CSCI567 Machine Learning (Spring 2021)

Sirisha Rambhatla

University of Southern California

Feb 10, 2021

Outline

1. Logistics
2. Review of last lecture
3. Linear Discriminant Analysis and Quadratic Discriminant Analysis
4. Relationship between Logistic Regression and LDA

Outline

1. Logistics
2. Review of last lecture
3. Linear Discriminant Analysis and Quadratic Discriminant Analysis
4. Relationship between Logistic Regression and LDA

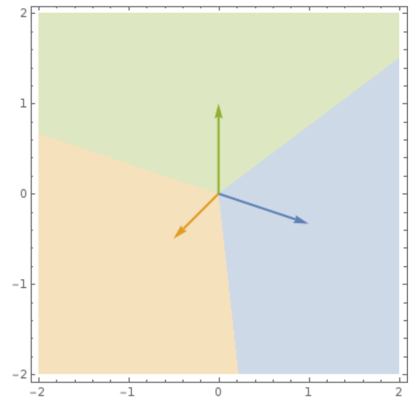
Logistics

- HW 2 was assigned. Solutions for HW 1 will be delayed, stay tuned!
- Please form the groups by Friday, let us know if cannot find a group.

Outline

1. Logistics
2. Review of last lecture
3. Linear Discriminant Analysis and Quadratic Discriminant Analysis
4. Relationship between Logistic Regression and LDA

Linear models: from binary to multiclass



$\boldsymbol{w}_1 = (1, -\frac{1}{3})$
$\boldsymbol{w}_2 = (-\frac{1}{2}, -\frac{1}{2})$
$\boldsymbol{w}_3 = (0, 1)$

- Blue class: $\{\boldsymbol{x} : 1 = \operatorname{argmax}_k \boldsymbol{w}_k^{\mathrm{T}}\boldsymbol{x}\}$
- Orange class: $\{\boldsymbol{x} : 2 = \operatorname{argmax}_k \boldsymbol{w}_k^{\mathrm{T}}\boldsymbol{x}\}$
- Green class: $\{\boldsymbol{x} : 3 = \operatorname{argmax}_k \boldsymbol{w}_k^{\mathrm{T}}\boldsymbol{x}\}$

$$\mathcal{F} = \left\{ f(\boldsymbol{x}) = \operatorname*{argmax}_{k \in [\mathsf{C}]} \; \boldsymbol{w}_k^{\mathrm{T}}\boldsymbol{x} \mid \boldsymbol{w}_1, \ldots, \boldsymbol{w}_{\mathsf{C}} \in \mathbb{R}^{\mathsf{D}} \right\}$$

Softmax + MLE = minimizing cross-entropy loss

Maximize probability of see labels $y_1, \ldots, y_{\mathsf{N}}$ given $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{\mathsf{N}}$

$$P(\boldsymbol{W}) = \prod_{n=1}^{\mathsf{N}} \mathbb{P}(y_n \mid \boldsymbol{x}_n; \boldsymbol{W}) = \prod_{n=1}^{\mathsf{N}} \frac{e^{\boldsymbol{w}_{y_n}^{\mathrm{T}}\boldsymbol{x}_n}}{\sum_{k \in [\mathsf{C}]} e^{\boldsymbol{w}_k^{\mathrm{T}}\boldsymbol{x}_n}}$$

By taking **negative log**, this is equivalent to minimizing

$$F(\boldsymbol{W}) = \sum_{n=1}^{\mathsf{N}} \ln \left(\frac{\sum_{k \in [\mathsf{C}]} e^{\boldsymbol{w}_k^{\mathrm{T}}\boldsymbol{x}_n}}{e^{\boldsymbol{w}_{y_n}^{\mathrm{T}}\boldsymbol{x}_n}} \right) = \sum_{n=1}^{\mathsf{N}} \ln \left(1 + \sum_{k \neq y_n} e^{(\boldsymbol{w}_k - \boldsymbol{w}_{y_n})^{\mathrm{T}}\boldsymbol{x}_n} \right)$$

This is the *multiclass logistic loss*, a.k.a *cross-entropy loss*.

Comparisons of multiclass-to-binary reductions

In big O notation,

Reduction	**#training points**	**test time**	**Idea**
OvA	CN	C	is class k or not?
OvO	CN	C^2	is class k or class k'?
ECOC	LN	L	is bit b on or off?
Tree	$(\log_2 \mathsf{C})\mathsf{N}$	$\log_2 \mathsf{C}$	belong to which half of the label set?

Outline

Revisiting Bayes optimal classifier

Tells us what to predict for $\boldsymbol{x}$, *knowing $\mathcal{P}(y|\boldsymbol{x})$*

Bayes optimal classifier: $f^*(\boldsymbol{x}) = \text{argmax}_{c \in [C]} \mathcal{P}(c|\boldsymbol{x})$.

