CS 2731 Introduction to Natural Language Processing

Session 6: Logistic regression, part 1

Michael Miller Yoder

September 18, 2023

University of Pittsburgh | School of Computing and Information

Course logistics

- Homework 1 was due last night at midnight
- Reading quiz
  - Threw out tricky question on “Feature normalization or standardization can be useful to compare values across **features** (not classes)”
- Project area and contribution form is **due Thu 09-21, 11:59pm**
  - Please plan meeting with your groups to discuss project ideas
  - If you don’t have any specific ideas, that’s fine! We will help you come up with some.
  - No need to submit anything on Canvas

Lecture overview: logistic regression part 1

- Hands-on example: Clickbait classification with Naive Bayes in scikit-learn
- Discriminative vs generative classifiers revisited
- Text classification with logistic regression
- Meet with project groups

Hands-on activity: Clickbait classification with Naive Bayes

Clickbait classification activity

- What steps are needed to go from labeled text data to a classifier?
- Colab notebook here: https://colab.research.google.com/drive/187yqGR_M_OVYrV28_nzjPY50mm6x-fxJ#scrollTo=ECHzSBjQVlAP

Discriminative vs generative classifiers

An Example: Classifying Orchids

Cymbidium

Phalaenopsis

Features: type (terrestrial, epiphitic), stem shape (sympodial, monopodial), leaf shape (pointed, rounded) roots (thin,thick, tuberous), flowers (dorsal sepal, other sepals, petals, labelum, pollinia)

Slide credit: David Mortensen

Two Options for Classification

Build up a model of each class

Cymbidiums Semi-epiphetic, lithophytic, or terrestrial. Pseduobulb. Flowers in spikes. Bowl-shaped labellum. Glaborous seed capsule.

Phalaenopsis Epiphetic. Long, course roots. Short, leafy stems. Flowers in racemes or panicles. Spread sepals and petals. Petals much larger than sepals.

Find features that distinguish them

	Cymbs	Phals
pseudobulbs	T	F
spikes	T	F
panicles	F	T
petals much larger than sepals	F	T

Slide credit: David Mortensen

Should we use generative or discriminative classifiers?

- **Generative:** builds a model of each of the categories, so it could generate instances of them
- **Discriminative:** weights most heavily the features that best discriminate between categories

Slide credit: David Mortensen

Finding the correct class for an orchid

We can classify orchids generatively or discriminatively, where x is the orchid and ℓ is the class label

Naive Bayes

$$\hat{\ell} = \operatorname*{argmax}_{\ell \in \mathcal{L}} P(\mathbf{x}|\ell)P(\ell)$$

(compute the likelihood times the prior)

Logistic Regression

$$\hat{\ell} = \operatorname*{argmax}_{\ell \in \mathcal{L}} P(\ell|\mathbf{x})$$

(compute the posterior directly)

We're going to classify documents in the same way!

Slide credit: David Mortensen

Logistic regression

What Goes into a (Discriminative) ML Classifier?

1. A feature representation
2. A classification function
3. An objective function
4. An algorithm for optimizing the objective function

Slide credit: David Mortensen

What Goes into Logistic Regression?

GENERAL	IN LOGISTIC REGRESSION
feature representation	represent each observation $\mathbf{x}^{(i)}$ as a **vector of features** $[x_1, x_2, \ldots x_n]$, as we did with orchids
classification function	**sigmoid function** (logistic function)
objective function	**cross-entropy loss**
optimization function	(stochastic) **gradient descent**

Slide credit: David Mortensen

The Two Phases of Logistic Regression

train learn **w** (a vector of weights, one for each feature) and b (a bias) using **stochastic gradient descent** and **cross-entropy loss**.

test given a test example x, we compute $p(y|x)$ using the learned weights w and b and return the label ($y = 1$ or $y = 0$) that has higher probability.

Slide credit: David Mortensen

Classification with logistic regression

Reminder: the Dot Product

We will see the dot product a lot. It is the **sum** of the element-wise **product** of two vectors of the same dimensionality.

$$\begin{bmatrix} 2 & 7 & 1 \end{bmatrix} \cdot \begin{bmatrix} 8 \\ 2 \\ 8 \end{bmatrix} = 2 \cdot 8 + 7 \cdot 2 + 1 \cdot 8 = 38 \tag{3}$$

Moving on...

