Lecture 6

Perceptron
LDA
Logistic Regression

Everything up to Lecture L7 - jan 29
EXCEPT gradient descent

HW1 - ~~1/28~~ 1/29 due (with new RNG fix)
HW3 - 1/28 out
2/4 due
Tue 2/10 Solutions

Quiz 1 2/12

LTI Linear

Lecture II: Linear regression and classification. Loss functions

Marina Meilă
mmp@uwaterloo.ca

With Thanks to Pascal Poupart & Gautam Kamath
Cheriton School of Computer Science
University of Waterloo

January 12, 2026

Linear predictors generalities

Loss functions

Least squares linear regression

- Linear regresssion as minimizing L_{LS}
- Linear regresssion as maximizing likelihood
- Linear Discriminant Analysis (LDA)
- QDA (Quadratic Discriminant Analysis)
- Logistic Regression
- The PERCEPTRON algorithm

Reading HTF Ch.: 2.1–5,2.9, 7.1–4 bias-variance tradeoff, Murphy Ch.: 1., 8.6[1], Bach Ch.:

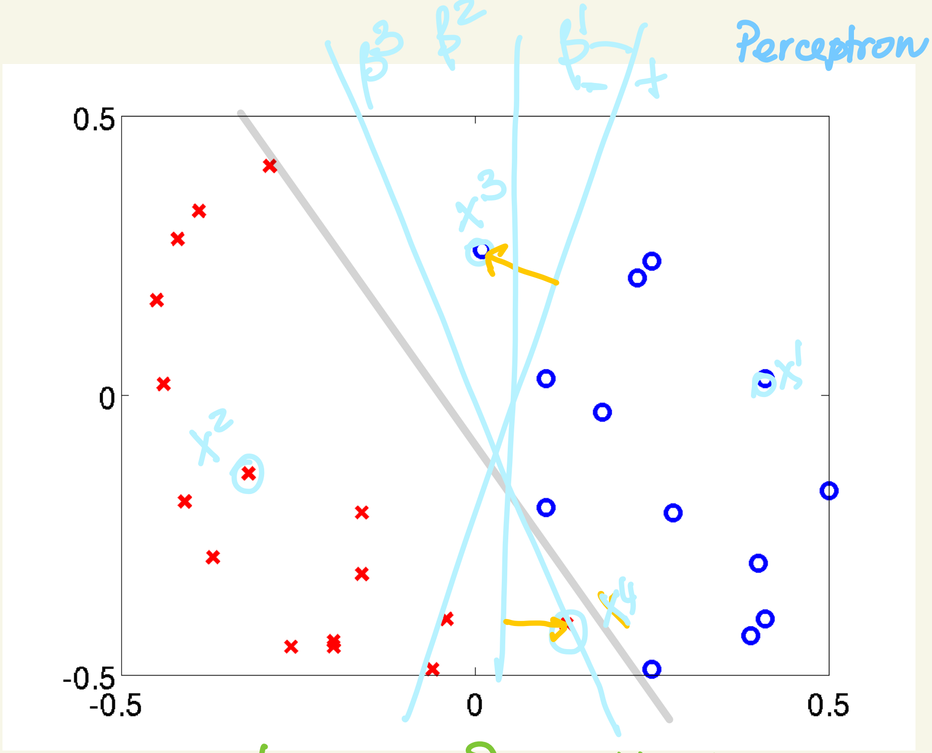
[1] Neither textbook is close to these notes except in a few places; take them as alternative perspectives or related reading

Linear classifier – linear decision bdary

Losses – L_{01} misclassification error → Perceptron ①

– Statistical loss ≈ Max Likelihood → Linear Discriminant Analysis LDA ②

– ——"—— L_{logit} → Logistic Regression ③



Linearly separable D ← training set

x^3 = mistake

β^3 : $L^{train}(\beta^3) = 0$

D separable: Alg ends in finite # steps

D NOT separable: No convergence

The PERCEPTRON algorithm

Fitting a linear predictor for classification, third approach.
Define $f(x) = \beta^T x$ and find β that classifies all the data correctly (when possible).
PERCEPTRON **Algorithm**

Input labeled training set $\mathcal{D}$

Initialize $\beta = 0$, for all i, $x^i \rightarrow \frac{x^i}{||x^i||}$ (normalize the inputs)

