Lecture 7: Generating Learning Algorithm

Tao LIN

SoE, Westlake University

October 21, 2025

1. Review of Last Week
   - Exponential Families
   - Generalized Linear Models

2. Generative Learning Algorithms
   - Discriminative vs. Generative Learning Algorithms
   - Gaussian Discriminant Analysis (GDA)
   - Linear Discriminant Analysis (LDA)
   - LDA and Logistic Regression
   - MLE for GDA
   - Naïve Bayes

Reading materials & Reference

Reading materials:

- Chapter 4, Stanford CS 229 Lecture Notes, `https://cs229.stanford.edu/notes2022fall/main_notes.pdf`
- Chapter 9, Probabilistic Machine Learning: an introduction.
- Learning From Data Lecture 4: Generative Learning Algorithms, `http://yangli-feasibility.com/`

Table of Contents

Table of Contents

Logistic Regression and Exponential Families

Logistic Regression models the probability of the two classes $\{0, 1\}$ by

$$p(1|\eta) = \sigma(\eta) \text{ and } p(0|\eta) = 1 - \sigma(\eta)\,, \tag{1}$$

where $\eta = \mathbf{x}^\top \mathbf{w}$.

Logistic Regression and Exponential Families

Logistic Regression models the probability of the two classes $\{0, 1\}$ by

$$p(1|\eta) = \sigma(\eta) \text{ and } p(0|\eta) = 1 - \sigma(\eta)\,, \tag{1}$$

where $\eta = \mathbf{x}^\top \mathbf{w}$. This can be compactly written as

$$p(y|\eta) = \frac{e^{\eta y}}{1 + e^{\eta}} \tag{2}$$

Logistic Regression and Exponential Families

Logistic Regression models the probability of the two classes $\{0, 1\}$ by

$$p(1|\eta) = \sigma(\eta) \text{ and } p(0|\eta) = 1 - \sigma(\eta)\,, \tag{1}$$

where $\eta = \mathbf{x}^\top \mathbf{w}$. This can be compactly written as

$$p(y|\eta) = \frac{e^{\eta y}}{1 + e^{\eta}} = \exp\left(\eta y - \ln(1 + e^{\eta})\right) \tag{2}$$

Logistic Regression and Exponential Families

Logistic Regression models the probability of the two classes $\{0, 1\}$ by

$$p(1|\eta) = \sigma(\eta) \text{ and } p(0|\eta) = 1 - \sigma(\eta)\,, \tag{1}$$

where $\eta = \mathbf{x}^\top \mathbf{w}$. This can be compactly written as

$$p(y|\eta) = \frac{e^{\eta y}}{1 + e^{\eta}} = \exp\left(\eta y - \ln(1 + e^{\eta})\right) \tag{2}$$

- The linear model predicts $\sigma(\eta)$ which is not the mean of the distribution.

Logistic Regression and Exponential Families

Logistic Regression models the probability of the two classes $\{0, 1\}$ by

$$p(1|\eta) = \sigma(\eta) \text{ and } p(0|\eta) = 1 - \sigma(\eta)\,, \tag{1}$$

where $\eta = \mathbf{x}^\top \mathbf{w}$. This can be compactly written as

$$p(y|\eta) = \frac{e^{\eta y}}{1 + e^{\eta}} = \exp\left(\eta y - \ln(1 + e^{\eta})\right) \tag{2}$$

- The linear model predicts $\sigma(\eta)$ which is not the mean of the distribution.
- Rather η is related to the mean μ by the non-linear relation $\eta = \ln \frac{\mu}{1-\mu}$ or $\mu = \sigma(\eta)$.

Logistic Regression and Exponential Families

Logistic Regression models the probability of the two classes $\{0, 1\}$ by

$$p(1|\eta) = \sigma(\eta) \text{ and } p(0|\eta) = 1 - \sigma(\eta)\,, \tag{1}$$

where $\eta = \mathbf{x}^\top \mathbf{w}$. This can be compactly written as

$$p(y|\eta) = \frac{e^{\eta y}}{1 + e^{\eta}} = \exp\left(\eta y - \ln(1 + e^{\eta})\right) \tag{2}$$

- The linear model predicts $\sigma(\eta)$ which is not the mean of the distribution.
- Rather η is related to the mean μ by the non-linear relation $\eta = \ln \frac{\mu}{1-\mu}$ or $\mu = \sigma(\eta)$.
- The relation between
  - η
  - μ

  makes possible to use linear model in this context.

Logistic Regression and Exponential Families

Logistic Regression models the probability of the two classes $\{0, 1\}$ by

$$p(1|\eta) = \sigma(\eta) \text{ and } p(0|\eta) = 1 - \sigma(\eta)\,, \tag{1}$$

where $\eta = \mathbf{x}^\top \mathbf{w}$. This can be compactly written as

$$p(y|\eta) = \frac{e^{\eta y}}{1 + e^{\eta}} = \exp\left(\eta y - \ln(1 + e^{\eta})\right) \tag{2}$$

- The linear model predicts $\sigma(\eta)$ which is not the mean of the distribution.
- Rather η is related to the mean μ by the non-linear relation $\eta = \ln \frac{\mu}{1-\mu}$ or $\mu = \sigma(\eta)$.
- The relation between
  - η (the parameter predicted by the linear model)
  - μ

  makes possible to use linear model in this context.

Logistic Regression and Exponential Families

Logistic Regression models the probability of the two classes $\{0, 1\}$ by

$$p(1|\eta) = \sigma(\eta) \text{ and } p(0|\eta) = 1 - \sigma(\eta)\,, \tag{1}$$

where $\eta = \mathbf{x}^\top \mathbf{w}$. This can be compactly written as

$$p(y|\eta) = \frac{e^{\eta y}}{1 + e^{\eta}} = \exp\left(\eta y - \ln(1 + e^{\eta})\right) \tag{2}$$

- The linear model predicts $\sigma(\eta)$ which is not the mean of the distribution.
- Rather η is related to the mean μ by the non-linear relation $\eta = \ln \frac{\mu}{1-\mu}$ or $\mu = \sigma(\eta)$.
- The relation between
  - η (the parameter predicted by the linear model)
  - μ (the distribution's mean)

  makes possible to use linear model in this context.

Logistic Regression and Exponential Families

Logistic Regression models the probability of the two classes $\{0, 1\}$ by

$$p(1|\eta) = \sigma(\eta) \text{ and } p(0|\eta) = 1 - \sigma(\eta)\,, \tag{1}$$

where $\eta = \mathbf{x}^\top \mathbf{w}$. This can be compactly written as

$$p(y|\eta) = \frac{e^{\eta y}}{1 + e^{\eta}} = \exp\left(\eta y - \ln(1 + e^{\eta})\right) \tag{2}$$

- The linear model predicts $\sigma(\eta)$ which is not the mean of the distribution.
- Rather η is related to the mean μ by the non-linear relation $\eta = \ln \frac{\mu}{1-\mu}$ or $\mu = \sigma(\eta)$.
- The relation between
  - η (the parameter predicted by the linear model)
  - μ (the distribution's mean)

  makes possible to use linear model in this context.

It is called the **link function**.

Exponential family — definition

A distribution belongs to the exponential family if it can be written in the form

$$p(y|\boldsymbol{\eta}) = \underbrace{h(y)}_{\geq 0} \exp\left[\boldsymbol{\eta}^\top \boldsymbol{\phi}(y) - A(\boldsymbol{\eta})\right] \tag{3}$$

[1] Assume that we are given independent samples from this distribution. We do know $\boldsymbol{\phi}(y)$ and $h(y)$ but not $\boldsymbol{\eta}$. In order to optimally estimate $\boldsymbol{\eta}$ given these samples, all we need is the empirical average of the $\boldsymbol{\phi}(y)$.

Exponential family — definition

A distribution belongs to the exponential family if it can be written in the form

$$p(y|\boldsymbol{\eta}) = \underbrace{h(y)}_{\geq 0} \exp\left[\boldsymbol{\eta}^\top \boldsymbol{\phi}(y) - A(\boldsymbol{\eta})\right] \tag{3}$$

- $\boldsymbol{\eta}$: natural parameter of the distribution

[1] Assume that we are given independent samples from this distribution. We do know $\boldsymbol{\phi}(y)$ and $h(y)$ but not $\boldsymbol{\eta}$. In order to optimally estimate $\boldsymbol{\eta}$ given these samples, all we need is the empirical average of the $\boldsymbol{\phi}(y)$.

Exponential family — definition

A distribution belongs to the exponential family if it can be written in the form

$$p(y|\boldsymbol{\eta}) = \underbrace{h(y)}_{\geq 0} \exp\left[\boldsymbol{\eta}^\top \boldsymbol{\phi}(y) - A(\boldsymbol{\eta})\right] \tag{3}$$

- $\boldsymbol{\eta}$: natural parameter of the distribution (encodes the parameters of the distribution in a natural form)

[1] Assume that we are given independent samples from this distribution. We do know $\boldsymbol{\phi}(y)$ and $h(y)$ but not $\boldsymbol{\eta}$. In order to optimally estimate $\boldsymbol{\eta}$ given these samples, all we need is the empirical average of the $\boldsymbol{\phi}(y)$.

Exponential family — definition

A distribution belongs to the exponential family if it can be written in the form

$$p(y|\boldsymbol{\eta}) = \underbrace{h(y)}_{\geq 0} \exp\left[\boldsymbol{\eta}^\top \boldsymbol{\phi}(y) - A(\boldsymbol{\eta})\right] \tag{3}$$

- $\boldsymbol{\eta}$: natural parameter of the distribution (encodes the parameters of the distribution in a natural form)
- $\boldsymbol{\phi}(y)$: sufficient statistics[1], containing all the relevant information

[1] Assume that we are given independent samples from this distribution. We do know $\boldsymbol{\phi}(y)$ and $h(y)$ but not $\boldsymbol{\eta}$. In order to optimally estimate $\boldsymbol{\eta}$ given these samples, all we need is the empirical average of the $\boldsymbol{\phi}(y)$.

Exponential family — definition

A distribution belongs to the exponential family if it can be written in the form

$$p(y|\boldsymbol{\eta}) = \underbrace{h(y)}_{\geq 0} \exp\left[\boldsymbol{\eta}^\top \boldsymbol{\phi}(y) - A(\boldsymbol{\eta})\right] \tag{3}$$

- $\boldsymbol{\eta}$: natural parameter of the distribution (encodes the parameters of the distribution in a natural form)
- $\boldsymbol{\phi}(y)$: sufficient statistics[1], containing all the relevant information to estimate η.

[1] Assume that we are given independent samples from this distribution. We do know $\boldsymbol{\phi}(y)$ and $h(y)$ but not $\boldsymbol{\eta}$. In order to optimally estimate $\boldsymbol{\eta}$ given these samples, all we need is the empirical average of the $\boldsymbol{\phi}(y)$.

Exponential family — definition

A distribution belongs to the exponential family if it can be written in the form

$$p(y|\boldsymbol{\eta}) = \underbrace{h(y)}_{\geq 0} \exp\left[\boldsymbol{\eta}^\top \boldsymbol{\phi}(y) - A(\boldsymbol{\eta})\right] \tag{3}$$

- $\boldsymbol{\eta}$: natural parameter of the distribution (encodes the parameters of the distribution in a natural form)
- $\boldsymbol{\phi}(y)$: sufficient statistics[1], containing all the relevant information to estimate η.
- $h(y)$: the base measure (a scaling factor independent of η)

[1] Assume that we are given independent samples from this distribution. We do know $\boldsymbol{\phi}(y)$ and $h(y)$ but not $\boldsymbol{\eta}$. In order to optimally estimate $\boldsymbol{\eta}$ given these samples, all we need is the empirical average of the $\boldsymbol{\phi}(y)$.

Exponential family — definition

A distribution belongs to the exponential family if it can be written in the form

$$p(y|\boldsymbol{\eta}) = \underbrace{h(y)}_{\geq 0} \exp\left[\boldsymbol{\eta}^\top \boldsymbol{\phi}(y) - A(\boldsymbol{\eta})\right] \tag{3}$$

- $\boldsymbol{\eta}$: natural parameter of the distribution (encodes the parameters of the distribution in a natural form)
- $\boldsymbol{\phi}(y)$: sufficient statistics[1], containing all the relevant information to estimate η.
- $h(y)$: the base measure (a scaling factor independent of η)
- $A(\boldsymbol{\eta})$: log-partition function, the quantity $e^{-A(\boldsymbol{\eta})}$ is used as a normalization constant:

[1] Assume that we are given independent samples from this distribution. We do know $\boldsymbol{\phi}(y)$ and $h(y)$ but not $\boldsymbol{\eta}$. In order to optimally estimate $\boldsymbol{\eta}$ given these samples, all we need is the empirical average of the $\boldsymbol{\phi}(y)$.