Slide credit: David Mortensen

Features in Logistic Regression

For feature x_i, weight w_i tells us how important x_i is

- x_i = "review contains awesome": $w_i = +10$
- x_j = "review contains abysmal": $w_j = -10$
- x_k = "review contains mediocre": $w_k = -2$

Slide credit: David Mortensen

Logistic Regression for One Observation x

input observation feature vector $x = [x_1, x_2, \ldots, x_n]$

weights one per feature $W = [w_1, w_2, \ldots, w_n]$ plus w_0, which is the **bias** b

output a predicted class $\hat{y} \in \{0, 1\}$

Slide credit: David Mortensen

How to Do Classification

For each feature x_i, weight w_i tells us the importance of x_i (and we also have the bias b that shifts where the function crosses the x-axis)

We'll sum up all the weighted features and the bias

$$z = \left(\sum_{i=1}^{n} w_i x_i \right) + b$$

$$z = \mathbf{w} \cdot \mathbf{x} + b$$

Slide credit: David Mortensen

A Most Important Formula

We compute

$$z = \mathbf{w} \cdot \mathbf{x} + b$$

If z is high, we say $y = 1$; if low, then $y = 0$.

orchids A classifiers for cymbidiums should return $y = 1$ when the input is a cymbidium and $y = 0$ otherwise.

sentiment A classifier for positive sentiment show return $y = 1$ when the input has positive sentiment (when the emotions of the writer towards the topic are positive) and $y = 0$ otherwise.

Remember this formula.

Slide credit: David Mortensen

But We Want a Probabilistic Classifier

What does "sum is high" even mean?

Can't our classifier be like Naive Bayes and give us a probability?

What we really want:

- $p(y = 1|x; \theta)$
- $p(y = 0|x; \theta)$

Where x is a vector of features and $\theta = (w, b)$ (the weights and the bias).

Slide credit: David Mortensen

The Problem: z isn't a Probability!

z is just a number:

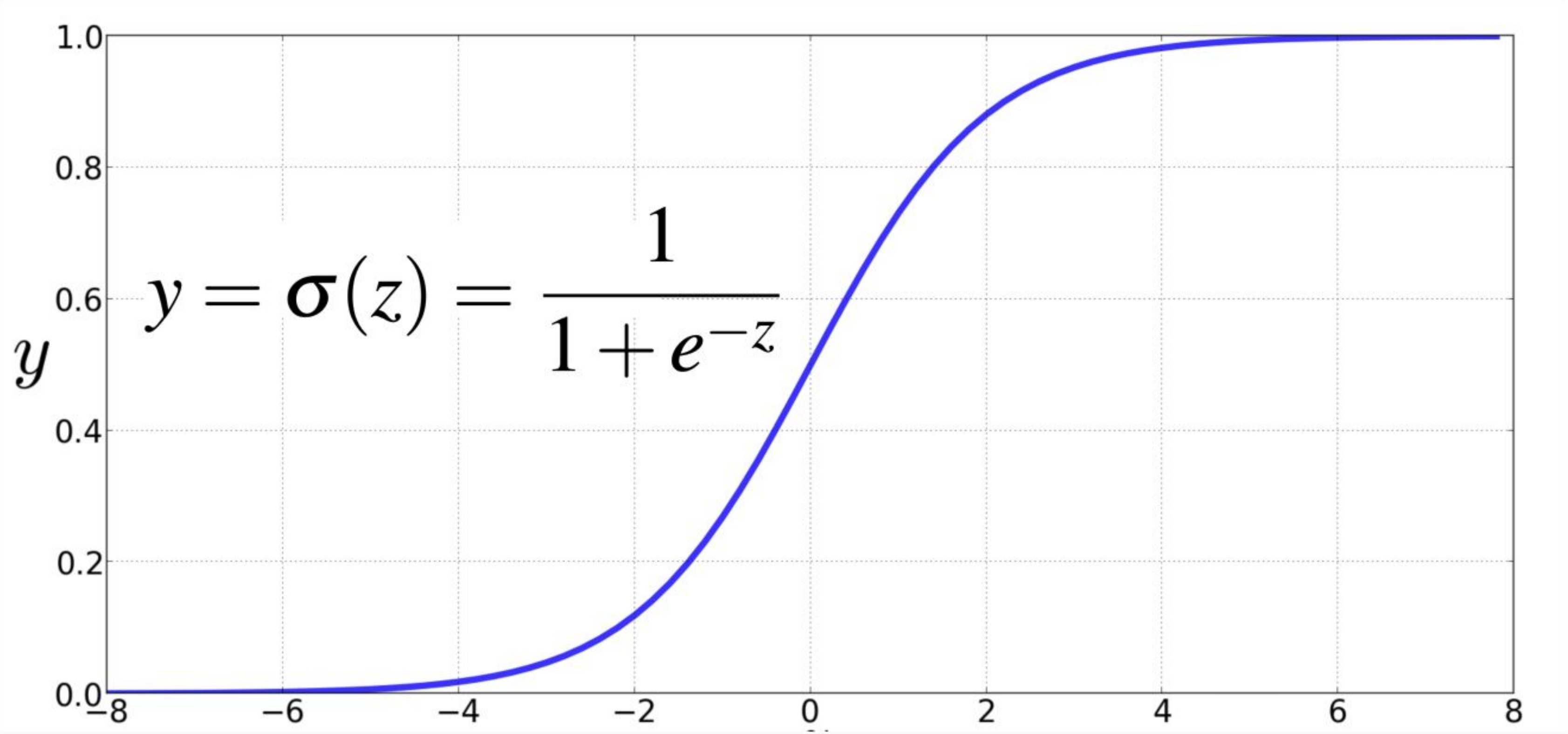
$$z = w \cdot x + b \tag{4}$$

Solution: use a function of z that goes from 0 to 1, like the **logistic function** or **sigmoid function**:

$$\hat{y} = \sigma(z) = \frac{1}{1 + e^{-z}} = \frac{1}{1 + \exp(-z)} \tag{5}$$

Slide credit: David Mortensen

The Sigmoid Function



Slide credit: David Mortensen

Logistic Regression in Three Easy Steps

1. Compute $w \cdot x + b$
2. Pass it through the sigmoid function: $\sigma(w \cdot x + b)$
3. Treat the result as a probability

Slide credit: David Mortensen

Making Probabilities with Sigmoids

$$
\begin{aligned}
P(y=1) &= \sigma(w \cdot x + b) \\
&= \frac{1}{1+\exp(-(w \cdot x + b))} \\
P(y=0) &= 1 - \sigma(w \cdot x + b) \\
&= 1 - \frac{1}{1+\exp(-(w \cdot x + b))} \\
&= \frac{\exp(-(w \cdot x + b))}{1+\exp(-(w \cdot x + b))}
\end{aligned}
$$

Slide credit: David Mortensen

From Probability to Classification

$$y = \begin{cases} 1 & P(y = 1|x) > 0.5 \\ 0 & \text{otherwise} \end{cases}$$

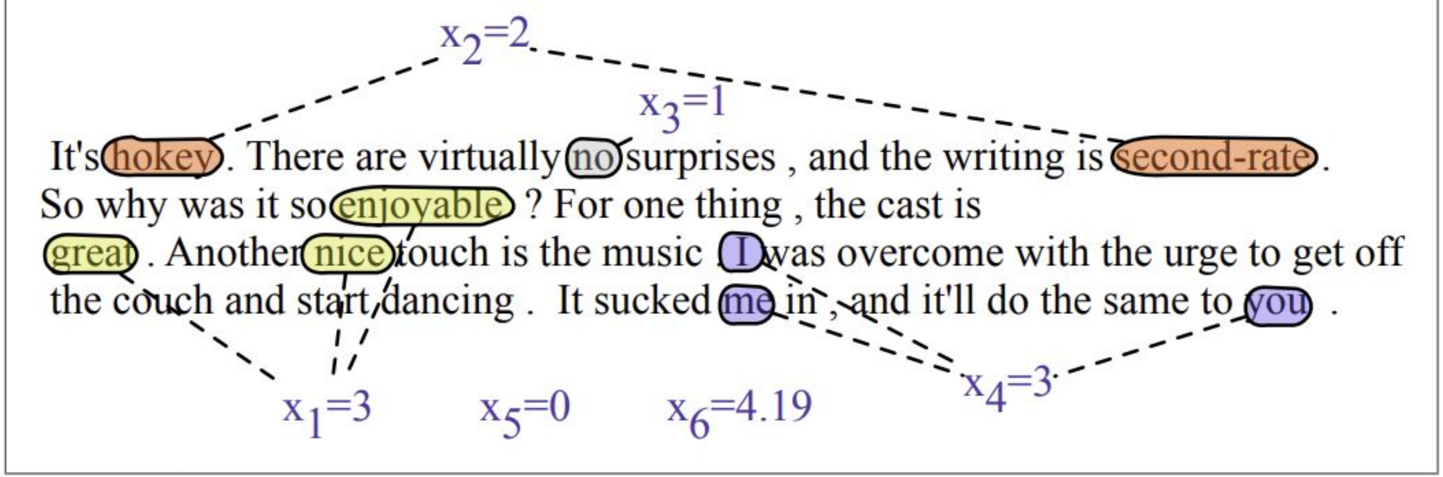
0.5 here is called the **decision boundary**