Repeat until no more mistakes

for $i = 1 : N$

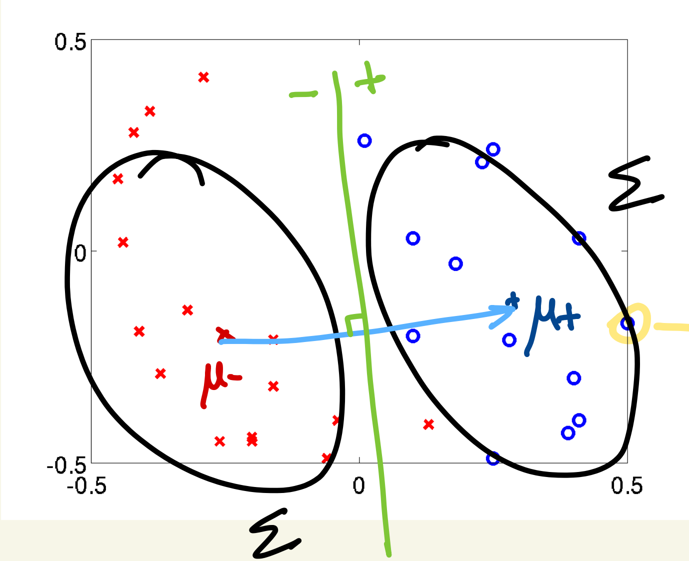
1. if $\mathrm{sgn}(\beta^T x^i) \neq y^i$ (a mistake)

   $\beta \leftarrow \beta + y^i x^i$

(Other variants exist)

LDA

Class + : μ_+
Class − : μ_- } $\Sigma = \Sigma_+ = \Sigma_-$



$\pi_+ = \frac{\#\{y^j = +\}}{n}$, $\pi_- = 1 - \pi_+$

1. Estimate from $\mathcal{D}$

2. New x want $P[y=+ \mid x]$ by Bayes' Rule

3. Predict $\hat{y} = + \Leftrightarrow P[y=+ \mid x] \geq \frac{1}{2}$

$\Updownarrow$

$P[y=+ \mid x] \geq P[y=- \mid x]$

Analysis

Generative model for classification (NOT G.M. for unsupervised)

$y^i \sim$ iid (π_+, π_-)

$\nearrow$ $y^i = +$, $x^i \sim N(\mu_+, \Sigma)$

$\searrow$ $y^i = -$, $x^i \sim N(\mu_-, \Sigma)$

simplify $\Sigma = \sigma^2 I$

Bayes

$P[y=+ \mid x] = \frac{\pi_+ e^{-\|x-\mu_+\|^2/2\sigma^2}}{Z}$

$P[y=- \mid x] = \frac{\pi_- e^{-\|x-\mu_-\|^2/2\sigma^2}}{Z}$ ← Ex

for what x : $\hat{y} = +$?

STATISTICS

↳ CALCULUS

1. ln

2. $\|x-\mu\|^2$

$$P[y=+\mid x] = \frac{\pi_+ e^{-\|x-\mu_+\|^2/2\sigma^2}}{Z}$$

$$P[y=-\mid x] = \frac{\pi_- e^{-\|x-\mu_-\|^2/2\sigma^2}}{Z}$$

1. $\hat{y}(x) = +$ iff $\ln \pi_+ - \frac{1}{2\sigma^2}\|x-\mu_+\|^2 \geq \ln \pi_- - \frac{1}{2\sigma^2}\|x-\mu_-\|^2$

2. $\|x-\mu\|^2 = x^T x + \mu_\pm^T \mu_\pm - 2\mu_\pm^T x$

$\|\mu\|^2$

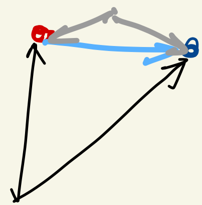
3. $+2\mu_+^T x - 2\mu_-^T x \geq -\ln \frac{\pi_+}{\pi_-}(2\sigma^2) - \|\mu_-\|^2 + \|\mu_+\|^2$

$$(\mu_+ - \mu_-)^T x \geq \sigma^2 \ln \frac{\pi_+}{\pi_-} + \frac{\|\mu_+\|^2 - \|\mu_-\|^2}{2}$$