Exponential family — definition

A distribution belongs to the exponential family if it can be written in the form

$$p(y|\boldsymbol{\eta}) = \underbrace{h(y)}_{\geq 0} \exp\left[\boldsymbol{\eta}^\top \boldsymbol{\phi}(y) - A(\boldsymbol{\eta})\right] \tag{3}$$

- $\boldsymbol{\eta}$: natural parameter of the distribution (encodes the parameters of the distribution in a natural form)
- $\boldsymbol{\phi}(y)$: sufficient statistics[1], containing all the relevant information to estimate η.
- $h(y)$: the base measure (a scaling factor independent of η)
- $A(\boldsymbol{\eta})$: log-partition function, the quantity $e^{-A(\boldsymbol{\eta})}$ is used as a normalization constant:

$$\text{(we need)} \int p(y|\boldsymbol{\eta}) dy = \int h(y) \exp\left[\boldsymbol{\eta}^\top \boldsymbol{\phi}(y) - A(\boldsymbol{\eta})\right] dy = 1 \tag{4}$$

$$\implies \int h(y) \exp\left[\boldsymbol{\eta}^\top \boldsymbol{\phi}(y)\right] dy = \int h(y) \exp\left[A(\boldsymbol{\eta})\right] dy = \exp\left[A(\boldsymbol{\eta})\right] \tag{5}$$

[1] Assume that we are given independent samples from this distribution. We do know $\boldsymbol{\phi}(y)$ and $h(y)$ but not $\boldsymbol{\eta}$. In order to optimally estimate $\boldsymbol{\eta}$ given these samples, all we need is the empirical average of the $\boldsymbol{\phi}(y)$.

Basic properties

- Cumulant $A(\boldsymbol{\eta})$ is convex.

Basic properties

- Cumulant $A(\boldsymbol{\eta})$ is convex.
- $\nabla A(\boldsymbol{\eta}) = \mathbb{E}\left[\boldsymbol{\phi}(y)\right]$

Basic properties

- Cumulant $A(\boldsymbol{\eta})$ is convex.
- $\nabla A(\boldsymbol{\eta}) = \mathbb{E}\left[\boldsymbol{\phi}(y)\right]$
- $\nabla^2 A(\boldsymbol{\eta}) = \mathbb{E}\left[\boldsymbol{\phi}(y)\boldsymbol{\phi}(y)^\top\right] - \mathbb{E}\left[\boldsymbol{\phi}(y)\right]\mathbb{E}\left[\boldsymbol{\phi}(y)\right]^\top$

Basic properties

- Cumulant $A(\boldsymbol{\eta})$ is convex.
- $\nabla A(\boldsymbol{\eta}) = \mathbb{E}\left[\boldsymbol{\phi}(y)\right]$
- $\nabla^2 A(\boldsymbol{\eta}) = \mathbb{E}\left[\boldsymbol{\phi}(y)\boldsymbol{\phi}(y)^\top\right] - \mathbb{E}\left[\boldsymbol{\phi}(y)\right]\mathbb{E}\left[\boldsymbol{\phi}(y)\right]^\top$
- There is a $1-1$ relationship between the "mean" $\boldsymbol{\mu} := \mathbb{E}\left[\boldsymbol{\phi}(y)\right]$ and natural parameter $\boldsymbol{\eta}$, defined using a so-called *link function* $\mathbf{g}$:

Basic properties

- Cumulant $A(\boldsymbol{\eta})$ is convex.
- $\nabla A(\boldsymbol{\eta}) = \mathbb{E}\left[\boldsymbol{\phi}(y)\right]$
- $\nabla^2 A(\boldsymbol{\eta}) = \mathbb{E}\left[\boldsymbol{\phi}(y)\boldsymbol{\phi}(y)^\top\right] - \mathbb{E}\left[\boldsymbol{\phi}(y)\right]\mathbb{E}\left[\boldsymbol{\phi}(y)\right]^\top$
- There is a $1-1$ relationship between the "mean" $\boldsymbol{\mu} := \mathbb{E}\left[\boldsymbol{\phi}(y)\right]$ and natural parameter $\boldsymbol{\eta}$, defined using a so-called *link function* $\mathbf{g}$:

$$\boldsymbol{\eta} = \mathbf{g}\left(\boldsymbol{\mu} := \mathbb{E}\left[\boldsymbol{\phi}(y)\right]\right) \iff \boldsymbol{\mu} = \mathbf{g}^{-1}(\boldsymbol{\eta}) = \nabla A(\boldsymbol{\eta}) \tag{6}$$

Table of Contents

Generalized Linear Models (GLM)

- Both linear and logistic regressions focus on the conditional relationship between X and Y

Generalized Linear Models (GLM)

- Both linear and logistic regressions focus on the conditional relationship between X and Y
  - LS: $Y \sim \mathcal{N}(\mathbf{x}^\top \mathbf{w}, \sigma^2)$

Generalized Linear Models (GLM)

- Both linear and logistic regressions focus on the conditional relationship between X and Y
  - LS: $Y \sim \mathcal{N}(\mathbf{x}^\top \mathbf{w}, \sigma^2)$
  - Logistic regression: $Y \sim \mathcal{B}(\sigma(\mathbf{x}^\top \mathbf{w}))$

Generalized Linear Models (GLM)

- Both linear and logistic regressions focus on the conditional relationship between X and Y
  - LS: $Y \sim \mathcal{N}(\mathbf{x}^\top \mathbf{w}, \sigma^2)$
  - Logistic regression: $Y \sim \mathcal{B}(\sigma(\mathbf{x}^\top \mathbf{w}))$
- Common feature of linear and logistic regression:

Generalized Linear Models (GLM)

- Both linear and logistic regressions focus on the conditional relationship between X and Y
  - LS: $Y \sim \mathcal{N}(\mathbf{x}^\top \mathbf{w}, \sigma^2)$
  - Logistic regression: $Y \sim \mathcal{B}(\sigma(\mathbf{x}^\top \mathbf{w}))$
- Common feature of linear and logistic regression:
  1. Model the conditional expectation as $\mu = f(\mathbf{x}^\top \mathbf{w})$

Generalized Linear Models (GLM)

- Both linear and logistic regressions focus on the conditional relationship between X and Y
  - LS: $Y \sim \mathcal{N}(\mathbf{x}^\top \mathbf{w}, \sigma^2)$
  - Logistic regression: $Y \sim \mathcal{B}(\sigma(\mathbf{x}^\top \mathbf{w}))$
- Common feature of linear and logistic regression:
  1. Model the conditional expectation as $\mu = f(\mathbf{x}^\top \mathbf{w})$
  2. Endow Y with a particular probability distribution having μ as parameter

Generalized Linear Models (GLM)

- Both linear and logistic regressions focus on the conditional relationship between X and Y
  - LS: $Y \sim \mathcal{N}(\mathbf{x}^\top \mathbf{w}, \sigma^2)$
  - Logistic regression: $Y \sim \mathcal{B}(\sigma(\mathbf{x}^\top \mathbf{w}))$
- Common feature of linear and logistic regression:
  1. Model the conditional expectation as $\mu = f(\mathbf{x}^\top \mathbf{w})$
  2. Endow Y with a particular probability distribution having μ as parameter
- The GLM framework extends these ideas to the general exponential family.

Constructing GLMs

To derive a GLM for a classification/regression problem (the conditional dist. of y given $\mathbf{x}$):

Constructing GLMs

To derive a GLM for a classification/regression problem (the conditional dist. of y given $\mathbf{x}$):

1. The natural parameter η and the observed inputs $\mathbf{x}$ are related linearly: $\eta = \mathbf{x}^\top \mathbf{w}$

Constructing GLMs

To derive a GLM for a classification/regression problem (the conditional dist. of y given $\mathbf{x}$):

1. The natural parameter η and the observed inputs $\mathbf{x}$ are related linearly: $\eta = \mathbf{x}^\top \mathbf{w}$
2. The conditional mean μ is represented as a function $f(\eta)$ of the linear combination $\boldsymbol{\eta}$

Constructing GLMs

To derive a GLM for a classification/regression problem (the conditional dist. of y given $\mathbf{x}$):

1. The natural parameter η and the observed inputs $\mathbf{x}$ are related linearly: $\eta = \mathbf{x}^\top \mathbf{w}$
2. The conditional mean μ is represented as a function $f(\eta)$ of the linear combination $\boldsymbol{\eta}$
3. The observed output y is assumed to be characterized by an exponential family distribution with conditional mean μ.

Constructing GLMs

To derive a GLM for a classification/regression problem (the conditional dist. of y given $\mathbf{x}$):

1. The natural parameter η and the observed inputs $\mathbf{x}$ are related linearly: $\eta = \mathbf{x}^\top \mathbf{w}$
2. The conditional mean μ is represented as a function $f(\eta)$ of the linear combination $\boldsymbol{\eta}$
3. The observed output y is assumed to be characterized by an exponential family distribution with conditional mean μ.

The condition probability is thus modeled as:

$$p(y|\mathbf{x}; \mathbf{w}) = h(y_n) \exp\left(\eta\phi(y) - A(\eta)\right) \qquad \text{for} \quad \eta = g \circ f(\mathbf{x}^\top \mathbf{w}) \tag{7}$$

Negative log-likelihood estimation

Note that:

$$\mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \ln p(y_n|\mathbf{x}_n^\top \mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} (\ln(h(y_n)) + \eta_n \phi(y_n) - A(\eta_n)) \tag{8}$$

Negative log-likelihood estimation

Note that:

$$\mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \ln p(y_n|\mathbf{x}_n^\top \mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} (\ln(h(y_n)) + \eta_n \phi(y_n) - A(\eta_n)) \tag{8}$$

If we rewrite this sum by using the matrix notation, we get

$$\nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} (\mathbf{x}_n \phi(y_n) - \nabla_{\mathbf{w}} A(\eta_n))$$

(11)

Negative log-likelihood estimation

Note that:

$$\mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \ln p(y_n|\mathbf{x}_n^\top \mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} (\ln(h(y_n)) + \eta_n \phi(y_n) - A(\eta_n)) \tag{8}$$

If we rewrite this sum by using the matrix notation, we get

$$\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} (\mathbf{x}_n \phi(y_n) - \nabla_{\mathbf{w}} A(\eta_n)) = -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - A'(\mathbf{x}_n^\top \mathbf{w})\mathbf{x}_n\right) \tag{9}$$

(11)

Negative log-likelihood estimation

Note that:

$$\mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \ln p(y_n|\mathbf{x}_n^\top \mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \left(\ln(h(y_n)) + \eta_n \phi(y_n) - A(\eta_n)\right) \tag{8}$$

If we rewrite this sum by using the matrix notation, we get

$$\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - \nabla_{\mathbf{w}} A(\eta_n)\right) = -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - A'(\mathbf{x}_n^\top \mathbf{w})\mathbf{x}_n\right) \tag{9}$$

$$= -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - \mathbb{E}\left[\phi(Y_n)\right]\mathbf{x}_n\right)$$

(11)

Negative log-likelihood estimation

Note that:

$$\mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \ln p(y_n|\mathbf{x}_n^\top \mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \left(\ln(h(y_n)) + \eta_n \phi(y_n) - A(\eta_n)\right) \tag{8}$$

If we rewrite this sum by using the matrix notation, we get

$$\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - \nabla_{\mathbf{w}} A(\eta_n)\right) = -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - A'(\mathbf{x}_n^\top \mathbf{w})\mathbf{x}_n\right) \tag{9}$$

$$= -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - \mathbb{E}\left[\phi(Y_n)\right]\mathbf{x}_n\right) = -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - g^{-1}(\mathbf{x}_n^\top \mathbf{w})\mathbf{x}_n\right) \tag{10}$$

$$\tag{11}$$

Negative log-likelihood estimation

Note that:

$$\mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \ln p(y_n|\mathbf{x}_n^\top \mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \left(\ln(h(y_n)) + \eta_n \phi(y_n) - A(\eta_n)\right) \tag{8}$$

If we rewrite this sum by using the matrix notation, we get

$$\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - \nabla_{\mathbf{w}} A(\eta_n)\right) = -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - A'(\mathbf{x}_n^\top \mathbf{w})\mathbf{x}_n\right) \tag{9}$$

$$= -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - \mathbb{E}\left[\phi(Y_n)\right]\mathbf{x}_n\right) = -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - g^{-1}(\mathbf{x}_n^\top \mathbf{w})\mathbf{x}_n\right) \tag{10}$$

$$= -\frac{1}{N}\mathbf{X}^\top \left[g^{-1}(\mathbf{X}\mathbf{w}) - \phi(\mathbf{y})\right] \tag{11}$$

Negative log-likelihood estimation

Note that:

$$\mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \ln p(y_n|\mathbf{x}_n^\top \mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \left(\ln(h(y_n)) + \eta_n \phi(y_n) - A(\eta_n)\right) \tag{8}$$

If we rewrite this sum by using the matrix notation, we get

$$\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - \nabla_{\mathbf{w}} A(\eta_n)\right) = -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - A'(\mathbf{x}_n^\top \mathbf{w})\mathbf{x}_n\right) \tag{9}$$

$$= -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - \mathbb{E}\left[\phi(Y_n)\right]\mathbf{x}_n\right) = -\frac{1}{N}\sum_{n=1}^{N} \left(\mathbf{x}_n \phi(y_n) - g^{-1}(\mathbf{x}_n^\top \mathbf{w})\mathbf{x}_n\right) \tag{10}$$

$$= -\frac{1}{N}\mathbf{X}^\top \left[g^{-1}(\mathbf{X}\mathbf{w}) - \phi(\mathbf{y})\right] \tag{11}$$

In the case of Logistic Regression:

$$\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}) = \frac{1}{N}\mathbf{X}^\top \left[\sigma(\mathbf{X}\mathbf{w}) - \mathbf{y}\right] \tag{12}$$

Last lecture:

- Exponential Families
- Generalized Linear Models

Last lecture:

- Exponential Families
- Generalized Linear Models

This lecture:

- Discriminative vs. Generative learning algorithms
- Gaussian Discriminant Analysis (GDA)
- Linear Discriminant Analysis (LDA)
- LDA and Logistic regression
- Naïve Bayes

Table of Contents

Table of Contents

Recap

Classification: we observe some data $\mathcal{S} = \{\mathbf{x}_n, y_n\}_{n=1}^{N} \in \mathcal{X} \times \underbrace{\mathcal{Y}}_{\text{Discrete set}}$

Goal: given a new observation $\mathbf{x}$, we want to predict its label y

How: relates input to a categorical variable

Recap

In previous lectures, we assume that we know the joint distribution $p(\mathbf{x}, y)$.