Slide credit: David Mortensen

Sentiment Classification: Movie Review

It’s hokey . There are virtually no surprises , and the writing is second-rate . So why was it so enjoyable ? For one thing , the cast is great . Another nice touch is the music . I was overcome with the urge to get off the couch and start dancing . It sucked me in , and it’ll do the same to you .

Slide credit: David Mortensen



Var	Definition	Value in Fig. 5.2
x_1	count(positive lexicon) $\in$ doc)	3
x_2	count(negative lexicon) $\in$ doc)	2
x_3	$\begin{cases} 1 & \text{if “no”} \in \text{doc} \\ 0 & \text{otherwise} \end{cases}$	1
x_4	count(1st and 2nd pronouns $\in$ doc)	3
x_5	$\begin{cases} 1 & \text{if “!”} \in \text{doc} \\ 0 & \text{otherwise} \end{cases}$	0
x_6	log(word count of doc)	$\ln(66) = 4.19$

Slide credit: Jurafsky & Martin

Classifying Sentiment for Input x

Var	Definition	Val
x_1	count(positive lexicon) $\in$ doc	3
x_2	count(negative lexicon) $\in$ doc	2
x_3	$\begin{cases} 1 & \text{if "no"} \in \text{doc} \\ 0 & \text{otherwise} \end{cases}$	1
x_4	count(1st & 2nd pronouns) $\in$ doc	3
x_5	$\begin{cases} 1 & \text{if "!"} \in \text{doc} \\ 0 & \text{otherwise} \end{cases}$	0
x_6	$\log$(word count of doc)	$\ln(66) = 4.19$

Suppose $w = [2.5, -0.5, -1.2, 0.5, 2.0, 0.7]$ **and** $b = 0.1$

Slide credit: David Mortensen

Performing the Calculations

$$
\begin{aligned}
p(+|x) = P(Y=1|x) &= \sigma(w \cdot x + b) \\
&= \sigma([2.5, -5.0, -1.2, 0.5.2.0, 0.7] \cdot [3, 2, 1, 3, 0, 4.19] + 0.1) \\
&= \sigma(0.833) \\
&= 0.70 \\
p(-|x) = P(Y=0|x) &= 1 - \sigma(w \cdot x + b) \\
&= 0.30
\end{aligned}
$$

Slide credit: David Mortensen

Performing the Calculations

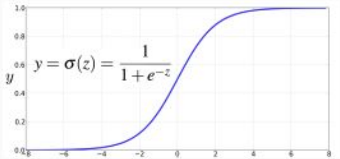
$$
\begin{aligned}
p(+|x) = P(Y=1|x) &= \sigma(w \cdot x + b) \\
&= \sigma([2.5, -5.0, -1.2, 0.5.2.0, 0.7] \cdot [3, 2, 1, 3, 0, 4.19] + 0.1) \\
&= \sigma(0.833) \\
&= 0.70 \;\; \text{(positive sentiment)} \\
p(-|x) = P(Y=0|x) &= 1 - \sigma(w \cdot x + b) \\
&= 0.30
\end{aligned}
$$

Slide credit: David Mortensen

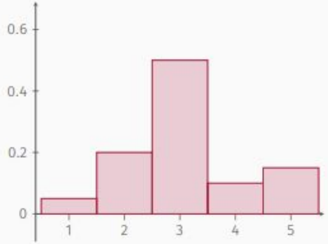
Multinomial logistic regression classification

Softmax is a Generalization of Sigmoid

Sigmoid makes its output look like a probability (forcing it to be between 0.0 and 1.0) and “squashes” it so that the output will tend to 0.0 or 1.0. Concerned about one class? Sigmoid is perfect.



For multiple classes, we do not want a probability—we want a probability **distribution**.

