# Transforming Classifier Scores into Accurate Multiclass Probability Estimates

Bianca Zadrozny & Charles Elkan

Presenter: Myle Ott

# Motivation
# (the same old story)

- Easy to rank examples in order of class-membership likelihood
- Hard (or at least not trivial) to turn these rankings into probabilities of class-membership
- Goal: find P($c$ | $x$): the probability of example $x$ belonging to class $c$

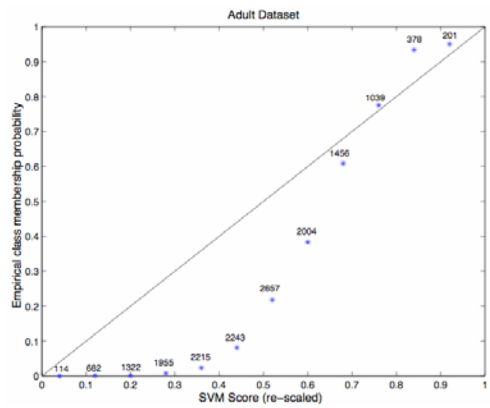
# Talking Points

From ranking scores:

- “Obtaining accurate two-class probability estimates”
  - Isotonic regression
- “Obtaining accurate multi-class probability estimates”

# “Obtaining accurate two-class probability estimates”

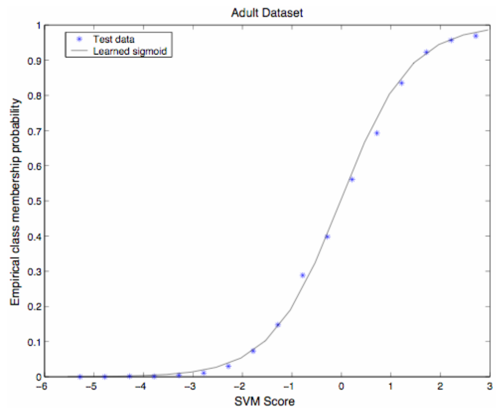
- Problem:




Interpreting re-scaled SVM scores as probabilities. (Note: rescaled based on the maximum and minimum seen distances from the hyperplane)

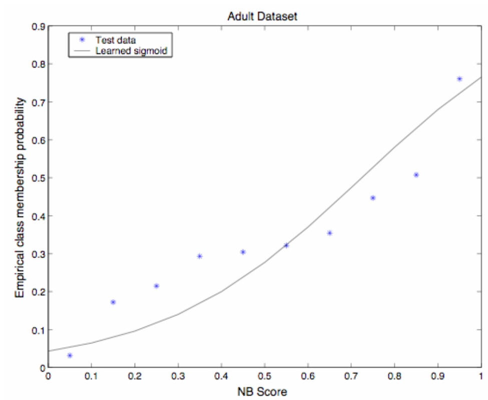
# Platt's Method

- Fit to a sigmoid:



# Naïve Bayes

- Doesn't work:




Platt's method applied to Naïve Bayes.

# Possible Solutions

- Binning
  - How many bins?
  - Why does it have to be a fixed number?
- Better method: isotonic (non decreasing) regression
  - Binning with variable number of bins

# Isotonic Regression

- Pair(Pool)-Adjacent Violators (PAV)

$\{x_i\}_{i=1}^{N}$ : training examples

$g(x_i)$ : value of the function to be learned via IR

$g^*$ : the isotonic regression

If $g$ is already isotonic, $g^* = g$. Otherwise, $\exists$ i s.t. $g(x_{i-1}) > g(x_i)$ (i.e. decreasing).

In this case, $x_{i-1}$ and $x_i$ are called pair(pool)-adjacent violators.

This is solved by replacing both $x_{i-1}$ and $x_i$ by their average.

If this new set of examples is isotonic, $g^*(x_{i-1}) = g^*(x_i) = \frac{g^*(x_{i-1}) + g^*(x_i)}{2}$, and $g^*(x_j) = g(x_j)$.

This process is repeated until an isotonic set of values is obtained.

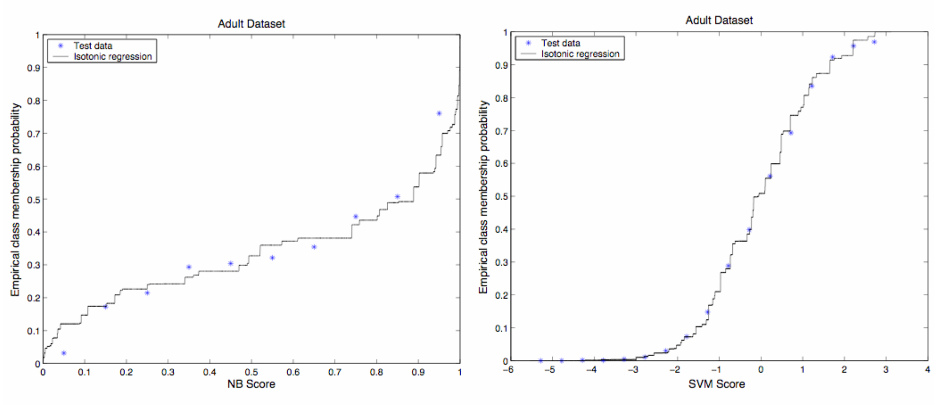
Make the set of training examples

# Isotonic Regression

- Making use of the PAV algorithm:
  - Sort examples according to score
  - Let $g(x_i)=0$ if $x_i$ is negative, 1 if $x_i$ is positive
  - Run PAV algorithm on $g$ to get $g^*$
  - $g^*$ is the isotonic regression
- Usually has pretty good results☐

Typically, this results in 0/1 probabilities if the sorted scores rank examples perfectly, baseline in the random case, and something pretty effective otherwise.

