Lecture 10: Logistic Regression

COMP90049
Introduction to Machine Learning

Semester 2, 2022

Lida Rashidi, CIS

Acknowledgement: Lea Frermann

Roadmap

Sofar...

- Naive Bayes + KNN
- Optimization (closed-form and iterative)
- Evaluation + Feature Selection

Today : more classification!

- Logistic Regression

Logistic Regression

Quick Refresher

Recall **Naive Bayes**

$$P(x, y) = P(y)P(x|y) \quad = \prod_{i=1}^{N} P(y^i) \prod_{m=1}^{M} P(x_m^i|y^i)$$

- a **probabilistic generative model** of the joint probability $P(x, y)$
- optimized to maximize the likelihood of the observed data
- **naive** due to unrealistic feature indepencence assumptions

Quick Refresher

Recall **Naive Bayes**

$$P(x, y) = P(y)P(x|y) \quad = \prod_{i=1}^{N} P(y^i) \prod_{m=1}^{M} P(x_m^i|y^i)$$

- a **probabilistic generative model** of the joint probability $P(x, y)$
- optimized to maximize the likelihood of the observed data
- **naive** due to unrealistic feature indepencence assumptions

For **prediction**, we apply **Bayes Rule** to obtain the conditional distribution

$$P(x, y) = P(y)P(x|y) = P(y|x)P(x)$$

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

$$\hat{y} = \operatorname*{argmax}_y P(y|x) \approx P(y)P(x|y)$$

How about we model $P(y|x)$ directly? $\rightarrow$ **Logistic Regression**

Logistic Regression on a high level

- Is a **binary** classification model
- Is a **probabilistic discriminative model** because it optimizes $P(y|x)$ directly
- Learns to optimally discriminate between inputs which belong to different classes
- No model of $P(x|y) \rightarrow$ no conditional feature independence assumption

Aside: Linear Regression

- Regression: predict a real-valued quantity y given features x, e.g.,

`housing price`	given	{`location`, `size`, `age`, ...}
`success of movie` ($)	given	{`cast`, `genre`, `budget`, ...}
`air quality`	given	{`temperature`, `timeOfDay`, `CO2`, ...}

Aside: Linear Regression

- Regression: predict a real-valued quantity y given features x, e.g.,

`housing price`	given	$\{$`location, size, age, ...`$\}$
`success of movie` (\$)	given	$\{$`cast, genre, budget, ...`$\}$
`air quality`	given	$\{$`temperature, timeOfDay, CO2, ...`$\}$

- **linear regression** is the simples regression model
- a real-valued $\hat{y}$ is predicted as a linear combination of weighted feature values

$$\begin{aligned}\hat{y} &= \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots \\ &= \theta_0 + \sum_i \theta_i x_i\end{aligned}$$

- The weights $\theta_0, \theta_1, \dots$ are model parameters, and need to be optimized during training
- Loss (error) is the sum of squared errors (SSE): $L = \sum_{i=1}^{N} (\hat{y}^i - y^i)^2$

Logistic Regression: Derivation I

- Let's assume a **binary** classification task, y is `true` (1) or `false` (0).
- We model **probabilites** $P(y = 1|x; \theta) = p(x)$ as a function of observations x under parameters θ. [What about $P(y = 0|x; \theta)$?]
- We want to use a **regression** approach

Logistic Regression: Derivation I

- Let's assume a **binary** classification task, y is `true` (1) or `false` (0).
- We model **probabilites** $P(y = 1|x; \theta) = p(x)$ as a function of observations x under parameters θ. [What about $P(y = 0|x; \theta)$?]
- We want to use a **regression** approach

Logistic Regression: Derivation I

- Let's assume a **binary** classification task, y is `true` (1) or `false` (0).
- We model **probabilites** $P(y = 1|x; \theta) = p(x)$ as a function of observations x under parameters θ. [What about $P(y = 0|x; \theta)$?]
- We want to use a **regression** approach

- How about: $p(x)$ as a linear function of x. Problem: probabilities are bounded in 0 and 1, linear functions are not.

$$p(x) = \theta_0 + \theta_1 x_1 + ...\theta_F x_F$$

Logistic Regression: Derivation I

- Let's assume a **binary** classification task, y is `true` (1) or `false` (0).
- We model **probabilites** $P(y = 1|x; \theta) = p(x)$ as a function of observations x under parameters θ. [What about $P(y = 0|x; \theta)$?]
- We want to use a **regression** approach

