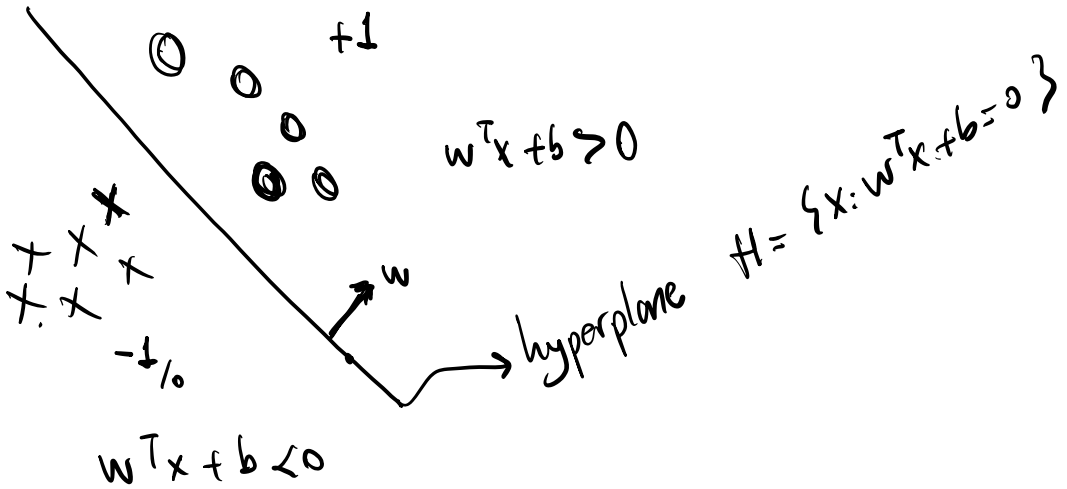
Lecture - 4

Recap for Perceptron.



P ← inputs with label 1
N ← " " " 0
Initialize $\underline{w}$ randomly.
while ! convergence
  Pick random $x \in P \cup N$
  if $x \in P$ and $\sum_{i=0}^{n} w_i x_i < 0$ then
    $w = w + x$
  if $x \in N$ and $\sum_{i=0}^{n} w_i x_i \geq 0$ then
    $w = w - x$

MLE, MAP, Naive Bayes Classifier

Maximum Likelihood Estimation (MLE)

$P(x)$

↳ we don't know P.

$P(H) = ?$

↳ I can sample data.

Toss 10 times $\Rightarrow n = 10$

$D = \{\underbrace{H, T, T, H, H, H, T, T, T, T}_{x_1 \; \cdots \; x_n}\}$

$P(H) = ? \; \frac{4}{10}$ $\quad n_H = 4$, $n_T = 6$ $\Rightarrow$

$$P(H) \approx \frac{n_H}{n_H + n_T} = \frac{4}{10}$$

How can we derive this formula

MLE

$P(D;\theta)$

↳ probability of observing heads.

↳ collected 10 flips.

$$\theta = \underset{\theta}{\arg\max}\ P(D;\theta)$$

Binomial Distribution $= Pr(X=k) = \binom{n}{k} p^k (1-p)^{n-k}$

- fixed number.
- independent Bernoulli trials.
- same probability of success.

$$P(D;\theta) = \binom{n_H + n_t}{n_H} \theta^{n_H} \cdot (1-\theta)^{n_T}$$

$10 = n_{//}$

MLE ⇒ frequentist approach.

$$\hat{\theta}_{MLE} = \arg\max P(D;\theta)$$

$$= \operatorname{argmax} \binom{n_H + n_T}{n_H} \theta^{n_H} (1-\theta)^{n_T}$$

$$= \operatorname{argmax}_{\theta} \log \underbrace{\binom{n_H + n_T}{n_H}}_{\text{constant}} + n_H \cdot \log(\theta) + n_T \cdot \log(1-\theta)$$

$$= \operatorname{argmax}_{\theta} \; n_H \cdot \log(\theta) + n_T \cdot \log(1-\theta)$$

Derivation

$$\frac{n_H}{\theta} - \frac{n_T}{1-\theta} = 0$$

$$\frac{n_H}{\theta} = \frac{n_T}{1-\theta} \Rightarrow n_H - \theta n_H = n_T \theta$$

$$n_H = n_T \theta + n_H \theta$$

$$\boxed{\frac{n_H}{n_H + n_T} = \theta}$$

n=10 → 2M → P(H) = 4/10

- When it fails?

