COL774 - ML

parags

Suyash Agrawal

04/01/18

1) Basic Concepts
2) Learning Specific algorithms
3) Hands on experience
4) Art of applying ML.

} What to expect

Group - 5, S.No 112
TA- Vishal

What is ML?
↳ • identifying patterns
• predict future behaviour

Why ML now?
↳ • Large amounts of data.
• Inc. in computation power
• Well founded models.

Types of ML algorithms -
↳ • Supervised Learning

05/01/18

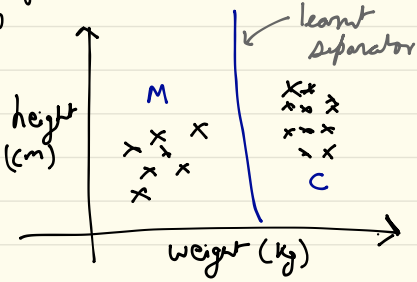
① Supervised Learning

$\langle x_1^{(i)}, \ldots, x_n^{(i)} \rangle$ $y^{(i)}$

$\langle x_1, \ldots, x_n \rangle$ y ?

$x_j^{(i)} \Rightarrow i^{th}$ feature vector of j^{th} example
$m \Rightarrow$ # of examples
$n \Rightarrow$ # of attributes

can be classification or regression



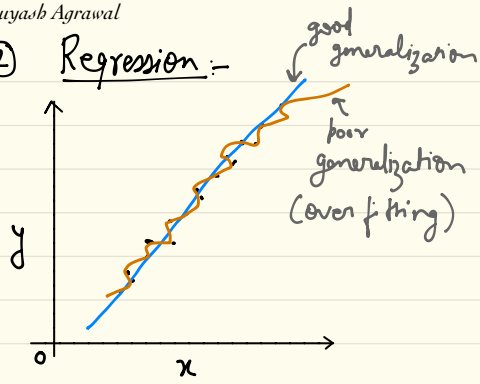
We are looking for a hypothesis $(h: X \to Y) \in H$
(all possible input combination; hypothesis class)

1) Which class of hypothesis?
2) Metric for being optimal
3) How to find the f^n?
4) Dealing with outliers.

Algos
- SVM
- Logistic Regression
- GDA

Suyash Agrawal

② Regression :-

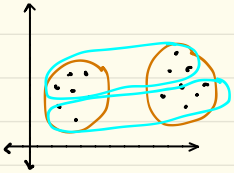


$h: X \rightarrow Y$

Area	Prid
2140	2
1100	1.5
⋮	⋮

③ Unsupervised Learning :-

$\langle x_1^{(1)}, \ldots, x_n^{(1)} \rangle$
$\vdots$
$\langle x_1^{(m)}, \ldots, x_n^{(m)} \rangle$ } ? ? ! ?

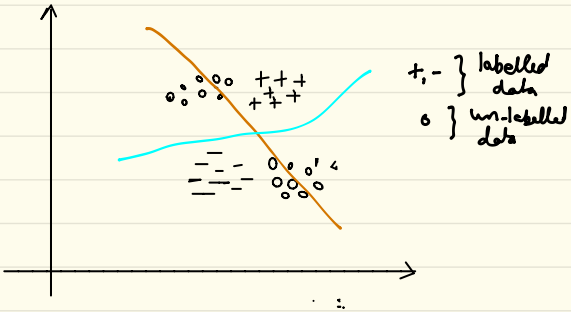


1) How many clusters?
2) Metric for quality of clustering

Algo :-
- K-Means

④ Semi Supervised Learning :-

$\langle x_1^{(1)}, \ldots, x_n^{(1)} \rangle$ $y^{(1)}$
$\vdots$
$\langle x_1^{(i)}, \ldots, x_n^{(i)} \rangle$?
$\langle x_1^{(m)}, \ldots, x_n^{(m)} \rangle$ y_m



⑤ Reinforcement Learning

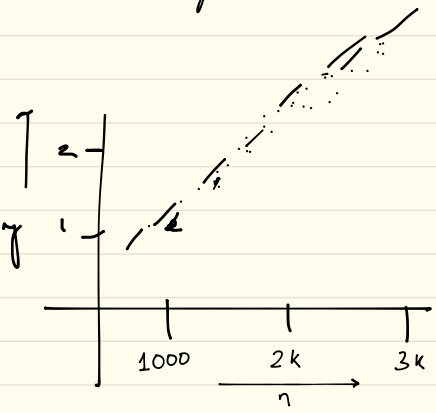
S- states , R- reward "Explore" / "Exploit"
A- actions

Suyash Agrawal

09/01/18

Video Show

Linear Regression



$h: X \longrightarrow Y$
X ↳ given

Aim: $h(x) \simeq y$

1) Sq. error is differentiable. } Why choose sq. error?
2) hypothesis class
   ↳ set of all linear functions

$$h(x) = \theta_1 x + \theta_0$$

$\langle x_1^{(i)}, \dots, x_n^{(i)} \rangle \; y^{(i)}$ ↳ one example

$\{x^{(i)}, y^{(i)}\}_{i=1}^{m}$ ← training examples

$h: X \longrightarrow Y$ | $h(x)$ should be as close to y as possible

$h \in H$ ↳ hypothesis class

$$h_\theta(x) = \underbrace{\theta^T}_{\theta = n\times 1} \underbrace{x}_{x = n\times 1} + \underbrace{\theta_0}_{\theta_0 = 1\times 1} = \theta_2 x_2 + \theta_1 x_1 + \theta_0 \quad (2D)$$

$$= \sum_{j=1}^{n} \theta_j x_j + \theta_0 x_0 \quad (x_0 \text{ ↳ dummy var. } x_0 = 1)$$

$$= \sum_{j=0}^{n} \theta_j x_j$$

We try to minimize :- $\text{Error} = \frac{1}{2} \sum_{i=1}^{m} \left[h_\theta(x^{(i)}) - y^{(i)}\right]^2$

- How to minimize the error? $\{\underset{\theta}{\text{argmin}}\, J(\theta)\}$

Suyash Agrawal

11/01/18

Linear Regression:

1) Hypothesis space
2) Loss function : $J(\theta) = \frac{1}{2}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$
3) Optimization Algo

Supervised Learning:-

Training data $\{x^{(i)}, y^{(i)}\}_{i=1}^{m}$ → Learning Algo } Linear Regression → Model : h_θ

$x \rightarrow h_\theta(x) \simeq y$

Vector Representation

- Default is column vector.

$\theta, x \in \mathbb{R}^{n+1}$ $x = \begin{bmatrix} x_0 \\ \vdots \\ x_n \end{bmatrix}$ $\theta = \begin{bmatrix} \theta_0 \\ \vdots \\ \theta_n \end{bmatrix}$

$h_\theta(x) = \theta^T x$

$J_\theta = \frac{1}{2}\sum_{i=1}^{m}[\theta^T x^{(i)} - y^{(i)}]^2$ squared error.

objective: $\underset{\theta}{\text{argmin}}\ J(\theta)$

Norm of a vector:-

$x \in \mathbb{R}^n$, $\underbrace{\|x\|_p}_{\text{p norm}}$ of x is defined as $= \left[\sum_{j=1}^{n}|x_j|^p\right]^{1/p}$

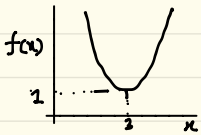
$\|x\|_2 \Rightarrow$ 2 norm (L2)

$\|x\|_1 \Rightarrow$ 1 norm (L1)

$L_0 \Rightarrow$ number of non zero entries

$L_\infty \Rightarrow \max_j |x_j|$

- How does one minimize a function?

↳ 

$f(x) = (x-3)^2 + 1$

$\underset{x}{\text{argmin}}\ f(x) = 3$

Methods:-

① Find $\frac{\partial f(x)}{\partial x}$ & find zero point

then use $\frac{\partial^2 f(x)}{\partial x^2} > 0$ to check minima

if $x \in \mathbb{R}^n$: $\nabla_x f(x) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix}$, at min / max $\nabla_x f(x) = 0$

Checking Min / Max:

Hessian (H) = $\begin{bmatrix} & & j & \\ & & \vdots & \\ i & \cdots & \frac{\partial^2 f}{\partial x_i x_j} & \cdots \\ & & \vdots & \end{bmatrix}$

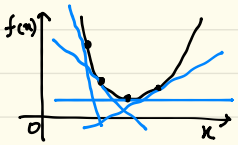
gives an idea of how fast gradient is changing & tells about curvature.

$f(x) = x_1^2 x_2 + x_2$

$H = \begin{bmatrix} 2x_2 & 2x_1 \\ 2x_1 & 0 \end{bmatrix}$

Suyash Agrawal

- **Gradient Descent**
  - It is not easy to find the roots of the gradient of function.
  - Move opposite to gradient.



$$x^{t+1} = x^t - \eta \left.\frac{\partial f(x)}{\partial x}\right|_{x^t}$$

η: Learning rate

Engineering Problem
1. Where to start
2. Learning rate

- As we move towards minima the step size decreases & progress slows down.

Multivariate :-

$$x^{t+1} = x^{(t)} - \eta \cdot \nabla_x f(x)$$

Is η a vector/scalar & Why?

12/01/18

- How to find $\arg\min_\theta J(\theta)$?
  - Compute $\frac{\partial J(\theta)}{\partial \theta_j}$

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{2}\sum_{i=1}^{m} 2\left[y^{(i)} - h_\theta(x^{(i)})\right]\cdot\left(-\underbrace{\frac{\partial h_\theta(x^{(i)})}{\partial \theta_j}}_{x_j^{(i)}}\right) \quad \text{(in linear regression)}$$

$$= \sum_{i=1}^{m}(-1)\cdot\left[y^{(i)} - h_\theta(x^{(i)})\right]\cdot x_j^{(i)}$$

$$\therefore \theta_j^{t+1} = \theta_j^t + \eta \cdot \sum_{i=1}^{m}\left[y^{(i)} - h_\theta(x^{(i)})\right]\cdot x_j^{(i)}$$

Linear Regression update rule.

Vector: $\theta^{t+1} = \theta^t + \eta \sum_{i=1}^{m}\left[y^{(i)} - h_\theta(x^{(i)})\right]x^{(i)}$

- Initialization
- Learning rate
- Convergence

} Engineering Questions

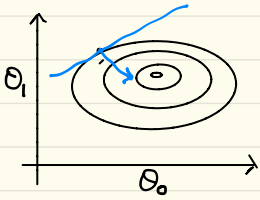
Pseudo code for Gradient descent

use:- $\theta^{t+1} = \theta^t - \eta \nabla_\theta J(\theta)$

- **Understanding $J(\theta)$**: (In case of linear regression)

$$J(\theta) = \frac{1}{2}\sum_{i=1}^{m}\left[\theta^T x^{(i)} - y^{(i)}\right]^2$$

quadratic in θ_j



Countours of J(θ)

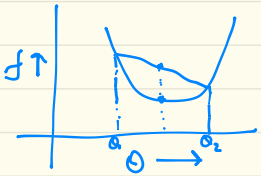
- **Initialization** :-
  - Randomly initializing close to zero works well (Practically).
- **Learning Rate** (η):
  - Start slow & gradually increase until divergence signs
  - Start large & gradually slow down.
- **Converge** :-
  - $|J(\theta^{t+1}) - J(\theta^t)| < \epsilon$
  - $|\theta^{t+1} - \theta^t| < \epsilon$ } Check L_∞ or L_2 norm.
  - $|\nabla_\theta J(\theta)| < \epsilon''$ }

If we choose ϵ to be very small then may take long time.

Lecture 07 - 16.01.18

- **Stochastic Gradient Descent (S.G.D)**

- **Convexity**: (Convex functions)



$f: R \to R$, f is convex iff:

$\forall\ \theta_1, \theta_2 \in R\ \ f(\alpha\theta_1 + (1-\alpha)\theta_2) \le \alpha f(\theta_1) + (1-\alpha) f(\theta_2)$

In strictly convex we have $<$

Convex functions have unique global minima. { atleast unique value }

- If f is cont. & differentiable: convexity $\Leftrightarrow f'(x) \ge 0$

  Concave $\Leftrightarrow f'(x) \le 0$

- **For higher dimensions**:

$f: \mathbb{R}^n \to \mathbb{R}^n$ is convex iff:

$f(\alpha\theta_1 + (1-\alpha)\theta_2) \le \alpha f(\theta_1) + (1-\alpha) f(\theta_2) \quad (0 \le \alpha \le 1) \quad \forall\ \theta_1, \theta_2 \in \mathbb{R}^n$

2nd order condition:

Hessian $H = \begin{bmatrix} & & j^{th} \\ & \vdots & \\ i^{th} \cdots & \frac{\partial^2 f}{\partial\theta_i\, \partial\theta_j} & \end{bmatrix}$

Convex $\Leftrightarrow H \succeq 0$ [H should be positive semi-definite]

Suyash Agrawal

Positive Semi definite -

Let $A \in \mathbb{R}^{n \times n}$ symmetric. A is +ve semi-definite if

$$\forall\, z \in \mathbb{R}^n \quad z^T A z \geq 0$$

Eg. $I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $\begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$

How to check if a matrix is positive semi-definite ?

↳ Eigen values ≥ 0 ?

We can prove, $J_\theta = \frac{1}{2} \sum_{i=1}^{m} \left[\theta^T x^{(i)} - y^{(i)}\right]^2$, is convex by calculating Hessian & showing it is +ve semi-definite.

H.W. Calculate Hessian of $J(\theta)$. $\Rightarrow X^T X$ $\quad \{\nabla_\theta J(\theta) = X^T X\theta - X^T y\}$

Gradient Descent

```
Θ = init()
t = 0
do {
    for j = 1 to n {
        Θ_j^{t+1} = Θ_j^t + η Σ_{i=1}^{m} [y^(i) - h_θ(x^(i))] · x_j^(i)
    }
    t += 1
} while (! converged)
```

m: examples, n: attributes

$O(mn)$ time expensive!

Stochastic Grad. Descent

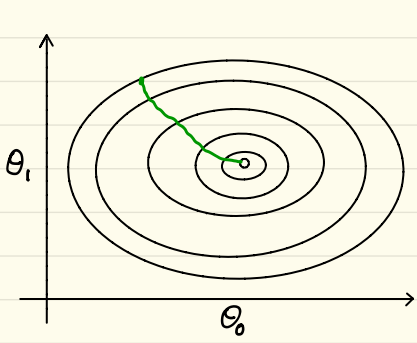
- Examples are iid.
- We divide training set in batches

```
Batch size = r
Θ = init()
t = 0 ; c = m/r
do {
    for b = 1 to c {
        for j = 1 to n {
            Θ_j^{t+1} = Θ_j^t + η Σ_{i=(b-1)r+1}^{br} [y^(i) - h_θ(x^(i))] · x_j^(i)
        }
        t += 1 ;
    }
} while (! converged)
```

$O(rn)$ time

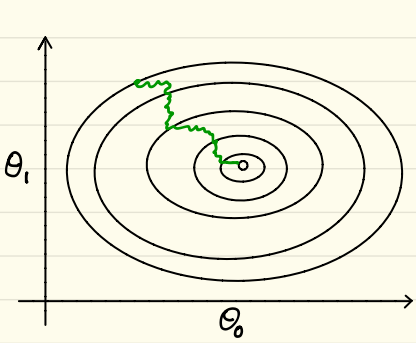
1 epoch: Seeing whole training data one time

Suyash Agrawal



Gradient Descent

$O(mnT)$



S.G.D.

$O(rnT')$

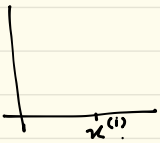
- $T \leq T'$ but iterations are faster & thus faster convergence
- Take smaller steps.
- Prove not converge to same minima.

Heuristic: $\eta \propto \frac{1}{t}$ or $\eta \propto \frac{1}{\sqrt{t}}$ η: learning rate, t: iterations done

Probabilistic Interpretation of Least Squares:

$$J_\theta = \frac{1}{2} \sum_{i=1}^{m} [\theta^T x^{(i)} - y^{(i)}]$$ (2) ← why? Also, why vertical dist & not dist from line?