β

linear

0 for $\pi_+ = \pi_-$



The perceptron algorithm and linearly separable data

- $\mathcal{D}$ is **linearly separable** iff there is a β_* so that $\operatorname{sgn}\beta_*^T x^i = y^i$ for all $i = 1, \dots N$.
  If one such β_* exists, then there are an infinity of them

Theorem

Let $\mathcal{D}$ be a linearly separable data set, and define

$$\gamma = \min_i \frac{|\beta_*^T x^i|}{||\beta_*||||x^i||}. \quad (39)$$

Then, the number of mistakes made by the PERCEPTRON *algorithm is at most $1/\gamma^2$.*

- Note that if we scale the examples to have norm 1, then γ is the minimum distance to the hyperplane $\beta_*^T x = 0$ in the data set.
  Exercise Show that if $\mathcal{D}$ is linearly separable, the scaling $x^i \rightarrow \frac{x^i}{||x^i||}$ leaves it linearly separable.
- If $\mathcal{D}$ is not linearly separable, the algorithm oscillates indefinitely.

Linear Discriminant Analysis (LDA)

Fitting a linear predictor for classification, first approach. (We are in the binary classification case)

- Assume each class is generated by a Normal distribution

$$P_{X|Y}(x|+) = \mathcal{N}(x;\mu_+,\Sigma_+), \quad P_{X|Y}(x|-) = \mathcal{N}(x;\mu_-,\Sigma_-) \quad \text{and } P_Y(1) = p$$

- Given x, what is the probability it came from class $+$?

$$P_{Y|X}(+|x) \;=\; \frac{P_Y(1)P_{X|Y}(x|+)}{P_Y(1)P_{X|Y}(x|+) + P_Y(-)P_{X|Y}(x|+-)} \quad \text{and} \quad P_{Y|X}(-|x) \;=\; 1-P_{Y|X}(+|x) \quad (19)$$

  This formula is true whether the distributions $P_{X|Y}$ are normal or not.
- We assign x to the class with maximum posterior probability.

$$\hat{y}(x) \;=\; \operatorname*{argmax}_{y\in\{\pm 1} P_{Y|X}(y|x) \quad (20)$$

  This too, holds true whether the distributions $P_{X|Y}$ are normal or not.

LDA – continued

Now we specialize to the case of normal class distribution. We assume in addition that $\Sigma_+ = \Sigma_- = K^{-1}$.

- Decision rule: $\hat{y} = 1$ iff $P_{Y|X}(+|x) > P_{Y|X}(-|x)$
- or equivalently iff

$$
\begin{aligned}
0 \leq f(x) &= \ln \frac{P_{Y|X}(+|x)}{P_{Y|X}(-|x)} && (21)\\
&= \ln \frac{p}{1-p} - \frac{1}{2}\left[x^T K x - 2\mu_+^T K x + \mu_+^T K \mu_+\right] \\
&\quad -\frac{1}{2}\left[x^T K x - 2\mu_-^T K x + \mu_-^T K \mu_-\right] && (22)\\
&= \left[K(\mu_+ - \mu_-)\right]^T x + \ln \frac{p}{1-p} + \frac{\mu_-^T K \mu_- - \mu_+^T K \mu_+}{2} && (23)\\
&= \beta^T x + \beta_0 && (24)
\end{aligned}
$$

- The above is a linear expression in x, hence this classifier is called **(Fisher's) Linear Discriminant**
- Note that if we change the variables to $x \leftarrow \sqrt{K}x$, $\mu_\pm \leftarrow \sqrt{K}\mu_\pm$, and if we shift the origin to $\frac{\mu_+ + \mu_-}{2}$ (24) becomes

$$2\mu_+^T x + \ln \frac{p}{1-p} \qquad (25)$$

This has a geometric interpretation

LDA Algorithm

LDA Algorithm

Train
1. Estimate μ_+ from data $\{(x^i, y^i),\ y^i = +1\}$
2. Estimate μ_- from data $\{(x^i, y^i),\ y^i = -1\}$
3. Estimate Σ jointly for both classes, calculate $K = \Sigma^{-1}$. Exercise Derive the formula for this estimate, in the Max Likelihood setting
4. Estimate $p = |\{(x^i, y^i),\ y^i = +1\}|/n$.

