Sigmoid convert to probability

SGD: stochastic gradient descent

Just running optimization may lead to overfitting (assumptions that does not generalize)

Cause of error:

* Assign weight to all features even little influence ones
* Assign very high weight

Add prior weight

Ignore no evidence: penalize nonzero values (L1 regularization)

Maximum entropy model: logistic regression classifier

Measure regularized log likelihood (by having regularization)

Linear regression classification

Regular expression tokenizer or just spaces to represent each index in the word vector

Inverse document frequency taking log for coherent range

Make sure your machine learning is sensitive to distinctions. How to weigh the feature. In the case of sentiment, certain additive may be more important such as boring/ exciting. Make sure vocabulary is treated accordingly. First, we can do nothing. The learner can see and count the word in the evidence collection. There are exceptions. One has to do with sparsity. You may not be able to get a big enough of data to get a big enough collection about rare words.

Prior knowledge can come in. But make sure the data is sparse or it will obscure before you build the vocabulary.

Ambiguity: a soda is cool may be just describing the temperature. You should Apply on processed text/ tagging information that says more about the document.

Fit transform in sklearn TFIDF set weight for other: use training data, processing , save, apply on dev, part of learning.

LDA is unsupervised.

Classification problem vs clustering problem.

Classifier: Confusion\_matrix

When half and half, can use accuracy and error

When incidents are rare, use precision and recall

Recall: True Positive over all yes things.

The yes answers are relatively rare. It’s often challenging to get high precision and recall but with the policy of “NO!” you can get high accuracy.

set higher precision/ threshold. Saying yes less often but saying yes more accurately.

set higher recall, set lower threshold.

Use ROC\_curve to visualize.

Naïve Bayes classifier

Base line: understanding the task.

Ceiling:

Measuring performance: partly random.

Optimizer’s curse: may get better result out of accident. Hands off test data.

Simulation of monte carol

Logistic regression classifier is better than Naïve Bayes classifier

Test the power of the experiment

The power of experimental result to distinguish between

Do I have data to draw conclusion

100

75% CL1 O Y N N Y swap last 2 col Y N Y Y 50%

0% CL2 O N N Y Y N N N Y 25%

Truth O Y N N N

Feature 1 labeled 1, feature 2 labeled 0.6, 0.4,0.3,0.2

1 0.6 0.4 0.3 0.2

T F F F

It takes three of the rest features to overcome

Brown corpus in NLTK

Word frequency scatter plot : apply loglog to give linear curve

Heavy tail distribution/ long tail (infrequent items)

Log normal distributions, appear to be linear

Word presence/absence vector more meaningful?

Supervised classification problem.

Pointwise mutual information

Auto complete

Meet + me/us/him

Does not necessarily mean meet+ me happen together.

San + Francisco this is from PMI two rare things go together

Bigram: nltk manages token stream , frequency of words, denominator

Wordnet: a database

Word 🡪 synsets set of synonyms

Meronym mero = part