Generating the data:-



Assume: underlying parameters θ

Input: $x^{(i)}$

① Apply Linear Model: $\theta^T x^{(i)}$

② $\epsilon^{(i)} \sim N(0, \sigma^2)$. (i.i.d noise)

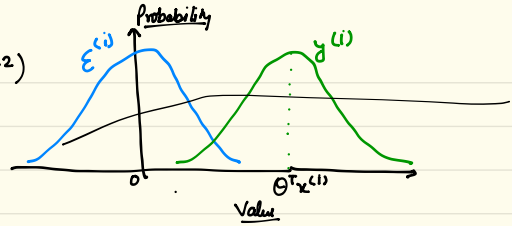
③ $y^{(i)} = \theta^T x^{(i)} + \epsilon^{(i)}$

$$\epsilon^{(i)} \sim N(\mu, \sigma^2)$$

$$P(\epsilon^{(i)}) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(\epsilon^{(i)} - \mu)^2}{2\sigma^2}}$$

Suyash Agrawal

$\therefore y^{(i)} | x^{(i)}; \theta \sim N(\theta^T x^{(i)}, \sigma^2)$

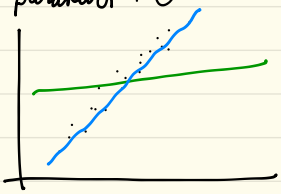


- **Maximum Likelihood Estimation :- (ML Estimation)**

Data = $\{x^{(i)}, y^{(i)}\}_{i=1}^m$, Estimate a set of parameter : θ

$$\theta_{ML} = \underset{\theta}{\text{argmax}}\ P(D; \theta)$$

max. likelihood



Blue line has more probability of generating the data than Green line

$$P(y^{(1)}, \cdots, y^{(m)} | x^{(1)}, \cdots, x^{(m)}; \theta) = \prod_{i=1}^{m} P(y^{(i)} | x^{(i)}; \theta) = \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}}$$

$P(y^{(1)}, \cdots, y^{(m)} | x^{(1)}, \cdots, x^{(m)}; \theta)$ = $L(\theta)$ = Likelihood

as i.i.d.

$$\underset{\theta}{\text{argmax}}\ L(\theta) = \underset{\theta}{\text{argmax}}\ LL(\theta)$$

Log Likelihood.

$$LL(\theta) = \sum_{i=1}^{m} \left(\log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) - \frac{1}{2}\left(\frac{y^{(i)} - \theta^T x^{(i)}}{\sigma}\right)^2 \right)$$

$$\therefore \underset{\theta}{\text{argmax}}\ LL(\theta) = \underset{\theta}{\text{argmin}} \sum_{i=1}^{m} \frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}$$

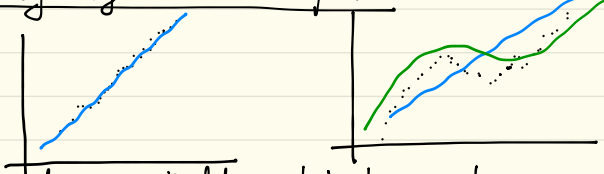
{ Removing the constants & taking argmin because of − }

ignore, because its a +ve constant

$$= \underset{\theta}{\text{argmin}}\ J(\theta)$$

$\therefore$ Least squared minimization is equivalent to max. likelihood estimation under the assumption of i.i.d gaussian noise.

- **Locally weighted Linear Regression :**



$$h_\theta(x) = \sum_{r=0}^{n} \theta_r x^r$$

Formulate as linear

$$\phi : x \to \langle x, x^2, x^3, \cdots, x^r \rangle$$

Kernel Trick?

We can also estimate data as being piece wise linear.

$$J(\theta) = \sum^{m} w^{(i)} [y^{(i)} - \theta^T x^{(i)}]^2 \quad ; \quad w^{(i)} = f(x^{(i)}, x) = e^{-\frac{(x - x^{(i)})^2}{2\sigma^2}}$$

• Locally weighted Linear Regression :

① Use Kernels : $x \to \phi(x)$

② $J(\theta) = \sum_{i=1}^{m} w^{(i)} \left[y^{(i)} - \theta^T x^{(i)}\right]^2$

$w^{(i)} = e^{-\frac{[x - x^{(i)}]^2}{2\tau^2}}$ (x: point of interest)

$\tau \to \infty$ } regular linear regression

$\tau \to 0$ } only x would have weight 1.

} ways to solve

- Underfitting when your hypothesis space is not able to fit the data.
- Lazy Learners : no work done during training { this , K-nn }
- For every point we create a new loss function $J(\theta)$
- This is a non-parametric method.
- As new point comes, we create a new $J(\theta)$ for this & minimize this to get θ & then we predict y as $\theta^T x$. i.e. for every point we learn a new θ.

- For $x \in \mathbb{R}^n$: $w^{(i)} = e^{-\frac{[(x^{(i)} - x)^T \Sigma^{-1} (x^{(i)} - x)]}{2}}$

- Linear Regression:

$J(\theta) = \frac{1}{2} \sum_{i=1}^{m} \left[y^{(i)} - \theta^T x^{(i)}\right]^2$

Analytical Solution for θ parameter:

$\nabla_\theta J(\theta) = 0$

- Design matrix: $X = \mathbb{R}^{m \times n} = \begin{bmatrix} \cdots & x^{(1)T} & \cdots \\ & \vdots & \\ \cdots & x^{(i)T} & \cdots \\ & \vdots & \\ \cdots & x^{(m)T} & \cdots \end{bmatrix}$, $\theta = \begin{bmatrix} \theta_0 \\ \vdots \\ \theta_n \end{bmatrix}$, $Y = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(m)} \end{bmatrix}$

$\therefore J(\theta) = \frac{1}{2} \left[(X\theta - Y)^T (X\theta - Y)\right] = \frac{1}{2} \left(\|X\theta - Y\|_2\right)^2$

① $\nabla_\theta \, a^T\theta = a$

$\nabla_\theta \, \theta^T a = a$ ($\theta^T a$: number; $a \in \mathbb{R}^n$)

② a) $\nabla_\theta \, \theta^T A \theta = 2A\theta$, $A \in \mathbb{R}^{n \times n}$ Symmetric } What if not symmetric?

b) $\nabla_\theta^2 (\theta^T A \theta) = 2A$ show.

(∇_θ^2: matrix of 2nd order derivatives, (Hessian))

$\nabla_\theta J(\theta) = \nabla_\theta \frac{1}{2}\left[(X\theta - Y)^T (X\theta - Y)\right]$ } Section 2.2 Notes 1

derive

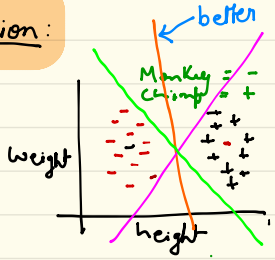
$\theta = \left[(X^T X)^{-1} X^T\right] Y$

pseudo inverse

Lecture 10 - 23.01.18

Logistic Regression:

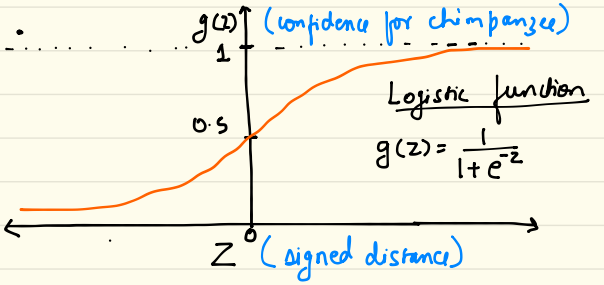
Binary classification:



Linear Classifier { Decision boundary is linear }

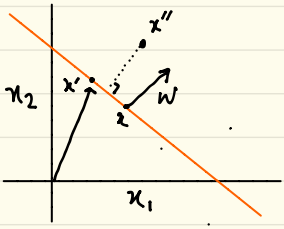
Metric for good line ?

↳ seperation should be large in classification as large seperation means more confidence



Equation of line: $\theta^T X = 0$

Suyash Agrawal



$(W, b) \equiv \theta \;\Big|\; w_2 x_2 + w_1 x_1 + b \equiv \theta_2 x_2 + \theta_1 x_1 + \theta_0 x_0 = 0$

$\therefore$ eqn: $W^T(x - x') = 0$

$= W^T x - \underbrace{W^T x'}_{b} = 0$

$\rightarrow W^T x + b = 0$

$W^T(x'' - x') \equiv$ un-normalized dist of x'' from line

$= W^T x'' + b$

$\therefore$ To calculate the dist of point from line $WX+b=0$, just put it in the equation — gives un-normalized signed distance.

- Process:

$$P(y=1 \mid x; \theta) = g(z) \;;\; z = \theta^T x$$

$$= \frac{1}{1 + e^{-(\theta^T x)}}$$

- Generative Process :

1) Compute $z = \theta^T x$
2) Compute $P(y=1 \mid x; \theta) = \frac{1}{1 + e^{-\theta^T x}}$ } Bernoulli (ϕ) $\therefore \phi = \frac{1}{1 + e^{-\theta^T x}}$
3) Sample $y \sim$ Bernoulli (ϕ)

↳ Generate random number in $[0, 1]$ & if $< \phi$ then 1 else 0

How to sample from gaussian dist ?

Suyash Agrawal

$$L(\theta) = \prod_{i=1}^{m} P(y^{(i)} | x^{(i)}; \theta)$$

$$\therefore \theta_{ML} = \arg\max_{\theta} (L(\theta)) = \prod_{i=1}^{m} P(y^{(i)} | x^{(i)}; \theta)$$

$$= \arg\max_{\theta} (\log(L(\theta))) = \sum_{i=1}^{m} \log\left(P(y^{(i)} | x^{(i)}; \theta)\right)$$

$$= \sum_{i=1}^{m} y^{(i)} \log(\phi) + (1-y^{(i)}) \log(1-\phi)$$

Logistic Regression (contd.):

Lecture 11 - 25.01.18

$$\frac{1}{1+e^{-z}} = g(z)$$

$$g'(z) = g(z)\cdot(1-g(z))$$

$$\arg\max_{\theta} (\log(L(\theta))) = \arg\max \sum_{i=1}^{m} \log\left(P(y^{i} | x^{i}; \theta)\right) = \arg\max \sum_{i=1}^{m} y^{(i)} \cdot \log\left(y^{(i)}=1 | x^{(i)}; \theta\right) + (1-y^{(i)}) \cdot \log\left(y^{(i)}=0 | x^{(i)}; \theta\right)$$

$L(\theta)$ is a concave function in θ.

$$= \arg\max \sum_{i=1}^{m} y^{(i)} \cdot \log\left(g(\theta^T x^{(i)})\right) + (1-y^{(i)}) \cdot \log\left(1-g(\theta^T x^{(i)})\right)$$

$$\therefore \frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{m} y^{(i)} \frac{g(\theta^T x^{(i)})\left(1-g(\theta^T x^{(i)})\right)}{g(\theta^T x^{(i)})} \cdot x_j^{(i)} + (1-y^{(i)}) \frac{-g(\theta^T x^{(i)})(1-g(\theta^T x^{(i)}))}{(1-g(\theta^T x^{(i)}))} \cdot x_j^{i}$$

$$= \sum_{i=1}^{m} \left[y^{(i)} \cdot (1-g(\theta^T x^{(i)})) - (1-y^{(i)}) \cdot g(\theta^T x^{(i)})\right] x_j^{(i)}$$

$$= \sum_{i=1}^{m} \left[y^{(i)} - g(\theta^T x^{(i)})\right] x_j^{(i)}$$

Here, we define $h_\theta(x) = P(y=1 | x; \theta) = g(\theta^T x)$

$$\therefore \frac{\partial LL(\theta)}{\partial \theta_j} = \sum_{i=1}^{m} \left[y^{(i)} - h_\theta(x^{(i)})\right] \cdot x_j^{(i)}$$

} similar to linear regression

In fact we see that in both case $h_\theta(x^{(i)})$ turns out to be the expected value of dist. of $y^{(i)}$

E.g. in linear reg: $\theta^T x^{(i)}$ ($E(x)$ for normal), in logistic: $g(\theta^T x^{(i)})$ → $E[x]$ for Bernoulli

Suyash Agrawal

$$\nabla_\theta LL(\theta) = \sum_{i=1}^{m} \left(y^{(i)} - h_\theta(x^{(i)})\right) \cdot x^{(i)}$$

- **Softmax Regression** : Multiclass classification

- **Generalized Linear Models (GLMs)**:

  This is a class of models → Linear Regression, Logistic Regression, ...

Training data: $\{y^{(i)}, x^{(i)}\}_{i=1}^{m}$

GLM :- Models : $P(y|x;\theta) = P(y;\eta)$ (η: natural parameter)

Conditions:

① $P(y;\eta) \in$ Exponential Family dist.

② $\eta = \theta^T x$ (η is linearly dependent on θ & x)

③ $h_\theta(x) = E[y;\eta]$

- **Exponential Family distributions**:

$$P(y;\eta) = b(y) \cdot \exp(\eta y - a(\eta)) = \frac{b(y) \cdot \exp(\eta y)}{\exp(a(\eta))}$$

- $y \sim N(\mu, 1) \Rightarrow P(y) = \frac{1}{\sqrt{2\pi}} \cdot e^{\frac{(y-\mu)^2}{2}} = \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{[y^2+\mu^2-2y\mu]}{2}}$

$\therefore \eta = \mu$

$b(y) = \frac{1}{\sqrt{2\pi}} \cdot e^{-y^2/2}$

$a(\eta) = \frac{\eta^2}{2}$

Normal dist also $\in$ Exp. family dist

- Similarly, we can show that Bernoulli dist also $\in$ Exp. family dist.
- Try with σ parameter in $N(\mu, \sigma^2)$

Suyash Agrawal

Numerical Algo: SVD

16/01/18

A: $m \times n$, $A: \mathbb{R}^n \to \mathbb{R}^m$

Null space $(A) = \{x : Ax = 0\}$, Range $(A) = \{Ax : x \in \mathbb{R}^n\}$

$\dim(\text{Null space}(A)) + \dim(\text{Range}(A)) = n$

$e_1, \ldots, e_k$ (basis) $\xrightarrow{\text{extend}}$ $e_{k+1}, \ldots, e_n$

$Ae_{k+1}, \ldots, Ae_n$ — basis of Range (A)

$\dim(\text{Range}(A)) = \text{Col. rank}(A)$

$$\begin{bmatrix} | & & | \\ A^1 & \cdots & A^n \\ | & & | \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = x_1 A^1 + \cdots + x_n A^n$$

2 norm of diag matrix:

$$A = \begin{bmatrix} d_1 & & 0 \\ & d_2 & \\ 0 & & \ddots \\ & & & d_n \end{bmatrix} \quad \|A\|_2 = \max_{\|x\|_2 = 1} \|Ax\|_2 = (d_1^2 x_1^2 + d_2^2 x_2^2 + \cdots d_n^2 x_n^2)^{1/2} = \max\{d_1, \ldots, d_n\}$$

① Suppose U is an invertible $n \times n$ matrix. Then, rank (AU) = rank (A)

range (A) = range (AU)

- range (A): $y = Ax = AU \cdot (U^{-1}x) \Rightarrow y \in$ range (AU)
- range (AU): $y = AUx = A(Ux)$, $y \in$ range (A)

② V: $m \times m$ inv. matrix : rank (VA) = rank (A)

null space (A) = null space (VA)

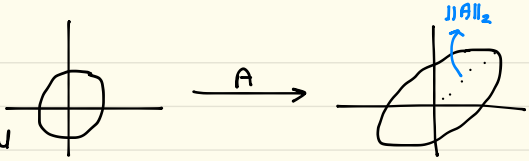
- null space (A): $Ax = 0$, $VAx = 0$, $x \in$ nullspace (VA)
- null space (VA): $VAx = 0$, $Ax = 0$ (multiply by V^{-1}), $\therefore x \in$ nullspace (A).

Suyash Agrawal

③ Suppose U is unitary $(U^TU = I)$

$\|AU\|_2 = \|A\|_2$

multiplying unitary = changing orthonormal basis.



$$\|AU\|_2 = \max_{\|x\|_2=1} \|AUx\|_2$$

$|AU|_1 = |A|$? ✗

$\|AU\|_F = \|A\|_F$ ✓ (Frobenius norm)

Let $y = Ux$, $\|y\|_2 = 1$

$$= \max_{\|y\|=1} \|Ay\|_2$$

$$= \|A\|_2$$

④ V: $m \times m$ unitary $\|VA\|_2 = \|A\|_2$ (as in rotation of ellipse the length of points remain same)

$\|Vy\|_2 = \|y\|_2$

$$\therefore \|VA\|_2 = \max_{\|x\|_2=1} \|VAx\|_2 = \max \|Ax\|_2 = \|A\|_2$$

Also, $\|VA\|_F = \|A\|_F$

$\|A\|_F^2 = \|A^1\|_2^2 + \|A^2\|_2^2 + \cdots + \|A^n\|_2^2$

$\|VA\|_F^2 = \|VA^1\|_2^2 + \cdots\cdots = \|A\|_F^2$

$\|A^1\|_2^2$

- **SVD Theorem:** A: $m \times n$ matrix

Then $\exists$ unitary matrices U: $m \times m$, V: $n \times n$

s.t. $A = U \Sigma V^T$

Σ: diag matrix : $\begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_{\min(m,n)} \end{bmatrix}$

$\sigma_1 \geq \sigma_2 \cdots \geq \sigma_{\min(m,n)} \geq 0$ (singular values) and are uniquely determined by A

$U = [U^1\ U^2 \cdots U^m]$ (right singular vectors)

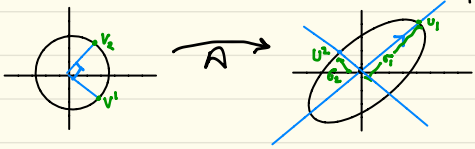
$V = [V^1\ V^2 \cdots V^n]$ (left singular vectors)

Suyash Agrawal

$\Rightarrow AV = U\Sigma$, Now $\|A\|_2 = \sigma_1$ { as V & U are unitary }

$\therefore AV^1 = \sigma_1 U^1$

$AV^2 = \sigma_2 U^2$



$\|x\|_2 = 1$, $X = \lambda_1 V^1 + \lambda_2 V^2$, $\lambda_1^2 + \lambda_2^2 = 1$

$AX = \lambda_1 \sigma_1 U^1 + \lambda_2 \sigma_2 U^2$

$\|AX\|_2 = (\lambda_1^2 \sigma_1^2 + \lambda_2^2 \sigma_2^2)^{1/2} \in [\sigma_2, \sigma_1]$

- If A is singular then $\sigma_2 = 0$ & ellipse will be a straight line.

- Rank (A) = rank (Σ) = number of non zero singular values

Proof: Define $\sigma_1 = \|A\|_2$

Let V^1 be the unit vector s.t. $\|AV^1\|_2 = \sigma_1$

Let U^1 be the unit vector s.t. $AV^1 = \sigma_1 U^1$

Let L: (set of vector $\perp$ to V^1) = $\{x : \langle x, V^1 \rangle = 0\}$

M: $\{x \mid \langle x, U^1 \rangle = 0\}$

Key claim: If $x \in L$, $Ax \in M$

Suppose the claim is false:

$\exists\, x \in L$, s.t.

