GENERAL ASSEMBLY

INTRO TO DATA SCIENCE

LECTURE 6: BAYESIAN INFERENCE

LAST TIME:

- PROBABILITY
- LOGISTIC REGRESSION

QUESTIONS?

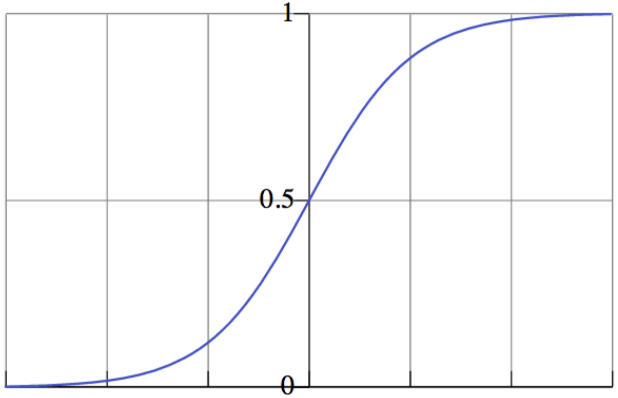
I. REVIEW LOGISTIC REGRESSION
II. BAYESIAN INFERENCE

EXERCISES:
III. IMPLEMENTING A SPAM FILTER

I. LOGISTIC REGRESSION

The **logistic function:**

$$E(y|x) = \pi(x) = \frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}}$$



The **logit function** is an important transformation of the logistic function. Notice that it returns the linear model!

$$g(x) = ln(\frac{\pi(x)}{1-\pi(x)}) = \alpha + \beta x$$

II. BAYESIAN INFERENCE

Bayes' theorem. Here it is again:

$$P(A|B) = P(B|A) * P(A) \,/\, P(B)$$

Some facts:

- This is a simple algebraic relationship using elementary definitions.
- It's interesting because it's kind of a "wormhole" between two different "interpretations" of probability.
- It's a very powerful computational tool.

This term is the **likelihood function**. It represents the joint probability of observing features $\{x_i\}$ given that that record belongs to class C.

$$P(\text{class } C \mid \{x_i\}) = \frac{P(\{x_i\} \mid \text{class } C) \cdot P(\text{class } C)}{P(\{x_i\})}$$

This term is the **prior probability** of C. It represents the probability of a record belonging to class C before the data is taken into account.

$$P(\text{class } C \mid \{x_i\}) = \frac{P(\{x_i\} \mid \text{class } C) \cdot P(\text{class } C)}{P(\{x_i\})}$$

This term is the **normalization constant.** It doesn't depend on C, and is generally ignored until the end of the computation.

$$P(\text{class } C \mid \{x_i\}) = \frac{P(\{x_i\} \mid \text{class } C) \cdot P(\text{class } C)}{P(\{x_i\})}$$

This term is the **posterior probability** of C. It represents the probability of a record belonging to class C after the data is taken into account.

$$P(\text{class } C \mid \{x_i\}) = \frac{P(\{x_i\} \mid \text{class } C) \cdot P(\text{class } C)}{P(\{x_i\})}$$

This term is the **posterior probability** of C. It represents the probability of a record belonging to class C after the data is taken into account.

$$P(\text{class } C \mid \{x_i\}) = \frac{P(\{x_i\} \mid \text{class } C) \cdot P(\text{class } C)}{P(\{x_i\})}$$

The goal of any Bayesian computation is to find ("learn") the posterior distribution of a particular variable.

Maximum likelihood estimator (MLE):

What parameters ***maximize*** the likelihood function?

$$P(\text{class } C \mid \{x_i\}) = \frac{P(\{x_i\} \mid \text{class } C) \cdot P(\text{class } C)}{P(\{x_i\})}$$

Maximum a posteriori estimate (MAP):

What parameters ***maximize*** the likelihood function **AND** prior?

$$P(\text{class } C \mid \{x_i\}) = \frac{P(\{x_i\} \mid \text{class } C) \cdot P(\text{class } C)}{P(\{x_i\})}$$

Suppose we have a dataset with features $x_1, \ldots, x_n$ and a class label c. What can we say about classification using Bayes' theorem?

Suppose we have a dataset with features $x_1, \ldots, x_n$ and a class label C. What can we say about classification using Bayes' theorem?

$$P(\text{class } C \mid \{x_i\}) = \frac{P(\{x_i\} \mid \text{class } C) \cdot P(\text{class } C)}{P(\{x_i\})}$$

Bayes' theorem can help us to determine the probability of a record belonging to a class, *given* the data we observe.

source: Data Analysis with Open Source Tools, by Philipp K. Janert. O'Reilly Media, 2011.

The idea of Bayesian inference, then, is to **update** our beliefs about the distribution of C using the data ("evidence") at our disposal.

$$P(\text{class } C \mid \{x_i\}) = \frac{P(\{x_i\} \mid \text{class } C) \cdot P(\text{class } C)}{P(\{x_i\})}$$

Then we can use the posterior for prediction.

Q: What piece of the puzzle we've seen so far looks like it could intractably difficult in practice?

