# Model Selection and Naïve Bayes

Machine Learning - 10601

Geoff Gordon, Miroslav Dudík

[partly based on slides of Tom Mitchell]

http://www.cs.cmu.edu/~ggordon/10601/

September 23, 2009

# Announcements

The New York Times

September 21,2009:

**Netflix awards $1 Million prize to a team of statisticians, machine-learning experts and computer engineers**


**“You’re getting Ph.D.’s for a dollar an hour,” Reed Hastings, chief of Netflix, said of the people competing for the prize.**

## How to win $1 Million

Goal:

(user,movie) -> rating $\in [1,5]$

Data:

100M (user,movie,date,rating) tuples

RESPONSE

Performance measure:

root mean squared error

on **withheld test set**

## How to win $1 Million

A part of the winning model is the "baseline model" capturing bulk of the information:

FEATURES

[Koren 2009]

LOSS ~ ERROR

$$\min_{b_*,c_*,\alpha_*} \sum_{(u,i)\in\mathcal{K}} \Big(r_{ui} - \mu - b_u - \alpha_u \cdot \mathrm{dev}_u(t_{ui}) - b_{u,t_{ui}} - (b_i + b_{i,\mathrm{Bin}(t_{ui})}) \cdot (c_u + c_{u,t_{ui}})\Big)^2 + \lambda_a b_u^2 + \lambda_b \alpha_u^2 + \lambda_c b_{u,t_{ui}}^2 + \lambda_d b_i^2 + \lambda_e b_{i,\mathrm{Bin}(t_{ui})}^2 + \lambda_f (c_u - 1)^2 + \lambda_g c_{u,t_{ui}}^2$$

USER

MOVIE

REGULARIZATION

REGULARIZATION COEFFICIENTS

# How to win $1 Million

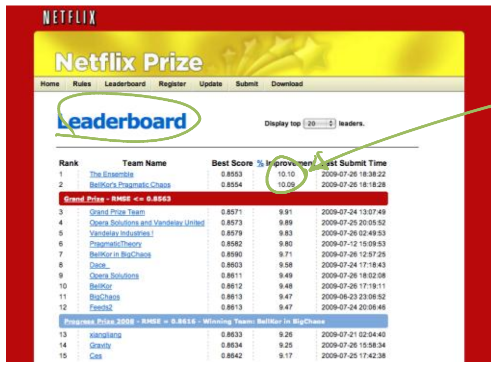
| training set | quiz set | test set |
|---|---|---|

FINAL ERROR ESTIMATE

USED FOR INTERIM RESULTS

GOAL: IMPROVE BY 10%

GIVEN TO CONTESTANTS



NETFLIX

Netflix Prize

Home | Rules | Leaderboard | Register | Update | Submit | Download

Leaderboard

Display top 20 leaders.

| Rank | Team Name | Best Score | % Improvement | Last Submit Time |
|---|---|---|---|---|
| 1 | The Ensemble | 0.8553 | 10.10 | 2009-07-26 18:38:22 |
| 2 | BellKor's Pragmatic Chaos | 0.8554 | 10.09 | 2009-07-26 18:18:28 |
| Grand Prize - RMSE <= 0.8563 | | | | |
| 3 | Grand Prize Team | 0.8571 | 9.91 | 2009-07-24 13:07:49 |
| 4 | Opera Solutions and Vandelay United | 0.8573 | 9.89 | 2009-07-25 20:05:52 |
| 5 | Vandelay Industries ! | 0.8579 | 9.83 | 2009-07-26 02:49:53 |
| 6 | PragmaticTheory | 0.8582 | 9.80 | 2009-07-12 15:09:53 |
| 7 | BellKor in BigChaos | 0.8590 | 9.71 | 2009-07-26 12:57:25 |
| 8 | Dace_ | 0.8603 | 9.58 | 2009-07-24 17:18:43 |
| 9 | Opera Solutions | 0.8611 | 9.49 | 2009-07-26 18:02:08 |
| 10 | BellKor | 0.8612 | 9.48 | 2009-07-26 17:19:11 |
| 11 | BigChaos | 0.8613 | 9.47 | 2009-06-23 23:06:52 |
| 12 | Feeds2 | 0.8613 | 9.47 | 2009-07-24 20:06:46 |
| Progress Prize 2008 - RMSE = 0.8616 - Winning Team: BellKor in BigChaos | | | | |
| 13 | xiangliang | 0.8633 | 9.26 | 2009-07-21 02:04:40 |
| 14 | Gravity | 0.8634 | 9.25 | 2009-07-26 15:58:34 |
| 15 | Ces | 0.8642 | 9.17 | 2009-07-25 17:42:38 |