$Ax = \alpha U^1 + \beta u$, $U \in M$, $\alpha \neq 0$, x is unit vector

Consider the vector $x' = x + \epsilon V^1$

We will show that $\dfrac{\|Ax'\|}{\|x'\|} > \sigma_1$

$Ax' = \alpha U^1 + \beta u + \epsilon \sigma_1 U^1$

$= (\alpha + \epsilon \sigma_1) U^1 + \beta U$

$\dfrac{\|Ax'\|^2}{\|x'\|^2} = \dfrac{(\alpha + \epsilon\sigma_1)^2 + \beta^2}{1 + \epsilon^2} > \dfrac{\sigma_1^2 \epsilon^2 + 2\alpha\epsilon\sigma_1}{1 + \epsilon^2}$

$= \sigma_1^2$

if we take $\epsilon = \dfrac{\sigma_1}{2\alpha}$

ML

30/01/18

GLM:

① $P(y;\eta) \in$ Exponential Family ($\eta = fn(\theta, x)$)

② $\eta = \theta^T x$ } linearity

③ $h_\theta(x) = E[y|x;\eta]$

- Exponential dist:

$$P(y;\eta) = b(y) \cdot \exp(\eta y - a(\eta))$$

① ↳ $y \sim N(\mu, 1)$ then $P(y) \in$ Exponential family.

$\eta = \mu$

② ↳ $y \sim$ Bernoulli (ϕ), then $P(y) \in$ Exp. family

H.W.↳ $\phi = \frac{1}{1+e^{-\eta}}$, $\eta = \theta^T x$

→ all of them is $y|x$

· Gradient update rule for GLMs :-

$$LL(\theta) = \sum_{i=1}^{m} \log P(y^i | x^i; \theta)$$

$$= \sum_{i=1}^{m} \log \left(P(y^i ; \eta^{(i)}) \right)$$

$$= \sum_{i=1}^{m} \log \left(b(y^{(i)}) \cdot \left[\exp \left(\eta^{(i)} y^{(i)} - a(\eta^{(i)}) \right) \right] \right)$$

$$= \sum_{i=1}^{m} \log \left(b(y^{(i)}) \right) + \sum_{i=1}^{m} \left[\eta^{(i)} \cdot y^{(i)} - a(\eta^{(i)}) \right]$$

$$\frac{\partial LL(\theta)}{\partial \theta_j} = 0 + \sum_{i=1}^{m} \left(y^{(i)} \cdot x_j^{(i)} - a'(\eta^{(i)}) \cdot x_j^{(i)} \right)$$

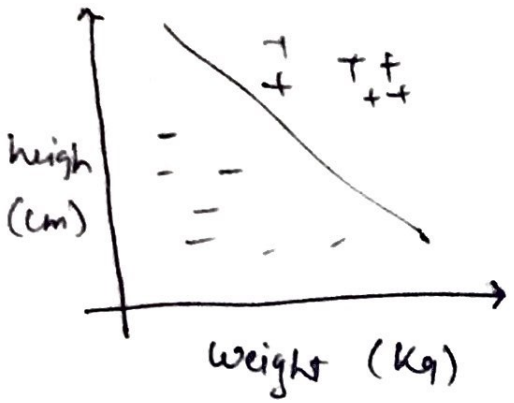
$$= \sum_{i=1}^{m} \left[y^{(i)} - a'(\eta^{(i)}) \right] \cdot x_j^{(i)}$$

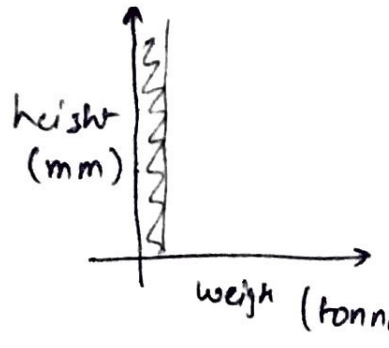
linear & logistic

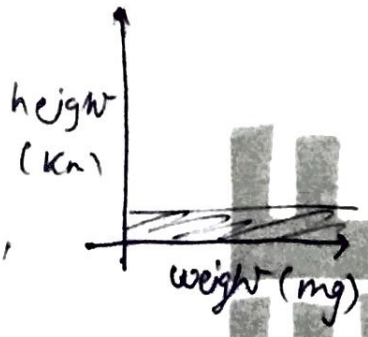
for $a'(\eta^{(i)})$ becomes $h(\theta^T x^{(i)})$

· Concavity ?

• Normalization: $\{x^{(i)}, y^{(i)}\}_{i=1}^{m}$







Make each attribute 0 mean & unit variance!

for $j = 1$ to n {

$$\mu_j = \frac{1}{m} \sum_{i=1}^{m} x_j^i$$

$$\sigma_j^2 = \frac{1}{m} \sum_{i=1}^{m} (x_j^i - \mu_j)^2$$

m or m-1? [Doesn't matter]

}

for $j = 1$ to n {

$$\forall i: \quad x_j^i = \left[\frac{x_j^i - \mu_j}{\sigma_j} \right]$$

}

$\text{Var}(X) = \sigma^2$

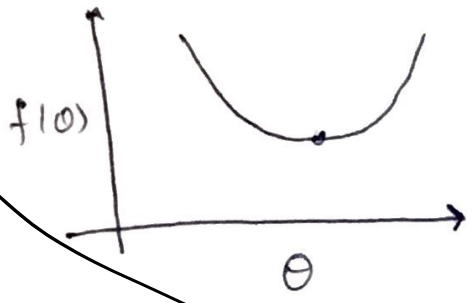
$\text{Var}\left(\frac{X}{k}\right) = \frac{\sigma^2}{k^2}$

★ We almost always should normalize the data we are working on [even in closed form sol^n].

• Should we also normalize y? [Not really needed].

- When not normalize? → When data comes from same metric, we don't do normalization. E.g. Naive Bayes classification we don't normalize data.

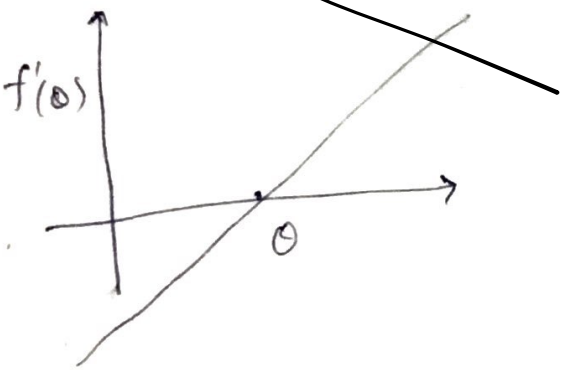
* Newton's Method.



$f(\theta) = (\theta-3)^2 + 1$

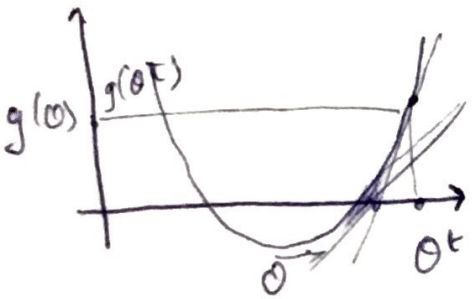
$f'(\theta) = 2(\theta-3)$

① Analytic
② Gradient Descent
③ Newton's meth.



To optimize we find zero of $f'(\theta)$

· Finding zero of a function:



- iterative process

$$\frac{0 - g(\theta^t)}{\theta^{t+1} - \theta^t} = g'(\theta^t)$$

$$\Rightarrow \boxed{\theta^{t+1} = \theta^t - \frac{g(\theta^t)}{g'(\theta^t)}}$$

Newton's update

- BFGS
- LBFGS

for Ottimizzatore:

$$\theta^{(t+1)} = \theta^{(t)} - \frac{f'(\theta^t)}{f''(\theta^t)} \quad \Big] \theta \in \mathbb{R}$$

$\theta \in \mathbb{R}^n$:

$$\theta^{(t+1)} = \theta^{(t)} - H^{-1} \nabla_\theta f(\theta^{(t)})$$

↳ Concrete Proof ?.. (See piazza)

- Hessian ~~non~~ singular ?
-

Suyash Agrawal

Lecture 13 - 01.02.18

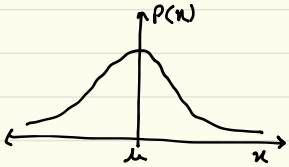
Last class:
- GLM
- Normalization
- Newton's Method

- **GDA: Gaussian Discriminant analysis** {Classification Algorithm}

- **Normal Distribution:**

$$X \in \mathbb{R}, \quad X \sim N(\mu, \sigma^2)$$

$$P(X=x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$



- For $X \in \mathbb{R}^n$:

$$X = (X_1, \cdots, X_n)$$

$$X \sim N(\mu, \Sigma)$$

(μ: n sized vector $\in \mathbb{R}^n$; Σ: covariance matrix (n×n))

$$P(X=x; \mu, \Sigma) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \cdot e^{-\frac{1}{2}((x-\mu)^T \Sigma^{-1} (x-\mu))}$$

- If X is a R.V. ($X \in \mathbb{R}$): $E[X] = \int P(X=x) \cdot x \, dx$

$$Var[X] = E\left[(X - E(X))^2\right] = E(X^2) - E(X)^2$$

- X_1, X_2 is R.V. $\Rightarrow$ $Cov(X_1, X_2) = E\left[(X_1 - E[X_1])(X_2 - E[X_2])\right]$

$$= E[X_1 \cdot X_2] - E[X_1] \cdot E[X_2]$$

- Independent $\Rightarrow$ 0 Covariance
  $\not\Leftarrow$

- If $X = (X_1, \cdots, X_n)$, then Σ = Covariance matrix

$$= \begin{bmatrix} & & i & \\ & & \vdots & \\ j & \cdots & \Sigma_{ij} & \end{bmatrix}$$

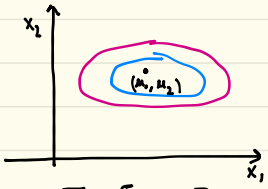
$$\Sigma_{ij} = Cov(X_i, X_j)$$

- **Properties of Σ:**
  - Symmetric
  - Positive semi definite {Prove}
  - $\Sigma_{ii} = Var(X_i)$
  - If Σ is diagonal, then all features are un-correlated.

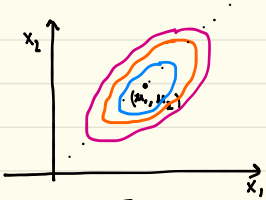
Suyash Agrawal

Eg: $X = (x_1, x_2)$, $\mu = (\mu_1, \mu_2)$

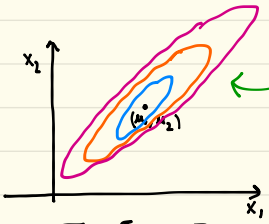
$$\Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$



$$\Sigma = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}$$



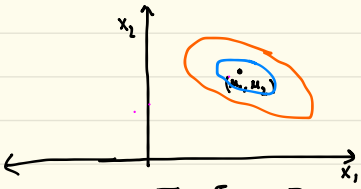
$$\Sigma = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$



$$\Sigma = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

← More skewed than in case of 0.5

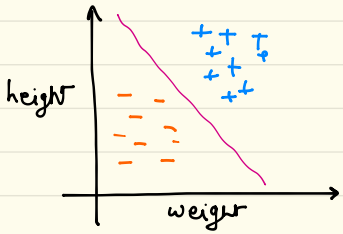
$$\Sigma = \begin{bmatrix} 3 & 0.5 \\ 0.5 & 1 \end{bmatrix}$$



$$\Sigma = \begin{bmatrix} 3 & -0.5 \\ -0.5 & 1 \end{bmatrix}$$

See the gda pdf [CS229N] for better visualization.

• Gaussian Discriminant Analysis:

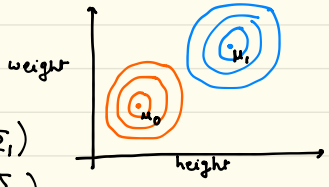


$\{x^{(i)}, y^{(i)}\}_{i=1}^{m}$

Logistic:
Given x_i we model $P(y^i | x^i; \theta)$
$y^{(i)} \sim$ Bernoulli (ϕ)
$P(y=1 | x; \theta) = \frac{1}{1 + e^{-\theta^T x}}$

- In GDA we try to generate the entire data {constrast to Logistic where we generated y^i given x_i}
- This is an example of Generative Model.
- $P(x, y; \theta)$
  - a) $y \sim$ Bernoulli (ϕ)
  - b) Model $P(x|y=1)$ & $P(x|y=0)$

    Here we make the assumption: $P(x|y=1) \sim N(\mu_1, \Sigma_1)$
    $P(x|y=0) \sim N(\mu_0, \Sigma_0)$



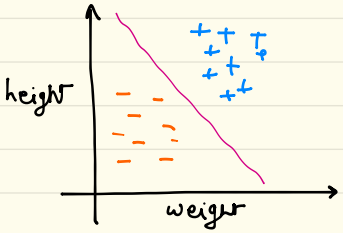
- $LL(\theta) = \sum_{i=1}^{m} \log\left(P(x^{(i)}, y^{(i)}; \theta)\right) = \sum_{i=1}^{m} \log\left(P(y^{(i)}; \theta) \cdot P(x^{(i)} | y^{(i)}; \theta)\right)$
- We try to find $\arg\max_{\theta} (LL(\theta))$

Suyash Agrawal

Lecture 14 - 02.02.18

MINOR: 2:25pm - 3:35pm, 1:10 hour, Closed Notes

· Gaussian Discriminant Analysis:



$\{x^{(i)}, y^{(i)}\}_{i=1}^{m}$

We want to model $P(y, x; \theta)$.

$$\cdot LL(\theta) = \sum_{i=1}^{m} \log\left(P(x^{(i)}, y^{(i)}; \theta)\right) = \sum_{i=1}^{m} \log\left(P(y^{(i)}; \theta) \cdot P\left(x^{(i)} \mid y^{(i)}; \theta\right)\right)$$

$$\theta = (\phi, \mu_0, \Sigma_0, \mu_1, \Sigma_1)$$

$$= \sum_{i=1}^{m} \log\left(P(y^i; \phi)\right) + \log\left(P(x^i \mid y^i; \theta)\right)$$

$$= \sum_{i=1}^{m} y^i \cdot \left[\log\left(P(y^i=1; \phi)\right) + \log\left(P(x^i \mid y^i=1; \mu_1, \Sigma_1)\right)\right] + (1-y^i) \cdot \left[\log\left(P(y^i=0; \phi)\right) + \log\left(P(x^i \mid y^i=0; \mu_0, \Sigma_0)\right)\right]$$

To maximize LL(θ): $\nabla_\phi LL(\theta) = 0$, $\nabla_{\mu_1} LL(\theta) = 0$

$\nabla_{\mu_1} LL(\theta) = 0$, $\nabla_{\Sigma_0} LL(\theta) = 0$, $\nabla_{\Sigma_1} LL(\theta) = 0$

Optimal parameters:

$$\phi = \frac{\sum_{i=1}^{m} 1\{y^i = 1\}}{m}$$

($1\{\langle bool \rangle\} = 1$ if bool is T, 0 otherwise)

$$\mu_1 = \frac{\sum_{i=1}^{m} 1\{y^i = 1\} \cdot x^{(i)}}{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}}, \quad \mu_0 = \frac{\sum_{i=1}^{m} 1\{y^i = 0\} \cdot x^i}{\sum_{i=1}^{m} 1\{y^i = 0\}}$$

$$\Sigma_1 = \frac{\sum_{i=1}^{m} 1\{y^i = 1\} (x^i - \mu_1)(x^i - \mu_1)^T}{\sum_{i=1}^{m} 1\{y^i = 1\}}$$

$$\Sigma_{1\,j,k} = \frac{\sum_{i=1}^{m} 1\{y^i = 1\} (x_j^i - \mu_j)(x_k^i - \mu_k)}{\sum_{i=1}^{m} 1\{y^i = 1\}}$$

Suyash Agrawal

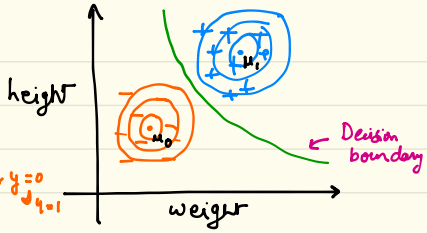
How to do classification using this?

↳ We want to calculate $P(y|x;\theta)$

$$= \frac{P(x|y;\theta)\cdot P(y;\theta)}{P(x;\theta)}$$

$P(x;\theta)$ → same for $y=0$ & $y=1$

↳ $P(x|y=1;\theta)\cdot P(y=1) + P(x|y=0;\theta)\cdot P(y=0)$



• Decision Boundary:

$$1 = \frac{P(y=1|x;\theta)}{P(y=0|x;\theta)} = \frac{P(x|y=1)\cdot P(y=1)}{P(x|y=0)\cdot P(y=0)} = \frac{\frac{\phi}{(2\pi)^{N/2}|\Sigma_1|^{1/2}}\cdot e^{-\frac{(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1)}{2}}}{\frac{(1-\phi)}{(2\pi)^{N/2}|\Sigma_0|^{1/2}}\cdot e^{-\frac{(x-\mu_0)^T\Sigma_0^{-1}(x-\mu_0)}{2}}}$$

Take denominator to other side & take log.

$$\log(1-\phi) - \log(|\Sigma_0|^{1/2}) - \frac{(x-\mu_0)^T\Sigma_0^{-1}(x-\mu_0)}{2} = \log(\phi) - \log(|\Sigma_1|^{1/2}) - \frac{(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1)}{2}$$

$$\frac{1}{2}\left[(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1) - (x-\mu_0)^T\Sigma_0^{-1}(x-\mu_0)\right] = \log\left(\frac{\phi}{1-\phi}\right) - \log\left(\frac{|\Sigma_1|^{1/2}}{|\Sigma_0|^{1/2}}\right)$$

Quadratic in x — constant (indep. of x)

- Quadratic Expression: $x^T\Sigma_1^{-1}x - x^T\Sigma_0^{-1}x$

∴ If $\Sigma_1 = \Sigma_0$ then quad. term = 0 & thus boundary is linear.

