Drs. Sha & Liu

{feisha,yanliu.cs}@usc.edu

September 16, 2014
Outline

1. Administration
2. Review of last lecture
3. Naive Bayes
A few announcements

- My office hour: weekly on Wed between 10am to 12pm
- Homework: to be released very soon (about later today)
- Reading assignments: on the course webpage
  - http://www-bcf.usc.edu/~feisha/csci567_fall2014/ (this section)
  - http://www-bcf.usc.edu/~liu32/fall2014.html (Prof. Liu's section)
Outline

1. Administration

2. Review of last lecture
   - Validation and cross-validation
   - Decision tree

3. Naive Bayes
Tuning hyper-parameters

**Examples:**
- $K$: the number of nearest neighbors in $K$-nearest neighbor classification
- The depth of the decision tree

**Selecting the optimal hyper-parameters**
- Cannot use the performance metrics on the training dataset: will be too optimistic
- Need an independent dataset
Validation and cross-validation

Development (or validation) data

- L samples/instances: \( D^{\text{DEV}} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_L, y_L)\} \)
- They are used to optimize hyperparameter(s).

Cross validation

- We split the training data into S equal parts.
- We use each part \textit{in turn} as a validation dataset and use the others as a training dataset.
- We choose the hyperparameter such that \textit{on average}, the model performing the best.

Training data, validation and test data should \textit{not} overlap!
A tree partitions the feature space

\[ x_1 > \theta_1 \]
\[ x_2 \leq \theta_2 \]
\[ x_1 \leq \theta_4 \]
\[ x_2 \leq \theta_3 \]

A B C D E

\[ x_1 > \theta_1 \]
\[ x_2 \leq \theta_2 \]
\[ x_1 \leq \theta_4 \]
\[ x_2 > \theta_3 \]

A B C D E
Learning a tree model

Three things to learn:

1. The structure of the tree.
2. The threshold values ($\theta_i$).
3. The values for the leaves ($A, B, \ldots$).
Use information gain to decide how to branch

**Which attribute to split?**

*Patrons?* is a better choice—gives information about the classification

Idea: use information gain to choose which attribute to split
Outline

1. Administration

2. Review of last lecture

3. Naive Bayes
   - Motivating example
   - Naive Bayes: informal definition
   - Parameter estimation
A daily battle

Great news: I will be rich!

FROM THE DESK OF MR. AMINU SALEH
DIRECTOR, FOREIGN OPERATIONS DEPARTMENT
AFRI BANK PLC
Afribank Plaza,
14th Floor
51/55 Broad Street,
P.M.B 12021 Lagos-Nigeria

Attention: Honorable Beneficiary,

IMMEDIATE PAYMENT NOTIFICATION

It is my modest obligation to write you this letter in regards to the authorization of your owed payment through our most respected financial institution (AFRI BANK PLC). I am Mr. Aminu Saleh, The Director, Foreign Operations Department, AFRI Bank Plc, NIGERIA. The British Government, in conjunction with the US Government, World Bank, United Nations Organization on foreign payment matters, has empowered my bank after much consultation and consideration, to handle all foreign payments and release them to their appropriate beneficiaries with the help of a representative from Federal Reserve Bank.

To facilitate the process of this transaction, please kindly re-confirm the following information below:

1) Your full Name and Address:
2) Phones, Fax and Mobile No.:
3) Profession, Age and Marital Status:
4) Copy of any valid form of your Identification:
How to tell spam from ham?

FROM THE DESK OF MR. AMINU SALEH
DIRECTOR, FOREIGN OPERATIONS DEPARTMENT
AFRI BANK PLC
Afribank Plaza,
14th Floor
51/55 Broad Street,
P.M.B 12021 Lagos-Nigeria

Attention: Honorable Beneficiary,

IMMEDIATE PAYMENT NOTIFICATION VALUED AT US$10 MILLION

Dear Dr. Sha,

I just would like to remind you of your scheduled presentation for CS597, Monday October 13, 12pm at OHE122.

If there is anything that you would need, please do not hesitate to contact me.

sincerely,

Christian Siagian
Intuition

How human solves the problem?