- For a given input $\mathbf{x}$, *the probability that the "correct" label is* y is $p(y|\mathbf{x})$.

Recap

In previous lectures, we assume that we know the joint distribution $p(\mathbf{x}, y)$.

- For a given input $\mathbf{x}$, *the probability that the "correct" label is* y is $p(y|\mathbf{x})$.
- Maximum A-Posteriori (MAP):

Recap

In previous lectures, we assume that we know the joint distribution $p(\mathbf{x}, y)$.

- For a given input $\mathbf{x}$, *the probability that the "correct" label is* y is $p(y|\mathbf{x})$.
- Maximum A-Posteriori (MAP): If we want to maximize the probability of guessing the correct label,

Recap

In previous lectures, we assume that we know the joint distribution $p(\mathbf{x}, y)$.

- For a given input $\mathbf{x}$, *the probability that the "correct" label is* y is $p(y|\mathbf{x})$.
- Maximum A-Posteriori (MAP): If we want to maximize the probability of guessing the correct label, then we should choose the decision rule

Recap

In previous lectures, we assume that we know the joint distribution $p(\mathbf{x}, y)$.

- For a given input $\mathbf{x}$, *the probability that the "correct" label is* y is $p(y|\mathbf{x})$.
- Maximum A-Posteriori (MAP): If we want to maximize the probability of guessing the correct label, then we should choose the decision rule

$$\hat{y}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x}) \tag{13}$$

Recap

In previous lectures, we assume that we know the joint distribution $p(\mathbf{x}, y)$.

- For a given input $\mathbf{x}$, *the probability that the "correct" label is* y is $p(y|\mathbf{x})$.
- Maximum A-Posteriori (MAP): If we want to maximize the probability of guessing the correct label, then we should choose the decision rule

$$\hat{y}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x}) \tag{13}$$

This classifier is also called the *Bayes classifier*: $f^\star = \arg\min_f L_{\mathcal{D}}(f)$.

Recap

In previous lectures, we assume that we know the joint distribution $p(\mathbf{x}, y)$.

- For a given input $\mathbf{x}$, *the probability that the "correct" label is* y is $p(y|\mathbf{x})$.
- Maximum A-Posteriori (MAP): If we want to maximize the probability of guessing the correct label, then we should choose the decision rule

$$\hat{y}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x}) \tag{13}$$

  This classifier is also called the *Bayes classifier*: $f^\star = \arg\min_f L_{\mathcal{D}}(f)$.
- Such an idea extends multi-label classification problem:

$$p(y = c|\mathbf{x}, \mathbf{w}) = \frac{p(\mathbf{x}|y = c, \mathbf{w}) \cdot p(y = c|\mathbf{w})}{\sum_{c'} p(\mathbf{x}|y = c', \mathbf{w}) \cdot p(y = c'|\mathbf{w})}, \tag{14}$$

Recap

In previous lectures, we assume that we know the joint distribution $p(\mathbf{x}, y)$.

- For a given input $\mathbf{x}$, *the probability that the "correct" label is* y is $p(y|\mathbf{x})$.
- Maximum A-Posteriori (MAP): If we want to maximize the probability of guessing the correct label, then we should choose the decision rule

$$\hat{y}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x}) \tag{13}$$

  This classifier is also called the *Bayes classifier*: $f^\star = \arg\min_f L_{\mathcal{D}}(f)$.
- Such an idea extends multi-label classification problem:

$$p(y = c|\mathbf{x}, \mathbf{w}) = \frac{p(\mathbf{x}|y = c, \mathbf{w}) \cdot p(y = c|\mathbf{w})}{\sum_{c'} p(\mathbf{x}|y = c', \mathbf{w}) \cdot p(y = c'|\mathbf{w})}, \tag{14}$$

  - $p(y = c'|\mathbf{w})$ is the prior over class labels,

Recap

In previous lectures, we assume that we know the joint distribution $p(\mathbf{x}, y)$.

- For a given input $\mathbf{x}$, *the probability that the "correct" label is* y is $p(y|\mathbf{x})$.
- Maximum A-Posteriori (MAP): If we want to maximize the probability of guessing the correct label, then we should choose the decision rule

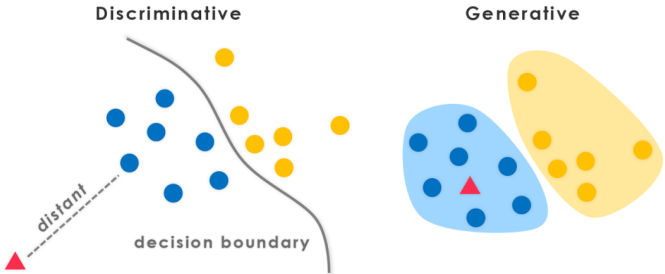
$$\hat{y}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x}) \tag{13}$$

  This classifier is also called the *Bayes classifier*: $f^\star = \arg\min_f L_{\mathcal{D}}(f)$.
- Such an idea extends multi-label classification problem:

$$p(y = c|\mathbf{x}, \mathbf{w}) = \frac{p(\mathbf{x}|y = c, \mathbf{w}) \cdot p(y = c|\mathbf{w})}{\sum_{c'} p(\mathbf{x}|y = c', \mathbf{w}) \cdot p(y = c'|\mathbf{w})}, \tag{14}$$

  - $p(y = c'|\mathbf{w})$ is the prior over class labels,
  - $p(\mathbf{x}|y = c', \mathbf{w})$ is called the class conditional density for class c'.

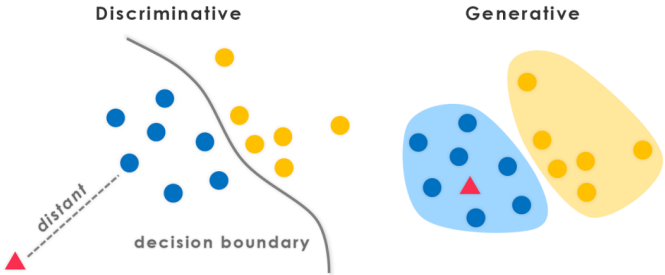
Discriminative v/s Generative Learning Algorithms



Discriminative Learning Algorithm:

Generative Learning Algorithm:

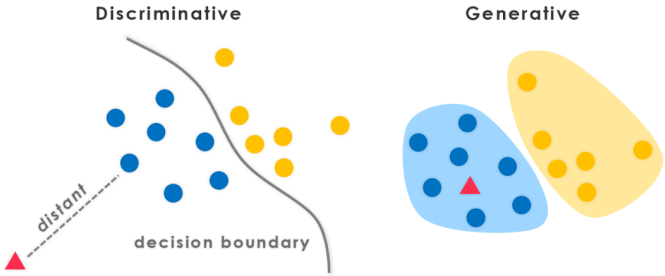
Discriminative v/s Generative Learning Algorithms



Discriminative Learning Algorithm:
Try to learn the conditional probability $p(y|\mathbf{x}, \mathbf{w})$ directly or learn mappings directly from $\mathcal{X}$ to $\mathcal{Y}$

Generative Learning Algorithm:

Discriminative v/s Generative Learning Algorithms



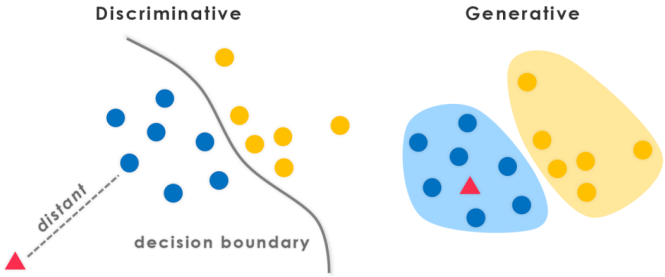
Discriminative Learning Algorithm:
Try to learn the conditional probability $p(y|\mathbf{x}, \mathbf{w})$ directly or learn mappings directly from $\mathcal{X}$ to $\mathcal{Y}$

- it relies on the decision rule

$$\hat{y}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x}, \mathbf{w})$$

Generative Learning Algorithm:

Discriminative v/s Generative Learning Algorithms



Discriminative Learning Algorithm:
Try to learn the conditional probability $p(y|\mathbf{x}, \mathbf{w})$ directly or learn mappings directly from $\mathcal{X}$ to $\mathcal{Y}$

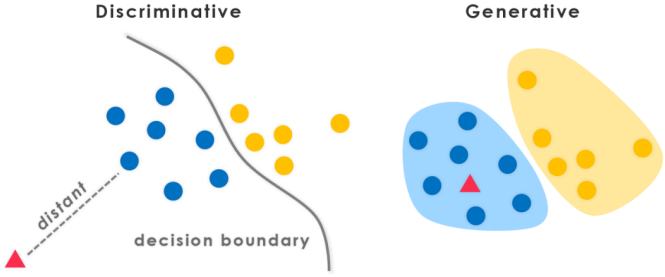
- it relies on the decision rule

$$\hat{y}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x}, \mathbf{w})$$

- e.g., linear regression, logistic regression

Generative Learning Algorithm:

Discriminative v/s Generative Learning Algorithms



Discriminative Learning Algorithm:
Try to learn the conditional probability $p(y|\mathbf{x}, \mathbf{w})$ directly or learn mappings directly from $\mathcal{X}$ to $\mathcal{Y}$

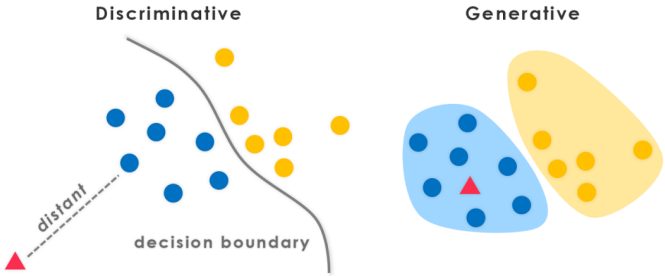
- it relies on the decision rule

$$\hat{y}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x}, \mathbf{w})$$

- e.g., linear regression, logistic regression

Generative Learning Algorithm:
Try to model the joint probability $p(\mathbf{x}, y|\mathbf{w})$

Discriminative v/s Generative Learning Algorithms



Discriminative Learning Algorithm:
Try to learn the conditional probability $p(y|\mathbf{x}, \mathbf{w})$ directly or learn mappings directly from $\mathcal{X}$ to $\mathcal{Y}$

- it relies on the decision rule

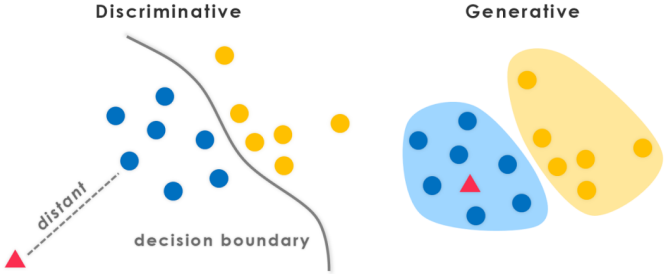
$$\hat{y}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x}, \mathbf{w})$$

- e.g., linear regression, logistic regression

Generative Learning Algorithm:
Try to model the joint probability $p(\mathbf{x}, y|\mathbf{w})$

- equivalently, it models $p(\mathbf{x}|y, \mathbf{w})$ and $p(y|\mathbf{w})$

Discriminative v/s Generative Learning Algorithms



Discriminative Learning Algorithm:
Try to learn the conditional probability $p(y|\mathbf{x}, \mathbf{w})$ directly or learn mappings directly from $\mathcal{X}$ to $\mathcal{Y}$

- it relies on the decision rule

$$\hat{y}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x}, \mathbf{w})$$

- e.g., linear regression, logistic regression

Generative Learning Algorithm:
Try to model the joint probability $p(\mathbf{x}, y|\mathbf{w})$

- equivalently, it models $p(\mathbf{x}|y, \mathbf{w})$ and $p(y|\mathbf{w})$
- learned models are transformed to $p(y|\mathbf{x}, \mathbf{w})$ later to classify data using Bayes' rule:
  $p(y|\mathbf{x}) = \frac{p(\mathbf{x}|y)p(y)}{p(\mathbf{x})}$