10.10%
10.09%

# FAQ: why quiz/test split?

**Why this whole quiz/test subset structure? Why not reveal a submission's RMSE on the test subset?**

We wanted a way of informing you and your competitive colleagues about your progress toward a prize while making it difficult for you to simply train and optimize against "the answer oracle". We also wanted a way for the judges to determine how robust your algorithm is. So we have you supply nearly 3 million predictions, then tell you and the world how you did on one half (the "quiz" subset) while we judge you on how you did on the other half (the "test" subset), without telling you that score or which prediction you make applies to which subset.

We wanted a way of informing you … about your progress … while making it difficult for you to simply train and optimize against "the answer oracle"

# FAQ: why quiz/test split?

**Leaderboard** Display top 20 leaders.

| Rank | Team Name | Best Score | % Improvement | Last Submit Time |
|---|---|---|---|---|
| 1 | The Ensemble | 0.8553 | 10.10 | 2009-07-26 18:38:22 |
| 2 | BellKor's Pragmatic Chaos | 0.8554 | 10.09 | 2009-07-26 18:18:28 |

**Grand Prize** - RMSE <= 0.8563

WON BECAUSE OF TIME

IMPROVEMENT ON TEST: 10.06%

OPTIMISTIC BIAS

# Two goals for withholding data

- model selection
  which FEATURES, what REGULARIZATION
- model assessment
  determining "true" error

| training set | validation set | test set |
|---|---|---|

EXTENDED TRAINING SET

What if data is scarce?

# Cross-validation

5 ~ 10

- split data randomly into **K** equal parts

| Part 1 | Part 2 | Part 3 |
|---|---|---|

- for each model setting:
  evaluate avg performance across **K** train-test splits

| | | | |
|---|---|---|---|
| evaluate error | Part 2 | Part 3 | → ERROR 1 |
| Part 1 | evaluate error | Part 3 | → ERROR 2 |
| Part 1 | Part 2 | evaluate error | → ERROR 3 |

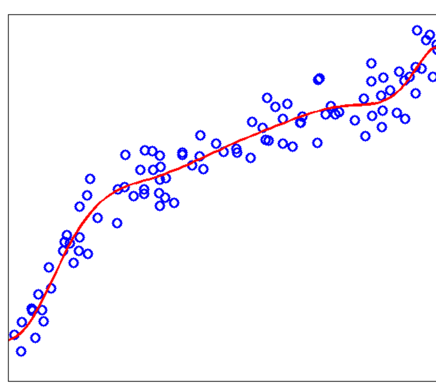
AVG = CROSS-VALID ERROR

- train the best model on the full data set

# The best model…

Depends on the size of the data set:

$$y \approx w_0 + w_1x + w_2x^2 + w_3x^3 + w_4x^4 + \ldots + w_{10}x^{10}$$



K-fold cross-validation trains on $\frac{K-1}{K}$ of the training data

MINOR PESSIMISTIC BIAS
(CROSS-VALIDATED MODELS SIMPLER THAN NECESSARY)

## Controlling model complexity

- limit the number of features
- add a "complexity penalty"

# Regularized estimation

$$\frac{\lambda}{2}\sum_j w_j^2$$

COMPLEXITY PENALTY

$$\min_w \left[\underline{\text{error}_{\text{train}}(w)} + \underbrace{\text{regularization}(w)}_{}\right]$$