T T T T T T T T

P(H) = 0

Frequentist vs Bayesian

P(D; θ)

↳ parameter

↳ Random variable :)

Simple fix : "2m" imaginary throws that would result in θ' (e.g. 0.5). Add "m" heads and "m" tails to your data.

$$\hat{\theta} = \frac{n_H + m_H}{n_H + n_T + m_H + m_T} = \frac{h_H + m}{n_H + n_T + 2m}$$

$$= \frac{n_H + 1}{n_H + n_T + 2}$$

⇒ for small "u", it incorporates your "prior belief" about what θ should be.

Smoothing

Laplace Smoothing : $P(x) = \frac{count(x) + \alpha}{n + \alpha V}$

Bayesian Approach

it combines prior belief (expressed as prior dist.) with observed data (likelihood.)

to update our belief in the form of posterior distribution.

$P(D;\theta) \Rightarrow P(\theta|D)$

θ: parameter

θ: random variable.

Maximum A Posteriori (MAP) Estimation.

Prior distribution = $P(\theta)$

likelihood function = $P(D|\theta)$ for observed data D.

MAP estimate $\hat{\theta}_{MAP}$ is.

$$\hat{\theta}_{MAP} = \underset{\theta}{\operatorname{argmax}}\ P(\theta|D)$$

Using Bayes Theorem → prior belief

$$\hat{\theta}_{MAP} = \arg\max_{\theta} \frac{P(D|\theta) \cdot P(\theta)}{P(D)}$$

$$P(\theta|D) = \frac{P(D|\theta) \cdot P(\theta)}{P(D)}$$ Bayes theorem.

$P(D)$ normalizer.

$N = \{H, H, T, \ldots\}$ coin toss

$h = 4$

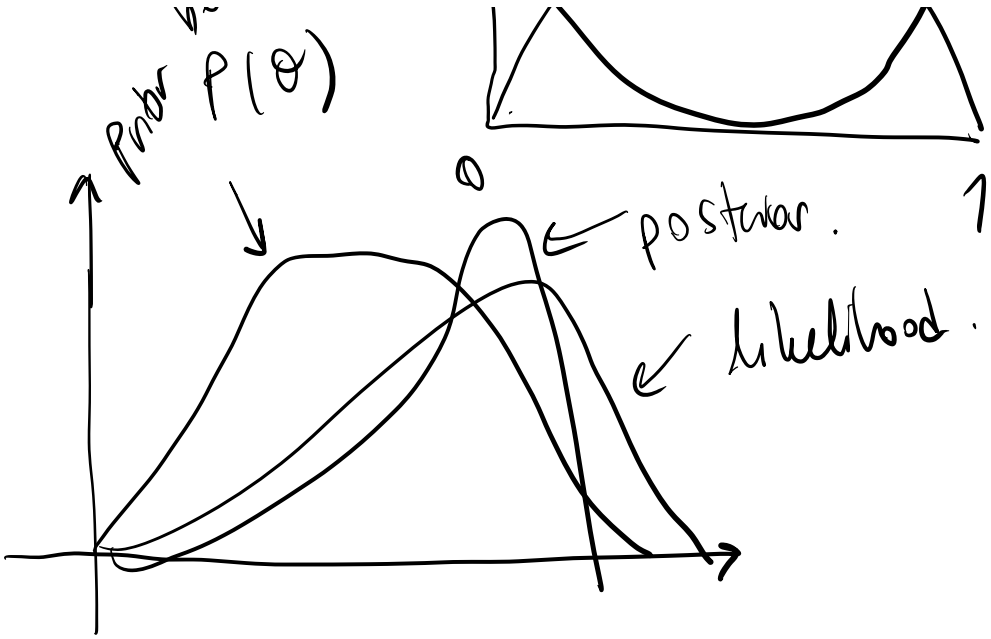
$t = 6$

$$\hat{\theta}_{MAP} = \arg\max_{\theta} P(D|\theta) \cdot P(\theta)$$

Beta distribution.