# Isotonic Regression



# “Obtaining accurate multi-class probability estimates”

- Problem:
  - Calibration methods (Platt’s method, isotonic regression, etc.) are designed for two-class problems

Because “[because] we are mapping between one-dimensional spaces […] it is easy to impose sensible restrictions on the shape of the function being learned” (bottom of page 3, section 4)

# “Obtaining accurate multi-class probability estimates”

- Solution:
  - Break the problem into many binary problems, calibrate them seperately, and then combine the probabilities
- Two ways:
  - One-against-all: each class one by one
  - All-pairs: try each possible “pair” of classes

One against all: for each class, the problem is predicting “class c” or “not class c (I.e. some other class)”

All pairs: try each possible combination (pair) of classes

# How do we “combine” the probabilities?

- One-against-all: since we have $P(c_i \mid x)$ for all $c_i$, just normalize the probabilities to 1.
- What about for all-pairs?
  - Construct a code matrix (a generalization of error-correcting output coding).

# Code Matrix

| | $b_1$ | $b_2$ | $b_3$ |
|---|---|---|---|
| $c_1$ | +1 | +1 | 0 |
| $c_2$ | -1 | 0 | +1 |
| $c_3$ | 0 | -1 | -1 |

b's represent various binary problems (all-pairs)

c's represent various classes

+1 indicates that the corresponding c is the positive class in the corresponding binary problem b

-1 … negative class

0 class not used in b

# Combining the Probabilities

$$r_b(x) = P(\bigvee_{c \in I} c | \bigvee_{c \in I \cup J} c,\ x) = \frac{\sum_{c \in I} P(c|x)}{\sum_{c \in I \cup J} P(c|x)}$$

- Where I and J are the sets of classes corresponding to M(-, b) = 1 and M(-, b) = -1, respectively

Essentially, rb(x) is equal to the probability of the positive class divided by the combined probabilities of the positive and negative classes (which should always be 1, right?) I only include this because it is included in the paper.

# Combining the Probabilities

- There are two methods for solving this problem:
  - Least-squares method with non-negativity constraints
  - Coupling, an iterative algorithm for minimizing log-loss instead of squared error

These methods are not explained in the paper, but references are given.

# Results (two-class)

| | MSE | | Error Rate | |
|---|---|---|---|---|
| Method | Training | Test | Training | Test |
| NB | 0.25112 | 0.25198 | 0.17100 | 0.17321 |
| Sigmoid NB | 0.21530 | 0.21515 | 0.15270 | 0.15190 |
| PAV NB | 0.20312 | 0.20452 | 0.14665 | 0.14831 |
| SVM | 0.28719 | 0.28684 | 0.15190 | 0.14968 |
| Sigmoid SVM | 0.20980 | 0.20962 | 0.15156 | 0.14993 |
| PAV SVM | 0.20815 | 0.20924 | 0.15115 | 0.15113 |

**Table 3: MSE and error rate on the Adult dataset.**

Major things to note: PAV (I.e. isotonic regression) works in a way comparable to Platt's method on SVMs and better for NB.

# Results (multi-class)

| Method | MSE | Error Rate |
|---|---|---|
| NB Normalization | 0.0326 | 0.1672 |
| NB Least-Squares | 0.0319 | 0.1672 |
| NB Coupling | 0.0304 | 0.1715 |
| PAV NB Normalization | 0.0241 | 0.1498 |
| PAV NB Least-Squares | 0.0260 | 0.1498 |
| PAV NB Coupling | 0.0260 | 0.1512 |
| BNB Normalization | 0.0163 | 0.0963 |
| BNB Least-Squares | 0.0164 | 0.0958 |
| BNB Coupling | 0.0160 | 0.1023 |
| PAV BNB Normalization | 0.0150 | 0.0946 |
| PAV BNB Least-Squares | 0.0150 | 0.0946 |
| PAV BNB Coupling | 0.0149 | 0.0935 |

**Table 4: MSE and error rate on Pendigits (test set)**

Major things to note: Normalization is very close in performance to least-squares and coupling.  PAV (I.e. isotonic regression) does help boost performance.

# Conclusion

- Isotonic regression works for various models (i.e. SVMs and NB) in two-class problems
- One-against-all with normalized probabilities works well for multi-class problems, although using some of the more sophisticated methods might perform slightly better