The posterior distribution on y given $\mathbf{x}$:

$$p(y|\mathbf{x}) = \frac{p(\mathbf{x}|y)p(y)}{p(\mathbf{x})} \tag{15}$$

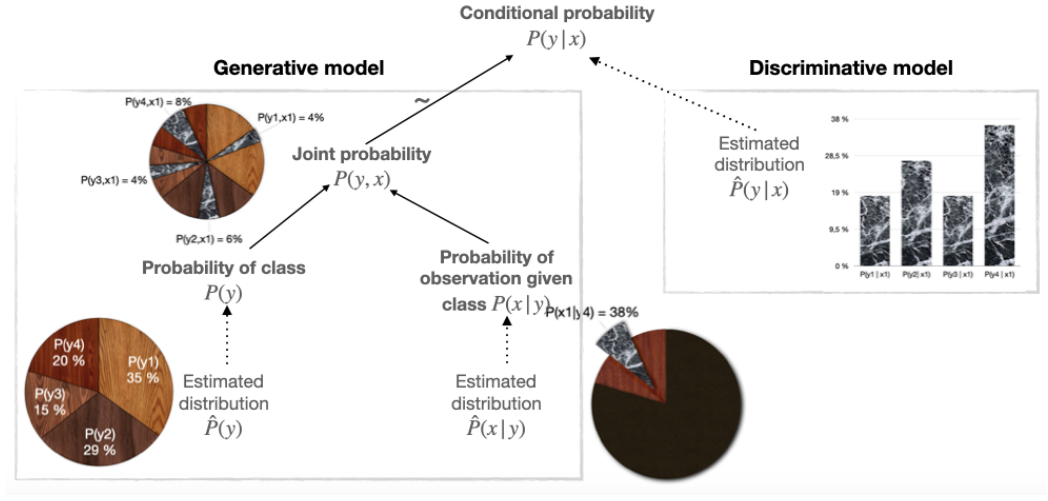
The posterior distribution on y given $\mathbf{x}$:

$$p(y|\mathbf{x}) = \frac{p(\mathbf{x}|y)p(y)}{p(\mathbf{x})} \tag{15}$$

Make predictions in a generative model (exercise):

$$\arg\max_{y\in\mathcal{Y}} p(y|\mathbf{x}, \mathbf{w}) = \arg\max_{y\in\mathcal{Y}} \frac{p(\mathbf{x}|y, \mathbf{w})p(y|\mathbf{w})}{p(\mathbf{x}|\mathbf{w})} = \arg\max_{y\in\mathcal{Y}} p(\mathbf{x}|y, \mathbf{w})p(y|\mathbf{w})$$

Discriminative v/s Generative Learning Algorithms (an alternative view)



From https://towardsdatascience.com/introduction-to-generative-and-discriminative-models-9c9ef152b9af/

Discriminative Models v.s. Generative Models (an alternative view)

Discriminative classification algorithms

- focus on the decision boundary
- more powerful with lots of examples

Generative classification algorithms

Discriminative Models v.s. Generative Models (an alternative view)

Discriminative classification algorithms

- focus on the decision boundary
- more powerful with lots of examples

Generative classification algorithms

- probabilistic “model” of each class

Discriminative Models v.s. Generative Models (an alternative view)

Discriminative classification algorithms

- focus on the decision boundary
- more powerful with lots of examples

Generative classification algorithms

- probabilistic “model” of each class
- decision boundary:
  - where one model becomes more likely

Discriminative Models v.s. Generative Models (an alternative view)

Discriminative classification algorithms

- focus on the decision boundary
- more powerful with lots of examples
- not designed to use unlabeled data
- only supervised tasks

Generative classification algorithms

- probabilistic “model” of each class
- decision boundary:
  - where one model becomes more likely

Discriminative Models v.s. Generative Models (an alternative view)

Discriminative classification algorithms

- focus on the decision boundary
- more powerful with lots of examples
- not designed to use unlabeled data
- only supervised tasks

Generative classification algorithms

- probabilistic “model” of each class
- decision boundary:
  - where one model becomes more likely
- natural use of unlabeled data

Discriminative Models v.s. Generative Models (an alternative view)

Discriminative classification algorithms

- focus on the decision boundary
- more powerful with lots of examples
- not designed to use unlabeled data
- only supervised tasks

Generative classification algorithms

- probabilistic “model” of each class
- decision boundary:
  - where one model becomes more likely
- natural use of unlabeled data

- algorithms:

Discriminative Models v.s. Generative Models (an alternative view)

Discriminative classification algorithms

- focus on the decision boundary
- more powerful with lots of examples
- not designed to use unlabeled data
- only supervised tasks

Generative classification algorithms

- probabilistic "model" of each class
- decision boundary:
  - where one model becomes more likely
- natural use of unlabeled data

- algorithms:
  - continuous input: Gaussian Discriminant Analysis

Discriminative Models v.s. Generative Models (an alternative view)

Discriminative classification algorithms

- focus on the decision boundary
- more powerful with lots of examples
- not designed to use unlabeled data
- only supervised tasks

Generative classification algorithms

- probabilistic "model" of each class
- decision boundary:
  - where one model becomes more likely
- natural use of unlabeled data

- algorithms:
  - continuous input: Gaussian Discriminant Analysis
  - discrete input: Naïve Bayes

Table of Contents

Gaussian Discriminant Analysis (GDA)

Recall that

$$p(y = c|\mathbf{x}, \mathbf{w}) = \frac{p(\mathbf{x}|y = c, \mathbf{w}) \cdot p(y = c|\mathbf{w})}{\sum_{c'} p(\mathbf{x}|y = c', \mathbf{w}) \cdot p(y = c'|\mathbf{w})},$$

- $p(y = c'|\mathbf{w})$
- $p(\mathbf{x}|y = c', \mathbf{w})$

Gaussian Discriminant Analysis (GDA)

Recall that

$$p(y = c|\mathbf{x}, \mathbf{w}) = \frac{p(\mathbf{x}|y = c, \mathbf{w}) \cdot p(y = c|\mathbf{w})}{\sum_{c'} p(\mathbf{x}|y = c', \mathbf{w}) \cdot p(y = c'|\mathbf{w})},$$

- $p(y = c'|\mathbf{w})$ is the prior over class labels,
- $p(\mathbf{x}|y = c', \mathbf{w})$

GDA assumes that $p(\mathbf{x}|y, \mathbf{w})$ is distributed according a multi-variate normal distribution.

Gaussian Discriminant Analysis (GDA)

Recall that

$$p(y = c|\mathbf{x}, \mathbf{w}) = \frac{p(\mathbf{x}|y = c, \mathbf{w}) \cdot p(y = c|\mathbf{w})}{\sum_{c'} p(\mathbf{x}|y = c', \mathbf{w}) \cdot p(y = c'|\mathbf{w})},$$

- $p(y = c'|\mathbf{w})$ is the prior over class labels,
- $p(\mathbf{x}|y = c', \mathbf{w})$ is called the class conditional density for class c'.

GDA assumes that $p(\mathbf{x}|y, \mathbf{w})$ is distributed according a multi-variate normal distribution.

Definition 1 (The density of a random vector from the multi-variate normal distribution)

The density of a random vector from the multi-variate normal/Gaussian distribution, with mean $\boldsymbol{\mu} \in \mathbb{R}^d$ and covariance $\boldsymbol{\Sigma} \in \mathbb{R}^{d \times d}$ is

$$\mathcal{N}(\mathbf{x} \,|\, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^d \, det(\boldsymbol{\Sigma})}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\}, \tag{16}$$

Definition 1 (The density of a random vector from the multi-variate normal distribution)

The density of a random vector from the multi-variate normal/Gaussian distribution, with mean $\boldsymbol{\mu} \in \mathbb{R}^d$ and covariance $\boldsymbol{\Sigma} \in \mathbb{R}^{d \times d}$ is

$$\mathcal{N}(\mathbf{x} \,|\, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^d \, det(\boldsymbol{\Sigma})}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\}, \tag{16}$$

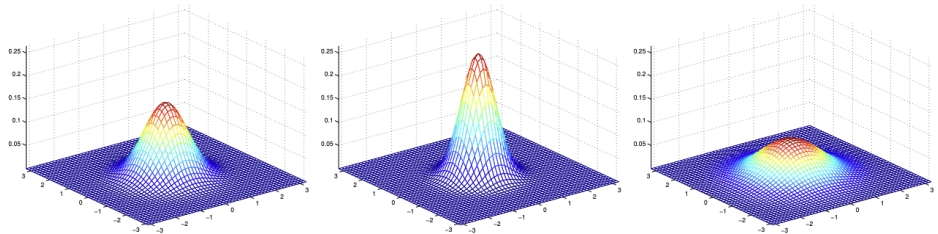
where $\boldsymbol{\Sigma}$ (exercise)

Definition 1 (The density of a random vector from the multi-variate normal distribution)

The density of a random vector from the multi-variate normal/Gaussian distribution, with mean $\boldsymbol{\mu} \in \mathbb{R}^d$ and covariance $\boldsymbol{\Sigma} \in \mathbb{R}^{d \times d}$ is

$$\mathcal{N}(\mathbf{x} \,|\, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^d \, det(\boldsymbol{\Sigma})}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\}, \tag{16}$$

where $\boldsymbol{\Sigma}$ (exercise) is symmetric and positive semi-definite matrix.



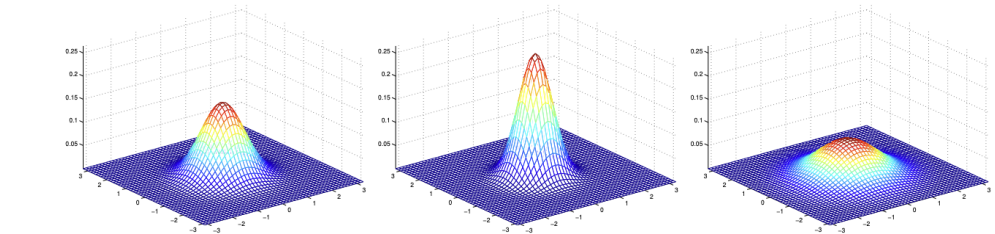
(Left): $\boldsymbol{\Sigma} = \mathbf{I}$; (Middle): $\boldsymbol{\Sigma} = 0.6\mathbf{I}$; (Right): $\boldsymbol{\Sigma} = 2\mathbf{I}$.

Definition 1 (The density of a random vector from the multi-variate normal distribution)

The density of a random vector from the multi-variate normal/Gaussian distribution, with mean $\boldsymbol{\mu} \in \mathbb{R}^d$ and covariance $\boldsymbol{\Sigma} \in \mathbb{R}^{d \times d}$ is

$$\mathcal{N}(\mathbf{x} \,|\, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^d \, det(\boldsymbol{\Sigma})}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\}, \tag{16}$$

where $\boldsymbol{\Sigma}$ (exercise) is symmetric and positive semi-definite matrix.



(Left): $\Sigma = \mathbf{I}$; (Middle): $\Sigma = 0.6\mathbf{I}$; (Right): $\Sigma = 2\mathbf{I}$. We can see that as

- Σ becomes larger, the Gaussian becomes more "spread-out"
- Σ becomes smaller, the Gaussian becomes more "compressed"

Definition 1 (The density of a random vector from the multi-variate normal distribution)

The density of a random vector from the multi-variate normal/Gaussian distribution, with mean $\boldsymbol{\mu} \in \mathbb{R}^d$ and covariance $\Sigma \in \mathbb{R}^{d \times d}$ is

$$\mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}, \Sigma) = \frac{1}{\sqrt{(2\pi)^d \, det(\Sigma)}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\}, \tag{16}$$

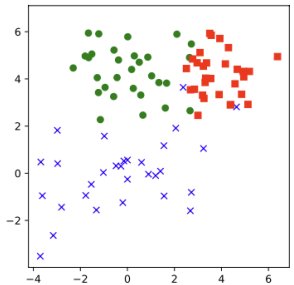
where Σ (exercise) is symmetric and positive semi-definite matrix.

Gaussian Discriminant Analysis (GDA) Model

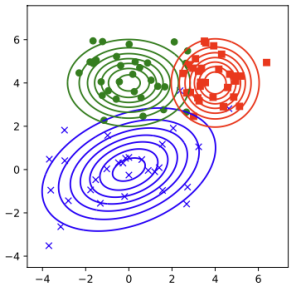
When we have a classification problem in which the input features $\mathbf{x}$ are continuous-valued RV,
$\Longrightarrow$ we can use the GDA model

Gaussian Discriminant Analysis (GDA) Model

When we have a classification problem in which the input features $\boldsymbol{x}$ are continuous-valued RV,
$\Longrightarrow$ we can use the GDA model



(a)

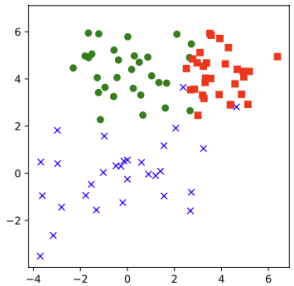


(b)

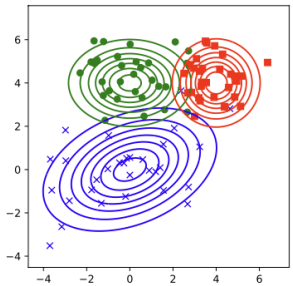
2d data from 3 different classes in (a), and we fit the full covariance Gaussian class-conditionals in (b).

