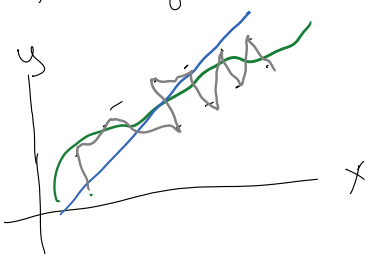
CSCC11 Tutorial

September 12, 2018 11:00 AM

T.A: Bryan Chan - email on course website



linear regression

create a function

$\hat{f}(x) = y^{*1}$ ← prediction

$\hat{g}(x) = y^{*2}$ ← prediction

how do we determine which is good? define error function

Error function to measure performance of the model.

Errorfunction choices

- Absolute function (F_1) → can't differentiate
- Least square (F_2)

Least squares:

- always non-negative ≥ 0
- Differentiable

In 1-D: were given data $\{(x_i, y_i)\}_{i=1}^{N}$

x_i: input, y_i: output

To formulate error:

$$E(w) = \sum_{i=1}^{n} (y_i - x_i \cdot w)^2$$

y_i: observation, $x_i \cdot w$: prediction

$$= \|\vec{y} - \vec{x}w\|_2^2 \quad \text{2 is 2nd norm}$$

$\vec{x} = [x_i]$

b is bias can be added.

find difference between observed output and input.

$= \|y - x w\|_{2 \leq 2^{nd}\ norm}$

$\vec{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad \vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$

Observed output & not input. find distance. Motivation of cross function

In multiple dimension, we have

$(\vec{x}_i, y_i)$, $x \in \mathbb{R}^d$

$E(\vec{w}) = \|\vec{y} - X\vec{w}\|^2_{2 \leq 2^{nd}\ norm}$, $X = \begin{bmatrix} \vec{x}_1^T \\ \vec{x}_2^T \\ \vec{x}_n^T \end{bmatrix}$, $\vec{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_d \end{bmatrix}$

expanding is the dot product of itself

$= (\vec{y} - X\vec{w})^T (\vec{y} - X\vec{w})$

$= \vec{w}^T X^T X w - 2\vec{y}^T X \vec{w} + \vec{y}^T \vec{y}$

quadratic

Optimize it: Goal to minimize it. Take gradient and set it to zero, $(\vec{0})$. This will guarantee a minimum

Linear Review

1. $(AB)^T = B^T A^T$
2. $(AB)^{-1} = B^{-1} A^{-1}$ ← assuming inverse exist
3. $(A^{-1})^T = (A^T)^{-1}$
4. $|AB| = |A||B|$

   ↑ determinant (AB)
5. $|A^{-1}| = \frac{1}{|A|}$
6. A square matrix is orthogonal if every column vector

is orthogonal (cuckoo product=0) and normalized

dot product of vector by itself is 1 ($\vec{x}^T x = 1$)

Assume matrix are symmetric

Understand what orthogonal, symmetric, and invertible means.

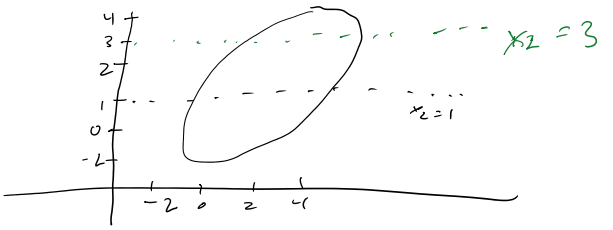
7. A square matrix "A" is not singular (invertible) if

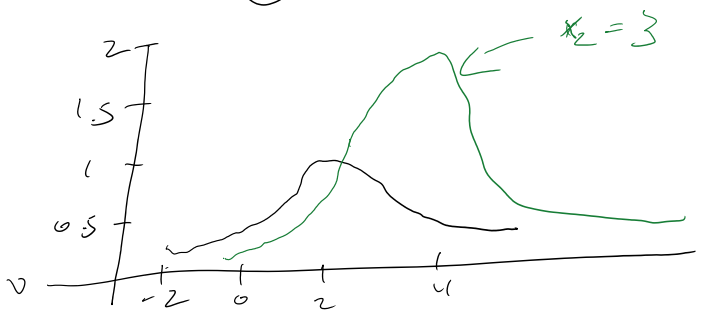
Conditional (2D-gaussian)

Let $\vec{x} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, $\vec{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}$, $C = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$, $\Lambda = C^{-1} = \begin{bmatrix} \Lambda_{11} & \Lambda_{12} \\ \Lambda_{21} & \Lambda_{22} \end{bmatrix}$

∴ the conditional distribution of x_1 given x_2 satisfies $x_1 | x_2 \sim \mathcal{N}(\mu_{x_1|x_2}, \Lambda_{11}^{-1})$, where where $\mu_{x_1|x_2} = \mu_1 - A_{11}^{-1} A_{12} (x_2 - \mu_2)$, note that Λ_{11}^{-1} is not simply C_{11}