H.W. Compute $P(y=1|x;\theta)$ with Σ_0 & Σ_1 as same

↳ Turns out to be $\frac{1}{1+e^{-\theta^Tx}}$

Suyash Agrawal

Lecture 15 - 09.02.18

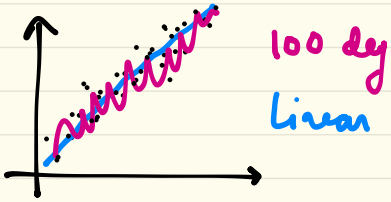
- Minor 1 Copy distribution : 1:00 - 2:30, SIT006, Grp 1,2 : 1-1:30 / 3,4 : 1:30-2 / 5,6 : 2-2:30, 09.02.18

- Question Discussion:

Q.3] Checking convergence of S.G.D.

- Take difference in cost of consecutive iterations — Too jittery
- Take difference of moving average over last r examples? — Smoother, but may still be jittery
- Have a band of ϵ when if the last β batches have error b/w this then we say that we have converged.

Q4) b) • If we have less data then if classes are really distributed as gaussian then GDA would give us much better generalization. E.g. linearreg vs 100 deg poly fit in a linearly spread data

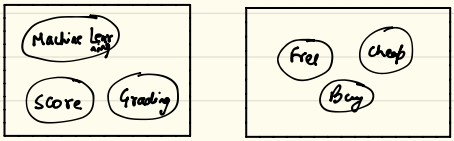


Suyash Agrawal

MINOR-2

• Naive Bayes:

E.g. Spam Classification:



i^{th} Example: $\langle x_1^{(i)}, x_2^{(i)}, \ldots, x_{|V|}^{(i)} \rangle$, $y^{(i)}$

$x_j^{(i)} \in \{0,1\}$ ⇒ j^{th} word present in i^{th} doc or not

$y^{(i)} \in \{0,1\}$ ⇒ i^{th} document is spam or not

Training Examples: $\{x^{(i)}, y^{(i)}\}_{i=1}^{m}$

$$P(y^{(i)} | x^{(i)}; \theta) = \frac{P(x^{(i)}, y^{(i)}; \theta)}{P(x^{(i)}; \theta)} = \frac{P(x^{(i)} | y^{(i)}; \theta) \cdot P(y^{(i)}; \theta)}{P(x^{(i)}; \theta)}$$

$y \sim \text{Bernoulli}(\phi)$

$x|y$, since $2^{|V|}$ is humongous, we make a very strong assumption: $x_j \perp x_k \mid y$

↳ Naive Bayes Assumption

$$\therefore P(x^{(i)} | y^{(i)}; \theta) = \prod_{j=1}^{|V|} P(x_j^{(i)} | y^{(i)}; \theta)$$ } due to assumption

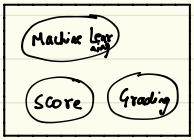
Suyash Agrawal

Lecture 15 - 13.02.18

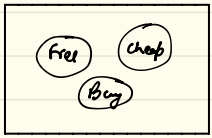
Last class:

- Minor 1 Discussion
- Naive Bayes Start

Naive Bayes:



$y=0$



$y=1$

i^{th} Example: $\langle x_1^{(i)}, x_2^{(i)}, \cdots, x_{|V|}^{(i)} \rangle$, $y^{(i)}$

$x_j^{(i)} \in \{0,1\} \Rightarrow j^{th}$ word present in i^{th} doc or not

$y^{(i)} \in \{0,1\} \Rightarrow i^{th}$ document is spam or not

Bernoulli Event Model.

We are making a generative model.

$y \sim$ Bernoulli (ϕ)

$x|(y=0) = (x_1, \cdots, x_{|V|}) | (y=0)$ Assumption: $x_1, \cdots, x_{|V|}$ are independent $|y$ → important.

$\therefore P(x|y) = P(x_1, \cdots, x_m | y) = \prod_{i=1}^{|V|} P(x_i|y)$ using assumption

Now, $\forall_j \quad x_j | y=1 \sim$ Bernoulli $(\theta_{j|y=1})$

$\forall_j \quad x_j | y=0 \sim$ Bernoulli $(\theta_{j|y=0})$

params $= 2|V|+1$

Note: Here we are giving up on the order of occurrence of words & frequency of occurring.

Log Likelihood:

Parameters: $\theta = \phi, \{\theta_{j|y=0}\}_{j=1}^{n}, \{\theta_{j|y=1}\}_{j=1}^{n}$

$$\begin{aligned} LL(\theta) &= \sum_{i=1}^{m} \log P(y^{(i)}, x^{(i)}; \theta) \\ &= \sum_{i=1}^{m} \log (P(y^{(i)}; \theta)) + \sum_{i=1}^{m} \log (P(x^{(i)}|y^{(i)}; \theta)) \\ &= \sum_{i=1}^{m} y^{(i)} \log \phi + (1-y^{(i)}) \log (1-\phi) \\ &\quad + \sum_{i=1}^{m} y^{(i)} \log \Big(\sum_{j=1}^{n} \underbrace{P(x_j^{(i)} | y^{(i)}=1; \theta_{j|y=1})}_{x_j^{(i)} \cdot \theta_{j|y=1} + (1-x_j^{(i)}) \cdot (1-\theta_{j|y=1})} \Big) + (1-y^{(i)}) \log \Big(\sum_{i=1}^{n} P(x_j^{(i)} | y^{(i)}=0; \theta_{j|y=0}) \Big) \end{aligned}$$

$\underset{\theta}{\text{argmax}}\ LL(\theta) \rightarrow \nabla_{\phi} LL(\theta) = 0, \ \forall_j \nabla_{\theta_{j|y=1}} LL(\theta) = 0, \ \forall_j \nabla_{\theta_{j|y=0}} LL(\theta) = 0$

After solving: $\phi = \frac{\sum_{i=1}^{m} 1\{y^{(i)}=1\}}{m}$, $\theta_{j|y=1} = \frac{\sum_{i=1}^{m} 1\{y^{(i)}=1 \wedge x_j^{(i)}=1\}}{\sum_{i=1}^{m} 1\{y^{(i)}=1\}}$, $\theta_{j|y=0} = \frac{\sum_{i=1}^{m} 1\{y^{(i)}=0 \wedge x_j^{(i)}=1\}}{\sum_{i=1}^{m} 1\{y^{(i)}=0\}}$

Prediction time:

x: document

$$\arg\max_y P(y|x) = \arg\max_y \frac{P(x|y)\cdot P(y)}{P(x)} = \arg\max_y P(y)\cdot P(x|y) = \arg\max_y \left[P(y)\cdot \prod_{j=1}^{n} P(x_j|y)\right]$$

$P(x)$: same $\forall y$

Warning: Never multiply. Always take log. to avoid underflow.

Note: For computing probabilities Naive Bayes may not be good, but for classification it works reasonably.

Note: Let us suppose that we have an exotic word $\overset{w_j}{\text{archi tecture}}$ (say "archi tecture") & it never occurred in our training data. $\therefore P(w_j|y=0) = 0$

Now if a new document contains this word w_j then $P(d|y=0)$ will turn out to be 0!

To fix this: {Laplace Smoothing}

$$\theta_{j|y=1} = \frac{\sum_{i=1}^{m} 1\{y^{(i)}=1 \wedge x_j^{(i)}=1\} + c}{\sum_{i=1}^{m} 1\{y^{(i)}=1\} + 2c}, \quad \theta_{j|y=0} = \frac{\sum_{i=1}^{m} 1\{y^{(i)}=0 \wedge x_j^{(i)}=1\} + c}{\sum_{i=1}^{m} 1\{y^{(i)}=0\} + 2c}$$

Intuition: Kind of adding two documents one will all the words & other with no words

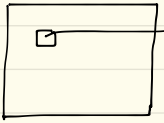
c is large $\Rightarrow$

c is small $\Rightarrow$

Suyash Agrawal

Lecture 16 - 15.02.18

- In Bernoulli event model we ignored the count of the words.
- Alternative:



Features: x_j : For position j in the document which word occurs $\in \{1, 2, \cdots, |V|\}$

$y \sim \text{Bernoulli}(\phi)$

$x_j|y \sim \text{Multinomial}\left(\{\theta^k_{j|y}\}_{k=1}^{|V|}\right)$, # param = $2(100 * (|V|-1)) + 1$

Since these are very large, we assume that all positions are independent

$\therefore$ Assume:

$$\{\theta^k_{j|y}\} = \{\phi^k_{j'|y}\} \ \forall j, j' \quad \Big| \quad \sum \phi^k_{j|y} = 1, \ \#\text{param}: 2(|V|-1)+1$$

Write expression for LL θ & solve.

Optimal Params:

$$\phi = \frac{\sum_{i=1}^{m} 1\{y^i = 1\}}{m}$$

Bag of Words Model

$$\theta^k_{|y=1} = \frac{\sum_{i=1}^{m} 1\{y^i = 1\}\, n^{(i)}_k}{\sum_{i=1}^{m} 1\{y^i = 1\}\, n^{(i)}}, \quad \theta^k_{|y=0} = \frac{\sum_{i=1}^{m} 1\{y^i = 0\}\, n^{(i)}_k}{\sum_{i=1}^{m} 1\{y^i = 0\}\, n^{(i)}}$$

$n^{(i)}_k$: # of times k^{th} word occurs in i^{th} document

We just need two arrays in order to implement this (each of $|V|$ size)

Laplace Smoothing:

$$\theta^k_{|y=0} = \frac{\sum_{i=1}^{m} 1\{y^i = 0\}\, n^{(i)}_k + c}{\sum_{i=1}^{m} 1\{y^i = 0\}\, n^{(i)} + c|V|}, \quad \theta^k_{|y=1} = \frac{\sum_{i=1}^{m} 1\{y^i = 1\}\, n^{(i)}_k + c}{\sum_{i=1}^{m} 1\{y^i = 1\}\, n^{(i)} + c|V|}$$

$+c|V|$: adding c document with all words in both classes.

① Linear Regression : $y \in \mathbb{R}$, $x \in \mathbb{R}^n$

② Classification:

	Logistic Regression	GDA	Naive Bayes
1)	$y \in \{0,1\}$ if y is discrete : Softmax	$y \in \{0,1\}$ $y \in \{0, \cdots, r\}$	$y \in \{0,1\}$ $y \in \{0, \cdots, r\}$
2)	$x \in \mathbb{R}^n$	$x \in \mathbb{R}^n$ GDA subsumes Naive bayes if $x \in \mathbb{R}^n$ in naive bayes & we use normal this is called gaussian naive bayes.	$\forall x_j \in \{0, \cdots, |V|\}$ $x_j \in \mathbb{R} \mid P(x_j|y)$ as some dist. [eg. normal]

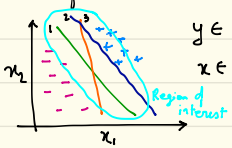
Logistic cannot give us $P(x)$!

Suyash Agrawal

Lecture 17 - 19.02.18

Support Vector Machines:

Classification:



$y \in \{-1, 1\}$

$x \in \mathbb{R}^n$

① is better than ② & ③

Methods seen till now:

- Logistic Regression {Discriminative}, $P(y=1|x) = \frac{1}{1+e^{-\theta^T x}}$
- GDA {Generative}: $P(y,x) = P(y) \cdot P(x|y)$ — $P(y)$: Bernoulli; $P(x|y)$: $N(\mu_0, \Sigma_0)$, $N(\mu_1, \Sigma_1)$; Quadratic if $\Sigma_0 \neq \Sigma_1$
- Naive Bayes {Generative}: $P(x,y) = P(y) \cdot P(x|y) = P(y) \cdot \prod P(x_i|y)$

Similarity: All of these are probabilistic models {define dist of data}. That's why we try to maximize log likelihood $LL(\theta)$.

- SVMs are more discriminative than Linear Models
- The hyperplane of interest as: $w^T x + b = 0$, $w \in \mathbb{R}^n$, $x \in \mathbb{R}^n$ instead of $\theta^T x = 0$, $\theta \in \mathbb{R}^{n+1}$, $x \in \mathbb{R}^{n+1}$
- Signed Distance of $x^{(i)}$: $\frac{(w^T x^{(i)} + b) \cdot y^{(i)}}{\|w\|} = \hat{\gamma}^{(i)}$, $y^{(i)} \in \{1, -1\}$ — Functional margin; geometric margin
- We currently are assuming that data is linearly separable with no outliers, but will relax this constraint later
- We try to: $\max_{w,b} \left(\min_{(i)} \gamma^{(i)} \right)$
- Now: $\max_{w,b} \left(\min_i \gamma^{(i)} \right) \equiv \max_{w,b} \gamma$, $\gamma \leq \gamma^{(i)} \ \forall i$

$$\gamma \leq y^{(i)} \cdot \frac{(w^T x^{(i)} + b)}{\|w\|}$$

$$\therefore \max_{w,b} \frac{\hat{\gamma}}{\|w\|}, \quad \hat{\gamma} \leq y^{(i)} (w^T x^{(i)} + b) \qquad \{\text{defined } \hat{\gamma} = \|w\| \cdot \gamma\}$$

If $\hat{\gamma}^*, w^*, b^*$ is a sol^n then $k\hat{\gamma}^*, kw^*, kb^*$ is also a sol^n.

Thus we take $\hat{\gamma}^* = 1$ {WLOG}

Hence, our optimization becomes:

$$\max_{w,b} \frac{1}{\|w\|}, \quad 1 \leq y^{(i)} \left(w^T x^{(i)} + b \right)$$

Changing max to min:

$$\min_{w,b} \|w\|, \quad 1 \leq y^{(i)} \left(w^T x^{(i)} + b \right)$$

$$\min_{w,b} \|w\| \Leftrightarrow \min_{w,b} \|w\|^2 = \min_{w,b} w^T w$$

∴ Final optimization:

$$\min_{w,b} \left[\frac{1}{2} w^T w \right], \quad y^{(i)} (w^T x^{(i)} + b) \geq 1$$

Convex Optimization Problem

Suyash Agrawal

Stephen Boyd {Book}

Convex Optimization Problems : (Constrained)

General form:

$$\min_{\omega} f(\omega) \quad \text{subject to:} \quad g_k(\omega) \le 0, \quad l_k(\omega) = 0$$

g_k: convex; l_k: affine $\{\omega^T x + b\}$

$k \in \{1, \cdots, r\}$

$l \in \{1, \cdots, s\}$

$\omega \in \mathbb{R}^n$

If form is this then feasible set is convex set.

C is convex if $\omega_1, \omega_2 \in C$ then $\alpha\omega_1 + (1-\alpha)\omega_2 \in C$

If f is convex then this is convex optimization.

Suyash Agrawal

Lecture 18 - 20.02.18

SVM:

SVM Objective: $\min_{w,b} \left[\frac{1}{2} w^T w\right], \quad y^{(i)}(w^T x^{(i)} + b) \geq 1$

Is SVM optimization a convex constrained optimization Problem? Yes

↳ $f(w) = \frac{1}{2} w^T w$ is convex

We have no equality constraints

Also, $y^{(i)}(w^T x^{(i)} + b)$ is linear & hence convex

} ∴ This is a convex optimization problem.

- We will go from primal space to dual space. {where it is easier to solve}
- Think on why can't we use SGD to optimize this?