- ~~How about: $p(x)$ as a linear function of x. Problem: probabilities are bounded in 0 and 1, linear functions are not.~~

Logistic Regression: Derivation I

- Let's assume a **binary** classification task, y is `true` (1) or `false` (0).
- We model **probabilites** $P(y=1|x;\theta) = p(x)$ as a function of observations x under parameters θ. [What about $P(y=0|x;\theta)$?]
- We want to use a **regression** approach

- ~~How about: $p(x)$ as a linear function of x. Problem: probabilities are bounded in 0 and 1, linear functions are not.~~
- How about: *log* $p(x)$ as a linear function of x. Problem: *log* is bounded in one direction, linear functions are not.

$$\log p(x) = \theta_0 + \theta_1 x_1 + \ldots \theta_F x_F$$

Logistic Regression: Derivation I

- Let's assume a **binary** classification task, y is `true` (1) or `false` (0).
- We model **probabilites** $P(y=1|x;\theta)=p(x)$ as a function of observations x under parameters θ. [What about $P(y=0|x;\theta)$?]
- We want to use a **regression** approach

- ~~How about: $p(x)$ as a linear function of x. Problem: probabilities are bounded in 0 and 1, linear functions are not.~~
- ~~How about: *log* $p(x)$ as a linear function of x. Problem: *log* is bounded in one direction, linear functions are not.~~

Logistic Regression: Derivation I

- Let's assume a **binary** classification task, y is `true` (1) or `false` (0).
- We model **probabilites** $P(y=1|x;\theta) = p(x)$ as a function of observations x under parameters θ. [What about $P(y=0|x;\theta)$?]
- We want to use a **regression** approach

- ~~How about: $p(x)$ as a linear function of x. Problem: probabilities are bounded in 0 and 1, linear functions are not.~~
- ~~How about: *log* $p(x)$ as a linear function of x. Problem: *log* is bounded in one direction, linear functions are not.~~
- How about: minimally modifying *log* $p(x)$ such that is is unbounded, by applying the **logistic** transformation

$$log\frac{p(x)}{1-p(x)} = \theta_0 + \theta_1 x_1 + ...\theta_F x_F$$

Logistic Regression: Derivation II

$$log \frac{p(x)}{1-p(x)} = \theta_0 + \theta_1 x_1 + ... \theta_F x_F$$

- also called the **log odds**
- the **odds** are defined as the fraction of success over the fraction of failures

$$odds = \frac{P(success)}{P(failures)} = \frac{P(success)}{1-P(success)}$$

- e.g., the odds of rolling a 6 with a fair dice are:

$$\frac{1/6}{1-(1/6)} = \frac{0.17}{0.83} = 0.2$$

Logistic Regression: Derivation III

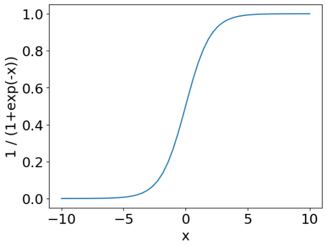
$$log \frac{P(x)}{1 - P(x)} = \theta_0 + \theta_1 x_1 + ... \theta_F x_F$$

If we rearrange and solve for $P(x)$, we get

$$P(x) = \frac{exp(\theta_0 + \theta_1 x_1 + ... \theta_F x_F)}{1 + exp(\theta_0 + \theta_1 x_1 + ... \theta_F x_F)} = \frac{exp(\theta_0 + \sum_{f=1}^{F} \theta_f x_f)}{1 + exp(\theta_0 + \sum_{f=1}^{F} \theta_f x_f)}$$

$$= \frac{1}{1 + exp(-(\theta_0 + \theta_1 x_1 + ... \theta_F x_F))} = \frac{1}{1 + exp(-(\theta_0 + \sum_{f=1}^{F} \theta_f x_f))}$$

- where the RHS is the inverse logit (or **logistic function**)
- we pass a regression model through the logistic function to obtain a valid probability predicton

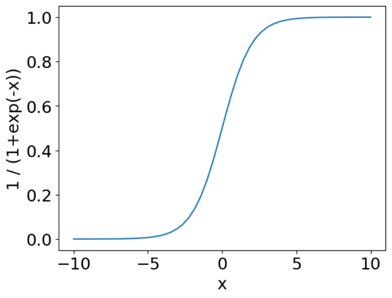


Logistic Regression: Interpretation

$$P(y|x;\theta) = \frac{1}{1 + exp(-(\theta_0 + \sum_{f=1}^{F} \theta_f x_f))}$$

A closer look at the logistic function

Most inputs lead to $P(y|x)$=0 or $P(y|x)$=1. That is intended, because all true labels are either 0 or 1.



