Shuwen Zhou

CS 1675

Homework 2

Winnow test:

Simple Test:

|  |  |  |  |
| --- | --- | --- | --- |
| Test file | True Output | Expected | correct |
| 1,y,y,y  2,y,y,n  3,n,n,y  4,n,n,n | F1,f2 | F1 | no |
| 1,n,y,y  2,n,y,n  3,y,n,y  4,y,n,n | none | !F1 | no |
| 1,y,y,y  3,n,n,n  3,n,n,y  4,n,y,n | F1,f2 | F1 ^ F2 |  |
| 1,y,n,y  2,y,y,n  3,n,n,n  4,y,y,y | F1,f2 | F1 U F2 |  |
| 1,n,n,y  2,n,y,n  3,y,n,n  4,n,y,y | none | !(F1 U F2) | no |

Real Data test:

|  |  |
| --- | --- |
| Artificial.data | F1,f3,f5,f15 |
| Votes.data | none |

Perceptron test:

Simple Test:

|  |  |  |  |
| --- | --- | --- | --- |
| Test file | True Output | Expected | correct |
| 1,y,y,y  2,y,y,n  3,n,n,y  4,n,n,n | F1 | F1 |  |
| 1,n,y,y  2,n,y,n  3,y,n,y  4,y,n,n | F2 | !F1 | no |
| 1,y,y,y  3,n,n,n  3,n,n,y  4,n,y,n | none | F1 ^ F2 | no |
| 1,y,n,y  2,y,y,n  3,n,n,n  4,y,y,y | F1,f2 | F1 U F2 |  |
| 1,n,n,y  2,n,y,n  3,y,n,n  4,n,y,y | none | !(F1 U F2) | no |

Real Data test:

|  |  |
| --- | --- |
| Artificial.data | F1,f3,f5,f7 |
| Votes.data | None |

10-Fold Cross Validation

|  |  |  |
| --- | --- | --- |
| Correct Rate | Perceptron | Winnow |
| Artificial.data | 62% | 75% |
| Votes.data | 56% | 0% |

Winnow can’t handle votes.data, why?

Human changes mind, the data never for sure. Winnow kills any feature if there’s one disagree with previous data. Thus, Winnow decide there’s no single feature that could determine voter’s mind. In comparison, Winnow works well in artificial data, since there’s a pattern to follow.

In the algorithm, if we change from if “else if misclassifying a negative training example x x(i) = 1 : w(i) = 0” , which is to zero out w, to half w x(1) = 1 : w(i) = w(i)/2, maybe we could get a better result, since it’s not too crucial. The modified Winnow get:

10-Fold Cross Validation with new Winnow:

|  |  |  |
| --- | --- | --- |
| Correct Rate | Perceptron | Winnow |
| Artificial.data | 62% | 57% |
| Votes.data | 56% | 61% |

Thus, comparing this chart by the above chart, Winnow’s performance even better than Perceptron on Votes.data, but its performance on Artificial.data decrease 18%. This could be considered as “critical” vs. “relax”.

**Reference:**

Java Library used:

algs4.jar

stdlib.jar

These are developed by the textbook Algorithms, 4th Edition by Robert Sedgewick and Kevin Wayne.

Usage see: http://algs4.cs.princeton.edu/code/

Reference see: http://algs4.cs.princeton.edu/home/

Used here only for file handling and easier coding.

Perceptron Learning Algorithm:

<http://www.cs.cmu.edu/~avrim/ML09/lect0126.pdf> by Avrim Blum, January 25, 2010, Carnegie Mellon University.

<http://www.cs.princeton.edu/courses/archive/spring13/cos511/scribe_notes/0404.pdf> by Andrej Risteski, April 4, 2013, Princeton University.

Winnow:

<http://people.cs.pitt.edu/~hwa/cs1675/01.23.2014.pdf> by: R. Hwa, January, 2014, University of Pittsburgh.

<https://class.coursera.org/ntumlone-001/> by by Hsuan-Tien Li, 2014. National Taiwan University.