NEG. LOG. LIKELIHOOD + NEG. LOG. PRIOR = $\boxed{\text{MAP}}$

$$\min_w \left[\underbrace{-\log p(\text{data}|w)}_{} - \underbrace{\log p(w)}_{}\right]$$

# BITS TO TRANSMIT THE DATA ONCE w IS KNOWN

$[-\log p(w)]$ bits NEEDED TO TRANSMIT INFO ABOUT w

MORE SURPRISING EVENTS NEED MORE BITS

$\boxed{\text{MINIMUM DESCRIPTION LENGTH}}$

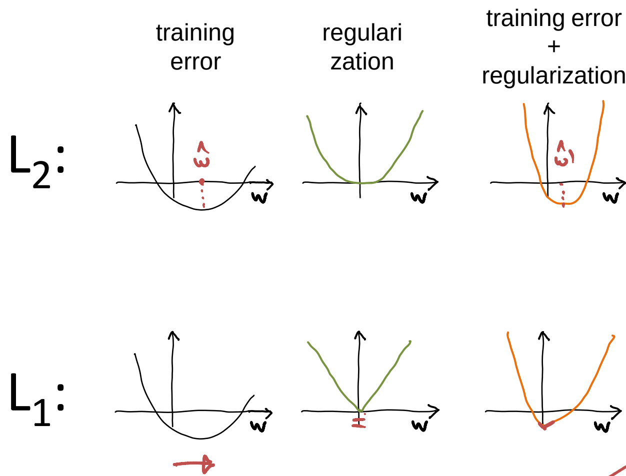
# Examples of regularization

$$\min_w \left[\frac{1}{2}\sum_n (y_n - \phi(x_n)\cdot w)^2 + \frac{1}{2}\lambda \underbrace{\|w\|_2^2}_{\sum_j w_j^2}\right]$$

RIDGE REGRESSION
$L_2$ PENALTY

$$\min_w \left[\frac{1}{2}\sum_n (y_n - \phi(x_n)\cdot w)^2 + \lambda \underbrace{\|w\|_1}_{\sum_j |w_j|}\right]$$

LASSO
$L_1$ PENALTY




SHRINKS ESTIMATE TOWARDS ZERO

OFTEN SHRINKS ESTIMATE TO ZERO

OFTEN YIELDS SPARSE SOLUTIONS

# $L_1$ vs $L_2$

$L_1$

- sparse solutions — E.G. MANY FEATURES IRRELEVANT
- more suitable when #features much larger than training set — #features can grow almost exponentially with training set size

$L_2$

- computationally better-behaved
- if we expect all or most features are relevant (and have enough data)

**How do you choose $\lambda$?** BY (CROSS) VALIDATION

# Announcements

**HW #3** out

due October 7

PROJECT PROPOSALS DUE

# Classification

Goal:

hypothesis

EXAMPLES DESCRIBED BY FEATURES

A SMALL # OF DISCR. VALS } LABELS CLASSES

learn a map **h: x ↦ y**

EMAIL ↦ SPAM/NOT SPAM

20x20 PXLS ↦ DIGITS

Data:

$(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2) \ldots, (\mathbf{x}_N, \mathbf{y}_N)$

Performance measure:

MISCLASSIFICATION ERROR

$\min_h [\#\text{ wrong predictions}] = \min_h \Pr[\text{wrong prediction}]$

All you need to know is **p(X,Y)**…

If you knew **p(X,Y)**, how would you classify an example **x**?