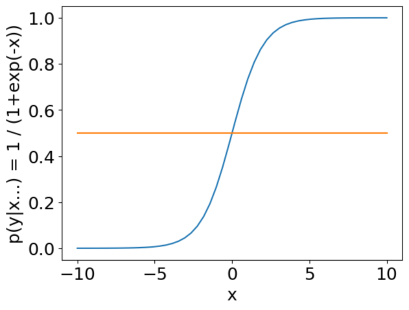
- $(\theta_0 + \sum_{f=1}^{F} \theta_f x_f) > 0$ means $y = 1$
- $(\theta_0 + \sum_{f=1}^{F} \theta_f x_f) \approx 0$ means most uncertainty
- $(\theta_0 + \sum_{f=1}^{F} \theta_f x_f) < 0$ means $y = 0$

Logistic Regression: Prediction

- The logistic function returns the probability of $P(y = 1)$ given an in put x

$$P(y = 1|x_1, x_2, ..., x_F; \theta) = \frac{1}{1 + \exp(-(\theta_0 + \sum_{f=1}^{F} \theta_f x_f))} = \sigma(x; \theta)$$

- We define a **decision boundary**, e.g., predict $y = 1$ if $P(y = 1|x_1, x_2, ..., x_F; \theta) > 0.5$ and $y = 0$ otherwise



Example!

$$P(y=1|x_1,x_2,\ldots,x_F;\theta) = \frac{1}{1+\exp(-(\theta_0+\sum_{f=1}^{F}\theta_f x_f))} = \frac{1}{1+\exp(-(\theta^T x))} = \sigma(\theta^T x)$$

Model parameters

$$\theta = [0.1,\ -3.5,\ 0.7,\ 2.1]$$

(Small) Test Data set

Outlook	Temp	Humidity	Class
rainy	*cool*	*normal*	0
sunny	*hot*	*high*	1

Feature Function

$$x_0 = 1 \text{ (bias term)}$$

$$x_1 = \begin{cases} 1 \text{ if outlook=sunny} \\ 2 \text{ if outlook=overcast} \\ 3 \text{ if outlook=rainy} \end{cases}$$

$$x_2 = \begin{cases} 1 \text{ if temp=hot} \\ 2 \text{ if temp=mild} \\ 3 \text{ if temp=cool} \end{cases}$$

$$x_3 = \begin{cases} 1 \text{ if humidity=normal} \\ 2 \text{ if humidity=high} \end{cases}$$

Example!

$$P(y=1|x_1, x_2, \ldots, x_F; \theta) = \frac{1}{1+\exp(-(\theta_0+\sum_{f=1}^{F}\theta_f x_f))} = \frac{1}{1+\exp(-(\theta^T x))} = \sigma(\theta^T x)$$

Model parameters

$$\theta = [0.1,\ -3.5,\ 0.7,\ 2.1]$$

(Small) Test Data set

Outlook	Temp	Humidity	Class
rainy	*cool*	*normal*	0
sunny	*hot*	*high*	1

Feature Function

$$x_0 = 1 \text{ (bias term)}$$

$$x_1 = \begin{cases} 1 \text{ if outlook=sunny} \\ 2 \text{ if outlook=overcast} \\ 3 \text{ if outlook=rainy} \end{cases}$$

$$x_2 = \begin{cases} 1 \text{ if temp=hot} \\ 2 \text{ if temp=mild} \\ 3 \text{ if temp=cool} \end{cases}$$

$$x_3 = \begin{cases} 1 \text{ if humidity=normal} \\ 2 \text{ if humidity=high} \end{cases}$$

Parameter Estimation

What are the four steps we would follow in finding the optimal parameters?