- **Theory of Langrangians:**

$$\underbrace{L(w,\alpha,\beta)}_{\text{Langrangian}} = f(w) + \sum_{k=1}^{r} \alpha_k g_k(w) + \sum_{l=1}^{s} \beta_l h_l(w)$$

α, β = Langrangian multipliers

Optimization Problem: $\min_w f(w)$

$\forall k \quad g_k(w) \leq 0$

$\forall e \quad h_e(w) = 0$

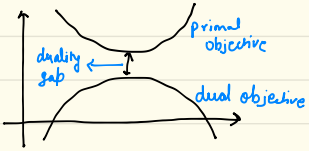
Primal objective: $\max_{\substack{\alpha,\beta \\ \alpha \geq 0}} L(w,\alpha,\beta) = \theta_p(w)$

form of $\theta_p(w) = \begin{cases} f(w) & \text{if } w \text{ is feasible} \quad \{h_i = 0 \ \& \ g_k \leq 0 \ \therefore \alpha_k = 0\} \\ \infty & \text{otherwise} \quad \{\text{make the corresponding } \alpha_k \text{ or } \beta_l \text{ large}\} \end{cases}$

Now, $\min_w \theta_p(w)$ is exactly same as our original optimization

$\min_w \underbrace{\theta_p(w)}_{\text{primal objective}} = \min_w \left[\max_{\alpha,\beta} (L(w,\alpha,\beta))\right]$

$\max_{\substack{\alpha,\beta \\ \alpha \geq 0}} \underbrace{\theta_D(\alpha,\beta)}_{\text{dual objective}} = \underbrace{\max_{\alpha,\beta} \left[\min_w L(w,\alpha,\beta)\right]}_{\text{dual problem}}$



$\theta_p(w) \geq \theta_D(\alpha,\beta)$

$\therefore \min_w \theta_p(w) \geq \max_{\alpha,w} \theta_D(\alpha,\beta)$

Suyash Agrawal

- **Theorem**: $\max_{\gamma} \min_{\omega} f(\omega, \gamma) \leq \min_{\omega} \max_{\gamma} f(\omega, \gamma)$

↳ Let $g(\gamma) = \min_{\omega} f(\omega, \gamma)$

and γ^* be s.t. $\gamma^* = \arg\max_{\gamma} g(\gamma)$

$\therefore g(\gamma^*) = \min_{\omega} f(\omega, \gamma^*)$

Now, $f(\omega, \gamma^*) \leq \max_{\gamma} f(\omega, \gamma) \quad \forall \omega$

$\min_{\omega} f(\omega, \gamma^*) \leq \min_{\omega} \max_{\gamma} f(\omega, \gamma)$

by def of $g(\gamma)$ ↳ $\max_{\gamma} \min_{\omega} f(\omega, \gamma) \leq \min_{\omega} \max_{\gamma} f(\omega, \gamma)$

Hence Proved

- Thus, $\max_{\substack{\alpha, \beta \\ \alpha \geq 0}} \theta_D(\alpha, \beta) = \max_{\substack{\alpha, \beta \\ \alpha \geq 0}} \left[\min_{\omega} L(\omega, \alpha, \beta) \right] \leq \min_{\omega} \left[\max_{\substack{\alpha, \beta \\ \alpha \geq 0}} L(\alpha, \omega, \beta) \right] = \min_{\omega} \theta_P(\omega)$

$\therefore d^* \leq p^*$

We will prove that in convex case: $d^* = p^*$

Suyash Agrawal

Lecture 19 - 22.02.18

· Slater's conditions:

① @ $f(\omega)$ is convex

ⓑ $g_k(\omega)$'s are convex $\forall k$

ⓒ $h_e(\omega)$'s are affine $\forall e$

② $\exists\, \omega$, $g_k(\omega)$'s are strictly feasible $\forall K$

$\exists \omega$, s.t. $g_k(\omega) < 0 \quad \forall k$

ⓓ $h_l(\omega) = 0 \quad \forall l$

·)

1) $\frac{1}{2}\omega^T\omega \rightarrow$ convex

$y^{(i)}(\omega^T x^{(i)} + b) \geq 1 \quad \forall i \rightarrow$ convex

2) Strict feasibility:

$\exists \omega$, s.t. $y^{(i)}(\omega^T x^{(i)} + b) > 1 \quad \forall i$ ✓

All the slater's conditions are satisfied.

$\therefore$ SVM objective satisfies all of slater's conditions.

· Now, Slaters conditions satisfied $\Rightarrow$

① $p^* = d^*$ primal optimal equals dual optimal

② $\exists\, \omega^*, \alpha^*, \beta^*$

ω^* is primal optimal point

α^*, β^* are dual optimal point

$$p^* = L(\omega^*, \alpha^*, \beta^*) = d^*$$

· KKT conditions (Karush Kuhn Tucker):

Consider a convex optimization problem

if $\exists\, \omega^*, \alpha^*, \beta^*$ s.t. $p^* = L(\omega^*, \alpha^*, \beta^*) = d^*$ $\Leftrightarrow$

① @ $\nabla_\omega L(\omega, \alpha^*, \beta^*)\big|_{\omega^*} = 0$

ⓑ $\nabla_\beta L(\omega^*, \alpha^*, \beta)\big|_{\beta^*} = 0$

} gradient variables

② $g_k(\omega^*) \leq 0 \quad \forall k$

$h_l(\omega^*) = 0 \quad \forall k$

} initial feasibility

ⓓ $\alpha_k^* \geq 0 \quad \forall k$

} dual feasibility

③ $\alpha_k^* g_k(\omega^*) = 0 \quad \forall K \Rightarrow g_k(\omega) < 0 \Rightarrow \alpha_k = 0$

↳ complementary slackness

SVM Objective: $\min_{w,b} \frac{1}{2} w^T w \quad , \quad y^{(i)} (w^T x^{(i)} + b) \geq 1$

$$\therefore L(w, b, \alpha) = \frac{1}{2} w^T w + \sum_{i=1}^{m} \alpha_i \left(1 - y^{(i)} (w^T x^{(i)} + b)\right)$$

$$\theta_P(w, b) = \min_{w,b} \left[\max_{\alpha \geq 0} L(w, b, \alpha) \right]$$

$$\theta_D(\alpha) = \max_{\alpha \geq 0} \left[\min_{w,b} L(w, b, \alpha) \right]$$

Now, $\nabla_w L(w, b, \alpha) = w - \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)}$

$$\nabla_w L(w, b, \alpha)\big|_{w^*} = 0 \Rightarrow w^* = \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)}$$

Similarly, $\nabla_b L(w, b, \alpha) = -\sum_{i=1}^{m} \alpha_i y^{(i)}$

$$\therefore \nabla_b L(w, b, \alpha)\big|_{b^*} = 0 \Rightarrow \sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

At $w^*, b^* \Rightarrow L(\alpha, w^*, b^*) = \frac{1}{2} w^{*T} w + \sum_{i=1}^{m} \alpha_i \left(1 - y^{(i)} (w^{*T} x^{(i)} + b)\right)$

$w^* = \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)}$ & $\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$

Substituting:

$$L(\alpha, w^*, b^*) = \frac{1}{2} \left(\sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)} \right)^T \left(\sum_{j=1}^{m} \alpha_j y^{(j)} x^{(j)} \right) + \sum_{i=1}^{m} \alpha_i - \left[\sum_{i=1}^{m} \alpha_i y^{(i)} \left(\sum_{j=1}^{m} \alpha_j y^{(j)} x^{(j)} \right)^T x^{(i)} + b \sum_{i=1}^{m} \alpha_i y^{(i)} \right]$$

$\sum_{j,i=1}^{m} \alpha_i \alpha_j y^{(i)} y^{(j)} x^{(i)T} x^{(j)}$ — $b \sum_{i=1}^{m} \alpha_i y^{(i)} = 0$

$$= \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y^{(i)} y^{(j)} x^{(i)T} x^{(j)}$$

Constraints: ① $\alpha_i \geq 0 \ \forall i$

② $\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$

Dual problem: $\max_{\alpha \geq 0} L(\alpha, w^*, b^*)$

Suyash Agrawal

Lecture 20 - 23.02.18

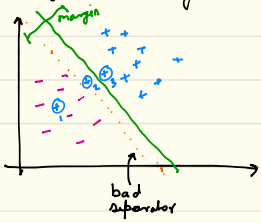
Primal Objective : $\min_{w,b} \frac{1}{2} w^T w$, $y^{(i)}(w^T x^{(i)} + b) \geq 1 \quad \forall i$

↓ Dual problem

Dual : $\max_{\alpha} \left(\sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \cdot \alpha_j \, y^{(i)} \cdot y^{(j)} \, x^{(i)T} x^{(j)} \right)$, $\alpha_i \geq 0 \ \forall i$ $\quad \sum_{i=1}^{m} \alpha_i y^{(i)} = 0$, $\omega = \sum_{i=1}^{m} \alpha_i y^{(i)} \cdot x^{(i)}$

SMO : Sequential Minimal Optimization , LIB SVM

Handling non-linearly separable data :



$\min_{w,b} \frac{1}{2} w^T w$, $y^{(i)}(w^T x^{(i)} + b) \geq 1$

How to handle cases like ① , ② & ③ ?

↳ $y^{(i)}(w^T x^{(i)} + b) \geq 1 - \epsilon_i$, $\epsilon_i \geq 0$ ↳ slack variables

But how to control ϵ's so that we get some sensible soln ? [else $w = 0$ is a soln]

↳ $\min_{w,b,\epsilon} \frac{1}{2} w^T w + c \sum_{i=1}^{m} \epsilon_i$ ↳ sum loss ; ↳ controls the effect of the penalty.

Note : $c \to \infty$ in linearly separable case.

In this case too slater's conditions are satisfied ∴ Primal soln = Dual soln.

$L(w, b, \epsilon, \alpha, \gamma) = \frac{1}{2} w^T w + c \sum \epsilon_i + \sum_{i=1}^{m} \alpha_i [1 - \epsilon_i - y^{(i)}(w^T x^{(i)} + b)] + \sum_{i=1}^{m} \gamma_i (-\epsilon_i)$

Dual objective : $\min_{w,b,\epsilon} (L(w, b, \epsilon, \alpha, \gamma))$

$\nabla_w L = 0 \Rightarrow w = \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)}$

$\nabla_b L = 0 \Rightarrow \sum_{i=1}^{m} \alpha_i y^{(i)} = 0$

$\nabla_\epsilon L = 0 \Rightarrow c - \alpha_i = \gamma_i$

$\max_{\alpha} \left[\sum \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y^{(i)} y^{(j)} x^{(i)T} x^{(j)} \right]$, $\alpha_i \geq 0$, $\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$, $0 \leq \alpha_i \leq c$

- $x \to \phi(x)$] ⇒ Kernel
-

Suyash Agrawal

- Solving Primal Problem:

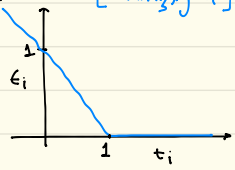
$$\min_{w,b} \frac{1}{2} w^T w + c \sum \epsilon_i \quad , \quad y^{(i)}(w^T x^{(i)} + b) \geq 1 - \epsilon_i \, , \quad \epsilon_i \geq 0$$

$$\hookrightarrow \epsilon_i \geq \underbrace{1 - y^{(i)}(w^T x^{(i)} + b)}_{t_i}$$

$$\therefore \min_{w,b,t} \frac{1}{2} w^T w + c \sum \epsilon_i \quad , \quad \epsilon_i \geq 1 - t_i \, , \; \epsilon_i \geq 0 \rightarrow \epsilon_i \geq \max(0, 1 - t_i)$$

$$t_i = y^{(i)}(w^T x^{(i)} + b) \qquad \hookrightarrow \epsilon_i = \max(0, 1 - t_i)$$

[as we are minimizing ϵ_i]



$\therefore$ We can transform to unconstraint problem:-

$$\min_{w,b} \frac{1}{2} w^T w + C \sum_{i=1}^{m} \max(0, 1 - t_i)$$

$$\text{where } t_i = y^{(i)}(w^T x^{(i)} + b)$$

But we can't use gradient descent because this is not differentiable.

Some possible soln:-

$$\epsilon_i = \begin{cases} 0 & t_i \geq 1 \\ 1 - t_i & t_i < 1 \end{cases} \qquad \nabla_{t_i} L = \begin{cases} 0 & t_i \geq 1 \\ -1 & t_i \leq 1 \end{cases}$$

↳ we are using left gradient. This is called subgradient.

$$\nabla_w L = w + C \sum_{i=1}^{m} \nabla_w [\max(0, 1 - t_i)]$$

$$= w + C \sum_{i=1}^{m} \nabla_t [\max(0, 1 - t_i)] \cdot \nabla_w t_i$$

$$= w + C \sum_{i=1}^{m} \begin{Bmatrix} 0 & t_i > 1 \\ -1 & t_i \leq 1 \end{Bmatrix} * (y^{(i)} x^{(i)})$$

Suyash Agrawal

Lecture 21 - 06.03.18

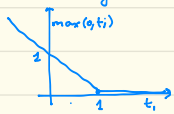
SVM:

$$\min_{w,b} \frac{1}{2} w^T w + C \sum_{i=1}^{m} \epsilon_i$$

Optimization methods:
- Primal space ⇒ Pegasos
- Dual space ⇒ SMO

$$\min_{w,b} \frac{w^T w}{2} + \sum_{i=1}^{m} \max(0, 1 - t_i), \text{ where } t_i = y^{(i)}(w^T x^{(i)} + b)$$

$\frac{w^T w}{2}$: regularizer; $\max(0, 1-t_i)$: hinge loss



Note: the gradient is not zero at minima!

Dual Formulation:

$$\frac{1}{2} w^T w + C \sum_{i=1}^{m} \epsilon_i \; ; \; y^{(i)}(w^T x^{(i)} + b) \geq 1 - \epsilon_i \; ; \; \epsilon_i \geq 0$$

$$\Downarrow$$

$$\max_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{j=1}^{m} \sum_{i=1}^{m} \alpha_i \alpha_j y^{(i)} y^{(j)} \left[x^{(i)T} x^{(j)} \right] \; ; \; 0 \leq \alpha_i \leq C \; ; \; \sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

} solve for α_i

Analysis of ϵ_i, α_i, γ_i & support vectors

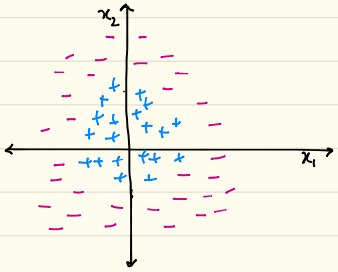
In region ① ⇒ $\alpha_i = 0$, $\gamma_i = C$

③ ⇒ $\alpha_i = C$, $\gamma_i = 0$, $\epsilon_i > 0$

② ⇒ $0 \leq \alpha_i \leq C$, $\gamma_i = 0$, $\epsilon_i = 0$

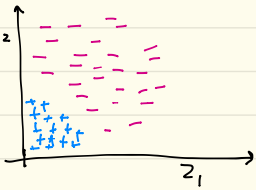
?

Kernels:



How to classify here? Can we use a linear separator?

$$\Rightarrow \quad z_1 = x_1^2, \quad z_2 = x_2^2 \longrightarrow$$



ϕ is kernel: $(x_1, x_2) \xrightarrow{\phi} (x_1^2, x_2^2)$

$(x_1, x_2) \in R^n \xrightarrow{\phi}$ Quadratic $\phi(x) \in R^N$ $\quad x_1^2, x_2^2, x_1 x_2, x_1, x_2 \in R^N$

Suyash Agrawal

Lecture 22 - 08.03.18

Kernels:

$(x_1, x_2) \xrightarrow{\phi} \phi(x_1, x_2)$

$\{x_1^2, x_2^2, x_1x_2, x_1, x_2\}$

$(x_1, x_2 \cdots x_n) \xrightarrow[\text{polynomial of degree } d]{\phi}$ terms of degree $\le d$ $\}$ # $= {}^{n+d}C_d$

$\in R^N$, $N = O(n^d)$

If we could compute: $K(x,z) = \phi(x)^T \cdot \phi(z)$ efficiently then we are o.k. ☺ $\}$ Kernel Trick

E.g.:

$K(x,z) = (x^Tz + c)^2$

Goal: $K(x,z)$ represents $\phi(x)^T\phi(z)$ for some $\phi(x)$ (feature transformation)

$\hookrightarrow (\sum_{i=1}^n x_iz_i)^2 + 2c\sum x_iz_i + c^2$

$= \left(\sum_{i=1}^n x_iz_i\right)\left(\sum_{j=1}^n x_jz_j\right) + \sum_{i=1}^n (\sqrt{2c}\,x_i)(\sqrt{2c}\,z_i) + c \cdot c$

$= \sum_{i=1}^n\sum_{j=1}^n x_ix_jz_iz_j + \sum_{i=1}^n (\sqrt{2c}\,x_i)(\sqrt{2c}\,z_i) + c^2$

$\phi(x) = \begin{bmatrix} x_1x_1 \\ x_1x_2 \\ \vdots \\ x_1x_n \\ x_2x_1 \\ x_2x_3 \\ \vdots \\ \sqrt{2c}\,x_1 \\ \vdots \\ \sqrt{2c}\,x_n \\ c \end{bmatrix}$ $\phi(z) = \begin{bmatrix} z_1z_1 \\ z_1z_2 \\ \vdots \\ z_1z_n \\ z_2z_1 \\ \vdots \\ \sqrt{2c}\,z_1 \\ \vdots \\ \sqrt{2c}\,z_n \\ c \end{bmatrix}$

- Similarly: $K(x,z) = \underbrace{(x^Tz + c)^d}_{O(n + \log d)}$, then $\phi(x)$ = degree d poly. on x $\Rightarrow O(n^d)$

SVM Objective: $\sum_{i=1}^m \alpha_i - \frac{1}{2}\sum_i\sum_j \alpha_i\alpha_j y^{(i)}y^{(j)}\underbrace{[x^{(i)T}x^{(j)}]}_{\phi(x^{(i)})^T\phi(x^{(j)})}$; $0 \le \alpha_i \le C$

$w = \sum_{i=1}^m \alpha_i y^{(i)}x^{(i)}$ $\}$ How to calculate w if $\phi(x)$ is exp.?
↳ We don't! we only need w^Tx of classification value is computable using kernel trick.

$h_{w,b}(x) = \begin{cases} 1 & [\text{if } w^T\phi(x) \ge 0] \to \left(\sum \alpha_i y^{(i)}\phi(x^{(i)})\right)^T\phi(x) \\ 0 & \text{otherwise} \end{cases}$

Similarly for b ← { Check! }

Suyash Agrawal

Kernel Function :

$x, z \in \mathbb{R}^n$, $K(x,z) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$

We say that $K(x,z)$ is a valid kernel function iff $\exists$ a feature transformation ϕ s.t. $K(x,z) = \phi(x)^T \phi(z)$

Another example of a Kernel fn:

$$K(x,z) = e^{-\frac{\|x-z\|^2}{2\sigma^2}} \implies \text{Gaussian / RBF Kernel}$$

↳ Similarity b/w two points.