Gaussian Discriminant Analysis (GDA) Model

When we have a classification problem in which the input features $\mathbf{x}$ are continuous-valued RV,
$\Longrightarrow$ we can use the GDA model



(a)



(b)

2d data from 3 different classes in (a), and we fit the full covariance Gaussian class-conditionals in (b).

$\Longrightarrow$ in which models $p(\mathbf{x}|y, \mathbf{w})$ uses a multi-variate normal distribution:

Gaussian Discriminant Analysis (GDA) Model

When we have a classification problem in which the input features $\mathbf{x}$ are continuous-valued RV,
$\Longrightarrow$ we can use the GDA model
$\Longrightarrow$ in which models $p(\mathbf{x}|y, \mathbf{w})$ uses a multi-variate normal distribution:

$$y \sim \text{Bernoulli}(\phi) \tag{17}$$
$$\mathbf{x}|y = 0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}) \tag{18}$$
$$\mathbf{x}|y = 1 \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}) \tag{19}$$

Gaussian Discriminant Analysis (GDA) Model

When we have a classification problem in which the input features $\mathbf{x}$ are continuous-valued RV,
$\Longrightarrow$ we can use the GDA model
$\Longrightarrow$ in which models $p(\mathbf{x}|y, \mathbf{w})$ uses a multi-variate normal distribution:

$$y \sim \text{Bernoulli}(\phi) \tag{17}$$
$$\mathbf{x}|y = 0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}) \tag{18}$$
$$\mathbf{x}|y = 1 \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}) \tag{19}$$

The corresponding class posterior ((Gaussian Discriminant Analysis: GDA)) therefore has the form

$$p(y = c|\mathbf{x}, \mathbf{w}) \propto \pi_c \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)\,, \tag{20}$$

where $\pi_c = p(y = c|\mathbf{w})$.

Gaussian Discriminant Analysis (GDA) Model

Write out the Probability Density Functions (PDFs):

$$p(y) = \phi^y (1-\phi)^{1-y} \tag{21}$$

$$p(\mathbf{x}|y=0) = \frac{1}{\sqrt{(2\pi)^d \, det(\mathbf{\Sigma}_0)}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_0)^\top \mathbf{\Sigma}_0^{-1} (\mathbf{x} - \boldsymbol{\mu}_0) \right\} \tag{22}$$

$$p(\mathbf{x}|y=1) = \frac{1}{\sqrt{(2\pi)^d \, det(\mathbf{\Sigma}_1)}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_1)^\top \mathbf{\Sigma}_1^{-1} (\mathbf{x} - \boldsymbol{\mu}_1) \right\} \tag{23}$$

Gaussian Discriminant Analysis (GDA) Model

Write out the Probability Density Functions (PDFs):

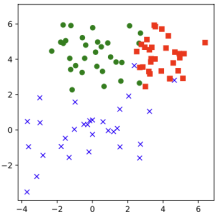
$$p(y) = \phi^y (1-\phi)^{1-y} \tag{21}$$

$$p(\mathbf{x}|y=0) = \frac{1}{\sqrt{(2\pi)^d \, det(\boldsymbol{\Sigma}_0)}} \exp\left\{ -\frac{1}{2} (\mathbf{x}-\boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}_0^{-1} (\mathbf{x}-\boldsymbol{\mu}_0) \right\} \tag{22}$$

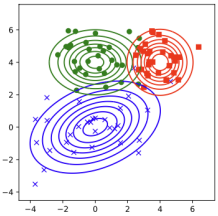
$$p(\mathbf{x}|y=1) = \frac{1}{\sqrt{(2\pi)^d \, det(\boldsymbol{\Sigma}_1)}} \exp\left\{ -\frac{1}{2} (\mathbf{x}-\boldsymbol{\mu}_1)^\top \boldsymbol{\Sigma}_1^{-1} (\mathbf{x}-\boldsymbol{\mu}_1) \right\} \tag{23}$$

The log posterior over class labels is given by

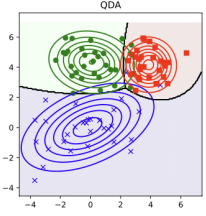
$$\log p(y=c|\mathbf{x}, \mathbf{w}) = \log \pi_c - \frac{1}{2} |2\pi \boldsymbol{\Sigma}_c| - \frac{1}{2} (\mathbf{x}-\boldsymbol{\mu}_c)^\top \boldsymbol{\Sigma}_c^{-1} (\mathbf{x}-\boldsymbol{\mu}_c) + \text{const}$$



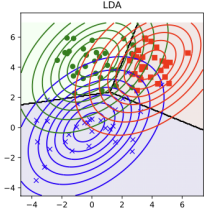
(a)



(b)



(a)



(b)

(a): GDA fits to data, where unconstrained covariances induce quadratic decision boundaries;
(b): LDA fits to data, where tied covariances induce linear decision boundaries.

What is LDA?

Table of Contents

Linear Discriminant Analysis (LDA)

LDA is a special case of GDA
in which the covariance matrices are **tied** or **shared** across classes (i.e., $\Sigma_c = \Sigma$)

Linear Discriminant Analysis (LDA)

LDA is a special case of GDA
in which the covariance matrices are **tied** or **shared** across classes (i.e., $\boldsymbol{\Sigma}_c = \boldsymbol{\Sigma}$)

The log posterior over class labels is given by

$$\log p(y = c|\mathbf{x}, \mathbf{w}) = \log \pi_c - \frac{1}{2}|2\pi\boldsymbol{\Sigma}_c| - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_c)^\top \boldsymbol{\Sigma}_c^{-1}(\mathbf{x} - \boldsymbol{\mu}_c) + \text{const}$$

Linear Discriminant Analysis (LDA)

LDA is a special case of GDA
in which the covariance matrices are **tied** or **shared** across classes (i.e., $\boldsymbol{\Sigma}_c = \boldsymbol{\Sigma}$)

The log posterior over class labels is given by

$$\begin{aligned}
\log p(y=c|\mathbf{x},\mathbf{w}) &= \log \pi_c - \frac{1}{2}|2\pi\boldsymbol{\Sigma}_c| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_c)^\top \boldsymbol{\Sigma}_c^{-1}(\mathbf{x}-\boldsymbol{\mu}_c) + \text{const} \\
&= \log \pi_c - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_c)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_c) + \text{const} \\
&= \underbrace{\log \pi_c - \frac{1}{2}\boldsymbol{\mu}_c^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_c}_{\gamma_c} + \mathbf{x}^\top \underbrace{\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_c}_{\boldsymbol{\beta}_c} + \underbrace{\text{const} - \frac{1}{2}\mathbf{x}^\top \boldsymbol{\Sigma}^{-1}\mathbf{x}}_{\kappa}
\end{aligned}$$

Linear Discriminant Analysis (LDA)

LDA is a special case of GDA
in which the covariance matrices are **tied** or **shared** across classes (i.e., $\boldsymbol{\Sigma}_c = \boldsymbol{\Sigma}$)

The log posterior over class labels is given by

$$\begin{aligned}
\log p(y=c|\mathbf{x},\mathbf{w}) &= \log \pi_c - \frac{1}{2}|2\pi\boldsymbol{\Sigma}_c| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_c)^\top \boldsymbol{\Sigma}_c^{-1}(\mathbf{x}-\boldsymbol{\mu}_c) + \text{const} \\
&= \log \pi_c - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_c)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_c) + \text{const} \\
&= \underbrace{\log \pi_c - \frac{1}{2}\boldsymbol{\mu}_c^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_c}_{\gamma_c} + \mathbf{x}^\top \underbrace{\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_c}_{\boldsymbol{\beta}_c} + \underbrace{\text{const} - \frac{1}{2}\mathbf{x}^\top \boldsymbol{\Sigma}^{-1}\mathbf{x}}_{\kappa} \\
&= \gamma_c + \mathbf{x}^\top \boldsymbol{\beta}_c + \kappa
\end{aligned}$$

Table of Contents

The connection between LDA and logistic regression

Recall that in LDA, we have

$$\log p(y = c|\mathbf{x}, \mathbf{w}) = \gamma_c + \mathbf{x}^\top \boldsymbol{\beta}_c + \kappa$$

The connection between LDA and logistic regression

Recall that in LDA, we have

$$\log p(y = c|\mathbf{x}, \mathbf{w}) = \gamma_c + \mathbf{x}^\top \boldsymbol{\beta}_c + \kappa$$

Then,

$$p(y = c|\mathbf{x}, \mathbf{w}) = \frac{e^{\gamma_c + \mathbf{x}^\top \boldsymbol{\beta}_c}}{\sum_{c'} e^{\gamma_{c'} + \mathbf{x}^\top \boldsymbol{\beta}_{c'}}} = \frac{e^{\mathbf{w}_c^\top [1, \mathbf{x}]}}{\sum_{c'} e^{\mathbf{w}_{c'}^\top [1, \mathbf{x}]}},$$

where $\mathbf{w}_c = [\gamma_c, \boldsymbol{\beta}_c]$.

The connection between LDA and logistic regression

Recall that in LDA, we have

$$\log p(y = c|\mathbf{x}, \mathbf{w}) = \gamma_c + \mathbf{x}^\top \boldsymbol{\beta}_c + \kappa$$

Then,

$$p(y = c|\mathbf{x}, \mathbf{w}) = \frac{e^{\gamma_c + \mathbf{x}^\top \boldsymbol{\beta}_c}}{\sum_{c'} e^{\gamma_{c'} + \mathbf{x}^\top \boldsymbol{\beta}_{c'}}} = \frac{e^{\mathbf{w}_c^\top [1, \mathbf{x}]}}{\sum_{c'} e^{\mathbf{w}_{c'}^\top [1, \mathbf{x}]}},$$

where $\mathbf{w}_c = [\gamma_c, \boldsymbol{\beta}_c]$.

To gain further insights, let's consider the binary case.

The posterior is given by

$$p(y = 1|\mathbf{x}, \mathbf{w})$$

The posterior is given by

$$p(y = 1|\mathbf{x}, \mathbf{w}) = \frac{e^{\gamma_1 + \boldsymbol{\beta}_1^\top \mathbf{x}}}{e^{\gamma_1 + \boldsymbol{\beta}_1^\top \mathbf{x}} + e^{\gamma_0 + \boldsymbol{\beta}_0^\top \mathbf{x}}}$$

The posterior is given by

$$p(y=1|\mathbf{x},\mathbf{w}) = \frac{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}}}{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}} + e^{\gamma_0+\boldsymbol{\beta}_0^\top \mathbf{x}}} = \frac{1}{1+e^{(\gamma_0-\gamma_1)+(\boldsymbol{\beta}_0-\boldsymbol{\beta}_1)^\top \mathbf{x}}}$$

The posterior is given by

$$p(y=1|\mathbf{x},\mathbf{w}) = \frac{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}}}{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}} + e^{\gamma_0+\boldsymbol{\beta}_0^\top \mathbf{x}}} = \frac{1}{1+e^{(\gamma_0-\gamma_1)+(\boldsymbol{\beta}_0-\boldsymbol{\beta}_1)^\top \mathbf{x}}} = \sigma\left((\boldsymbol{\beta}_1-\boldsymbol{\beta}_0)^\top \mathbf{x} + (\gamma_1-\gamma_0)\right)$$

The posterior is given by

$$p(y=1|\mathbf{x},\mathbf{w}) = \frac{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}}}{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}} + e^{\gamma_0+\boldsymbol{\beta}_0^\top \mathbf{x}}} = \frac{1}{1+e^{(\gamma_0-\gamma_1)+(\boldsymbol{\beta}_0-\boldsymbol{\beta}_1)^\top \mathbf{x}}} = \sigma\left((\boldsymbol{\beta}_1-\boldsymbol{\beta}_0)^\top \mathbf{x} + (\gamma_1-\gamma_0)\right)$$

Note that

$\gamma_1 - \gamma_0$

The posterior is given by

$$p(y=1|\mathbf{x},\mathbf{w}) = \frac{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}}}{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}} + e^{\gamma_0+\boldsymbol{\beta}_0^\top \mathbf{x}}} = \frac{1}{1+e^{(\gamma_0-\gamma_1)+(\boldsymbol{\beta}_0-\boldsymbol{\beta}_1)^\top \mathbf{x}}} = \sigma\left((\boldsymbol{\beta}_1-\boldsymbol{\beta}_0)^\top \mathbf{x} + (\gamma_1-\gamma_0)\right)$$

Note that

$$\gamma_1 - \gamma_0 = (\log \pi_1 - \frac{1}{2}\boldsymbol{\mu}_1^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_1) - (\log \pi_0 - \frac{1}{2}\boldsymbol{\mu}_0^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_0)$$