Objective Function

Mimimize the **Negative conditional log likelihood**

$$\mathcal{L}(\theta) = -P(Y|X;\theta) = -\prod_{i=1}^{N} P(y^i|x^i;\theta)$$

note that

$$P(y = 1|x;\theta) = \sigma(\theta^T x)$$
$$P(y = 0|x;\theta) = 1 - \sigma(\theta^T x)$$

Objective Function

Mimimize the **Negative conditional log likelihood**

$$\mathcal{L}(\theta) = -P(Y|X;\theta) = -\prod_{i=1}^{N} P(y^i|x^i;\theta)$$

note that

$$P(y=1|x;\theta) = \sigma(\theta^T x)$$
$$P(y=0|x;\theta) = 1 - \sigma(\theta^T x)$$

so

$$\begin{aligned}\mathcal{L}(\theta) = -P(Y|X;\theta) &= -\prod_{i=1}^{N} P(y^i|x^i;\theta) \\ &= -\prod_{i=1}^{N} (\sigma(\theta^T x^i))^{y^i} * (1 - \sigma(\theta^T x^i))^{1-y^i}\end{aligned}$$

Objective Function

Mimimize the **Negative conditional log likelihood**

$$\mathcal{L}(\theta) = -P(Y|X;\theta) = -\prod_{i=1}^{N} P(y^i|x^i;\theta)$$

note that

$$P(y=1|x;\theta) = \sigma(\theta^T x)$$
$$P(y=0|x;\theta) = 1 - \sigma(\theta^T x)$$

so

$$\mathcal{L}(\theta) = -P(Y|X;\theta) = -\prod_{i=1}^{N} P(y^i|x^i;\theta)$$
$$= -\prod_{i=1}^{N} (\sigma(\theta^T x^i))^{y^i} * (1 - \sigma(\theta^T x^i))^{1-y^i}$$

take the log of this function

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Take 1st Derivative of the Objective Function

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial D} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C} \times \frac{\partial C}{\partial D}$

Take 1st Derivative of the Objective Function

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial D} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C} \times \frac{\partial C}{\partial D}$

Also

- Derivative of sum = sum of derivatives $\rightarrow$ focus on 1 training input
- Compute $\frac{\partial \mathcal{L}}{\partial \theta_j}$ for each θ_j individually, so focus on 1 θ_j

THE UNIVERSITY OF MELBOURNE

Take 1st Derivative of the Objective Function

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial D} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C} \times \frac{\partial C}{\partial D}$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j} = \frac{\partial \log \mathcal{L}(\theta)}{\partial p} \times \frac{\partial p}{\partial z} \times \frac{\partial z}{\partial \theta_j} \quad \text{where } p = \sigma(\theta^T x) \text{ and } z = \theta^T x$$

Take 1st Derivative of the Objective Function

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial D} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C} \times \frac{\partial C}{\partial D}$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j} = \frac{\partial \log \mathcal{L}(\theta)}{\partial p} \times \frac{\partial p}{\partial z} \times \frac{\partial z}{\partial \theta_j} \quad \text{where } p = \sigma(\theta^T x) \text{ and } z = \theta^T x$$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial p} = -\left(\frac{y}{p} - \frac{1-y}{1-p}\right)$$

(because $\mathcal{L}(\theta) = -[ylogp + (1 - y)log(1 - p)]$

Take 1st Derivative of the Objective Function

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial D} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C} \times \frac{\partial C}{\partial D}$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j} = \frac{\partial \log \mathcal{L}(\theta)}{\partial p} \times \frac{\partial p}{\partial z} \times \frac{\partial z}{\partial \theta_j} \quad \text{where } p = \sigma(\theta^T x) \ \text{ and } \ z = \theta^T x$$

$$\frac{\partial p}{\partial z} = \frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$$

Take 1st Derivative of the Objective Function

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial D} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C} \times \frac{\partial C}{\partial D}$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j} = \frac{\partial \log \mathcal{L}(\theta)}{\partial p} \times \frac{\partial p}{\partial z} \times \frac{\partial z}{\partial \theta_j} \quad \text{where } p = \sigma(\theta^T x) \text{ and } z = \theta^T x$$

$$\frac{\partial z}{\partial \theta_j} = \frac{\partial \theta^T x}{\partial \theta_j} = x_j$$