Predict Now apply (24) to classify new data x

Logistic Regression

Fitting a linear predictor for classification, another approach.
Let $f(x) = \beta^T x$ model the **log odds** of class 1

$$f(X) = \ln \frac{P(Y=1|X)}{P(Y=-1|X)} \quad (31)$$

$P(Y=1|X)$ → p_x; ← log-odds

Then

- $\hat{y} = 1$ iff $P(Y=1|X) > P(Y=-1|X)$
  - just like in the previous case! so what's the difference?

1. Max Likelihood

x → f(x); x → p (training point)

$$f = \ln \frac{p}{1-p} \Rightarrow p = ? \quad \text{Ex}$$

$$p = \frac{e^f}{1+e^f} = P[y=+|x]$$

$$1-p = \frac{1}{1+e^f} = P[y=-|x]$$

$y_* = 1 \Leftrightarrow y = +1$

$0 \Leftrightarrow y = -1$

$$P[y_*^i | x^i] = \frac{e^{y_* f}}{1+e^f} \quad (f \leftarrow \beta)$$

2. Likelihood

$$L(\beta) = \prod_{i=1}^{n} P[y_*^i \mid x^i]$$

$$\ell(\beta) = \sum_{i=1}^{n} \ln P[y_*^i \mid x^i] = \sum_{i=1}^{n} \left[y_*^i f(x^i) - \ln\left(1 + e^{f(x^i)}\right) \right]$$

β

STAT

Calculus + Opt.

$$\arg\max_{\beta} L(\beta) = \hat{\beta}$$

$\beta \in \mathbb{R}^d$

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

$\nabla(a^T z) = a$

3. $\nabla \ell \equiv \partial \ell / \partial \beta$

$n = 1 \qquad (x, y) : \ell = y_* f - \ln(1 + e^f)$

$$\mathbb{R}^d \ni \frac{\partial \ell}{\partial \beta} = \frac{\partial \ell}{\partial f} \cdot \frac{\partial f}{\partial \beta}$$

$\mathbb{R}$ $\qquad \mathbb{R}^d$

$$\frac{\partial \ell}{\partial f} = y_* - \frac{e^f}{1 + e^f}$$

$p = P[y = 1 \mid x]$

$$\Rightarrow \frac{\partial \ell}{\partial \beta} = \nabla_\beta \ell = \left(y_* - \frac{e^f}{1 + e^f} \right) x$$

$w \in \mathbb{R}$

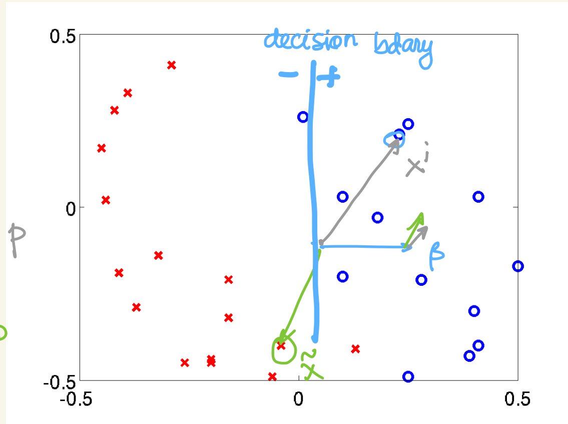
$y_* = 1 \Rightarrow w = 1 - p$

$\tilde{y}_* = 0 \Rightarrow w = -p$

$$\frac{\partial f}{\partial \beta} \equiv \nabla_\beta f = \nabla_\beta (x^T \beta) = x$$

$n > 1 \qquad \nabla_\beta \ell = \sum_{i=1}^{n} \left(y_*^i - \frac{e^{f(x^i)}}{1 + e^{f(x^i)}} \right) x^i$

w_i



Logistic Regression

Fitting a linear predictor for classification, another approach.
Let $f(x) = \beta^T x$ model the **log odds** of class 1

$$f(X) \;=\; \frac{P(Y=1|X)}{P(Y=-1|X)} \tag{31}$$

Then

- $\hat{y} = 1$ iff $P(Y=1|X) > P(Y=-1|X)$
  - just like in the previous case! so what's the difference?
  - Answer: We don't assume each class is Gaussian, so we are in a more general situation than LDA
- What is $p(x) = P(Y=1|X=x)$ under our linear model?