The posterior is given by

$$p(y=1|\mathbf{x},\mathbf{w}) = \frac{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}}}{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}} + e^{\gamma_0+\boldsymbol{\beta}_0^\top \mathbf{x}}} = \frac{1}{1+e^{(\gamma_0-\gamma_1)+(\boldsymbol{\beta}_0-\boldsymbol{\beta}_1)^\top \mathbf{x}}} = \sigma\left((\boldsymbol{\beta}_1-\boldsymbol{\beta}_0)^\top \mathbf{x} + (\gamma_1-\gamma_0)\right)$$

Note that

$$\gamma_1-\gamma_0 = (\log\pi_1 - \frac{1}{2}\boldsymbol{\mu}_1^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_1) - (\log\pi_0 - \frac{1}{2}\boldsymbol{\mu}_0^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_0) = -\frac{1}{2}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_0) + \log\frac{\pi_1}{\pi_0}$$

The posterior is given by

$$p(y=1|\mathbf{x},\mathbf{w}) = \frac{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}}}{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}} + e^{\gamma_0+\boldsymbol{\beta}_0^\top \mathbf{x}}} = \frac{1}{1+e^{(\gamma_0-\gamma_1)+(\boldsymbol{\beta}_0-\boldsymbol{\beta}_1)^\top \mathbf{x}}} = \sigma\left((\boldsymbol{\beta}_1-\boldsymbol{\beta}_0)^\top \mathbf{x} + (\gamma_1-\gamma_0)\right)$$

Note that

$$\gamma_1-\gamma_0 = (\log \pi_1 - \frac{1}{2}\boldsymbol{\mu}_1^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_1) - (\log \pi_0 - \frac{1}{2}\boldsymbol{\mu}_0^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_0) = -\frac{1}{2}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_0) + \log\frac{\pi_1}{\pi_0}$$

If we define

$$\mathbf{w} = \boldsymbol{\beta}_1 - \boldsymbol{\beta}_0 = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)$$

$$\mathbf{x}_0 = \frac{1}{2}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_0) - (\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)\frac{\log\frac{\pi_1}{\pi_0}}{(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)}$$

The posterior is given by

$$p(y=1|\mathbf{x},\mathbf{w}) = \frac{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}}}{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}} + e^{\gamma_0+\boldsymbol{\beta}_0^\top \mathbf{x}}} = \frac{1}{1+e^{(\gamma_0-\gamma_1)+(\boldsymbol{\beta}_0-\boldsymbol{\beta}_1)^\top \mathbf{x}}} = \sigma\left((\boldsymbol{\beta}_1-\boldsymbol{\beta}_0)^\top \mathbf{x} + (\gamma_1-\gamma_0)\right)$$

Note that

$$\gamma_1-\gamma_0 = (\log \pi_1 - \frac{1}{2}\boldsymbol{\mu}_1^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_1) - (\log \pi_0 - \frac{1}{2}\boldsymbol{\mu}_0^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_0) = -\frac{1}{2}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_0) + \log\frac{\pi_1}{\pi_0}$$

If we define

$$\mathbf{w} = \boldsymbol{\beta}_1 - \boldsymbol{\beta}_0 = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)$$

$$\mathbf{x}_0 = \frac{1}{2}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_0) - (\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)\frac{\log\frac{\pi_1}{\pi_0}}{(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)}$$

then we have $\mathbf{w}^\top \mathbf{x}_0 = -(\gamma_1-\gamma_0)$,

The posterior is given by

$$p(y=1|\mathbf{x},\mathbf{w}) = \frac{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}}}{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}} + e^{\gamma_0+\boldsymbol{\beta}_0^\top \mathbf{x}}} = \frac{1}{1+e^{(\gamma_0-\gamma_1)+(\boldsymbol{\beta}_0-\boldsymbol{\beta}_1)^\top \mathbf{x}}} = \sigma\left((\boldsymbol{\beta}_1-\boldsymbol{\beta}_0)^\top \mathbf{x} + (\gamma_1-\gamma_0)\right)$$

Note that

$$\gamma_1-\gamma_0 = (\log \pi_1 - \frac{1}{2}\boldsymbol{\mu}_1^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_1) - (\log \pi_0 - \frac{1}{2}\boldsymbol{\mu}_0^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_0) = -\frac{1}{2}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_0) + \log\frac{\pi_1}{\pi_0}$$

If we define

$$\mathbf{w} = \boldsymbol{\beta}_1 - \boldsymbol{\beta}_0 = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)$$

$$\mathbf{x}_0 = \frac{1}{2}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_0) - (\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)\frac{\log\frac{\pi_1}{\pi_0}}{(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)}$$

then we have $\mathbf{w}^\top \mathbf{x}_0 = -(\gamma_1-\gamma_0)$, and hence

$$p(y=1|\mathbf{x},\mathbf{w}) = \sigma\left(\mathbf{w}^\top(\mathbf{x}-\mathbf{x}_0)\right).$$

The posterior is given by

$$p(y=1|\mathbf{x},\mathbf{w}) = \frac{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}}}{e^{\gamma_1+\boldsymbol{\beta}_1^\top \mathbf{x}} + e^{\gamma_0+\boldsymbol{\beta}_0^\top \mathbf{x}}} = \frac{1}{1+e^{(\gamma_0-\gamma_1)+(\boldsymbol{\beta}_0-\boldsymbol{\beta}_1)^\top \mathbf{x}}} = \sigma\left((\boldsymbol{\beta}_1-\boldsymbol{\beta}_0)^\top \mathbf{x} + (\gamma_1-\gamma_0)\right)$$

Note that

$$\gamma_1-\gamma_0 = (\log \pi_1 - \frac{1}{2}\boldsymbol{\mu}_1^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_1) - (\log \pi_0 - \frac{1}{2}\boldsymbol{\mu}_0^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_0) = -\frac{1}{2}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_0) + \log\frac{\pi_1}{\pi_0}$$

If we define

$$\mathbf{w} = \boldsymbol{\beta}_1 - \boldsymbol{\beta}_0 = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)$$

$$\mathbf{x}_0 = \frac{1}{2}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_0) - (\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)\frac{\log\frac{\pi_1}{\pi_0}}{(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)}$$

then we have $\mathbf{w}^\top \mathbf{x}_0 = -(\gamma_1-\gamma_0)$, and hence

$$p(y=1|\mathbf{x},\mathbf{w}) = \sigma\left(\mathbf{w}^\top(\mathbf{x}-\mathbf{x}_0)\right) .$$

This has the same form as binary logistic regression, and the MAP decision rule is

$$\hat{y}(\mathbf{x}) = 1 \text{ iff } \mathbf{w}^\top \mathbf{x} > c\,, \text{ where } c = \mathbf{w}^\top \mathbf{x}_0$$

Geometry interpretation of LDA

Consider the MAP decision rule:

$$\hat{y}(\mathbf{x}) = 1 \text{ iff } \mathbf{w}^\top \mathbf{x} > c\,, \text{ where } c = \mathbf{w}^\top \mathbf{x}_0$$

Geometry interpretation of LDA

Consider the MAP decision rule:

$$\hat{y}(\mathbf{x}) = 1 \text{ iff } \mathbf{w}^\top \mathbf{x} > c\,, \text{ where } c = \mathbf{w}^\top \mathbf{x}_0$$

If $\pi_0 = \pi_1 = 0.5$, then $c = \frac{1}{2}\mathbf{w}^\top(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_0)$.

Geometry interpretation of LDA

Consider the MAP decision rule:

$$\hat{y}(\mathbf{x}) = 1 \text{ iff } \mathbf{w}^\top \mathbf{x} > c\,, \text{ where } c = \mathbf{w}^\top \mathbf{x}_0$$

If $\pi_0 = \pi_1 = 0.5$, then $c = \frac{1}{2}\mathbf{w}^\top(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_0)$.

To interpret this equation geometrically, suppose $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}$

Geometry interpretation of LDA

Consider the MAP decision rule:

$$\hat{y}(\mathbf{x}) = 1 \text{ iff } \mathbf{w}^\top \mathbf{x} > c\,, \text{ where } c = \mathbf{w}^\top \mathbf{x}_0$$

If $\pi_0 = \pi_1 = 0.5$, then $c = \frac{1}{2}\mathbf{w}^\top(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_0)$.

To interpret this equation geometrically, suppose $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}$
$\implies \mathbf{w} = \sigma^{-2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)$, which is parallel to a line joining the two centroids, $\boldsymbol{\mu}_0$ and $\boldsymbol{\mu}_1$.

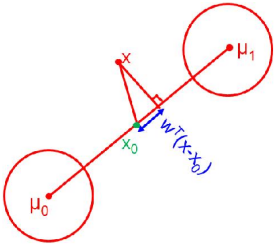
Geometry interpretation of LDA

Consider the MAP decision rule:

$$\hat{y}(\mathbf{x}) = 1 \text{ iff } \mathbf{w}^\top \mathbf{x} > c\,, \text{ where } c = \mathbf{w}^\top \mathbf{x}_0$$

If $\pi_0 = \pi_1 = 0.5$, then $c = \frac{1}{2}\mathbf{w}^\top(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_0)$.

To interpret this equation geometrically, suppose $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}$
$\implies \mathbf{w} = \sigma^{-2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)$, which is parallel to a line joining the two centroids, $\boldsymbol{\mu}_0$ and $\boldsymbol{\mu}_1$.



Geometry of LDA in the 2 class case where $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \mathbf{I}$.

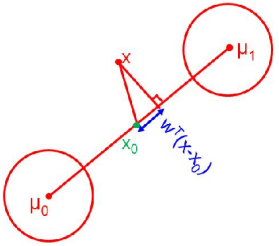
Geometry interpretation of LDA

Consider the MAP decision rule:

$$\hat{y}(\mathbf{x}) = 1 \text{ iff } \mathbf{w}^\top \mathbf{x} > c\,, \text{ where } c = \mathbf{w}^\top \mathbf{x}_0$$

If $\pi_0 = \pi_1 = 0.5$, then $c = \frac{1}{2}\mathbf{w}^\top(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_0)$.

To interpret this equation geometrically, suppose $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}$
$\Longrightarrow \mathbf{w} = \sigma^{-2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)$, which is parallel to a line joining the two centroids, $\boldsymbol{\mu}_0$ and $\boldsymbol{\mu}_1$.
$\Longrightarrow$ we can classify a point by projecting it onto this line, and check its distance to $\boldsymbol{\mu}_0$ and $\boldsymbol{\mu}_1$.



Geometry of LDA in the 2 class case where $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \mathbf{I}$.

Some remarks

GDA

- maximizes the **joint likelihood** $\prod_{i=1}^{n} p(\mathbf{x}^{(i)}, y^{(i)})$
- modeling assumptions:
  $\mathbf{x}|y = b \sim \mathcal{N}(\boldsymbol{\mu}_b, \boldsymbol{\Sigma}), y \sim \text{Bernoulli}(\phi)$
- when modeling assumptions are correct, GDA is asymptotically efficient[a] and data efficient.

[a] in the limit of very large training sets (large n), there is no algorithm that is strictly better than GDA.

Logistic Regression

- maximizes the **conditional likelihood** $\prod_{i=1}^{n} p(y^{(i)}|\mathbf{x}^{(i)})$
- modeling assumptions:
  $p(y|\mathbf{x})$ is a logistic function; no restriction on $p(\mathbf{x})$
- more robust and less sensitive to incorrect modeling assumptions.

Table of Contents

Maximimum Likelihood Estimation for GDA

The likelihood function of a GDA model is as follows:

$$p(\mathcal{D}|\mathbf{w}) = \prod_{i=1}^{N} \text{Cat}(y_i|\boldsymbol{\pi}) \prod_{c=1}^{C} \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)^{1\{y_i=c\}}$$

Maximimum Likelihood Estimation for GDA

The likelihood function of a GDA model is as follows:

$$p(\mathcal{D}|\mathbf{w}) = \prod_{i=1}^{N} \text{Cat}(y_i|\boldsymbol{\pi}) \prod_{c=1}^{C} \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)^{1\{y_i=c\}}$$

Hence the log-likelihood is given by

$$\log p(\mathcal{D}|\mathbf{w}) = \left[\sum_{i=1}^{n}\sum_{c=1}^{C} 1\{y_i = c\} \log \pi_c\right] + \sum_{c=1}^{C}\left[\sum_{i:y_i=c} \log \mathcal{N}\left(\mathbf{x}_i|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c\right)\right]$$

Maximimum Likelihood Estimation for GDA

The likelihood function of a GDA model is as follows:

$$p(\mathcal{D}|\mathbf{w}) = \prod_{i=1}^{N} \text{Cat}(y_i|\boldsymbol{\pi}) \prod_{c=1}^{C} \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)^{1\{y_i=c\}}$$

Hence the log-likelihood is given by

$$\log p(\mathcal{D}|\mathbf{w}) = \left[\sum_{i=1}^{n}\sum_{c=1}^{C} 1\{y_i = c\} \log \pi_c\right] + \sum_{c=1}^{C}\left[\sum_{i:y_i=c} \log \mathcal{N}\left(\mathbf{x}_i|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c\right)\right]$$

where we can optimize $\boldsymbol{\pi}$ and the $(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$ terms separately.

