menu

› Course Shortcuts

› Student Lounge

› Q&A

› Estimating Distributions from Data

› Classification with the Naive Bayes Algorithm

Module Introduction: Classification with the Naive Bayes Algorithm

Limitations of High Dimensional Space in MLE

MLE Curse of Dimensionality

Formalize the Curse of Dimensionality

Capturing All Possibilities in d Dimensions

Naive Bayes Assumption

Data Sets Where the Naive Bayes Assumption Holds

Naive Bayes Classifier

Derivation of Naive Bayes Classifier

Naive Bayes Cheat Sheet

Determine Probability With Categorical Naive Bayes

Categorical Naive Bayes Classifiers

Naive Bayes in Action

Module Wrap-up: Classification with the Naive Bayes Algorithm

› Building a Baby Name Classifying System

› Course Resources

Estimating Probability Distributions ›

MLE Curse of Dimensionality

In the previous module, you saw that MLE can be used to estimate the parameters of a distribution from a data set. For the binomial distribution you can directly estimate the labels given an input. However, there is a major problem with this approach on highly dimensional data sets: the more dimensions in your data set, the less likely it becomes that any two feature vectors will match exactly. You can only construct probabilities for vectors whose exact combination of features have already been “seen” in the training set.

☆ **Key Points**

You can only construct probabilities for vectors whose exact combination of features have already been “seen” in the training set.

Directly estimating a Binomial or Bernoulli distribution with MLE is only useful if there are many training inputs with feature vectors identical to a test input.

Spam Email Example

Let’s suppose you are trying to build an email classifier. For the initial implementation of this email classifier, you’ll use a very simple set of three features:

1. Does the email contain the word “bacon”?
2. Does the email come from a recognized IP address?
3. Does the email contain any misspelled words?

Suppose the (very-limited) training data set looks like this, where each id is an email, 0 represents “no”, 1 represents “yes”, and the label tells us if the email is spam or not.

ID	Contains word "bacon"	Recognized IP	Misspelled words	Label
1	0	0	1	spam
2	0	0	1	spam
3	1	0	0	spam
4	0	0	1	spam
5	1	0	1	spam
6	1	1	1	spam
7	0	0	1	spam
8	0	0	1	spam
9	1	0	1	spam
10	0	0	1	spam
11	0	1	0	not spam
12	0	1	1	not spam
13	1	1	1	not spam
14	0	1	1	not spam
15	0	1	0	not spam

What would be the most likely label for an email whose feature vector is (0,0,0) based on a Bernoulli distribution fit with MLE on the given data?

○ Spam

○ Not spam

✓ Undefined

Correct! There is no exact match for this feature vector in our data.

1/1

◂ Previous

Next ▸