Take 1st Derivative of the Objective Function

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial D} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C} \times \frac{\partial C}{\partial D}$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j} = \frac{\partial \log \mathcal{L}(\theta)}{\partial p} \times \frac{\partial p}{\partial z} \times \frac{\partial z}{\partial \theta_j} \quad \text{where } p = \sigma(\theta^T x) \text{ and } z = \theta^T x$$

$$= -\frac{y}{p} - \frac{1-y}{1-p} \times \sigma(z)[1 - \sigma(z)] \times x_j$$

Take 1st Derivative of the Objective Function

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial D} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C} \times \frac{\partial C}{\partial D}$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j} = \frac{\partial \log \mathcal{L}(\theta)}{\partial p} \times \frac{\partial p}{\partial z} \times \frac{\partial z}{\partial \theta_j} \quad \text{where } p = \sigma(\theta^T x) \text{ and } z = \theta^T x$$

$$= -\frac{y}{p} - \frac{1-y}{1-p} \times \sigma(z)[1 - \sigma(z)] \times x_j$$

$$= \left[\sigma(\theta^T x) - y\right] \times x_j$$

Logistic Regression: Parameter Estimation III

The derivative of the log likelihood wrt. a single parameter θ_j for **all** training examples

$$\frac{\log \mathcal{L}(\theta)}{\partial \theta_j} = \sum_{i=1}^{N} \left(\sigma(\theta^T x^i) - y^i \right) x_j^i$$

- Now, we would set derivatives to zero (**Step 3**) and solve for θ (**Step 4**)
- Unfortunately, that's not straightforward here (as for Naive Bayes)
- Instead, we will use an iterative method: **Gradient Descent**

Logistic Regression: Parameter Estimation III

The derivative of the log likelihood wrt. a single parameter θ_j for **all** training examples

$$\frac{\log \mathcal{L}(\theta)}{\partial \theta_j} = \sum_{i=1}^{N} \left(\sigma(\theta^T x^i) - y^i \right) x_j^i$$

- Now, we would set derivatives to zero (**Step 3**) and solve for θ (**Step 4**)
- Unfortunately, that's not straightforward here (as for Naive Bayes)
- Instead, we will use an iterative method: **Gradient Descent**

$$\theta_j^{(new)} \leftarrow \theta_j^{(old)} - \eta \frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j}$$

$$\theta_j^{(new)} \leftarrow \theta_j^{(old)} - \eta \sum_{i=1}^{N} \left(\sigma(\theta^T x^i) - y^i \right) x_j^i$$

Multinomial Logistic Regression

- So far we looked at problems where either $y = 0$ or $y = 1$ (e.g., spam classification: $y \in \{\texttt{play}, \texttt{not\_play}\}$)

$$
\begin{aligned}
P(y = 1|x;\theta) = \sigma(\theta^T x) \quad &= \frac{exp(\theta^T x)}{1 + exp(\theta^T x)} \\
P(y = 0|x;\theta) = 1 - \sigma(\theta^T x) \quad &= 1 - \frac{exp(\theta^T x)}{1 + exp(\theta^T x)}
\end{aligned}
$$

Multinomial Logistic Regression

- So far we looked at problems where either $y = 0$ or $y = 1$ (e.g., spam classification: $y \in \{\texttt{play}, \texttt{not\_play}\}$)

$$P(y=1|x;\theta) = \sigma(\theta^T x) \qquad = \frac{exp(\theta^T x)}{1 + exp(\theta^T x)}$$

$$P(y=0|x;\theta) = 1 - \sigma(\theta^T x) \qquad = 1 - \frac{exp(\theta^T x)}{1 + exp(\theta^T x)}$$

- But what if we have more than 2 classes, e.g., $y \in \{\texttt{positive}, \texttt{negative}, \texttt{neutral}\}$
- we predict the probability of each class c by passing the input representation through the **softmax** function, a generalization of the sigmoid

$$p(y=c|x;\theta) = \frac{exp(\theta_c x)}{\sum_k exp(\theta_k x)}$$

- we learn a parameter vector θ_c for each class c

Example! Multi-class with 1-hot features

$$p(y = c|x; \theta) = \frac{exp(\theta_c x)}{\sum_k exp(\theta_k x)}$$

Model parameters

$$\theta_{c0} = [0.1, \ -3.5, \ 0.7, \ 2.1]$$
$$\theta_{c1} = [0.6, \ 2.5, \ 2.7, \ -2.1]$$
$$\theta_{c2} = [3.1, \ 1.5, \ 0.07, \ 3.6]$$