Spam emails

concentrated use of a lot of words like “money”, “free”, “bank account”, “viagara”

Ham emails

word usage pattern is more spread out
Simple strategy: count the words

Bag-of-word representation of documents (and textual data)

\[
\begin{pmatrix}
\text{free} & 100 \\
\text{money} & 2 \\
\vdots & \vdots \\
\text{account} & 2 \\
\vdots & \vdots 
\end{pmatrix}
\]

\[
\begin{pmatrix}
\text{free} & 1 \\
\text{money} & 1 \\
\vdots & \vdots \\
\text{account} & 2 \\
\vdots & \vdots 
\end{pmatrix}
\]
Weighted sum of those telltale words:

$$\begin{pmatrix}
100 \times 0.2 \\
2 \times 0.3 \\
\vdots
\end{pmatrix}$$

Different weights for spam and ham: representing how compatible the word usage pattern is to different category

$$\begin{pmatrix}
100 \times 0.01 \\
2 \times 0.02 \\
\vdots
\end{pmatrix}$$

$$= 3.2$$

$$= 1.03$$
Our intuitive model of classification

Assign weight to each word

Compute compatibility score to “spam”

\[ \text{# of “free” } \times a_{\text{free}} + \text{# of “account” } \times a_{\text{account}} + \text{# of “money” } \times a_{\text{money}} \]

Compute compatibility score to “ham”:

\[ \text{# of “free” } \times b_{\text{free}} + \text{# of “account” } \times b_{\text{account}} + \text{# of “money” } \times b_{\text{money}} \]

Make a decision:

if spam score > ham score then spam

else ham
How we get the weights?

Learning from experience

get a lot of spams

get a lot of hams

But what to optimize?
Naive Bayes model for identifying spams

Class label: binary

\[ y = \{ \text{spam}, \text{ham} \} \]

Features: word counts in the document (Bag-of-word)

Ex: \[ x = \{('free', 100), ('lottery', 10), ('money', 10), , ('identification', 1)\} \]

Each pair is in the format of \((w_i, \#w_i)\), namely, a unique word in the dictionary, and the number of times it shows up.
Naive Bayes model for identifying spams

\[ p(x|y) = \prod_{i} p(w_i|y)^{#w_i} \]

These conditional probabilities are model parameters.
Spam writer’s vocabulary

Features: word counts in the document

Ex: \( x = \{(‘free’, 100), (‘identification’, 2), (‘lottery’, 10), (‘money’, 10), \ldots \} \)

Model: Naive Bayes (NB)

\[
p(x|\text{spam}) = p(‘free’|\text{spam})^{100} p(‘identification’|\text{spam})^{2} \\
\quad p(‘lottery’|\text{spam})^{10} p(‘money’|\text{spam})^{10} \ldots \\
\quad \neq p(x|\text{ham})
\]

Parameters to be estimated:
\( p(‘free’|\text{spam}), p(‘free’|\text{ham}), \text{etc} \)
Naive Bayes

Why the name “naive”?  

Strong assumption of conditional independence:

\[ p(w_i, w_j | y) = p(w_i | y) p(w_j | y) \]

How to estimate model parameters?  

Use maximum likelihood estimation (soon)
Does this correspond to our intuitive model of classification?

Yes. It does!

Let us consider the Bayes optimal classifier under this assumed probabilistic distribution

\[ p(x|y) = p(w_1|y)^{w_1} p(w_2|y)^{w_2} \cdots p(w_m|y)^{w_m} \]
\[ = \prod_i p(w_i|y)^{w_i} \]
For any document $x$, we need to compute

$$p(\text{spam}|x) \quad \text{and} \quad p(\text{ham}|x)$$
Naive Bayes classification rule

For any document $x$, we need to compute

$$p(\text{spam} | x) \quad \text{and} \quad p(\text{ham} | x)$$

Using Bayes rule, this gives rise to

$$p(\text{spam} | x) = \frac{p(x | \text{spam}) p(\text{spam})}{p(x)}, \quad p(\text{ham} | x) = \frac{p(x | \text{ham}) p(\text{ham})}{p(x)}$$
**Naive Bayes classification rule**

For any document $x$, we need to compute

$$p(\text{spam}|x) \quad \text{and} \quad p(\text{ham}|x)$$

Using Bayes rule, this gives rise to

$$p(\text{spam}|x) = \frac{p(x|\text{spam})p(\text{spam})}{p(x)} , \quad p(\text{ham}|x) = \frac{p(x|\text{ham})p(\text{ham})}{p(x)}$$

It is convenient to compute the logarithms, so we need only to compare

$$\log[p(x|\text{spam})p(\text{spam})] \quad \text{versus} \quad \log[p(x|\text{ham})p(\text{ham})]$$

as the denominators are the same.
Classifier in the linear form of compatibility scores

\[
\begin{align*}
\log[p(x|\text{spam})p(\text{spam})] &= \log \left[ \prod_i p(w_i|\text{spam})^\#w_i p(\text{spam}) \right] \\
&= \sum_i \#w_i \log p(w_i|\text{spam}) + \log p(\text{spam})
\end{align*}
\]
Classifier in the linear form of compatibility scores

\[
\log[p(x|\text{spam})p(\text{spam})] = \log \left[ \prod_i p(w_i|\text{spam})^\#w_i p(\text{spam}) \right] = \sum_i \#w_i \log p(w_i|\text{spam}) + \log p(\text{spam}) \quad (1)
\]

Similarly, we have

\[
\log[p(x|\text{ham})p(\text{ham})] = \sum_i \#w_i \log p(w_i|\text{ham}) + \log p(\text{ham})
\]

Namely, we are back to the idea of comparing weighted sum of \# of word occurrences!