For ex





Diagonalization

Given a covariance matrix with eigen vectors $\vec{u}_1, \vec{u}_2$ and corresponding eigenvalues λ_1, λ_2

let $U = \begin{bmatrix} | & | \\ u_1 & u_2 \\ | & | \end{bmatrix}$, $S = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$,

then $CU = US$, $C\vec{u}_1 = \lambda_1 S$

$C\vec{u}_2 = \lambda_2 S$,

since U is orthonormal, orthonormal if not

$C = USU^{-1} = USU^T$ take the inverse you get the transpose

$\Rightarrow C^{-1} = (USU^{-1})^{-1} = US^{-1}U^{-1}$

$= US^{-1}U^T$

$\left(-\frac{1}{2}(\vec{x}-\vec{u})^T C^{-1}(\vec{x}-\vec{u})\right)$, sub C^{-1} in,

$-\frac{1}{2}(\vec{x}-\vec{u})^T US^{-1}\underbrace{U^T(\vec{x}-\vec{u})}$

change of variable

$y = U^T(\vec{x}-\vec{u}) \Rightarrow \left(-\frac{1}{2}\vec{y}^T S^{-1}\vec{y}\right)$

y is still a gaussian, with mean $= 0$, and covariance of S.

So S is diagonal matrix $\Rightarrow y_1, y_2$ are $\perp$

if covariance is diagonal then $y_1 \perp y_2$

$\Rightarrow$ Eigen values of C are the variances along the principle directions given by the eigenvectors used for the diagonalization.

Positive Definite

A $d \times d$ matrix A is positive definite iff $\forall \vec{z} \in \mathbb{R}^d - \{\vec{0}\}$, $\vec{z}^T A \vec{z} > 0$

if $\vec{z} A \vec{z} \geq 0$, then positive semi definite

why is $C > 0$?

1) Covariance is the second moment of a PDF (shifted by $\vec{u}$)

$\Rightarrow C = E((\vec{x}-\vec{\mu})(\vec{x}-\vec{\mu})^T)$, $E(a^n)$

If C is full rank (invertible) then

$$\vec{z}^T C \vec{z} = \vec{z}^T E((\vec{x}-\vec{\mu})(\vec{x}-\vec{\mu})^T)\vec{z}$$

$$= E(z^T(x-\mu)(x-\mu)^T z)$$

$$= E(u^2), \quad u = (\vec{x}-\vec{\mu})^T z$$

$$> 0$$

mean will always be positive because of the square.

$\Rightarrow C$ is positive definite.
if $C \succeq 0$, eigen ≥ 0

Why?

$C\vec{z} = \lambda\vec{z}$, $\vec{z}$ eigen vector (unit norm)
$\Leftrightarrow z^T C z = \lambda > 0$, λ eigen value

$\therefore \lambda \geq 0$

Block Matricies (Reference: matrix identity)

A block matrix is a matrix that is interpreted as having been breaking into sections called blocks.

$$U = \left[\begin{array}{c|c} U_1 & U_2 \end{array}\right]$$

2×1 2×1

Useful in higher dimensions, use it to marginalize and condition

$p(x_1, x_2)$

$$p(x_1) = \int_p (x_1, x_2)\, dx_2$$
$$p(x_2) = \int p(x_2, x_1)\, dx_1$$

For the case of gaussians

Covariance

$$\Sigma = \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \quad \begin{matrix} A \in \mathbb{R}^{m \times m} \\ D \in \mathbb{R}^{n \times n} \end{matrix}$$

$$B = C^T$$

if $\Sigma = \begin{bmatrix} A & 0 \\ 0 & D \end{bmatrix}$ then $\Sigma^{-1} = \begin{bmatrix} A^{-1} & 0 \\ 0 & D^{-1} \end{bmatrix}$

$$\det = \det(A) \det(D)$$

October 3, 2018 11:10 AM

Regression - Quiz

helpful tips:
1 generate noiseless data set
create own data set/training set
penalty for efficiency.
↳ around 5 seconds

Bayes's Rule

$$P(\vec{w}|D) = \underbrace{P(D|\vec{w})}_{\text{likelihood}} \times \frac{\overbrace{P(\vec{w})}^{\text{prior}}}{\underbrace{P(D)}_{\text{evidence}}}$$

posterior

$P(D) = \int P(D, w)\,dw$
$= \int P(D|w)P(w)\,dw$

$\vec{w}$ parameter of given model

D = training data.

$$\Rightarrow P(\vec{w}|D,M) = \frac{P(D|\vec{w},M) \times P(\vec{w}|M)}{P(D|M)}$$

We care about different models.

In a1, we are trial and erroring. But there is a more rigorous way using Bayes's Rule.