$h(x) = \arg\max_y p(x,y) = \arg\max_y p(y|x)$

Why?

$$\text{error} = \int_{x,y} \delta(h(x) \neq y)\, p(x,y)\, dx\, dy$$

1 if statement true

$p(x,y) = p(y|x)\,p(x)$

$$\Pr[\text{error}|x] = \int_y \delta(h(x) \neq y)\, p(y|x)\, dy$$

minimized if $h(x) = \arg\max_y p(y|x)$

$y = 0 \cdots$
$y = 1 = h(x)$
$\vdots$
$y = \cdots$

# How many parameters need to be estimated?

**Y** binary = ENJOY / NOT ENJOY

$\boxed{X}$ described by **M** binary features $X_1, X_2, \ldots, X_M$

6

Data:

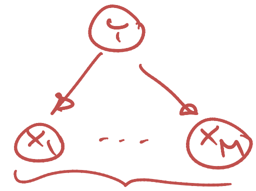
| Sky | Temp | Humid | Wind | Water | Forecst | EnjoySpt |
|---|---|---|---|---|---|---|
| Sunny | Warm | Normal | Strong | Warm | Same | Yes |
| Sunny | Warm | High | Strong | Warm | Same | Yes |
| Rainy | Cold | High | Strong | Warm | Change | No |
| Sunny | Warm | High | Strong | Cool | Change | Yes |

**p(X,Y)** described by $2^6 \cdot 2 - 1$ numbers $\cdots\ 2^M \cdot 2 - 1$

# choices for X    # choices for Y

# Naïve Bayes Assumption

- features of **X** conditionally independent given class **Y**




DIFFERENT FEATURES OF THE SAME EXAMPLE

HOW MANY PARAMS ?

$2M+1 \ll 2 \cdot 2^M - 1$

= DECISION:

$$h(x) = \arg\max_y \left[ p(y) \prod_j (x_j|y) \right]$$

OPTIMAL IF OUR ASSUMPTION CORRECT

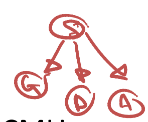
$$p(Y) \prod_j p(X_j|Y)$$

1

1 per each value of Y, and each j

## Example: Live in Sq Hill?

$p(S,G,D,A)$




- **S=1** iff live in Sq Hill
- **G=1** iff shop in Sq Hill Giant Eagle
- **D=1** iff drive to CMU
- **A=1** iff owns a Mac

$P(S=1) = \frac{9+1}{9+26+2} = .27$ $\quad p(S=0) = .73$

$P(G=1|S=0) = \frac{7+1}{26+2} = .29$ $\quad p(G=0|S=0) = .71$

$P(G=1|S=1) = \frac{7+1}{9+2} = .73$

$P(D=1|S=0) = \frac{2+1}{26+2} = .11$

$P(D=1|S=1) = \frac{1}{9+2} = .09$

$P(A=1|S=0) = \frac{12+1}{26+2} = .46$

$P(A=1|S=1) = \frac{1+1}{9+2} = .18$

G = 1
D = 0
A = 0

PREDICT $\rightsquigarrow$ S = 1

$p(S=1, G=1, D=0, A=0)$
$= .27 \times .73 \times .91 \times .82 = .15$

$p(S=0, \ldots)$
$= .73 \times .29 \times .89 \times .54 = .10$

## Naïve Bayes Assumption

- usually incorrect…
- Naïve Bayes often performs well, even when the assumption is violated [see Domingos-Pazzani 1996]

IN MISCLASS ERROR
(USUALLY NOT GOOD FOR PROBABILITIES)

## Learning to classify text documents

- which emails are spam? Y = SPAM/NOT SPAM
- which emails promise an attachment?
- which web pages are student home pages?

What are the features of **X**?