(Small) Test Data set

Outlook	Temp	Humidity	Class
rainy	*cool*	*normal*	0 (don't play)
sunny	*cool*	*normal*	1 (maybe play)
sunny	*hot*	*high*	2 (play)

Feature Function

$$x_0 = 1 \ \text{(bias term)}$$
$$x_1 = \begin{cases} 1 \text{ if outlook=sunny} \\ 2 \text{ if outlook=overcast} \\ 3 \text{ if outlook=rainy} \end{cases}$$
$$x_2 = \begin{cases} 1 \text{ if temp=hot} \\ 2 \text{ if temp=mild} \\ 3 \text{ if temp=cool} \end{cases}$$
$$x_3 = \begin{cases} 1 \text{ if humidity=normal} \\ 2 \text{ if humidity=high} \end{cases}$$

$$x_0 = 1 \ \text{(bias term)}$$
$$x_1 = \begin{cases} [100] \text{ if outlook=sunny} \\ [010] \text{ if outlook=overcast} \\ [001] \text{ if outlook=rainy} \end{cases}$$
$$x_2 = \begin{cases} [100] \text{ if temp=hot} \\ [010] \text{ if temp=mild} \\ [001] \text{ if temp=cool} \end{cases}$$
$$x_3 = \begin{cases} [10] \text{ if humidity=normal} \\ [01] \text{ if humidity=high} \end{cases}$$

Example! Multi-class with 1-hot features

$$p(y = c|x; \theta) = \frac{exp(\theta_c x)}{\sum_k exp(\theta_k x)}$$

(Small) Test Data set

Outlook	Temp	Humidity	Class
0 0 1	0 0 1	1 0	0
1 0 0	0 0 1	1 0	1
1 0 0	1 0 0	0 1	2

Feature Function

$$x_0 = 1 \text{ (bias term)}$$

$$x_1 = \begin{cases} [100] & \text{if outlook=sunny} \\ [010] & \text{if outlook=overcast} \\ [001] & \text{if outlook=rainy} \end{cases}$$

$$x_2 = \begin{cases} [100] & \text{if temp=hot} \\ [010] & \text{if temp=mild} \\ [001] & \text{if temp=cool} \end{cases}$$

$$x_3 = \begin{cases} [10] & \text{if humidity=normal} \\ [01] & \text{if humidity=high} \end{cases}$$

Logistic Regression: Final Thoughts

Pros

- Probabilistic interpretation
- No restrictive assumptions on features
- Often outperforms Naive Bayes
- Particularly suited to frequency-based features (so, popular in NLP)

Cons

- Can only learn *linear* feature-data relationships
- Some feature scaling issues
- Often needs a lot of data to work well
- Regularisation a nuisance, but important since overfitting can be a big problem

THE UNIVERSITY OF MELBOURNE

Summary

- Derivation of logistic regression
- Prediction
- Derivation of maximum likelihood

References

Cosma Shalizi. *Advanced Data Analysis from an Elementary Point of View*. Chapters 11.1 and 11.2. Online Draft.
`https://www.stat.cmu.edu/~cshalizi/ADAfaEPoV/ADAfaEPoV.pdf`

Dan Jurafsky and James H. Martin. *Speech and Language Processing*. Chapter 5. Online Draft V3.0.
`https://web.stanford.edu/~jurafsky/slp3/`

Optional: Detailed Parameter Estimation

Step 2 Differentiate the loglikelihood wrt. the parameters

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial C} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C}$

Optional: Detailed Parameter Estimation

Step 2 Differentiate the loglikelihood wrt. the parameters

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial C} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C}$

Also

- Derivative of sum = sum of derivatives $\rightarrow$ focus on 1 training input
- Compute $\frac{\partial \mathcal{L}}{\partial \theta_j}$ for each θ_j individually, so focus on 1 θ_j

Optional: Detailed Parameter Estimation

Step 2 Differentiate the loglikelihood wrt. the parameters

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial C} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C}$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j} = \frac{\partial \log \mathcal{L}(\theta)}{\partial p} \times \frac{\partial p}{\partial z} \times \frac{\partial z}{\partial \theta_j} \quad \text{where } p = \sigma(\theta^T x) \text{ and } z = \theta^T x$$