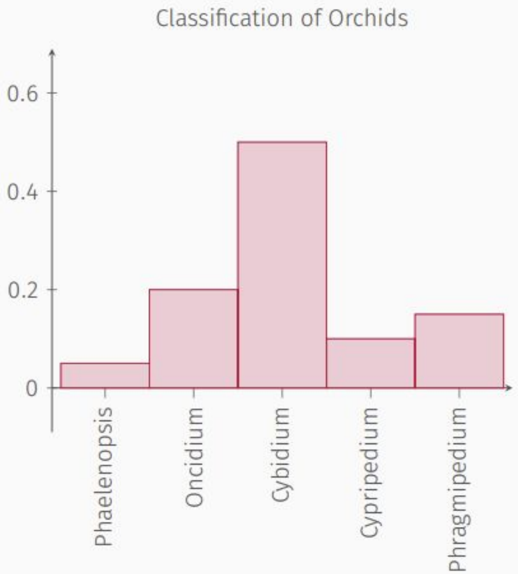


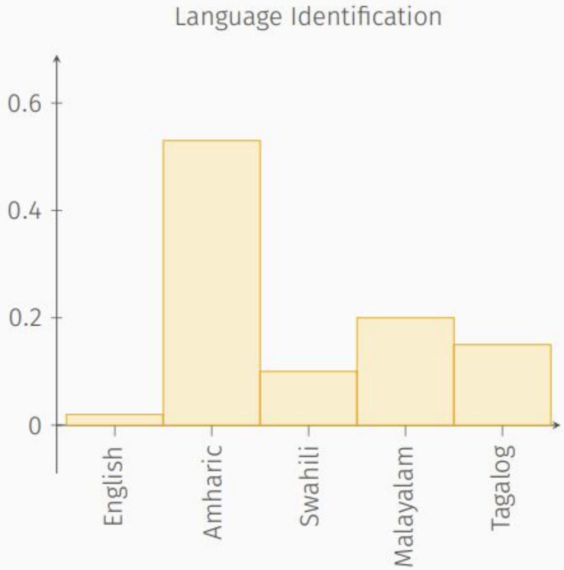
Instead of a sigmoid function, we will use SOFTMAX.

Slide credit: David Mortensen

What is a Probability Distribution?

A probability distribution is a function giving the probabilities that different possible outcomes of an experiment will occur. Our probability distributions will usually be over DISCRETE RANDOM VARIABLES.





Slide credit: David Mortensen

The Softmax Function

The formula for the softmax function is

$$\text{softmax}(\mathbf{z}_i) = \frac{\exp(\mathbf{z}_i)}{\sum_{j=1}^{K} \exp(\mathbf{z}_j)} \quad 1 \leq i \leq K$$

where K is the number of dimensions in the input vector $\mathbf{z}$. Compare it to the formula for the sigmoid function:

$$\hat{y} = \sigma(z) = \frac{1}{1 + \exp(-z)}$$

The formulas are very similar, but sigmoid is a function from a scalar to a scalar, whereas softmax is a function from a vector to a vector.

Slide credit: David Mortensen

Computing z

Remember that, to compute z in logistic regression, we used the formula

$$z = \mathbf{w}\mathbf{x} + b$$

where $\mathbf{w}$ is a vector of weights, $\mathbf{x}$ is a vector of features, and b is a scalar bias term. Thus, z is a scalar. For multinomial logistic regression, we need a vector $\mathbf{z}$ instead of a scalar z. Our formula will be

$$\mathbf{z} = \mathbf{W}\mathbf{x} + \mathbf{b}$$

where W is a matrix with the shape $[K \times f]$ (where K is the number of output classes and f is the number of input features). In other words, there is an element in $\mathbf{W}$ for each combination of class and feature. $\mathbf{x}$ is a vector of features. $\mathbf{b}$ is a vector of biases (one for each class).

Slide credit: David Mortensen

A Summary Comparison of Logistic Regression and Multinomial Logistic Regression

Logistic regression is

$$\hat{y} = \sigma(\mathbf{w}\mathbf{x} + b)$$

where y is, roughly, a probability.

Multinomial logistic regression (or SOFTMAX REGRESSION) is

$$\hat{\mathbf{y}} = \text{softmax}(\mathbf{W}\mathbf{x} + \mathbf{b})$$

where $\hat{\mathbf{y}}$ is a PROBABILITY DISTRIBUTION over classes, $\mathbf{W}$ is a class $\times$ feature weight matrix, $\mathbf{x}$ is a vector of features, and $\mathbf{b}$ is a vector of biases.

Slide credit: David Mortensen

Questions?

Project area and contribution form
due Thu, Sep 21