TEXT

# Feature $X_j$ is the $j$th word

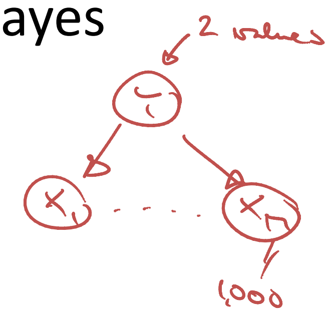
**Article from rec.sport.hockey**

```
Path: cantaloupe.srv.cs.cmu.edu!das-news.harvard.e
From: xxx@yyy.zzz.edu (John Doe)
Subject: Re: This year's biggest and worst (opinic
Date: 5 Apr 93 09:53:39 GMT

I can only comment on the Kings, but the most
obvious candidate for pleasant surprise is Alex
Zhitnik. He came highly touted as a defensive
defenseman, but he's clearly much more than that.
Great skater and hard shot (though wish he were
more accurate). In fact, he pretty much allowed
the Kings to trade away that huge defensive
liability Paul Coffey. Kelly Hrudey is only the
biggest disappointment if you thought he was any
good to begin with. But, at best, he's only a
mediocre goaltender. A better choice would be
Tomas Sandstrom, though not through any fault of
his own, but because some thugs in Toronto decided
```

ABOUT SPORTS

# Assumption #1: Naïve Bayes




1000 positions in document

10,000 dictionary

$\approx 10{,}000^{1{,}000}$ # TO MODEL FULLY
NUMBERS

=

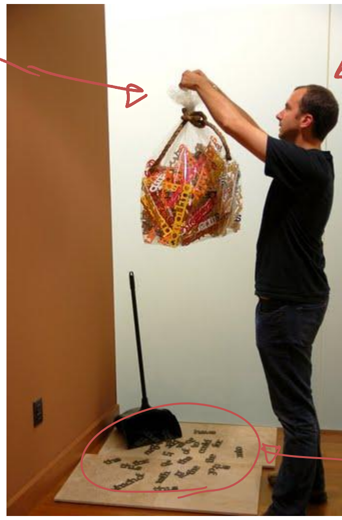
$p(Y) \leftarrow$ 1 value

$p(x_j \mid Y=0) \leftarrow 10{,}000 - 1$
$\phantom{p(x_j \mid} Y=1 \leftarrow 10{,}000 - 1$ } 1,000

$\approx 10{,}000 \times 2 \times 1{,}000 \approx 20M$ NUMBERS
TO MODEL AS NAIVE BAYES

# Assumption #2: “Bag of words”

BAG OF WORDS



JOE T

$p(x_j|Y) = P(x_k|Y)$

for all $j \neq k$

NAIVE BAYES: 20M

$(10{,}000-1)\times 2+1$

$\approx$ 20K NUMBERS

BAG OF WORDS

$\underbrace{\text{“the”, “a”, “and”, “of”, ...}}_{\text{STOP WORDS}}$

# “Bag of words” approach


| | |
|---|---|
| aardvark | 0 |
| about | 2 |
| all | 2 |
| Africa | 1 |
| apple | 0 |
| anxious | 0 |
| ... | |
| gas | 1 |
| ... | |
| oil | 1 |
| … | |
| Zaire | 0 |

10,000 ENTRIES

## Twenty NewsGroups

Given 1000 training documents from each group
Learn to classify new documents according to
which newsgroup it came from

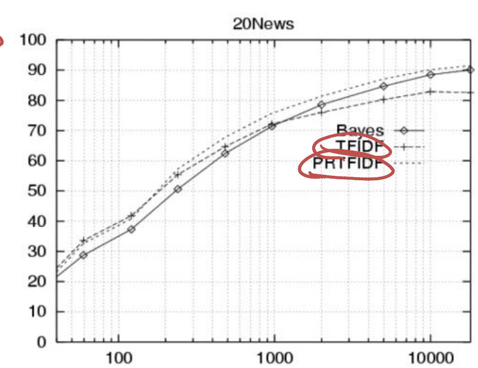
| | |
|---|---|
| comp.graphics | misc.forsale |
| comp.os.ms-windows.misc | rec.autos |
| comp.sys.ibm.pc.hardware | rec.motorcycles |
| comp.sys.mac.hardware | rec.sport.baseball |
| comp.windows.x | rec.sport.hockey |
| alt.atheism | sci.space |
| soc.religion.christian | sci.crypt |
| talk.religion.misc | sci.electronics |
| talk.politics.mideast | sci.med |
| talk.politics.misc | |
| talk.politics.guns | |