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)} \leftarrow \text{constant}$$

↳ Maximize posterior (give the "best" model)

Ex/ Estimating Gaussian Distribution

Suppose we are learning a gaussian distribution from n training data $\{\vec{x}_i\}_{i=1}^N$ and we want to know the best parameter $(\vec{\mu}, \Sigma)$ for this distribution.

We already have the likelihood $= P(\vec{X}_{i=N}|\vec{\mu}, \Sigma)$

$= P(\vec{x}_1, \vec{x}_2, \ldots \vec{x}_N | \vec{\mu}, \Sigma)$ (i.i.d)

$= \prod_{i=1}^{N} P(\vec{X}_{i=N}|\vec{\mu}, \Sigma)$

$= \prod_{i=1}^{N} \frac{1}{\sqrt{(2\pi)^D |\Sigma|}} \exp\left(-\frac{1}{2}(\vec{x}_i - \vec{\mu})^T \Sigma^{-1} (\vec{x}_i - \vec{\mu})\right)$, note $(2\pi)^D$ ← D is Dimension

⇒ Because we restricted M = Gaussian, we

only need to maximize the likelihood (MLE).

$\Rightarrow$ Maximizing the above is complicated so instead, take the negative log-likelihood, minimizing it

$$\rightarrow L(\vec{\mu}, \Sigma) = -\ln\left(p(\vec{x}_{1:N} | \vec{\mu}, \Sigma)\right)$$

$$= -\sum_{i=1}^{N} p(\vec{x}_i | \vec{\mu}, \Sigma)$$

$$= \sum_{i=1}^{N} \left[\frac{(\vec{x}_i - \vec{\mu})^T \Sigma^{-1} (\vec{x}_i - \vec{\mu})}{2} + \frac{N}{2} \ln|\Sigma| + \frac{ND}{2} \ln(2\pi) \right]$$

$$\mu^* = \arg\max_{\mu} p(\vec{x}_{1:N} | \vec{\mu}, \Sigma) = \arg\min_{\mu} L(\vec{\mu}, \Sigma)$$

$$\Sigma^* = \arg\max_{\Sigma} p(\vec{x}_{1:N} | \vec{\mu}, \Sigma) = \arg\min_{\Sigma} L(\vec{\mu}, \Sigma)$$

can solve them by

$$\frac{\partial L}{\partial \vec{\mu}} = 0 \qquad \frac{\partial L}{\partial \Sigma} = 0$$

$$\vec{\mu}^* = \frac{1}{N} \sum_{i=1}^{N} \vec{x}_i$$

$$\Sigma^* = \frac{1}{N} (\vec{x}_i - \vec{\mu}^*)(\vec{x}_i - \vec{\mu}^*)^T$$

Entropy & Information Theory

- Entropy measures uncertainty of a distribution.
- Entropy is a log measure of uncertainty associated with a r.v.
- Use entropy for decision tree, K-L divergence, cross entropy
  (K-L divergence: common measure of distance or ML.)
- Give a discrete r.v. y, which takes on K values, entropy is defined as $h = -E_{p(c)}[\log p(c)] = -\sum_{c=1}^{K} p(c) \log(p(c))$
  ($E_{p(c)}$: expectation)

Say we have $K=8$, $p(c=i) = \frac{1}{8}$ $\forall i = 1, \ldots, 8$

$H = -8\left(\frac{1}{8}\log\left(\frac{1}{8}\right)\right) = 3$ ← minimum bound

This is used for data compression too.

1 hour, 4 – 5 questions. upto classification.

Short questions

- understand formulate least square
- understand entropy
- decision tree
- gcc, linear regression
- map and MLE

} short answer

Understand definition.

Optimization

We have $E(w)$, Find $\arg\min_w E(w)$ ← weight that will minimize error

To find $\arg\min_w E(w)$, take the gradient and isolate for w.

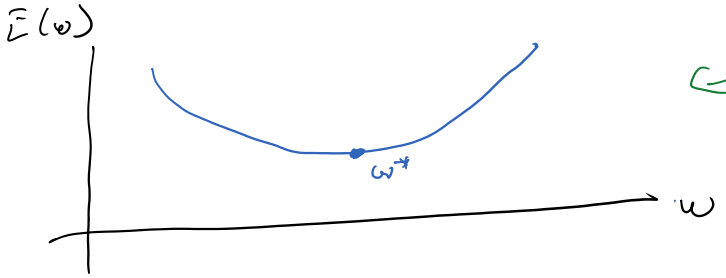
L.S regression - can get closed form solution