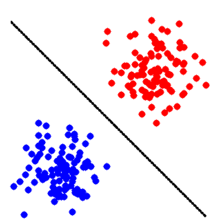
But the main issue was that in practice we don't know what $\mathcal{P}(y|\boldsymbol{x})$ is!

What do we know?

Ok, so we know that by **Bayes theorem** for a C class classification task

$$\mathcal{P}(y = c|X = \boldsymbol{x}) = \frac{\mathcal{P}(X = \boldsymbol{x}|y = c)\mathcal{P}(y = c)}{\mathcal{P}(X = \boldsymbol{x})}$$

Let's consider a Binary Classification task, $C = \{0, 1\}$. *What is the decision boundary?*



$$\mathcal{P}(y = 0|X = \boldsymbol{x}) = \mathcal{P}(y = 1|X = \boldsymbol{x})$$

The main bottleneck is *not knowing $\mathcal{P}(X = \boldsymbol{x}|y = c)$*

$$\mathcal{P}(y = c|X = \boldsymbol{x}) = \frac{\mathcal{P}(X = \boldsymbol{x}|y = c)\mathcal{P}(y = c)}{\mathcal{P}(X = \boldsymbol{x})}.$$

LDA makes **two simplifying assumptions**:

- Let $\mathcal{P}(X = \boldsymbol{x}|y = c) \sim \mathcal{N}(\boldsymbol{\mu}_c, \Sigma_c)$, and
- Let all class covariances be the same i.e. $\Sigma_c = \Sigma$ for all $c \in [C]$

If so, the *decision boundary* (for binary classification) is given by

$$\mathcal{P}(y = 0|X = \boldsymbol{x}) = \mathcal{P}(y = 1|X = \boldsymbol{x})$$

$$\frac{\mathcal{P}(X = \boldsymbol{x}|y = 0)\mathcal{P}(y = 0)}{\mathcal{P}(X = \boldsymbol{x})} = \frac{\mathcal{P}(X = \boldsymbol{x}|y = 1)\mathcal{P}(y = 1)}{\mathcal{P}(X = \boldsymbol{x})}$$

The main bottleneck is *not knowing* $\mathcal{P}(X = \boldsymbol{x}|y = c)$

$$\mathcal{P}(y = c|X = \boldsymbol{x}) = \frac{\mathcal{P}(X = \boldsymbol{x}|y = c)\mathcal{P}(y = c)}{\mathcal{P}(X = \boldsymbol{x})}.$$

LDA makes **two simplifying assumptions**:

- Let $\mathcal{P}(X = \boldsymbol{x}|y = c) \sim \mathcal{N}(\boldsymbol{\mu}_c, \Sigma_c)$, and
- Let all class covariances be the same i.e. $\Sigma_c = \Sigma$ for all $c \in [C]$

If so, the *decision boundary* (for binary classification) is given by

$$\mathcal{P}(y = 0|X = \boldsymbol{x}) = \mathcal{P}(y = 1|X = \boldsymbol{x})$$

$$\mathcal{P}(X = \boldsymbol{x}|y = 0)\mathcal{P}(y = 0) = \mathcal{P}(X = \boldsymbol{x}|y = 1)\mathcal{P}(y = 1)$$

Now, $\mathcal{P}(X = \boldsymbol{x}|y = 0) \sim \mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0)$ **and** $\mathcal{P}(X = \boldsymbol{x}|y = 1) \sim \mathcal{N}(\boldsymbol{\mu}_1, \Sigma_1)$**.**

For $\boldsymbol{\mu}_c \in \mathbb{R}^d$ and $\Sigma_c^{-1} \in \mathbb{R}^{d \times d}$, we have

$$\mathcal{P}(X = \boldsymbol{x}|y = c) = \frac{1}{(2\pi)^{d/2}|\Sigma_c|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_c)^\top \Sigma_c^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_c)\right)$$

The *decision boundary* is given by

$$\mathcal{P}(X = \boldsymbol{x}|y = 0)\mathcal{P}(y = 0) = \mathcal{P}(X = \boldsymbol{x}|y = 1)\mathcal{P}(y = 1).$$

Substituting for $\mathcal{P}(X = \boldsymbol{x}|y = c)$, and taking $\log(\cdot)$ and simplifying[1]

$$\log\left(\frac{\mathcal{P}(y = 0)}{\mathcal{P}(y = 1)}\right) - \frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_0)^\top \Sigma_0^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_0) = -\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_1)^\top \Sigma_1^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_1)$$

For simplicity of this exposition we have ignored the $1/(2\pi)^{d/2}|\Sigma_c|^{1/2}$ terms since these will just add to the constants.