Q) How to prove that $K(x,z)$ is a valid kernel?

- Find $\phi(x)$ s.t. $K(x,z) = \phi(x)^T \phi(z)$
- Mercer's Theorem

- ## Mercer's Theorem :

$K(x,z) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ on a finite dimensional space then for K to be valid it is necessary & sufficient that:

Let $\{x^{(1)}, \dots, x^{(m)}\}$ be a finite set of points in the original space & Kernel matrix:

$$K^{(m)} = \begin{bmatrix} & & j & \\ & & \vdots & \\ i & \cdots & K_{ij}^{(m)} = K(x^{(i)}, x^{(j)}) & \end{bmatrix} \quad \text{then :}$$

1) $K^{(m)}$ is symmetric
2) $K^{(m)}$ is positive semi definite

} Mercer's Thm conditions.

Proof that this is necessary :

① $K^{(m)}$ is symmetric : $K_{ij}^{(m)} = \phi(x^{(i)})^T \phi(x^{(j)})$

$= \phi(x^{(j)})^T \phi(x^{(i)})$

$= K_{ji}^{(m)}$

② $K^{(m)}$ is +ve semi definite :

if $u^T K^{(m)} u \geq 0 \quad \forall u$ then now

$$u^T K^{(m)} u = \sum_{i,j=1}^{m} u_i K_{ij}^{(m)} u_j$$

$$= \sum_{i,j=1}^{m} u_i \phi(x^{(i)})^T \phi(x^{(j)}) u_j$$

$$= \sum_{i,j=1}^{m} u_i \left(\sum_k \phi(x^{(i)})_k \phi(x^{(j)})_k \right) u_j$$

$$= \sum_{i,j=1}^{m} \sum_k u_i \phi(x^{(i)})_k \phi(x^{(j)})_k u_j$$

$$= \sum_k \left(\sum_i u_i \phi(x^{(i)})_k \right) \left(\sum_j u_j \phi(x^{(j)})_k \right)$$

$$= \sum_k \left[\sum_i u_i \phi(x^{(i)})_k \right]^2 \geq 0 \quad ; \text{ Hence Proved}$$

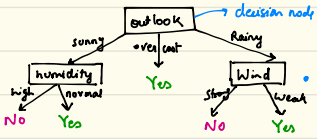
Suyash Agrawal

Decision Trees:

Example: Learning to play Tennis:

x_1 (outlook) x_2 (humidity) x_3 (wind) y ⟨Y/N⟩

Outlook ∈ {sunny, overcast, rain}
humidity ∈ {high, normal}
wind ∈ {high, normal}



• When to stop? Overfitting?

Example Tree

1) Internal Nodes → Decision Nodes
2) Leaves → Labels
3) Test single attribute at each node { Training difficult with multiple attributes even though they may have more advantage due to look ahead }

Note: Multi attribute has same representation power as single attribute decision node.

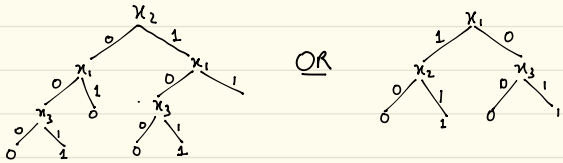
Claim: A decision tree can represent any Boolean function over n variables

↳ One way is to expand the tree fully!

But we can also do better {shorter tree}

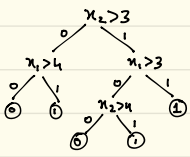
Eg: $(x_1 \wedge x_2) \vee (\neg x_1 \wedge x_3)$, built shortest possible tree

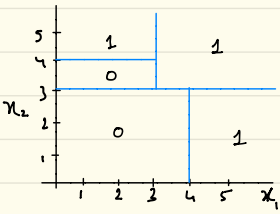
↳



Not Shortest

- How to handle continous data? → Binning / Learning the splitter
- Variations: testing same attribute again down the path {in case of continous / multivalued & binary tree}
- Decision surface:





For oblique we need to test two attributes at a time.

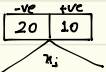
Suyash Agrawal

Lecture 24 - 13.03.18

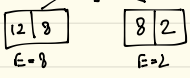
```
Func growTree (D) {
    if (y^(i) == 0) ∀ i
        return Leaf (D, 0)
    if (y^(i) == 1) ∀ i
        return Leaf (D, 1)
    x_j = chooseBestAttr(D);
    D_0 = {x^(i), y^(i)}_{i=1}^m  s.t. x_j^(i) = 0
    D_1 = {x^(i), y^(i)}_{i=1}^m  s.t. x_j^(i) = 1
    return new Node (D, x_j, growTree(D_0), growTree(D_1));
}
```

Training error:

Let us say that error is # of misclassified labels.

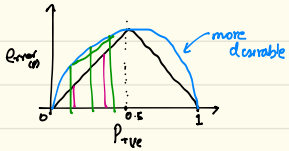


L1:



$\therefore J^{L1} = J_0^{L1} + J_1^{L1}$
$= 8 + 2 = 10$

But this error does not ensure that data labels are segregated!

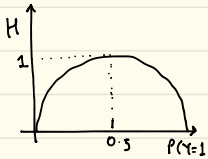


Entropy:

Y: discrete valued R.V.

$H(Y) = \sum_y P(Y=y)\,[-\log_2(P(Y=y))]$

$H(Y)$: Entropy (Expected surprise); $-\log_2(P(Y=y))$: Surprise

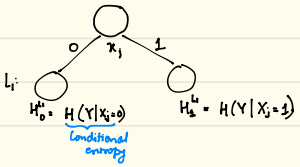


If Y = Bernoulli R.V.

Now, let us use entropy as our error metric.

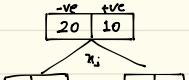
☆ Split on the feature which maximizes $H(Y) - H(Y|X_j)$

$H(Y|X_j) = \sum_{x_j} P(X_j = x_j) \cdot H(Y|X_j = x_j)$

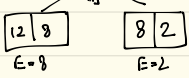


$I(Y, X_j) = H(Y) - H(Y|X_j)$ (symmetric) $= I(X_j, Y)$ — Information Gain

$\therefore X_j^* = \underset{x_j}{\text{argmax}} \left[H(Y) - H(Y|X_j)\right]$



L1:



$H(Y) = \frac{2}{3}\left[-\log\left(\frac{2}{3}\right)\right] + \frac{1}{3}\left(-\log\left(\frac{1}{3}\right)\right) = 0.9183$

$H(Y|X_j=0) = \frac{3}{5}\left(-\log\frac{3}{5}\right) + \frac{2}{5}\left(-\log\frac{2}{5}\right) = 0.9710$

$H(Y|X_j=1) = 4/5\left(-\log 4/5\right) + \frac{1}{5}\left(-\log\frac{1}{5}\right) = 0.72$

$H(Y|X_j) = \frac{2}{3} \cdot H(Y|X_j=0) + \frac{1}{3} H(Y|X_j=1) = 0.8878$

Suyash Agrawal

Lecture 25 - 15.03.18

- Slides Today!
- Patho Logical case for decision trees → Parity with noise.
- Definition of over fitting

- **Random Forests:**

Decision trees are unstable learners → High variance

Based on small changes in data, the model can drastically change

Soln: Bagging

Bagging:

$D = \{x^{(i)}, y^{(i)}\}_{i=1}^{m}$

function generateBootstrapSamples(D) {

  For k=1 to r {

    $D_k = \{\}$

    For i = 1 to m {

      $c \sim$ uniformly($\{1 \cdots m\}$)

      $D_k \leftarrow \{x^{(c)}, y^{(c)}\} \cup D_k$

    }

  }

}

Sampling with replacement

For prediction → Classification: Voting
→ Regression: Average

Lecture 26 - 16.03.18

Forest Man?
Random Art?
Random Forest.

Random Forests:

Bag of Decision Trees

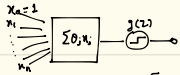
1) # estimators (# of trees)
2) max-features → Select best feat. from a random sample (√ size max feat.) of all features
3) Boot strap
4) Out of bag error. } H.W.

Trade Off (bias-variance): Accuracy of individual tree high n_s V/S Low variance of model low n_s

Suyash Agrawal

Lecture 27 - 20.03.18

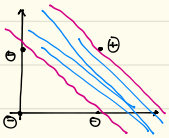
Basic unit: Perceptron



$$g(\theta^T x) = \begin{cases} 1 & \text{if } \theta^T x \le 0 \\ 0 & \text{o.w.} \end{cases}$$

$\therefore$ This perceptron learns linear boundary $[\theta^T x \ge 0]$.

E.g.: $f(x_1, x_2) = x_1 \wedge x_2 \qquad x_1, x_2 \in \{0,1\}$

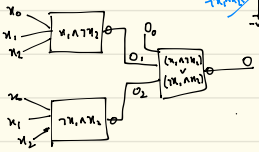


$h_\theta(x) = I\{\theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 \ge 0\}$

Optimal parameter (if slope is constrained to -1): $\theta_0 = 1$, $\theta_1 = 1$, $\theta_2 \in (-2, -1)$

- if $f(x_1, x_2) = x_1 \vee x_2 \Rightarrow$ if $\theta_2 = 1$, $\theta_1 = 1$ then $\theta_0 \in (-1, 0)$
- if $f(x) = \neg x \Rightarrow$ if $\theta_1 = -1$ then $\theta_0 \in (0, 1)$
- if $f(x) = x \Rightarrow$ if $\theta_1 = 1$ then $\theta_0 \in (-1, 0)$
- if $f(x_1, x_2) = x_1 \oplus x_2$

Cannot Learn this!!! But $x_1 \oplus x_2 = (x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge x_2)$ & this can be represented by composition of perceptrons



Input layer | Layer 1 | Layer 2 | Output layer

- How do we learn a perceptron?

$$J(\theta) = \sum_{i=1}^{m} \left(y^{(i)} - h_\theta(x^{(i)})\right)^2$$

→ Use Gradient descent. But problems:-
- ⁘ $h_\theta(x)$ is non differentiable! [Also subgradient method won't work. Why?]
- ⁘ Not convex, but we don't care ☺

Delta Rule (adhoc heuristic algo for training):

init (θ);

do {

$x^{(i)} \leftarrow$ wrongly classified examples

for all j {

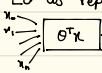
$$\Delta\theta_j = \eta\left[y^{(i)} - h_\theta(x^{(i)})\right] x_j^{(i)}$$
$$\theta_j = \theta_j + \Delta\theta_j;$$

}

} (until convergence)

Note: This actually converges if our data is linearly separable for a sufficiently small η.

Let us replace our step function to sigmoid function. (to make loss differentiable)



$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}$$

Suyash Agrawal

Training of a perceptron:

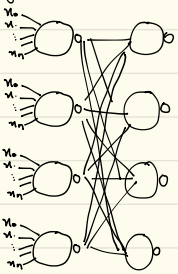
$$J(\theta) = \frac{1}{2} \sum_{i=1}^{m} (y^{(i)} - h_\theta(x^{(i)}))^2$$

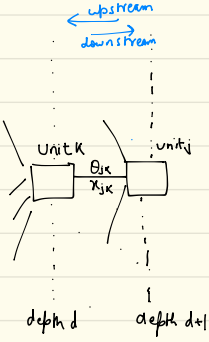
$$\frac{\partial J(\theta)}{\partial \theta_j} = -\frac{1}{2} \sum_{i=1}^{m} 2 (y^{(i)} - h_\theta(x^{(i)})) \cdot \frac{\partial}{\partial \theta_j} (h_\theta(x^{(i)}))$$

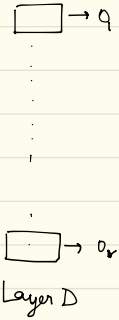
$$= -\sum_{i=1}^{m} [y^{(i)} - h_\theta(x^{(i)})] \underbrace{(h_\theta(x^{(i)}))(1 - h_\theta(x^{(i)}))}_{\text{extra term as compared to logistic}} x_j^{(i)}$$

[Check sign] [not convex in general]

Training a general network:







Note: hidden layer: one layers hidden from the output

x_{jk} = output from unit k to unit j

θ_{jk} = parameter associated with input x_{jk}

net_j = net linear input associated with unit j $= \sum_{k \in \text{upnbr}(j)} \theta_{jk} x_{jk}$

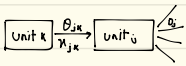
↳ upneighbours of unit j

O_j = output of unit j $= g(net_j) = \frac{1}{1 + e^{-net_j}}$

$$J(\theta) = \sum_{i=1}^{m} \sum_{d=1}^{N} [y_d^{(i)} - O_d^{(i)}]^2$$

↳ Indirectly dependent on param in hidden layer

① $$\frac{\partial J(\theta)}{\partial \theta_{jk}} = \frac{\partial J(\theta)}{\partial net_j} \cdot \frac{\partial net_j}{\partial \theta_{jk}} = \frac{\partial J(\theta)}{\partial net_j} x_{jk}$$



$$\therefore \frac{\partial J(\theta)}{\partial \theta_{jk}} = \frac{\partial J(\theta)}{\partial net_j} x_{jk} = -\underbrace{\delta_j}_{\text{we are defining this}} x_{jk}$$

Note: We will assume : $m = 1$

② $$\frac{\partial O_j}{\partial net_j} = O_j (1 - O_j), \quad \forall k \in \text{upnbr}(j) \quad x_{jk} = O_k$$

Suyash Agrawal

Ⓐ Unit $j \in$ output layer

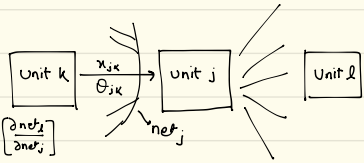
$$\frac{\partial J(\theta)}{\partial net_j} = \frac{\partial}{\partial net_j}\left[\frac{1}{2}\sum_{c=1}^{r}(y_c - o_c)^2\right] = -1\cdot(y_j - o_j)\, o_j(1-o_j)$$

Note:

$f(y_1, \dots y_r)$, where each y_c is a f^n of x

$$\frac{\partial f(y_1, \dots y_r)}{\partial x} = \sum_{e=1}^{r} \frac{\partial f(y_1, \dots y_r)}{\partial y_e}\frac{\partial y_e}{\partial x}$$

Ⓑ Unit $j \in$ hidden layer



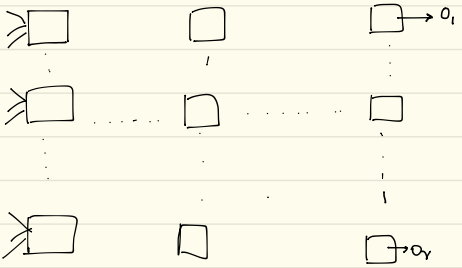
$$\frac{\partial J(\theta)}{\partial net_j} = \sum_{l \in downstr(j)} \left[\frac{\partial J(\theta)}{\partial net_l}\right]\left[\frac{\partial net_l}{\partial net_j}\right]$$

$$= \sum_{l \in downstr(j)} -\delta_l \cdot \frac{\partial \theta_{lj} O_j}{\partial net_j} = \sum_{l \in downstr(j)} -\delta_l\, \theta_{lj}\, O_j(1-O_j)$$

$$\therefore \delta_j = \sum_{l \in downstr(j)} \delta_l\, \theta_{lj}\, O_j(1-O_j)$$

Lecture 29 - 23.03.18

- Layer by layer training:



Training:

① Forward Propagation

② Back prop

③ Weight update

- Slides
  - Auto encoder

Suyash Agrawal

Lecture 30 - 03.04.18

Deep Learning:

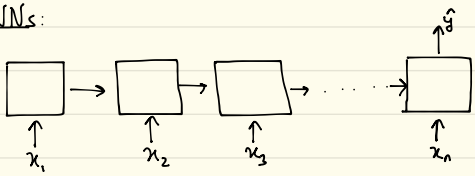
"Deep" neural network

Architectures:

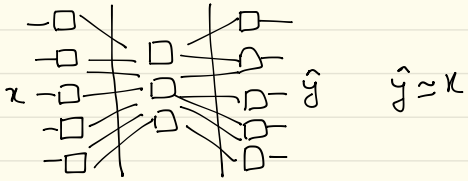
1) CNNs

Lecture 31 - 05.04.18

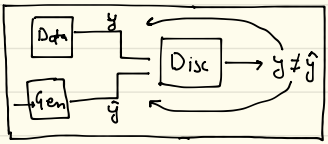
- RNNs:



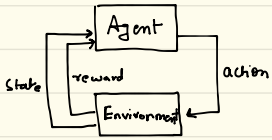
- Auto encoders:



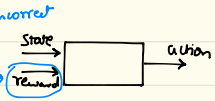
- GANs: (Generative Adversarial Networks)



- Deep Q-learning: {Reinforcement learning}

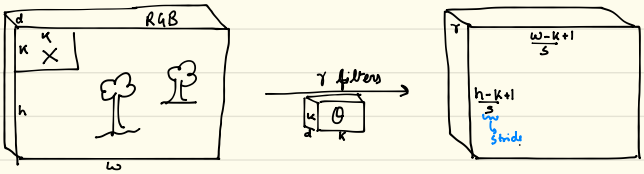


Example: Alpha Go



Prof: Balraman Ravindran {IIT M}

•) CNNs:

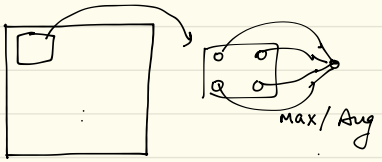


$g\left(\sum_{i,j} x_{ij}\theta_{ij} + b\right)$, g is typically ReLU.

We typically pad our images to make input & output dimensions same.