Maximimum Likelihood Estimation for GDA

The likelihood function of a GDA model is as follows:

$$p(\mathcal{D}|\mathbf{w}) = \prod_{i=1}^{N} \text{Cat}(y_i|\boldsymbol{\pi}) \prod_{c=1}^{C} \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)^{1\{y_i=c\}}$$

Hence the log-likelihood is given by

$$\log p(\mathcal{D}|\mathbf{w}) = \left[\sum_{i=1}^{n}\sum_{c=1}^{C} 1\{y_i = c\} \log \pi_c\right] + \sum_{c=1}^{C}\left[\sum_{i:y_i=c} \log \mathcal{N}\left(\mathbf{x}_i|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c\right)\right]$$

where we can optimize $\boldsymbol{\pi}$ and the $(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$ terms separately.
MLE estimation:

- $\hat{\pi}_c = \frac{N_c}{N}$
- $\hat{\boldsymbol{\mu}}_c = \frac{1}{N_c} \sum_{i:y_i=c} \mathbf{x}_n$
- $\hat{\boldsymbol{\Sigma}}_c = \sum_{i:y_i=c} (\mathbf{x}_i - \hat{\boldsymbol{\mu}}_c)(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_c)^\top$

A different version of derivatives for MLE of LDA

Log-likelihood of the data:

$\ell(\phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma})$

A different version of derivatives for MLE of LDA

Log-likelihood of the data:

$$\ell(\phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) = \log \prod_{i=1}^{n} p(\mathbf{x}^{(i)}, y^{(i)}; \phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma})$$

A different version of derivatives for MLE of LDA

Log-likelihood of the data:

$$
\begin{aligned}
&\ell(\phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) \\
&= \log \prod_{i=1}^{n} p(\mathbf{x}^{(i)}, y^{(i)}; \phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) \\
&= \log \prod_{i=1}^{n} p(\mathbf{x}^{(i)} | y^{(i)}; \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) p(y^{(i)}; \phi)
\end{aligned}
$$

A different version of derivatives for MLE of LDA

Log-likelihood of the data:

$$
\begin{aligned}
&\ell(\phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) \\
&= \log \prod_{i=1}^{n} p(\mathbf{x}^{(i)}, y^{(i)}; \phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) \\
&= \log \prod_{i=1}^{n} p(\mathbf{x}^{(i)} | y^{(i)}; \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) p(y^{(i)}; \phi) \\
&= \sum_{i=1}^{n} \log p(\mathbf{x}^{(i)} | y^{(i)}; \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) + \sum_{i=1}^{n} \log p(y^{(i)}; \phi)
\end{aligned}
$$

A different version of derivatives for MLE of LDA

Log-likelihood of the data:

$$
\begin{aligned}
&\ell(\phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) \\
&= \log \prod_{i=1}^{n} p(\mathbf{x}^{(i)}, y^{(i)}; \phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) \\
&= \log \prod_{i=1}^{n} p(\mathbf{x}^{(i)} | y^{(i)}; \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) p(y^{(i)}; \phi) \\
&= \sum_{i=1}^{n} \log p(\mathbf{x}^{(i)} | y^{(i)}; \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) + \sum_{i=1}^{n} \log p(y^{(i)}; \phi) \\
&= \sum_{i=1}^{n} \log \frac{1}{\sqrt{(2\pi)^d \, det(\boldsymbol{\Sigma})}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_{y^i})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}_{y^{(i)}}) \right\} + \sum_{i=1}^{n} y^{(i)} \log \phi + \sum_{i=1}^{n} (1 - y^{(i)}) \log(1 - \phi)
\end{aligned}
$$

Log-likelihood of the data:

$$
\begin{aligned}
&\ell(\phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) \\
&= \log \prod_{i=1}^{n} p(\mathbf{x}^{(i)}, y^{(i)}; \phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) \\
&= \sum_{i=1}^{n} \log \frac{1}{\sqrt{(2\pi)^d \, det(\boldsymbol{\Sigma})}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_{y^i})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}_{y^{(i)}}) \right\} + \sum_{i=1}^{n} y^{(i)} \log \phi + \sum_{i=1}^{n} (1 - y^{(i)}) \log(1 - \phi)
\end{aligned}
$$

Log-likelihood of the data:

$$
\begin{aligned}
&\ell(\phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) \\
&= \log \prod_{i=1}^{n} p(\mathbf{x}^{(i)}, y^{(i)}; \phi, \boldsymbol{\mu}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) \\
&= \sum_{i=1}^{n} \log \frac{1}{\sqrt{(2\pi)^d \, det(\boldsymbol{\Sigma})}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_{y^i})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}_{y^{(i)}}) \right\} + \sum_{i=1}^{n} y^{(i)} \log \phi + \sum_{i=1}^{n} (1 - y^{(i)}) \log(1 - \phi)
\end{aligned}
$$

By maximizing ℓ w.r.t. the parameters, the maximumn likelihood estimate of the parameters:

$$
\begin{aligned}
\phi &= \frac{1}{n} \sum_{i=1}^{n} 1\{y^{(i)} = 1\} \\
\boldsymbol{\mu}_b &= \frac{\sum_{i=1}^{n} 1\{y^{(i)} = b\} \mathbf{x}^{(i)}}{\sum_{i=1}^{n} 1\{y^{(i)} = b\}} \text{ for } b = 0, 1 \\
\boldsymbol{\Sigma} &= \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_{y^{(i)}})(\mathbf{x}^{(i)} - \boldsymbol{\mu}_{y^{(i)}})^\top
\end{aligned}
$$

Table of Contents

In GDA, the feature vectors $\mathbf{x}$ were continuous, real-valued vectors.

In GDA, the feature vectors $\mathbf{x}$ were continuous, real-valued vectors.

Let's talk about a different learning algorithm in which x_j's are discrete-valued.

In GDA, the feature vectors $\mathbf{x}$ were continuous, real-valued vectors.

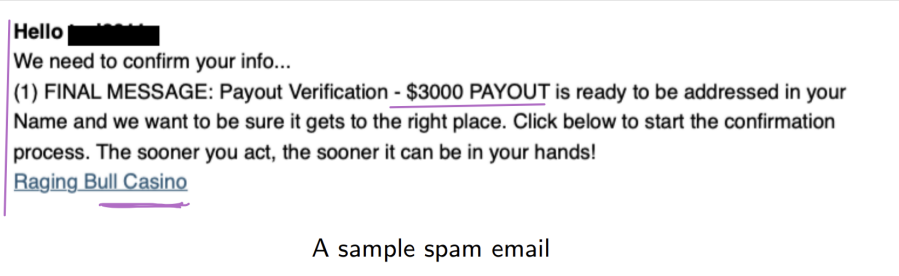
Let's talk about a different learning algorithm in which x_j's are discrete-valued.

$\Longrightarrow$ **Naïve Bayes**

Naïve Bayes: Motivating Example

Example 2 (Spam filter (document classification))

Classify email message $\mathbf{x}$ to spam ($y = 1$) and non-spam ($y = 0$) classes.



A sample spam email

Example: Spam Filter

We would like to represent an email via a feature vector.

Binary text features: Given a dictionary of size n, represent a message composed of dictionary words as $\mathbf{x} \in \{0,1\}^n$:

$$x_i = \begin{cases} 1 & i\text{-th dictionary word is in message} \\ 0 & \text{otherwise} \end{cases}$$

$$x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} \begin{matrix} \text{a} \\ \text{aardvark} \\ \text{aardwolf} \\ \vdots \\ \text{buy} \\ \vdots \\ \text{zygmurgy} \end{matrix}$$

Example: Spam Filter

We would like to represent an email via a feature vector.

Binary text features: Given a dictionary of size n, represent a message composed of dictionary words as $\mathbf{x} \in \{0,1\}^n$:

$$x_i = \begin{cases} 1 & i\text{-th dictionary word is in message} \\ 0 & \text{otherwise} \end{cases}$$

To build a generative model, we have to model $p(\mathbf{x}|y)$

$$x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} \begin{matrix} \text{a} \\ \text{aardvark} \\ \text{aardwolf} \\ \vdots \\ \text{buy} \\ \vdots \\ \text{zygmurgy} \end{matrix}$$

Naïve Bayes assumption and Naïve Bayes classifier

Probability of observing email $x_1, \ldots, x_n$ given spam class y:

$$p(x_1, \ldots, x_n|y) = p(x_1|y)p(x_2|x_1, y), \ldots, p(x_n|x_1, \ldots x_{n-1}, y)$$

Naïve Bayes assumption and Naïve Bayes classifier

Probability of observing email $x_1, \ldots, x_n$ given spam class y:

$$p(x_1, \ldots, x_n|y) = p(x_1|y)p(x_2|x_1, y), \ldots, p(x_n|x_1, \ldots x_{n-1}, y)$$

To model $p(\mathbf{x}|y)$, we will therefore make a very strong assumption.

Assumption 1 (Naïve Bayes assumption)

x_i's are conditionally independent given y:

$$p(x_i|y, x_1, \ldots, x_{i-1}) = p(x_i|y)$$

Naïve Bayes assumption and Naïve Bayes classifier

Probability of observing email $x_1, \ldots, x_n$ given spam class y:

$$p(x_1, \ldots, x_n|y) = p(x_1|y)p(x_2|x_1, y), \ldots, p(x_n|x_1, \ldots x_{n-1}, y)$$

To model $p(\mathbf{x}|y)$, we will therefore make a very strong assumption.

Assumption 1 (Naïve Bayes assumption)

x_i's are conditionally independent given y:

$$p(x_i|y, x_1, \ldots, x_{i-1}) = p(x_i|y)$$

As a result,

$$p(x_1, \ldots, x_n|y) = p(x_1|y)p(x_2|y) \ldots p(x_n|y) = \prod_{i=1}^{n} p(x_i|y)$$

Naïve Bayes Parameters

The Spam filter problem indeed is a Multi-variate Bernoulli event model

Naïve Bayes Parameters

The Spam filter problem indeed is a Multi-variate Bernoulli event model

Namely, $\mathbf{x}|y$ is generated from n independent Bernoulli trials

$$p(\mathbf{x}, y) = p(y)p(\mathbf{x}|y) = p(y)\prod_{i=1}^{n} p(x_i|y)$$

Naïve Bayes Parameters

The Spam filter problem indeed is a Multi-variate Bernoulli event model

Namely, $\mathbf{x}|y$ is generated from n independent Bernoulli trials

$$p(\mathbf{x}, y) = p(y)p(\mathbf{x}|y) = p(y)\prod_{i=1}^{n} p(x_i|y)$$

- $y \sim \text{Bernoulli}(\phi_y)$

Naïve Bayes Parameters

The Spam filter problem indeed is a Multi-variate Bernoulli event model

Namely, $\mathbf{x}|y$ is generated from n independent Bernoulli trials

$$p(\mathbf{x}, y) = p(y)p(\mathbf{x}|y) = p(y)\prod_{i=1}^{n} p(x_i|y)$$

- $y \sim$ Bernoulli(ϕ_y)
  assume email class (spam v.s. no-spam) is randomly generated with prior $p(y) = \phi_y^y(1-\phi_y)^{1-y}$

Naïve Bayes Parameters

The Spam filter problem indeed is a Multi-variate Bernoulli event model

Namely, $\mathbf{x}|y$ is generated from n independent Bernoulli trials

$$p(\mathbf{x}, y) = p(y)p(\mathbf{x}|y) = p(y)\prod_{i=1}^{n} p(x_i|y)$$

- $y \sim$ Bernoulli(ϕ_y)
  assume email class (spam v.s. no-spam) is randomly generated with prior $p(y) = \phi_y^y(1-\phi_y)^{1-y}$
- $x_i|y = b \sim$ Bernoulli$(\phi_{i|y=b}), b = 0, 1$:

Naïve Bayes Parameters

The Spam filter problem indeed is a Multi-variate Bernoulli event model

Namely, $\mathbf{x}|y$ is generated from n independent Bernoulli trials

$$p(\mathbf{x}, y) = p(y)p(\mathbf{x}|y) = p(y)\prod_{i=1}^{n} p(x_i|y)$$

- $y \sim \text{Bernoulli}(\phi_y)$
  assume email class (spam v.s. no-spam) is randomly generated with prior $p(y) = \phi_y^y(1-\phi_y)^{1-y}$
- $x_i|y = b \sim \text{Bernoulli}(\phi_{i|y=b}), b = 0, 1$:
  given $y = b$, each word x_i is included in the message independently with $p(x_i = 1|y = b) = \phi_{i|y=b}$.

Naïve Bayes Parameters

The Spam filter problem indeed is a Multi-variate Bernoulli event model

Namely, $\mathbf{x}|y$ is generated from n independent Bernoulli trials

$$p(\mathbf{x}, y) = p(y)p(\mathbf{x}|y) = p(y)\prod_{i=1}^{n} p(x_i|y)$$

- $y \sim \text{Bernoulli}(\phi_y)$
  assume email class (spam v.s. no-spam) is randomly generated with prior $p(y) = \phi_y^y(1-\phi_y)^{1-y}$
- $x_i|y = b \sim \text{Bernoulli}(\phi_{i|y=b}), b = 0, 1$:
  given $y = b$, each word x_i is included in the message independently with $p(x_i = 1|y = b) = \phi_{i|y=b}$.