Remember the likelihood function?

$$P(\{x_i\}|C) = P(\{x_1, x_2, \ldots, x_n\})|C)$$

Remember the likelihood function?

$$P(\{x_i\}\,|\,C) = P(\{x_1, x_2, \ldots, x_n\})\,|\,C)$$

Observing this exactly would require us to have enough data for every possible combination of features to make a reasonable estimate.

Q: What piece of the puzzle we've seen so far looks like it could intractably difficult in practice?

A: Estimating the full likelihood function.

Q: So what can we do about it?

Q: So what can we do about it?

A: Make a simplifying assumption. In particular, we assume that the features x_i are conditionally independent from each other:

Q: So what can we do about it?

A: Make a simplifying assumption. In particular, we assume that the features x_i are conditionally independent from each other:

$$P(\{x_i\}|C) = P(x_1, x_2, \ldots, x_n|C) \approx P(x_1|C) * P(x_2|C) * \ldots * P(x_n|C)$$

Q: So what can we do about it?

A: Make a simplifying assumption. In particular, we assume that the features x_i are conditionally independent from each other:

$$P(\{x_i\}|C) = P(x_1, x_2, \ldots, x_n|C) \approx P(x_1|C) * P(x_2|C) * \ldots * P(x_n|C)$$

This "naïve" assumption simplifies the likelihood function to make it tractable.

$$P(\{x_i\}|C) = P(x_1, x_2, \ldots, x_n|C) \approx P(x_1|C) * P(x_2|C) * \ldots * P(x_n|C)$$

Q: Given that we can compute this value, what do we do with it?

$$P(\{x_i\}|C) = P(x_1, x_2, \ldots, x_n|C) \approx P(x_1|C) * P(x_2|C) * \ldots * P(x_n|C)$$

Q: Given that we can compute this value, what do we do with it?

A: In our training phase, we 'learn' the probability of seeing our training examples under each class.

$$P(\{x_i\}|C) = P(x_1, x_2, \ldots, x_n|C) \approx P(x_1|C) * P(x_2|C) * \ldots * P(x_n|C)$$

Q: Given that we can compute this value, what do we do with it?

A: In our training phase, we ‘learn’ the probability of seeing our training examples under each class.

Then we use Bayes Theorem to compute P(class | inputs)

$$P(\text{class } C \mid \{x_i\}) = \frac{P(\{x_i\} \mid \text{class } C) \cdot P(\text{class } C)}{P(\{x_i\})}$$

Maximum a posteriori estimate (MAP):

What **LABEL *maximizes*** the likelihood function **AND** prior?

$$P(\text{class } C \mid \{x_i\}) = \frac{P(\{x_i\} \mid \text{class } C) \cdot P(\text{class } C)}{P(\{x_i\})}$$

Example: Text Classification

Does this news article talk about politics?

Training Set: Collection of New Articles

Example: Text Classification

Does this news article talk about politics?

Training Set: Collection of New Articles

Article 1: The computer contractor who exposed….
Article 2: The parents of a missing U.S. journalist in Syria…

Q: What are my features?

Q: What are my features?

A: The text in the documents.

Q: What are my features?

A: The text in the documents.

Q: How to I represent them?

Q: What are my features?

A: The text in the documents.

Q: How to I represent them?
A: Binary occurrence? Word counts?

the,	computer,	contractor,	exposed,	parents,	missing,	Syria,	U.S.
1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1

computer,	contractor,	exposed,	parents,	missing,	Syria,	U.S.
1	1	1	0	0	0	0
0	0	0	1	1	1	1

We can make some alterations

1) Drop stop words (commonly occurring words that don't have meaning)

computer,	contractor,	exposed,	parents,	missing,	Syria,	U.S.,	**POL**
1	1	1	0	0	0	0	0
0	0	0	1	1	1	1	1

Our goal is to compute compute P (POL = T | words in the text)

We need to **learn** P(word | POL) i.e. P (Syria | POL)

computer	contractor	exposed	parents	missing	Syria	U.S.	**POL**
1	1	1	0	0	0	0	0
0	0	0	1	1	1	1	1

Once we've learned P(computer | POL), P(U.S. | POL) on our training set, we want to label our test set

computer,	contractor,	exposed,	parents,	missing,	Syria,	U.S.,	**POL**
1	1	1	0	0	0	0	0
0	0	0	1	1	1	1	1

The correct label, POL = True or POL = False is the one that maximize our posterior.

computer,	contractor,	exposed,	parents,	missing,	Syria,	U.S.,	**POL**
1	1	1	0	0	0	0	0
0	0	0	1	1	1	1	1

Compute probability in each class:

P (POL = T | {x}) = P ({x} | POL = T) * P(POL=T)

P (POL = F | {x}) = P ({x} | POL = F) * P(POL=F)

computer,	contractor,	exposed,	parents,	missing,	Syria,	U.S.,	**POL**
1	1	1	0	0	0	0	0
0	0	0	1	1	1	1	1

Article 2: The parents of a missing U.S. journalist in Syria...

P (POL = T | {x}) = P ({x} | POL = T) * P(POL=T)

= P(Syria | POL=T) * P(journalist | POL=T) * P(parents | POL=T) ...

* P(POL=T)

III. SPAM FILTER