$$\ln\frac{p}{1-p} \;=\; f, \quad \frac{p}{1-p} \;=\; e^f, \quad p \;=\; \frac{e^f}{1+e^f} \quad 1-p \;=\; \frac{1}{1+e^f} \tag{32}$$

An alternative "symmetric" expression for $p, 1-p$ is

$$p \;=\; \frac{e^{f/2}}{e^{f/2}+e^{-f/2}}, \quad 1-p \;=\; \frac{e^{-f/2}}{e^{f/2}+e^{-f/2}}. \tag{33}$$

Estimating the parameters by Max Likelihood

- Denote $y_* = (1-y)/2 \in \{0,1\}$
- The likelihood of a data point is $P_{Y|X}(y|x) = \frac{e^{y_* f(x)}}{1+e^{f(x)}}$
- The log-likelihood is $l(\beta; x) = y_* f(x) - \ln(1+e^{f(x)})$
- $\frac{\partial l}{\partial f} = y_* - \frac{e^f}{1+e^f} = y_* - \frac{1}{1+e^{-f}}$
  This is a scalar, and $\operatorname{sgn}\frac{\partial l}{\partial f} = y$
- We have also $\frac{\partial f(x)}{\partial \beta} = x$
- Now, the gradient of l w.r.t the parameter vector β is

$$\frac{\partial l}{\partial \beta} = \frac{\partial l}{\partial f}\frac{\partial f}{\partial \beta} = \left(y_* - \frac{1}{1+e^{-f(x)}}\right)x \tag{34}$$

Interpretation: The infinitezimal change of β to increase log-likelihood for a single data point is along the direction of x, with the sign of y

Estimating the parameters by Max Likelihood

- Denote $y_* = (1-y)/2 \in \{0,1\}$
- The likelihood of a data point is $P_{Y|X}(y|x) = \frac{e^{y_* f(x)}}{1+e^{f(x)}}$
- The log-likelihood is $l(\beta;x) = y_* f(x) - \ln(1+e^{f(x)})$
- $\frac{\partial l}{\partial f} = y_* - \frac{e^f}{1+e^f} = y_* - \frac{1}{1+e^{-f}}$
  This is a scalar, and $\operatorname{sgn}\frac{\partial l}{\partial f} = y$
- We have also $\frac{\partial f(x)}{\partial \beta} = x$
- Now, the gradient of l w.r.t the parameter vector β is

$$\frac{\partial l}{\partial \beta} = \frac{\partial l}{\partial f}\frac{\partial f}{\partial \beta} = \left(y_* - \frac{1}{1+e^{-f(x)}}\right)x \tag{34}$$

  Interpretation: The infinitezimal change of β to increase log-likelihood for a single data point is along the direction of x, with the sign of y
- Log-likelihood of the data set $\mathcal{D}$

$$l(\beta;\mathcal{D}) = \frac{1}{N}\sum_{i=1}^{d} l(\beta;(x^i,y^i)) \tag{35}$$

- The optimal β maximizes $l(\beta;\mathcal{D})$ and therefore

$$\frac{\partial l(\beta;\mathcal{D})}{\partial \beta} = \frac{1}{N}\sum_{i=1}^{d}\left(y_*^i - \frac{1}{1+e^{-f(x^i)}}\right)x^i = 0 \tag{36}$$

- Unfortunately, (36) does not have a closed form solution!
  We maximize the (log)likelihood by iterative methods (e.g. gradient ascent) to obtain the β of the classifier.

The gradient – an alternative formula

- We use the original y values instead of y_*
- Note that

$$P_{Y|X}(y|x) = \frac{1}{1 + e^{-yf(x)}} = \phi(yf(x)) \tag{37}$$

- with $\phi' = \phi(1-\phi)$
- Then, $\frac{\partial \ln P_{Y|X}(y|x)}{\partial f} = \frac{\partial \ln \phi(yf)}{\partial f} = \frac{y\phi(yf)(1-\phi(yf))}{\phi(yf)} = y(1-\phi(yf))$
- The gradient of the log-likelihood of the data is now

$$\frac{\partial l(\beta;\mathcal{D})}{\partial \beta} = \frac{1}{N}\sum_{i=1}^{d}\left(1 - \underbrace{\phi(e^{yf(x^i)})}_{P_{Y|X}(y_i|x^i,\beta)}\right) y_i x^i \tag{38}$$