NAÏVE BAYES CLASSIFIER

Introduction to Machine Learning Experiment 1

Spring 2019

GOAL

- Implement a Naïve Bayes classifier and test it on a real dataset
- Basic ideas:
  - How to implement and apply a machine learning algorithm on a practical dataset
  - How to evaluate its performance
  - How to analyze your results
- NOTE: all these parts are very important, and should be included in your code/report

NAÏVE BAYES CLASSIFIER

- Assume that : $P(y|x_1, \dots, x_n) \propto P(y) \prod_{i=1}^{n} P(x_i|y)$
- Training:
  - Estimate $P(y)$ and $P(x_i|y)$
- Test:
  - Output $\hat{y} = \mathrm{argmax}_y P(y) \prod_{i=1}^{n} P(x_i|y)$

DATA

- The Chinese E-mail Data set:

  https://plg.uwaterloo.ca/~gvcormac/treccorpus06/

- Aims at determining whether an e-mail is spam
- Each file is an e-mail, including main text and some meta information
- We will provide original version and the other one with word segmentation (all in utf-8 encoding)

DATA (CONT.)

- Files:
  - ./data/: email files (64,620 in total)
  - ./data_cut/: email files with word segmentation (64,620 in total)
  - ./label/index: labels, each row contains a label (spam/ham) and a relative path to corresponding email file

HOW TO EVALUATE THE PERFORMANCE

- Train your classifier on training set and test its performance on test set (5-fold cross validation)

k-折叠交叉验证就是将训练集的1/k作为测试集，每个模型训练k次，测试k次，错误率为k次的平均，最终选择平均率最小的模型Mi。

- At least report the average accuracy:
  - $Accuracy = \frac{\text{number of correctly classified records}}{\text{number test records}}$
- You are welcome to learn about, and then use other evaluation metrics (e.g. precision, recall or F1)

HOW TO ANALYZE YOUR RESULTS

- What is the issue that you encounter?
- How do you address the issue?
  - how do you design the experiment?
  - how do you modify the algorithm?
- Does your solution work or not?
  - Does the classification performance improve?
- And finally try to explain why your solution works (or why it does not)

ISSUE 1: THE SIZE OF TRAINING SET

- How does the size of training set influence the classification performance?
- Suggested solution:
  - Sample 5%, 50% and 100% from the whole training set to train your model
  - Repeat the random sampling (5 times) and report min/max/average accuracy

ISSUE 2: ZERO-PROBABILITIES

- Suppose on training set, no records with $x_i = k, y = c$
- Then $\hat{P}(y = c|x_1, \dots, x_i = k, \dots, x_n) = 0$
- (why is this an issue? When does it likely to happen?)
- Possible solution:
  - Smoothing: $\hat{P}(x_i = k|y = c) = \frac{\#\{y=c, x_i=k\}+\alpha}{\#\{y=c\}+M\alpha}$
  - M is the number of unique class label

ISSUE 3: SPECIFIC FEATURES

- Are there any specific features except for bag-of-words?
- Hints:
  - Received from …
  - Time
  - Priority/Mailer

REQUIREMENT

- Implement a Naïve Bayes classifier (30% of the overall score)
- Address all 3 mentioned issues:
  - Issue 1 (30%)
  - Issue 2 & 3 (2*20%=40%)
- NOTE: the score is not based on the performance (i.e. the accuracy) of your model, but how you implement the algorithm, evaluate its performance and analyze your results.

SUBMISSION

A zipped file that contains:

- Source Code:
  - With necessary comments
  - Make sure the TA can understand/run your code and reproduce your main results (set random seed w.r.t. 5-fold partition)
- README
  - A text file (in utf8 encoding)
  - Includes your name, student id and contact information (the TA may give you feedbacks if you submit before the deadline:))
- Report
  - A PDF file
  - Experiment design/results/analysis/discussion
  - Don’t just copy&paste the source code

DEADLINE & OTHER INFORMATION

- Deadline: 2019.03.28 Thursday 23:59
  - Upload the zipped file, with your name and student id in the filename, to learn.tsinghua.edu.cn
  - Late submissions WILL NOT BE ACCEPTED
    - If there's any special circumstance, ask for permission in advance
    - Anyway, the score of late submission will get a discount
  - Copy of code or report WILL DEFINITLY NOT BE PERMITTED
- Contact the TA:
  - Chenyang Wang
  - THUwangcy@gmail.com
  - 17888802343

REFERENCE

- http://scikit-learn.org/stable/modules/naive_bayes.html

  (for basic theory and smoothing)

- Just for reference, you should implement the core algorithm all by yourself