Lecture 32 - 06.04.18

• Pooling:



∴ for 2X2 pooling ⇒ 500x500 $\xrightarrow{\text{pool}}$ 250x250

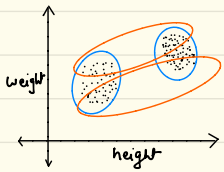
We can use either avg or max pooling

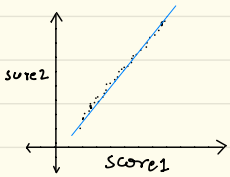
• Slides

Suyash Agrawal

Lecture 33 - 10.04.18

- **Unsupervised Learning:**
  - $\{x^{(i)}, \cancel{y^{(i)}}\}_{i=1}^{m}$





1) Clustering
2) Density estimation: $p(x) = \sum_{z} P(x|z) \cdot P(z)$
3) Dimensionality Reduction: PCA {Principal Component Analysis}

- **K-Means:**

Good clustering:

- Points in the same cluster are close by
- Points in different clusters are far apart

Let us assume that we know the number of clusters = k

Parameters:

- $\mu = (\mu_1, \cdots, \mu_k)$, $i \in \{1 \cdots k\}$
- $c = (c^{(1)}, c^{(2)}, \cdots, c^{(m)})$, $i \in \{1 \cdots m\}$

Objective:

$$\min_{\mu, c} \sum_{i=1}^{m} \| x^{(i)} - \mu_{c^{(i)}} \|^2$$

- Given cluster assignment we can find μ
- Given μ we can find cluster assignments.

Algorithm:

- Randomly initialize μ
- While not converged:
  - Given μ find c {E-step}
  - Given c find μ {M-step}

- Points in each cluster should be as close to their representatives as possible. This automatically implies that representatives are ~~for~~ far apart.

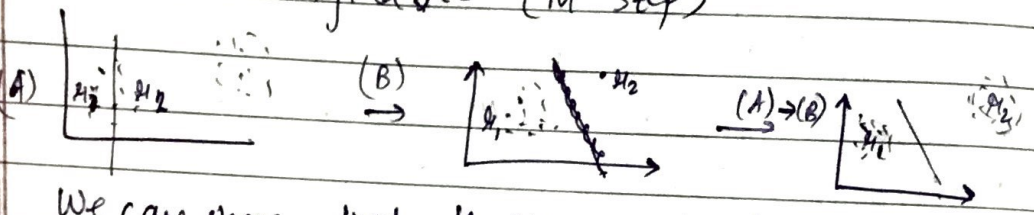
★ If we know the assignments, we can compute the representatives. We can do this by minimising sum of distances of representatives from mean of each clusters. ~~Given~~

★ K-Means Algorithm:-

1) Given μ's compute cluster assignments.
2) Given c's compute ~~do~~ cluster representatives.

~~This~~ k is ~~any~~ a fixed number. Our algorithm should first compute k and then find out the clustering.

→ (A) Given μ's find c's (E-step)
→ (B) Given c's find μ's (M-step)



★ We can show that K-Means finds local minima of

$$\underset{c,\mu}{\operatorname{argmin}}\ J(c,\mu)$$

12/04/2018

→ InitParams(); {$\mu_1 \dots \mu_k$ → parameters}
→ (A) → AssignPoints(); → E step
→ (B) → EstimateParams() → M steps
Repeat until convergence

★ $$J(c,\mu) = \underset{c,\mu}{\operatorname{argmin}} \sum_{i=1}^{m} \| x^{(i)} - \mu_{c^{(i)}} \|^2 \qquad c = (c^{(1)} \dots c^{(m)}) \quad \mu = (\mu^{(1)} \dots \mu^{(k)})$$

Here c is discrete, so we cannot apply gradient descent.
$\mu_k \in \mathbb{R}^n$

★ Coordinate Descent :-

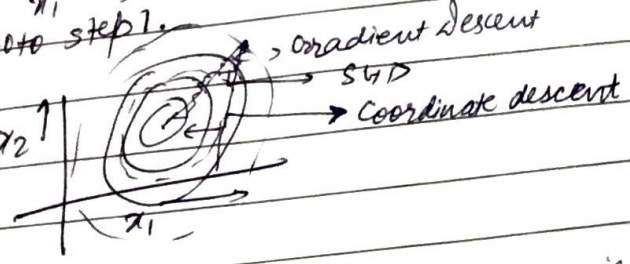
$$\underset{x_1,x_2}{\operatorname{argmin}}\ f(x_1,x_2)$$ → fix everything except one direction

Step 1: $\min_{x_2} f(x_1, x_2)$ → stops at tangent to contour
(x_1 fixed)

Step 2: Then fix x_2 and move along x_1.

$$\min_{x_1} f(x_1, x_2')$$

goto step 1.



★ Convergence criteria - when there is no direction to move (no improvement in any direction). This has similar guarantees as gradient descent. When to use it? Single variable optimization can be easier (analytical solution available). Apply coordinate descent where gradient ~~descent~~ is difficult to compute while it may be possible to solve entire coordinate descent analytically.

★ Block Coordinate Descent :- Fix some coordinates and descend on others. (some maybe discrete while other continuous)

★ Q: For SVM, we use block coordinate descent. Why can't we use coordinate descent in SMO?
Because of similar guarantees, block coordinate descent also converges to local optima.

★ $J(c, \mu) = \underset{c, \mu}{\operatorname{argmin}} \sum_{i=1}^{m} [\|x^{(i)} - \mu_{c^{(i)}}\|^2]$

① Fix μ's

$$\underset{c^{(1)} \dots c^{(m)}}{\operatorname{argmin}} \sum_{i=1}^{m} \|x^{(i)} - \mu_{c^{(i)}}\|^2$$

$$\equiv \forall i \quad \underset{c^{(i)}}{\operatorname{argmin}} \|x^{(i)} - \mu_{c^{(i)}}\|^2 \rightarrow E \text{ step}$$

② Fix $c^{(i)}$s,

$$\underset{\mu_1 \dots \mu_k}{\operatorname{argmin}} \sum_{i=1}^{m} \|x^{(i)} - \mu_{c^{(i)}}\|^2 = \underset{\mu_1 \dots \mu_k}{\operatorname{argmin}} \sum_{i=1}^{m} \sum_{K=1}^{k} 1\{c^{(i)} = K\} \|x^{(i)} - \mu_K\|^2$$

$$\equiv \forall K \quad \underset{\mu_K}{\operatorname{argmin}} \sum_{i=1}^{m} 1\{c^{(i)} = K\} \|x^{(i)} - \mu_K\|^2$$

(K-Medians chooses one of the points as representatives)

* $\mu_k \in \mathbb{R}^n$ We can apply SGD or CGD.

$\Rightarrow \mu_k = \dfrac{\sum_{i=1}^{m} 1\{c^{(i)}=k\} x^{(i)}}{\sum_{i=1}^{m} 1\{c^{(i)}=k\}} \quad \forall k$

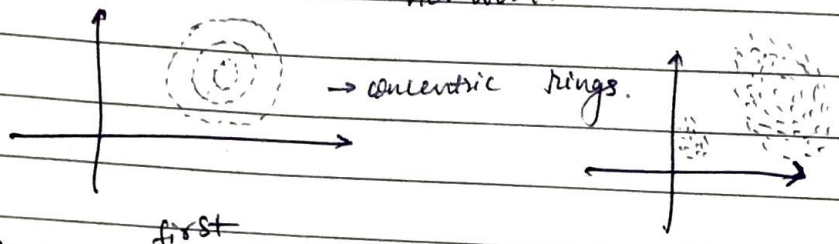
↳ Mean

* K-Means: ① Start with some initial cluster means.

repeat until converges
- (A) Find c's given μ's.
- (B) Find μ's given c's.

→ Convergence: No change in c or μ in consecutive steps.
This converges to the local optima.

Q: When does K-Means not work?



* In K-Means first pick a centroid among one of the points.
* Clusters must be of similar size and well separated.
K-Means assumes "K". Some algorithms figure out k and then proceed.

13/04/2018

Suyash Agrawal

Lecture 35 - 13.04.18

- Some other clustering algorithm apart from k-Means: • DBSCAN
  • Chinese Restaurant Process

- **Gaussian Mixture Models (GMM):**

$z^{(i)} \sim \text{Multinoulli}(\phi)$

$x^{(i)} | z^{(i)} \sim \text{Normal}(\mu_k, \Sigma_k)$ (given $z^{(i)} = k$)

$$\theta_{ML} = \arg\max_\theta \prod_{i=1}^{m} P(x^{(i)}; \theta)$$

$$= \arg\max_\theta \prod_{i=1}^{m} \left(\sum_{j=1}^{k} \underbrace{P(x^{(i)}, j; \theta)}_{P(x^{(i)}|j;\theta) \cdot P(j;\theta)} \right)$$

Since there is a Σ here we can't have closed form solution. Also gradient descent is bad because of Σ inside a log. Thus we use iterative style optimization (EM)

Algorithm:

① Init Params () ~ $(\phi, \{\mu_k, \Sigma_k\}_{k=1}^{K})$

repeat

Ⓐ Estimate $z^{(i)}$s given the parameters:

↳ $$P(z^{(i)} | x^{(i)}; \theta) = \frac{P(x^{(i)} | z^{(i)}; \theta) \cdot P(z^{(i)}; \theta)}{\sum_{z^{(i)}} P(x^{(i)} | z^{(i)}; \theta) \cdot P(z^{(i)}; \theta)}$$ } E step

Ⓑ Update the parameters:

↳ $$\phi_k = \sum_{i=1}^{m} 1\{z^{(i)} = k\} / m$$

$$\mu_k = \frac{\sum_{i=1}^{m} 1\{z^{(i)} = k\} x^{(i)}}{\sum_{i=1}^{m} 1\{z^{(i)} = k\}}$$

$$\Sigma_k = \frac{\sum_{i=1}^{m} 1\{z^{(i)} = k\} (x^{(i)} - \mu_k)(x^{(i)} - \mu_k)^T}{\sum_{i=1}^{m} 1\{z^{(i)} = k\}}$$

$P(z^{(i)} = k) \Rightarrow$ Soft EM

Since otherwise we throw away info & impose a hard bound.

} M step

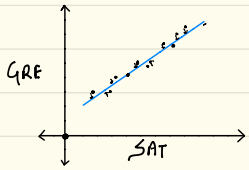
- **k Means vs GMM:**

① ϕ : uniform
② Σ = identity
③ Do hard assignment

K Means	GMM
① Init Params (); // set of centroids	① Init params () // set ϕ, $\{\mu_k, \Sigma_k\}$
Hard Ⓐ E-Step: Assign cluster to every point	Ⓐ E-step: Compute $P(z^{(i)} \| x^{(i)}; \theta)$ Soft
Ⓑ M-step: Compute $(\mu_1 \cdots \mu_k)$ given $(c^{(1)}, \ldots, c^{(m)})$	Ⓑ M-step: Compute ϕ, $\{\mu_k \Sigma_k\}$ given $P(z^{(i)} \| x^{(i)}; \theta)$

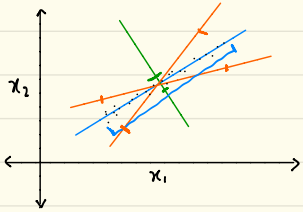
Suyash Agrawal

Principal Component Analysis:



- Dimensionality Reduction
- Difference with Autoencoders?
- Eigen faces: nxn image to ~50 dimensional vector.

Advantages:
- Discern Pattern
- Reduce computation
- Reducing noise



We try to maximize variability while projecting.

- Given: $\{x^{(i)}\}_{i=1}^{m}$, $x^{(i)} \in \mathbb{R}^n$
- To find: $\{u_1, \dots u_k\}$ where $u_i \in \mathbb{R}^n$, $k << n$

$$u_i \perp u_j \quad \forall i,j \quad \& \quad \|u_i\|^2 = 1$$

$$x^{(i)} \approx \sum_{i=1}^{k} \lambda_i u_i$$

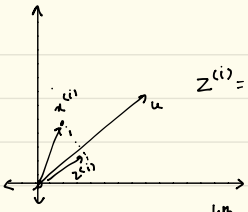
- Normalizing Data:

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_j^{(i)} \quad , \quad \forall j: \; x_j^{(i)} \longleftarrow \frac{x_j^{(i)} - \mu_j}{\sigma_j}$$

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m} (x_j^{(i)} - \mu_j)^2$$

We may not want to preprocess if all attributes are from same dimension. {E.g. not in image pixels}

Suyash Agrawal



$$z^{(i)} = (u^T x^{(i)})\, u$$

- We can either try to maximize variance or minimize dist. projected

$$\frac{1}{m}\sum (z^{(i)})^2 = \frac{\sum (x^{(i)T} u)^2}{m} = \frac{1}{m}\sum (x^{(i)T} u)^T (x^{(i)T} u)$$

$$= u^T \left[\frac{1}{m} \sum_{i=1}^{m} x^{(i)} x^{(i)T} \right] u$$

$\Sigma \sim$ empirical cov matrix

$$\Sigma = \begin{bmatrix} & & k^{th} & \\ & & & \\ j^{th} & & \text{Cov}(x_j, x_k) & \\ & & & \end{bmatrix}$$

$$\Sigma_{jk} = \frac{1}{m} \sum_{i=1}^{m} x_j^{(i)} \cdot x_k^{(i)} \quad , \quad x_j^{(i)} \leftarrow \frac{(x_j^{(i)} - u_j)}{\sigma_j}$$

PCA (single component) $\Rightarrow \arg\max_u \left[u^T \Sigma u \right]$

Find $\{u_1 \cdots u_m\}$ s.t. $u_{l_1} \perp u_{l_2}$, $\|u_l\|^2 = 1$

$\hookrightarrow$ Projection will be: $z_1^{(i)}, \cdots z_k^{(i)}$

$$\therefore (u_1, \cdots u_k) = \underset{u_1 \cdots u_k}{\arg\max} \sum_{l=1}^{k} \left[u_l^T \Sigma u_l \right] \text{, where } u_{l_1} \perp u_{l_2} \text{ \& } \|u_l\|^2 = 1$$

Solved using SVD. Eigenvalues based computation

Lecture 37 - 19.04.18

Class skipped due to Imagine Cup finals.

19/04/2018

Exam Q

★ Maximising variance is equivalent to minimising sum of squared distances from projected line.

$\{u_1 \dots u_k\}$ → orthonormal basis

$$\underset{u_1,\dots,u_k}{\operatorname{argmax}} \sum_{l=1}^{k} (u_l^T \Sigma u) \qquad \{u_1 \dots u_k\} \rightarrow \text{orthonormal basis}$$

$\|u_l\|^2 = 1$

$$\Sigma = \frac{1}{m} \sum_{i=1}^{m} x^{(i)} x^{(i)T} \qquad (n\times 1 \times 1 \times n \equiv n \times n)$$

$$\text{cov}(X_j, X_k) = E[(X_j - E[X_j])(X_k - E[X_k])]$$

($E[X_j] \neq 0$, $E[X_k] \neq 0$)

X = Design Matrix $(M \times n)$

$\Rightarrow \Sigma = [X^T X]$

$$\Rightarrow \underset{u_1 \dots u_k}{\operatorname{argmax}} \sum_{l=1}^{k} u_l^T [X^T X] u_l$$

★ $A \in R^{n \times n}$

λ :- eigenvalue of A if $\exists x$ s.t. $Ax = \lambda x$ ($x \in R^n \setminus \{0\}$, $x \neq 0$, $\lambda \in R$, eigenvector; +ve semi-definite)

$$(A - \lambda I)x = 0$$

$\Rightarrow \det(A - \lambda I) = 0$

↳ Polynomial in λ (degree n)

★ Use eigenvalue decomposition:

$A = Q \Lambda Q^{-1}$ ($\Lambda \rightarrow$ diagonal matrix

$Q^{-1} = Q^T$, Q: rows & columns are $\perp^r$ to each other, norm of each row & col. is 1, Q : orthonormal)

$O(n^3)$

$A = X^T X$ (→ +ve semi-definite)

$$\Rightarrow \underset{u_1 \dots u_k}{\operatorname{argmax}} \sum_{l=1}^{k} u_l^T (X^T X) u_l$$

$u_1 \dots u_k$ are principal eigenvectors of $X^T X$ (takes those eigenvectors whose eigenvalues are maximum)

Here, pick k principal eigenvectors.

★ $X \in R^{m \times n}$ $O(n^2 m^2)$ in $X^T X$ computation

★ SVD:- Singular Value Decomposition

$X \in R^{m \times n}$

$X = U D V^T$ (n-m entries 0 and rest m entries along diagonal non-zero)

D is diagonal matrix.

$m[\ U\ ]\ m[\ D\ ]\ n[\ V\ ]$

Columns of U are orthonormal & eigenvectors of XX^T.

Columns of V are eigenvectors of $X^T X$.

$O(m^2 n)$ time

★ If n is smaller do SVD on X^T else on X.

Exam Question → D:- entries of D are square roots of eigenvalues of $X^T X$ or XX^T.

$X^T X = (U D V^T)(V D U^T)$

$= U D V^T V D U^T$

$= U D^2 U^T$

$X^T X = V D^2 V^T$ → eigenvalues

20/04/2016

If $m < n$ → fat matrix

$X \in R^{m \times n}$

★ SVD:- $X = U D V^T$ $D \in R^{m \times n}$

$U \in R^{m \times m}$ $D \in R^{m \times n}$ $V^T \in R^{n \times n}$

$m[\ \ ]\ [\ \ ]\ n[\ \ ]$

U:- orthonormal (all rows & columns ⊥ to each other and norm = 1)

★ Columns of u are eigenvectors of XX^T

columns of v are eigenvectors of $X^T X$.