Step 2 Differentiate the loglikelihood wrt. the parameters

$$\boxed{\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1-y^i)\log(1-\sigma(\theta^T x^i))}$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1-\sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial C} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C}$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j} = \frac{\partial \log \mathcal{L}(\theta)}{\partial p} \times \frac{\partial p}{\partial z} \times \frac{\partial z}{\partial \theta_j} \quad \text{where } p = \sigma(\theta^T x) \text{ and } z = \theta^T x$$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial p} = -\left(\frac{y}{p} - \frac{1-y}{1-p}\right)$$

(because $\mathcal{L}(\theta) = -[ylogp + (1-y)log(1-p)]$

Step 2 Differentiate the loglikelihood wrt. the parameters

$$\boxed{\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))}$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial C} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C}$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j} = \frac{\partial \log \mathcal{L}(\theta)}{\partial p} \times \frac{\partial p}{\partial z} \times \frac{\partial z}{\partial \theta_j} \quad \text{where } p = \sigma(\theta^T x) \text{ and } z = \theta^T x$$

$$\frac{\partial p}{\partial z} = \frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$$

Step 2 Differentiate the loglikelihood wrt. the parameters

$$\boxed{\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))}$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial C} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C}$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j} = \frac{\partial \log \mathcal{L}(\theta)}{\partial p} \times \frac{\partial p}{\partial z} \times \frac{\partial z}{\partial \theta_j} \quad \text{where } p = \sigma(\theta^T x) \text{ and } z = \theta^T x$$

$$\frac{\partial z}{\partial \theta_j} = \frac{\partial \theta^T x}{\partial z} = x_j$$

Step 2 Differentiate the loglikelihood wrt. the parameters

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial C} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C}$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j} = \frac{\partial \log \mathcal{L}(\theta)}{\partial p} \times \frac{\partial p}{\partial z} \times \frac{\partial z}{\partial \theta_j} \qquad \text{where } p = \sigma(\theta^T x) \text{ and } z = \theta^T x$$

$$= -\left[\frac{y}{p} - \frac{1-y}{1-p}\right] \times \sigma(z)[1 - \sigma(z)] \times x_j \qquad \text{[[combine 3 derivatives]]}$$

$$= -\left[\frac{y}{p} - \frac{1-y}{1-p}\right] \times p[1-p] \times x_j \qquad [[\ \sigma(z) = p]]$$

$$= -\left[\frac{y(1-p)}{p(1-p)} - \frac{p(1-y)}{p(1-p)}\right] \times p[1-p] \times x_j \qquad [[\ \times \frac{1-p}{1-p} \text{ and } \frac{p}{p}\ ]]$$

$$= -\left[y(1-p) - p(1-y)\right] \times x_j \qquad \text{[[cancel terms]]}$$

Optional: Detailed Parameter Estimation

Step 2 Differentiate the loglikelihood wrt. the parameters

$$\log \mathcal{L}(\theta) = -\sum_{i=1}^{N} y^i \log \sigma(\theta^T x^i) + (1 - y^i) \log(1 - \sigma(\theta^T x^i))$$

Preliminaries

- The derivative of the logistic (sigmoid) function is $\frac{\partial \sigma(z)}{\partial z} = \sigma(z)[1 - \sigma(z)]$
- The chain rule tells us that $\frac{\partial A}{\partial C} = \frac{\partial A}{\partial B} \times \frac{\partial B}{\partial C}$

$$\frac{\partial \log \mathcal{L}(\theta)}{\partial \theta_j} = \frac{\partial \log \mathcal{L}(\theta)}{\partial p} \times \frac{\partial p}{\partial z} \times \frac{\partial z}{\partial \theta_j} \quad \text{where } p = \sigma(\theta^T x) \text{ and } z = \theta^T x$$

$$= -\Big[y(1-p) - p(1-y)\Big] \times x_j \qquad \text{[[copy from last slide]]}$$

$$= -\Big[y - yp - p + yp\Big] \times x_j \qquad \text{[[multiply out]]}$$

$$= -\Big[y - p\Big] \times x_j \qquad \text{[[-yp+yp=0]]}$$

$$= \Big[p - y\Big] \times x_j \qquad \text{[[-[y-p] = -y+p = p-y]]}$$

$$= \Big[\sigma(\theta^T x) - y\Big] \times x_j \qquad [[p = \sigma(z),\, z = \theta^T x\ ]]$$