$[0, 1]$

$$P(\theta) = \frac{\theta^{\alpha-1} \cdot (1-\theta)^{\beta-1}}{B(\alpha, \beta)}$$



MLE

$$\hat{\theta}_{MAP} \quad P(D|\theta) \cdot P(\theta)$$

$$= \binom{n_H + n_T}{n_H} \theta^{n_H} (1-\theta)^{n_T} \cdot \theta^{\alpha-1} (1-\theta)^{\beta-1}$$

$$= \theta^{n_H + \alpha - 1} \cdot (1-\theta)^{n_T + \beta - 1}$$

$$= \hat{\theta}_{MAP} = \frac{n_H + \alpha - 1}{n_H + n_T + (\alpha + \beta - 2)}$$

- The MAP estimate is identical to MLE, with $\alpha - 1$ heads (belief) and $\beta - 1$ tails (belief)

- As $n \to \infty$ $\hat{\theta}_{MAP} \to \theta_{MLE}$ as $\alpha - 1$ and $\beta - 1$ become irrelevant compared to very large n_H, n_T.

Ex/ $D = \{(\vec{x}_1, y_1) \dots (x_n, y_n)\}$

drawn from some unknown $P(X, Y)$

for example (x, y)

$$\hat{P}(x, y) = \frac{\sum_{i=1}^{n} I(x_i = x \wedge y_i = y)}{n}$$

(x = spam email, y = spam)

I need ~~~~

How can we estimate $\hat{P}(y|x) = ?$ (x = email, y = spam)

$$p(y|x) = \frac{P(y, x)}{P(x)} = \frac{\sum_{i=1}^{n} I(x_i = \vec{x} \wedge y_i = y)}{\sum_{i=1}^{n} I(x_i = x)}$$

($\vec{x}$ = ?)

$$P(y = y \mid X_1 = x_1, X_2 = x_2 \dots X_d = x_d)$$

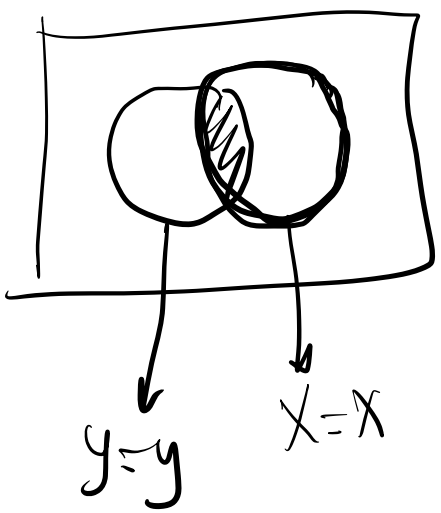
MLE is only good if there are many example (training vector) with the

same identical features of X!

In high dimensional spaces it never happens

$P(X)$, Dr $P(Y,X)$ → single feature.

$P(Y=y \mid X=x)$



$X=x$ / $y=y$

$$\frac{P(Y=y, X=x)}{P(X)}$$

↓

$$\frac{\sum_{i=1}^{n} I\left(x_i = x \wedge y_i = y\right)}{\sum_{i=1}^{n} I\left(x_i = x\right)}$$

- how many times do we see X and out of those how many times be also see $y = y$.

In high dimension.

$$P\left(Y = y \mid X_1 = x_1, X_2 = x_2 \dots X_d = x_d\right)$$

- how many times, I exactly see the same data point in my dataset?

Naive Bayes

→ Features are independent given the label.

n features, $X = (x_1 \dots x_n)$

likelihood

$$P(C_k|X) = \frac{P(X|C_k) \times P(C_k)}{P(X)}$$

(C_k: $P(C_k)$)

feature independence assumption

n-dimension space

$$P(X|C_k) = P(x_1, x_2, x_3, x_4 \dots x_n | C_k)$$

$$\approx P(x_1|C_k) \times P(x_2|C_k) \times \dots \times P(x_n|C_k)$$

1st p

Plugging into Bayes' Theorem.

$$P(C_K|X) \propto \frac{P(X_1|C_K) \times P(X_2|C_K) \cdots \times P(X_n|C_K) \times P(C_K)}{P(X)}$$

Classification Decisions

- For a given obs. X, we compute $P(C_K|X)$ for each class C_K, and assign the obs. to the class that max. value:

$$\hat{C}(X) = \operatorname{argmax} P(C_K) \times P(X_1|C_K) \times P(X_2|C_K) \cdots P(X_n|C_K)$$

$$P(\text{spam} \mid \vec{X}) =$$

$$P(\neg \text{spam} \mid \vec{X}) =$$

Example.