BAG OF WORDS + NAIVE BAYES

Naive Bayes: 89% classification accuracy

## Learning Curve for 20 Newsgroups

1-misclass error



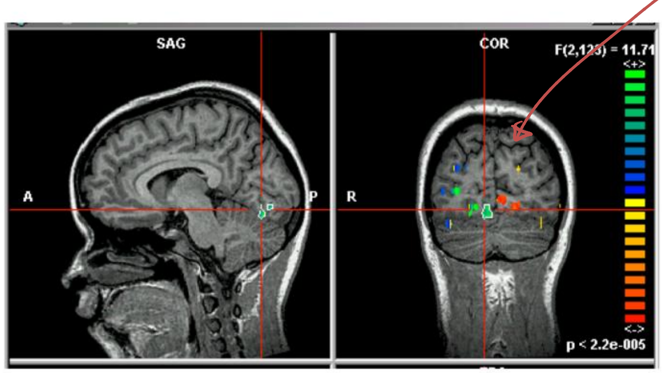

Accuracy vs. Training set size (1/3 withheld for test)

# training examples

## What if we have continuous $X_i$ ?

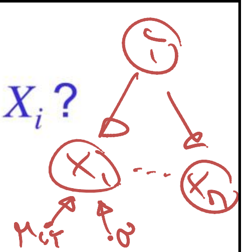
Eg., image classification: $X_i$ is i^th pixel

INTENSITY OF VOXELS ∝ BRAIN ACTIVITY




## What if we have continuous $X_i$ ?

Eg., image classification: $X_i$ is i^th pixel




Gaussian Naïve Bayes (GNB): assume

$$P(X_i = x \mid Y = y_k) = \frac{1}{\sigma_{ik}\sqrt{2\pi}} \; e^{\frac{-(x-\mu_{ik})^2}{2\sigma_{ik}^2}}$$

EACH CLASS k

EACH VOXEL (FEATURE i)

Sometimes assume variance

- is independent of Y (i.e., $\sigma_i$),
- or independent of $X_i$ (i.e., $\sigma_k$)
- or both (i.e., $\sigma$)

## Gaussian Naïve Bayes Algorithm – continuous $X_i$ (but still discrete Y)

- Train Naïve Bayes (examples)
  for each value $y_k$
  estimate $\pi_k \equiv P(Y = y_k)$
  for each attribute $X_i$ estimate
  class conditional mean $\mu_{ik}$, variance $\sigma_{ik}$
- Classify ($X^{new}$)

$$Y^{new} \leftarrow \arg\max_{y_k} \; P(Y = y_k) \prod_i P(X_i^{new} | Y = y_k)$$

$$Y^{new} \leftarrow \arg\max_{y_k} \; \pi_k \prod_i Normal(X_i^{new}, \mu_{ik}, \sigma_{ik})$$

## Estimating Parameters: $Y$ discrete, $X_i$ continuous

Maximum likelihood estimates:

P(X|Y)

jth training example

$$\hat{\mu}_{ik} = \frac{1}{\sum_j \delta(Y^j = y_k)} \sum_j X_i^j \delta(Y^j = y_k)$$

AVERAGE ACROSS EXAMPLES LABELED W/ K

ith feature

kth class

δ(z)=1 if z true, else 0

CAN ALSO APPLY SMOOTHING

$$\hat{\sigma}_{ik}^2 = \frac{1}{\sum_j \delta(Y^j = y_k)} \sum_j (X_i^j - \hat{\mu}_{ik})^2 \delta(Y^j = y_k)$$

EMPIRICAL VARIANCE among examples from class k

DISCRETE

P(Y) ← (with LAPLACE SMOOTHING) EMPIRICAL FREQ

# What you should know about Naïve Bayes

**Naïve Bayes**

- assumption
- why we use it

**Text classification**

- bag of words model

**Gaussian Naïve Bayes**

- each feature a Gaussian given the class