lang H is insufficient to repr. funct

$$variance = \int_x E_D[(h_D(x) - E_D[h_D(x)])^2]\ p(x) dx$$

esp. when training set size is small

Error Decomposition: Summary

Expected true error of learned P(y|x) for regression (and similarly for classification) has three sources:

1. Unavoidable error: even with perfect estimate
   – non-determinism in world prevents perfect predictions
2. Bias:
   – even with infinite training data, hypothesis h(x) might not equal true f(x). E.g., if learner's hypothesis representation cannot represent the true f(x)
3. Variance
   – Whenever we have only finite training data, the sample of just n training examples might represent an empirical distribution that varies from the true P(Y|X). i.e., if we collect many training sets of size n, the empirical distribution they represent will vary about P(Y|X).