features ↑ label ↓

	outlook	temperature	humidity	wind	play football
1	Sunny	hot	high	weak	N
2	S	h	H	strong	N
3	overcast	h	H	W	Y
4	rainy	mild	H	W	Y
5	R	Cold	normal	W	Y
6	R	C	N	S	N
7	O	C	N	S	Y
8	S	M	High	W	N
9	S	C	N	W	Y
10	R	M	N	W	Y
11	S	M	N	S	Y
12	O	M	H	S	Y
13	O	H	N	W	Y
14	R	M	H	S	N

test data ↓

X : { outlook : sunny,
temper : mild,
humidity : normal,
windy : weak
}

$y = ?$ Y/N

Yes

$P(\text{yes} \mid X) \propto P(X \mid y) \cdot P(y)$

$\propto \underbrace{P(x_1 \mid y)} \cdot P(x_2 \mid y) \cdot P(x_3 \mid y) \cdot P(x_4 \mid y) \cdot P(y)$

$\propto P(\text{sunny} \mid \text{yes}) \cdot P(\text{mild} \mid y) \cdot P(\text{normal} \mid y) \cdot P(\text{weak} \mid y) \cdot P(y)$

$2/9 \cdot 4/9 \cdot 6/9 \cdot 6/9 \cdot 9/14$

$\simeq 0.02$

$P(NO \mid X) \simeq P(Sunny \mid N) \cdot P(Hi\,gh \mid N) \cdot P(Normal \mid N) \cdot P(weak \mid N) \cdot P(N)$

$\Rightarrow P(no \mid x) \simeq 0.006$

$P(yes \mid x) > P(no \mid x)$

$X = (x_1, \ldots x_d)$ → d features , Y is the label $\{0,1\}$

$\underset{y}{\text{argmax}}\ P(Y=y \mid X=(x_1 \ldots x_d))$

Bayes theorem $\Rightarrow P(Y \mid X) = \frac{P(X \mid Y) \cdot P(Y)}{P(X)}$

$X = (Outlook, Temp, humidity, windy)$ and Y=play

$$P(Y = \text{play}) = \frac{\#\,\text{play}}{\#\,\text{play} + \#\,\neg\text{play}} = \frac{9}{14}$$

↓

$$p(X = \text{sunny} \mid Y = \text{play}) = \frac{\#\,\text{play} \;\&\; \#\,\text{sunny}}{\#\,\text{play}}$$

$$= 2/9$$

$\Rightarrow P(X_1 = 1, X_2 = 2 \text{ ---- } \mid Y = \text{play})$

→ think all features are independent!

$$P(X_1, X_2 \mid Y) = P(X_1 \mid Y) \cdot P(X_2 \mid Y)$$

conditional independence.

$$P(Y \mid X) = \frac{\prod_{i=1}^{d} P(X_i \mid Y) \cdot P(Y)}{P(X)}$$

$$P(x|y) = P(x_1|y) \times P(x_2|y) \times \cdots \times P(x_n|y).$$

$$P(y|x) \propto \prod_{i=1}^{d} P(x_i|y) \times P(y)$$

Types of NB Classifiers.

- Bernoulli NB → $X = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}$
- Multinomial NB → $\begin{bmatrix} 5 \\ 2 \\ 3 \\ 2 \\ 1 \\ \vdots \end{bmatrix}$
- Gaussian Naive Bayes.

t_1	label
125	N
100	N
70	N
120	Y
95	N
60	N
220	Y
85	N
75	N
90	Y

Test data

$X = 120$

$Y = ?$

$\mu = 110$

$\sigma^2 = 2975$

$$P(x|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_i)^2}{2\sigma_y^2}\right)$$

$P(\text{yes} \mid 120)$?

$$P(\text{No} \mid 120) = \frac{1}{\sqrt{2\pi 2975}} \exp\left(-\frac{(120-110)^2}{2 \times 2975}\right)$$

Adv / Disadv. of ND

Adv

- Simplicity
- Scalability.
  - fast training and

- Fast training algorithm
- Online updates possible

- Works with high dimensionals.
  ↳ text classification problems.
- Low training work well.
- probabilistic output.

Disadvantages

1- Naive assumption

2 - Data Distribution Ass.:

Gaussion, Bernolli, ...

3 - Zero Frequcy problem.