Entries of D → sqrt of eigenvalues of XX^T.

Proof → Assume $X^T X$ is full rank.

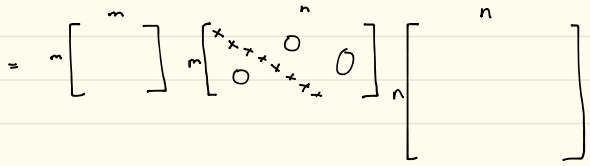
$X^T X = V D^T U^T U D V^T = V (D^T D) V^T$ → Eigen value decom. of $X^T X$

Suyash Agrawal

Lecture 38 - 20.04.18

SVD :

$X = U D V^T \in R^{m \times n} \quad m < n$



U, V are orthonormal.

Colums of U are eigen vectors of XX^T

" " V " " " of X^TX

Assume XX^T is full rank.

$X^TX = V D^T U^T U D V^T = V (D^T D) V^T$

$XX^T = U D V^T V D^T U^T = U (D D^T) U^T$

} eigen value decomposition

$D^T D = R^{n \times m} = diag(D_{11}, D_{22}, \cdots, D_{mm}, 0, 0, \cdots 0)$

EM (Expectation Maximization):

Given $\{x^{(i)}\}_{i=1}^m$

Suppose: We have model for $\{x^{(i)}, y^{(i)}\}_{i=1}^m$

$P(x^{(i)}, y^{(i)}; \theta)$: Model

If we had labelled data :

$\{x^{(i)}, y^{(i)}\}_{i=1}^m$

$\theta_{ML} = \arg\max_\theta \prod_{i=1}^m P(x^{(i)}, y^{(i)}; \theta)$

$= \arg\max_\theta \sum_{i=1}^m \log (P(x^{(i)}, y^{(i)}; \theta))$

Our case: (EM)

$\{x^{(i)}\}_{i=1}^m$

$\theta_{ML} = \arg\max_\theta \prod_{i=1}^m P(x^{(i)}; \theta)$

$= \arg\max_\theta \prod_{i=1}^m \left[\sum_{z^{(i)}} P(x^{(i)}, z^{(i)}; \theta)\right]$

$= \arg\max_\theta \sum_{i=1}^m \log \left(\sum_{z^{(i)}} P(x^{(i)}, z^{(i)}; \theta)\right)$

EM:

$L(A)$: if we knew θ's estimate $z^{(i)}$'s

$L(B)$: if we knew $z^{(i)}$'s estimate θ's

Suyash Agrawal

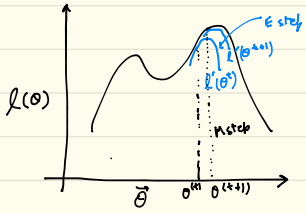
EM (Expectation Maximization):

$\{x^{(i)}\}_{i=1}^m$, hidden : $\{z^{(i)}\}_{i=1}^M$

$$\theta_{ML} = \arg\max_\theta \prod_{i=1}^m P(x^{(i)};\theta)$$
$$= \arg\max_\theta \prod_{i=1}^m \left[\sum_{z^{(i)}} P(x^{(i)}, z^{(i)};\theta)\right]$$
$$= \arg\max_\theta \sum_{i=1}^m \log\left(\sum_{z^{(i)}} P(x^{(i)}, z^{(i)};\theta)\right)$$

$\ell'(\theta)$

$\ell(\theta) \geq \ell'(\theta)$

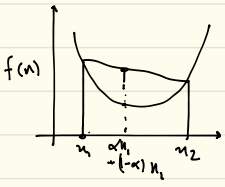


① Jensen's Inequality:

$f: \mathbb{R}^n \to \mathbb{R}$

$f(\alpha x_1 + (1-\alpha)x_2) \leq \alpha f(x_1) + (1-\alpha) f(x_2)$

$P(x)$: over x



if f is convex:

$$f(E_p(x)) \leq E_p[f(x)]$$

if f is concave:

$$f(E_p(x)) \geq E_p(f(x))$$

if f is strictly concave:

$$f(E_p(x)) > E_p(f(x)) \quad \text{unless } \exists c \text{ st } P(X=c) = 1$$

Also,

$$E[f(x)] = f(E(x)) \text{ iff } \exists c,\ P(X=c) = 1$$

$$\ell(\theta) = \sum_{i=1}^m \left[\log\left(\sum_{z^{(i)}} \frac{P(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})} \cdot Q_i(z^{(i)})\right)\right]$$

(log: concave; $\log$ → f, $\sum_{z^{(i)}}$ → E)

where $Q_i(z^{(i)})$ is some distribution over $z^{(i)}$.

$$\geq \sum_{i=1}^m \left[\sum_{z^{(i)}} \left(\log\left(\frac{P(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}\right) Q_i(z^{(i)})\right)\right] = \ell'(\theta)$$

Suyash Agrawal

If the quantity inside the log becomes constant then inequality becomes equality.

$$\frac{P(x^{(i)}, z^{(i)}; \theta)}{Q_i(z^{(i)})} = c$$

$$\Rightarrow \sum_{z^{(i)}} P(x^{(i)}, z^{(i)}; \theta) = c \cdot \sum_{z^{(i)}} Q_i(z^{(i)})$$

$$P(x^{(i)}; \theta) = c \cdot 1$$

$$\therefore c = P(x^{(i)}; \theta)$$

$$Q_i(z^{(i)}) = \frac{P(x^{(i)}, z^{(i)}; \theta)}{P(x^{(i)}; \theta)} = P(z^{(i)} | x^{(i)}; \theta)$$

$Q_i(z^{(i)})$ is filling in missing values (by computing probability for $z^{(i)}$'s)

EM:

Intuition:

① If we know θ's, then we can compute $z^{(i)}$'s :- $Q_i(z^{(i)}) \Leftarrow P(z^{(i)} | x^{(i)}; \theta)$

② If we know $z^{(i)}$'s $| Q_i(z^{(i)})$, then we can compute $\arg\max_\theta \sum_{z^{(i)}} \log\left(\frac{P(x^{(i)}, z^{(i)}; \theta)}{Q_i(z^{(i)})}\right) Q_i(z^{(i)})$
$\equiv \arg\max_\theta l(\theta)$

Formal:

① $t \leftarrow 0$

$\theta^{(t)} \leftarrow$ initParams()

do {

Estep $\forall i$: $Q_i(z^{(i)}) = P(z^{(i)} | x^{(i)}; \theta)$

Mstep $\theta^{(t+1)} = \arg\max_\theta \sum_{i=1}^{m} \sum_{z^{(i)}} Q_{(i)}^{t}(z^{(i)}) \times \log\left(\frac{P(x^{(i)}, z^{(i)}; \theta)}{Q_i^{(t)}(z^{(i)})}\right)$ (constant)

} (until convergence)

Suyash Agrawal

Learning Theory:

$(x^{(i)}, y^{(i)}) \sim D$ → underlying distribution.

$y^{(i)} \in \{0, 1\}$, $h \in H$ → hypothesis space

$\mathcal{E}^{(i)}(h) = 1\{h(x^{(i)}) \neq y^{(i)}\}$

$\hat{\mathcal{E}}(h) = \frac{1}{m} \sum_{i=1}^{m} 1\{h(x^{(i)}) \neq y^{(i)}\}$]⇒ Training Error

$\mathcal{E}(h) = \underset{(x,y)\sim D}{E}[1\{h(x) \neq y\}]$]⇒ generalization error

Goal : Minimize $\mathcal{E}(h)$ ⟶ ① Minimize $\hat{\mathcal{E}}(h)$
② Minimize $|\hat{\mathcal{E}}(h) - \mathcal{E}(h)|$

$P(|\mathcal{E}(h) - \hat{\mathcal{E}}(h)| > \gamma) \leq \delta$

PAC-L ≡ Probably Approximately Correct
① γ → error
② δ → probability of error
③ m
④ $|H|$:- important

Setting : ① Binary Classification
② H is finite {We will generalize later}

Coin Toss:

ϕ: Probability of Head (1)

$\therefore P(Z = 1) = \phi$ {1 ≡ Head}

$Z \sim \text{Bernoulli}(\phi)$

$E(Z) = \phi \cdot 1 + (1-\phi) \cdot 0 = \phi$

$\text{Var}(Z) = E(Z^2) - E^2(Z) = \phi - \phi^2 = \phi(1-\phi)$

Trial 1: 51 H 49 T ⇒ $\hat{\phi} = 0.51$
Trial 2: 45 H 55 T ⇒ $\hat{\phi} = 0.45$
↓ γ times
⋮

$\hat{\phi}$: Random Variable ! {How?}

$\hat{\phi} = \frac{1}{m} \sum_{i=1}^{m} Z^{(i)}$ } Estimator of ϕ,

$E(\hat{\phi}) = E\left(\frac{1}{m}\sum Z^{(i)}\right) = \frac{1}{m}\sum_{i=1}^{m} E(Z^{(i)}) = \frac{\phi \cdot m}{m} = \phi$ } Unbiased (since $E(\hat{\phi}) = \phi$)

↳ we don't need $Z^{(i)}$'s to be independent

$\text{Var}(\hat{\phi}) = \text{Var}\left(\frac{1}{m}\sum_{i=1}^{m} Z^{(i)}\right) = \frac{1}{m^2}\sum_{i=1}^{m} \text{Var}(Z^{(i)}) = \frac{m\phi(1-\phi)}{m^2} = \frac{\phi(1-\phi)}{m}$ {As $m \uparrow$ $\text{Var}(\hat{\phi}) \downarrow$}

Suyash Agrawal

Pac learning:

$Z \sim$ Bernoulli (ϕ)

$\hat{\phi} = \frac{1}{m}\sum_{i=1}^{m} Z^{(i)}$

$E[\hat{\phi}] = \phi$, $Var(\hat{\phi}) = \frac{\phi(1-\phi)}{m}$

$P(|\phi - \hat{\phi}| > \gamma) \leq \underbrace{2 \cdot e^{-2\gamma^2 m}}_{\delta}$] $\Rightarrow$ Chernoff Bound | Hoefdding inequality

$\mathcal{E}(h) = \underset{(x,y)\sim D}{E}[1\{h(x) \neq y\}]$

$Z = 1\{h(x) \neq y\} \in \{0,1\}$

$E_D[Z] = \mathcal{E}(h) = \phi$

$\hat{\phi} = \frac{1}{m}\sum_{i=1}^{m} Z^{(i)} = \hat{\mathcal{E}}(h)$

$\therefore P_r(|\mathcal{E}(h) - \hat{\mathcal{E}}(h)| \geq \gamma) \leq \delta = 2 \cdot e^{-2\gamma^2 m}$] independent of H!

$P_r(\exists h \in H, |\mathcal{E}(h) - \hat{\mathcal{E}}(h)| > \gamma) \leq \delta \equiv P(\forall h \in H, |\mathcal{E}(h) - \hat{\mathcal{E}}(h)| \leq \gamma) \geq 1 - \delta$

Define : $A_k :- |\mathcal{E}(h_k) - \hat{\mathcal{E}}(h_k)| > \gamma$

$\forall k :- P_r(A_k) \leq \delta$

$P_r(\bigcup_k A_k) \leq \sum_{k=1}^{K} P_r(A_k)$

$\therefore P_r(\exists h \in H, |\mathcal{E}(h) - \hat{\mathcal{E}}(h)| > \gamma) \leq \delta = k \cdot 2e^{-2\gamma^2 m}$ } Uniform Convergence Bound

As, $\delta \geq 2k e^{-2\gamma^2 m}$

$\Rightarrow m \geq \frac{1}{2\gamma^2}\log\left(\frac{2k}{\delta}\right)$ or $\gamma \geq \sqrt{\frac{1}{2m}\cdot\log\left(\frac{2k}{\delta}\right)}$

What about best hypothesis?

$\hat{h}^*$:- Best hypothesis on the training set

$\hat{h}^* = \underset{h\in H}{\operatorname{argmin}}\ \hat{\mathcal{E}}(h)$ } Best in practise

$h^* = \underset{h\in H}{\operatorname{argmin}}\ \mathcal{E}(h)$ } Best in principle

$\mathcal{E}(\hat{h}^*) \leq \hat{\mathcal{E}}(\hat{h}^*) + \gamma \leq \hat{\mathcal{E}}(h^*) + \gamma \leq \mathcal{E}(h^*) + 2\gamma$ } with probability atleast $1-\delta$.

($\hat{h}^*$ was best on training)

$\hat{\mathcal{E}}(\hat{h}^*) + \gamma$: due to uniform convergence bound

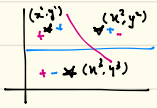
$\hat{\mathcal{E}}(h^*) + \gamma$: using uniform convergence bound

Suyash Agrawal

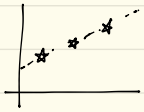
Lecture 42 - 28.04.18

VC Dimension (Vapnik Chervonenkis Dimension):

$y \in \{0,1\}$, $h \in$ set of all hypothesis



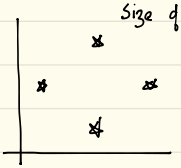
These points are shattered by the space of linear separator



Are not shattered by linear separator. (E.g. config + - +)

VC dimension: "capacity" of a hypothesis space to classify a set of points.

Size of the largest set that can be shattered



$h \in$ linear classifier

$|H|$ is ∞

$\Rightarrow$ These points can't be shattered by a linear separator

Infact, there is no set of 4 points which can be shattered by linear sep.

$h \in$ quadratic classifier

$|H|$ is infinite

These set of points is shattered by a quad. classifier. Even 5 points can be done.

} This can't be shattered by quad. classifier

The game is to find a set of K points s.t. for any possible labelling, we are able to shatter them using our hypothesis.

VC dimension of linear separator is 3, while that of quad. classifier is atleast 5 (in 2-D space)

Definition:

Let H be a hypothesis class.

Let $S = \{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$ where $x^{(i)} \in X \rightarrow$ instance space

Further, let $h \in H$ be boolean valued

$h: X \rightarrow \{0,1\}$, $h(x) \in \{0,1\}$

S is shattered by H iff $\exists\, h \in H$ s.t. h can realize all possible labelling over S.

Let $\ell = \{\ell^{(1)}, \ldots, \ell^{(m)}\}$ be a labelling of S

$\forall\, \ell$, $\exists h \in H$ s.t. $h(x^{(i)}) = \ell^{(i)}$ $\forall i = 1, \ldots, m$

Suyash Agrawal

VC Dimension: Given a hypothesis space H of instance space X, it is the size of largest set S that is shattered by H.

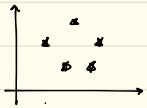
Example:

$H \equiv$ set of all possible axis || rectangles

$X \equiv \mathbb{R}^2$

$h_R \in H$, $h_R(x) = \begin{cases} 1 & \text{if } x \text{ lies inside } R \\ 0 & \text{o.w.} \end{cases}$

4 points can be shattered



5 points can't be shattered

[assign + to topmost, leftmost, rightmost, bottom most & the last one -].

$\therefore$ VC dim = 4

- VC dimension of degree k polynomial over $x \in \mathbb{R}$ $\rightarrow k+1$. ?

PAC bounds:

Finite $|H|$, $k = |H|$

① $m \geq \frac{1}{2\gamma^2} \log\left(\frac{2k}{\delta}\right)$

② $\gamma \geq \sqrt{\frac{1}{2m} \log\left(\frac{2k}{\delta}\right)}$

③ $\delta \geq 2k e^{-2\gamma^2 m}$

Uniform convergence bound.

Infinite $|H|$, $d = VC(H)$

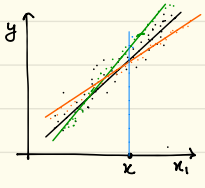
$m \geq \frac{1}{\gamma}\left(4 \log\left(\frac{2}{\delta}\right) + 8d \log\left(\frac{13}{\delta}\right)\right)$

$\therefore m = O_{\gamma,\delta}(d) \Rightarrow$ linear in d

$\gamma \geq O\left(d/m \log m/d + \frac{1}{m} \log 1/\delta\right)$

Suyash Agrawal

- **Bias Variance Tradeoff:**



$x^{(i)} \in \mathbb{R}^n$, $y^{(i)} \in \mathbb{R}$

$$\theta^* = \arg\min_{\theta} \frac{1}{2m} \left[y^{(i)} - h_\theta(x^{(i)})\right]^2$$

$$E_D\left[J(\theta)\right] = E_D\left[(h_\theta(x) - y)^2\right]$$ } expected error on the data point x.

$h_{\bar{\theta}}(x) = E_D\left[h_\theta(x)\right]$

$\left[y - h_{\bar{\theta}}(x)\right] \Rightarrow$ Bias

$E_D\left[\left(h_\theta(x) - h_{\bar{\theta}}(x)\right)^2\right] \Rightarrow$ Variance

$E_D\left[J(\theta)\right] = (\text{Bias})^2 + \text{Variance}$] $\Rightarrow$ Proof?

(Bias: inherent capacity to fit training; Variance: variation caused by different training samples)

Now, high degree poly will have high variance {as slight change in training data can change pred quite a bit}

But, they also tend to have low bias since they are able to learn well.

High bias $\Rightarrow$ underfitting

High variance $\Rightarrow$ overfitting

- **Summary:**

$\{x^{(i)}, y^{(i)}\}_{i=1}^{m} \Rightarrow$ Supervised

Classification

Logistic | GDA / Naive Bayes | SVM | Decision Trees | Neural Nets | Deep Learning

① Hypothesis Space

② Optimization metric / Loss fn

③ Optimization fn