Let's simplify this

$$\log \frac{\mathcal{P}(y=0)}{\mathcal{P}(y=1)} - \frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_0)^\top \Sigma_0^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_0) = -\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_1)^\top \Sigma_1^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_1)$$

$$\log \frac{\mathcal{P}(y = 0)}{\mathcal{P}(y = 1)} - \frac{1}{2}\boldsymbol{x}^\top \Sigma_0^{-1}\boldsymbol{x} + \boldsymbol{\mu}_0^\top \Sigma_0^{-1}\boldsymbol{x} - \frac{1}{2}\boldsymbol{\mu}_0^\top \Sigma_0^{-1}\boldsymbol{\mu}_0 = -\frac{1}{2}\boldsymbol{x}^\top \Sigma_1^{-1}\boldsymbol{x} + \boldsymbol{\mu}_1^\top \Sigma_1^{-1}\boldsymbol{x} - \frac{1}{2}\boldsymbol{\mu}_1^\top \Sigma_1^{-1}\boldsymbol{\mu}_1$$

Setting $\Sigma_0 = \Sigma_1 = \Sigma$,

$$\boldsymbol{w}^\top \boldsymbol{x} + w_0 = 0$$

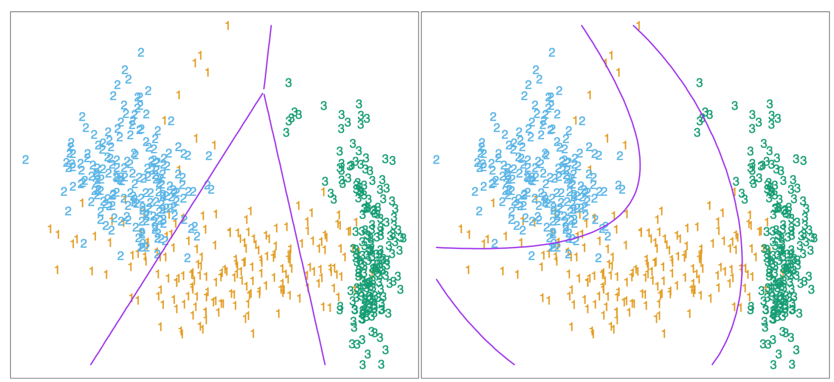
Here,

$$\boldsymbol{w}^\top = \boldsymbol{\mu}_0^\top \Sigma^{-1} - \boldsymbol{\mu}_1^\top \Sigma^{-1}$$

and

$$w_0 = \log \frac{\mathcal{P}(y = 0)}{\mathcal{P}(y = 1)} - \frac{1}{2}\boldsymbol{\mu}_0^\top \Sigma^{-1}\boldsymbol{\mu}_0 + \frac{1}{2}\boldsymbol{\mu}_1^\top \Sigma^{-1}\boldsymbol{\mu}_1$$

What do the decision boundaries look like?



The decision boundaries are a quadratic when Σ's are not the same, this is known as *Quadratic Discriminant Analysis*!

Outline

For **Logistic Regression** we assumed:

$$\mathcal{P}(y=1|X=\boldsymbol{x}) = \sigma(\boldsymbol{w}^\top \boldsymbol{x}) = \frac{1}{1+e^{-\boldsymbol{w}^\top \boldsymbol{x}}}$$

This can be written as follows, in terms of *log-odds*

$$\log \frac{\mathcal{P}(y=1|X=\boldsymbol{x})}{\mathcal{P}(y=0|X=\boldsymbol{x})} = \boldsymbol{w}^\top \boldsymbol{x}$$

The log-odds can be modeled as a linear function of x.

For **LDA** we can write the *log-odds* as

$$\begin{aligned}
\log \frac{\mathcal{P}(y=1|X=\boldsymbol{x})}{\mathcal{P}(y=0|X=\boldsymbol{x})} &= \log \frac{\mathcal{P}(X=\boldsymbol{x}|y=1)\mathcal{P}(y=1)}{\mathcal{P}(X=\boldsymbol{x}|y=0)\mathcal{P}(y=0)} \\
&= \log \frac{\mathcal{P}(y=1)}{\mathcal{P}(y=0)} + (\boldsymbol{\mu}_1^\top \Sigma^{-1} - \boldsymbol{\mu}_0^\top \Sigma^{-1})\boldsymbol{x} - \frac{1}{2}\boldsymbol{\mu}_1^\top \Sigma^{-1} \boldsymbol{\mu}_1 + \frac{1}{2}\boldsymbol{\mu}_0^\top \Sigma^{-1} \boldsymbol{\mu}_0 \\
&= \boldsymbol{w}^\top \boldsymbol{x}
\end{aligned}$$

LDA satisfies the assumptions of the Logistics Regression model!

LDA imposes additional assumptions on the data, i.e., it **assumes that the class conditional densities are Gaussian**.