I.e., $\quad p(x_i|y = b) = \phi_{i|y=b}^{x_i}(1-\phi_{i|y=b})^{1-x_i}$

Naïve Bayes Parameters

The Spam filter problem indeed is a Multi-variate Bernoulli event model

Namely, $\mathbf{x}|y$ is generated from n independent Bernoulli trials

$$p(\mathbf{x}, y) = p(y)p(\mathbf{x}|y) = p(y)\prod_{i=1}^{n} p(x_i|y)$$

- $y \sim \text{Bernoulli}(\phi_y)$
  assume email class (spam v.s. no-spam) is randomly generated with prior $p(y) = \phi_y^y(1-\phi_y)^{1-y}$
- $x_i|y = b \sim \text{Bernoulli}(\phi_{i|y=b}), b = 0, 1$:
  given $y = b$, each word x_i is included in the message independently with $p(x_i = 1|y = b) = \phi_{i|y=b}$.

$$\text{I.e.,} \qquad p(x_i|y = b) = \phi_{i|y=b}^{x_i}(1-\phi_{i|y=b})^{1-x_i}$$

Model parameters:

- $\phi_y \in \mathbb{R}$
- $\phi_{i|y=1}, \phi_{i|y=0}$ for $i = 1, \ldots, n$

Naïve Bayes Parameter Learning

Likelihood of i.i.d. training data $(\mathbf{x}^{(1)}, y^{(1)}), \ldots, (\mathbf{x}^{(n)}, y^{(n)})$:

$$\ell(\phi_y, \phi_{i|y=1}, \phi_{i|y=0}) = \prod_{i=1}^{n} p(\mathbf{x}^{(i)}, y^{(i)})$$

Naïve Bayes Parameter Learning

Likelihood of i.i.d. training data $(\mathbf{x}^{(1)}, y^{(1)}), \ldots, (\mathbf{x}^{(n)}, y^{(n)})$:

$$\ell(\phi_y, \phi_{i|y=1}, \phi_{i|y=0}) = \prod_{i=1}^{n} p(\mathbf{x}^{(i)}, y^{(i)})$$

Maximum likelihood estimation of parameters:

$$\phi_y = \frac{1}{n} \sum_{i=1}^{n} 1\{y^{(i)} = 1\} \qquad \text{(\% of spam emails)}$$

$$\phi_{i|y=b} = \frac{\sum_{i=1}^{n} 1\{x_j^{(i)} = 1, y^{(i)} = b\}}{\sum_{i=1}^{n} 1\{y^{(i)} = b\}} \text{ for } b = 1, 0$$

(\% of spam (non-spam) emails containing j-th dictionary word)

Naïve Bayes Parameter Prediction

Given new example with feature $\mathbf{x}$, compute the posterior probability

$$p(y = 1|\mathbf{x})$$

Naïve Bayes Parameter Prediction

Given new example with feature $\mathbf{x}$, compute the posterior probability

$$p(y = 1|\mathbf{x}) = \frac{p(\mathbf{x}|y = 1)p(y = 1)}{p(\mathbf{x})}$$

Naïve Bayes Parameter Prediction

Given new example with feature $\mathbf{x}$, compute the posterior probability

$$\begin{aligned} p(y=1|\mathbf{x}) &= \frac{p(\mathbf{x}|y=1)p(y=1)}{p(\mathbf{x})} \\ &= \frac{p(\mathbf{x}|y=1)p(y=1)}{p(\mathbf{x}|y=1)p(y=1)+p(\mathbf{x}|y=0)p(y=0)} \end{aligned}$$

Naïve Bayes Parameter Prediction

Given new example with feature $\mathbf{x}$, compute the posterior probability

$$
\begin{aligned}
p(y=1|\mathbf{x}) &= \frac{p(\mathbf{x}|y=1)p(y=1)}{p(\mathbf{x})} \\
&= \frac{p(\mathbf{x}|y=1)p(y=1)}{p(\mathbf{x}|y=1)p(y=1)+p(\mathbf{x}|y=0)p(y=0)} \\
&= \frac{\prod_{j=1}^{d} p(x_j|y=1)p(y=1)}{\left(\prod_{j=1}^{d} p(x_j|y=1)\right)p(y=1)+\left(\prod_{j=1}^{d} p(x_j|y=0)\right)p(y=0)}
\end{aligned}
$$

Naïve Bayes Parameter Prediction

Given new example with feature $\mathbf{x}$, compute the posterior probability

$$\begin{aligned} p(y=1|\mathbf{x}) &= \frac{p(\mathbf{x}|y=1)p(y=1)}{p(\mathbf{x})} \\ &= \frac{p(\mathbf{x}|y=1)p(y=1)}{p(\mathbf{x}|y=1)p(y=1)+p(\mathbf{x}|y=0)p(y=0)} \\ &= \frac{\prod_{j=1}^{d} p(x_j|y=1)p(y=1)}{\left(\prod_{j=1}^{d} p(x_j|y=1)\right)p(y=1)+\left(\prod_{j=1}^{d} p(x_j|y=0)\right)p(y=0)} \end{aligned}$$

Choose label $y=1$ (spam) if $p(y=1|x) > T$, where $T \in [0,1]$ is a threshold, e.g., $T=0.5$.

Naïve Bayes Parameter Prediction

Given new example with feature $\mathbf{x}$, compute the posterior probability

$$
\begin{aligned}
p(y=1|\mathbf{x}) &= \frac{p(\mathbf{x}|y=1)p(y=1)}{p(\mathbf{x})} \\
&= \frac{p(\mathbf{x}|y=1)p(y=1)}{p(\mathbf{x}|y=1)p(y=1)+p(\mathbf{x}|y=0)p(y=0)} \\
&= \frac{\prod_{j=1}^{d} p(x_j|y=1)p(y=1)}{\left(\prod_{j=1}^{d} p(x_j|y=1)\right)p(y=1)+\left(\prod_{j=1}^{d} p(x_j|y=0)\right)p(y=0)}
\end{aligned}
$$

Choose label $y=1$ (spam) if $p(y=1|x) > T$, where $T \in [0,1]$ is a threshold, e.g., $T=0.5$.
$\Longrightarrow$ T trade-off between wrongly blocked non-spam (FPs) v.s. wrongly blocked spams (FNs).

Naïve Bayes Parameter Prediction

Given new example with feature $\mathbf{x}$, compute the posterior probability

$$
\begin{aligned}
p(y=1|\mathbf{x}) &= \frac{p(\mathbf{x}|y=1)p(y=1)}{p(\mathbf{x})} \\
&= \frac{p(\mathbf{x}|y=1)p(y=1)}{p(\mathbf{x}|y=1)p(y=1)+p(\mathbf{x}|y=0)p(y=0)} \\
&= \frac{\prod_{j=1}^{d} p(x_j|y=1)p(y=1)}{\left(\prod_{j=1}^{d} p(x_j|y=1)\right)p(y=1)+\left(\prod_{j=1}^{d} p(x_j|y=0)\right)p(y=0)}
\end{aligned}
$$

Choose label $y=1$ (spam) if $p(y=1|x) > T$, where $T \in [0,1]$ is a threshold, e.g., $T=0.5$.
$\Longrightarrow$ T trade-off between wrongly blocked non-spam (FPs) v.s. wrongly blocked spams (FNs).

In case of taking the x_j value in $\{1,\ldots,k_j\}$,
we can model $p(x_j|y)$ as multi-nomial distribution, rather than as Bernoulli.

Laplace smoothing

Issue with Naïve Bayes prediction:

- Suppose word x_j hasn't been seen in the training data,

Laplace smoothing

Issue with Naïve Bayes prediction:

- Suppose word x_j hasn't been seen in the training data,

(exercise) $\phi_{j|y=1} = \phi_{j|y=0}$

Laplace smoothing

Issue with Naïve Bayes prediction:

- Suppose word x_j hasn't been seen in the training data,

$$\text{(exercise) } \phi_{j|y=1} = \phi_{j|y=0} = \frac{\sum_{i=1}^{n} 1\{x_j^{(i)} = 1, y^{(i)} = b\}}{\sum_{i=1}^{n} 1\{y^{(i)} = b\}} = 0\,, \qquad \text{for } b = 1, 0$$

Laplace smoothing

Issue with Naïve Bayes prediction:

- Suppose word x_j hasn't been seen in the training data,

$$\text{(exercise) } \phi_{j|y=1} = \phi_{j|y=0} = \frac{\sum_{i=1}^{n} 1\{x_j^{(i)} = 1, y^{(i)} = b\}}{\sum_{i=1}^{n} 1\{y^{(i)} = b\}} = 0\,, \qquad \text{for } b = 1, 0$$

- We cannot compute class posterior

$$p(y = 1|\mathbf{x}) = \frac{\prod_{j=1}^{d} p(x_j|y = 1)p(y = 1)}{\left(\prod_{j=1}^{d} p(x_j|y = 1)\right) p(y = 1) + \left(\prod_{j=1}^{d} p(x_j|y = 0)\right) p(y = 0)} = \frac{0}{0}$$

Laplace smoothing

It is statistically a bad idea to estimate the probability of some event to be zero, just because you haven't seen it before in your finite training set.

Laplace smoothing

It is statistically a bad idea to estimate the probability of some event to be zero, just because you haven't seen it before in your finite training set.

To estimate the mean of a multi-nomial random variable z taking values in $\{1, \ldots, k\}$:

Laplace smoothing

It is statistically a bad idea to estimate the probability of some event to be zero, just because you haven't seen it before in your finite training set.

To estimate the mean of a multi-nomial random variable z taking values in $\{1, \ldots, k\}$:

- we can parameterize our multi-nomial with $\phi_j = p(z = j)$.

Laplace smoothing

It is statistically a bad idea to estimate the probability of some event to be zero, just because you haven't seen it before in your finite training set.

To estimate the mean of a multi-nomial random variable z taking values in $\{1, \ldots, k\}$:

- we can parameterize our multi-nomial with $\phi_j = p(z = j)$.
- given a set of n independent observations $\{z^{(1)}, \ldots, z^{(n)}\}$, the maximum likelihood estimates are

$$\phi_j = \frac{\sum_{i=1}^{n} 1\{z^{(i)} = j\}}{n},$$

Laplace smoothing

It is statistically a bad idea to estimate the probability of some event to be zero, just because you haven't seen it before in your finite training set.

To estimate the mean of a multi-nomial random variable z taking values in $\{1, \ldots, k\}$:

- we can parameterize our multi-nomial with $\phi_j = p(z = j)$.
- given a set of n independent observations $\{z^{(1)}, \ldots, z^{(n)}\}$, the maximum likelihood estimates are

$$\phi_j = \frac{\sum_{i=1}^{n} 1\{z^{(i)} = j\}}{n},$$

with the risk that some of the ϕ_j's might end up as zero.

Laplace smoothing

It is statistically a bad idea to estimate the probability of some event to be zero, just because you haven't seen it before in your finite training set.

To estimate the mean of a multi-nomial random variable z taking values in $\{1, \ldots, k\}$:

- we can parameterize our multi-nomial with $\phi_j = p(z = j)$.
- given a set of n independent observations $\{z^{(1)}, \ldots, z^{(n)}\}$, the maximum likelihood estimates are

$$\phi_j = \frac{\sum_{i=1}^{n} 1\{z^{(i)} = j\}}{n},$$

  with the risk that some of the ϕ_j's might end up as zero.
- Laplace smoothing can be used to eliminate this issue:

$$\phi_j = \frac{1 + \sum_{i=1}^{n} 1\{z^{(i)} = j\}}{k + n} \qquad \text{where}$$

  - $\phi_j \neq 0$ for all j
  - $\sum_{j=1}^{k} \phi_j = 1$

Naïve Bayes with Laplace smoothing

Apply Laplace smoothing to $\phi_{j|y=b}$ for $b \in \{0, 1\}$

Naïve Bayes with Laplace smoothing

Apply Laplace smoothing to $\phi_{j|y=b}$ for $b \in \{0, 1\}$

$$p(x_j = 1|y = b) = \phi_{j|y=b} = \frac{\sum_{i=1}^{n} 1\{x_j^{(i)} = 1, y^{(i)} = b\} + 1}{\sum_{i=1}^{n} 1\{y^{(i)} = b\} + 2}$$

Naïve Bayes with Laplace smoothing

Apply Laplace smoothing to $\phi_{j|y=b}$ for $b \in \{0, 1\}$

$$p(x_j = 1|y = b) = \phi_{j|y=b} = \frac{\sum_{i=1}^{n} 1\{x_j^{(i)} = 1, y^{(i)} = b\} + 1}{\sum_{i=1}^{n} 1\{y^{(i)} = b\} + 2}$$

- In practice we don't apply Laplace smoothing to $\phi_y = p(y = 1)$, which is greater than 0.

Last lecture:

- Exponential Families
- Generalized Linear Models

Last lecture:

- Exponential Families
- Generalized Linear Models

This lecture:

- Discriminative vs. Generative learning algorithms
- Gaussian Discriminant Analysis (GDA)
- Linear Discriminant Analysis (LDA)
- LDA and Logistic regression
- Naïve Bayes