$$w^* = (X^T X)^{-1} X^T y$$

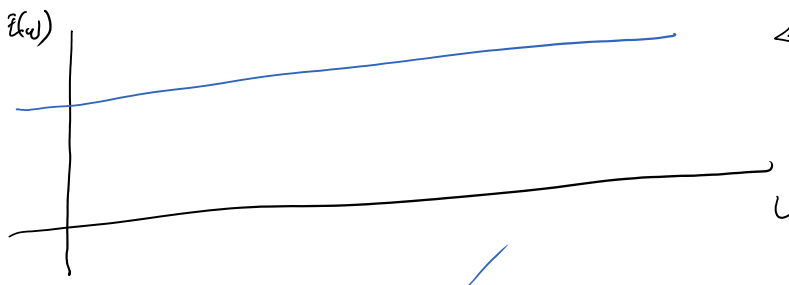
computation for inverse is costly

L.S regression cost function

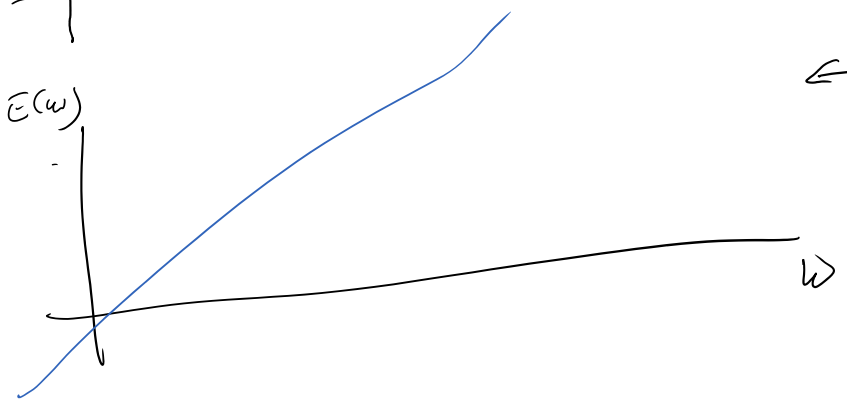
$$E(w) = w^T X^T X w - 2 y X w + y^T y$$



← draw in terms of parameter space



← no point, cause you won't be able to learn



← no minimum can go negative infinity.

L2 loss is popular because you can derive it easily.

Gradient Decent (uses first order Taylor expansion)

- Can approximate error function

$$\approx E(\vec{w}_0) + (\vec{w} - \vec{w}_0) \frac{\partial E(\vec{w}_0)}{\partial \vec{w}}$$

- Goal: to minimize error

$\Rightarrow \min E(\bar{w})$

$\Rightarrow \min_{\vec{h}} E(\bar{w}_0 + \bar{h}) \approx \min_{h} E(\bar{w}_0) + \bar{h} \frac{\partial E(\bar{w}_0)}{\partial h}$

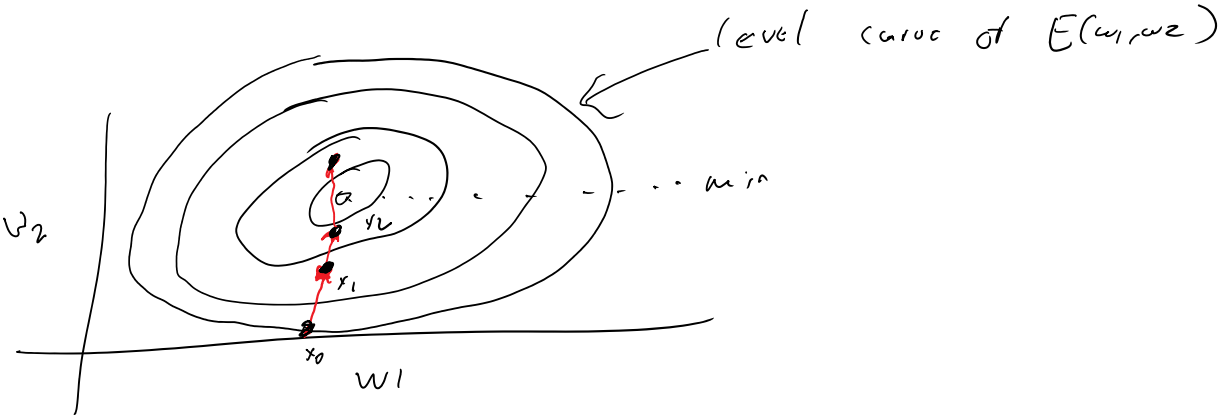
$\Rightarrow \min_{\vec{h}} \frac{\partial E(\bar{w}_0)}{\partial h}$ ← take gradient

$\Rightarrow \bar{h}^* = -\nabla E(\bar{w}_0)$

$\Rightarrow w_{i+1} = w_i - \alpha \frac{\partial E(\bar{w}_i)}{\partial \bar{w}}$ } optimization step

α: step size

$\vec{w_0} = ?$ → anything except 0



Gradient decent is used to get local minimum.

set step size as a hyper parameter