\(\log p(\text{spam})\) and \(\log p(\text{ham})\) are called “priors” or “bias” (they are not in our intuition but they are crucially needed)
Mini-summary

What we have shown
By making a probabilistic model (i.e., Naive Bayes), we are able to derive a decision rule that is consistent with our intuition

Our next step is to leverage this link to learn the rule from the data
Formal definition of Naive Bayes

**General case**
Given a random variable $X \in \mathbb{R}^D$ and a dependent variable $Y \in [C]$, the Naive Bayes model defines the joint distribution

$$P(X = x, Y = y) = P(Y = y)P(X = x|Y = y)$$  \hspace{1cm} (3)

$$= P(Y = y) \prod_{d=1}^{D} P(X_d = x_d|Y = y)$$  \hspace{1cm} (4)
Special case (i.e., our model of spam emails)

**Assumptions**

- All $X_d$ are categorical variables from the same domain — $x_d \in [K]$, for example, the index to the unique words in a dictionary.

- $P(X_d = x_d | Y = y)$ depends only on the value of $x_d$, not $d$ itself, namely, orders are not important (thus, we only need to count).

**Simplified definition**

$$P(X = x, Y = c) = P(Y = c) \prod_k P(k | Y = c)^{z_k} = \pi_c \prod_k \theta^{z_k}_{ck}$$

where $z_k$ is the number of times $k$ in $x$.

*Note that we only need to enumerate in the product, the index to the $x_d$’s possible values. On the previous slide, however, we enumerate over $d$ as we do not have the assumption there that order is not important.*
Learning problem

Training data

\[ \mathcal{D} = \{(x_n, y_n)\}_{n=1}^{N} \rightarrow \mathcal{D} = \{ (\{z_{nk}\}_{k=1}^{K}, y_n)\}_{n=1}^{N} \]

Goal

Learn \( \pi_c, c = 1, 2, \ldots, C, \) and \( \theta_{ck}, \forall c \in [C], k \in [K] \) under the constraint

\[ \sum_c \pi_c = 1 \]

and

\[ \sum_k \theta_{ck} = \sum_k P(k | Y = c) = 1 \]

as well as those quantities should be nonnegative.
Our hammer: maximum likelihood estimation

Log-Likelihood of the training data

\[ \mathcal{L} = \log P(D) = \log \prod_{n=1}^{N} \pi_{y_n} P(x_n|y_n) \] (5)

\[ = \log \prod_{n=1}^{N} \left( \pi_{y_n} \prod_{k} \theta_{y_{nk}} \right) \] (6)

\[ = \sum_{n} \left( \log \pi_{y_n} + \sum_{k} z_{nk} \log \theta_{y_{nk}} \right) \] (7)

\[ = \sum_{n} \log \pi_{y_n} + \sum_{n,k} z_{nk} \log \theta_{y_{nk}} \] (8)
Our hammer: maximum likelihood estimation

Log-Likelihood of the training data

\[
\mathcal{L} = \log P(\mathcal{D}) = \log \prod_{n=1}^{N} \pi_{y_n} P(x_n | y_n) 
\]

\[
= \log \prod_{n=1}^{N} \left( \pi_{y_n} \prod_{k} \theta_{y_n k}^{z_{nk}} \right) 
\]

\[
= \sum_{n} \left( \log \pi_{y_n} + \sum_{k} z_{nk} \log \theta_{y_n k} \right) 
\]

\[
= \sum_{n} \log \pi_{y_n} + \sum_{n,k} z_{nk} \log \theta_{y_n k} 
\]

Optimize it!

\[
(\pi_c^*, \theta_{ck}^*) = \arg \max \sum_{n} \log \pi_{y_n} + \sum_{n,k} z_{nk} \log \theta_{y_n k} 
\]
Details

Note the separation of parameters in the likelihood

\[
\sum_{n} \log \pi_{y_n} + \sum_{n,k} z_{nk} \log \theta_{y_n,k}
\]

which implies that \(\{\pi_c\}\) and \(\{\theta_{ck}\}\) can be estimated separately.

Reorganize terms

\[
\sum_{n} \log \pi_{y_n} = \sum_{c} \log \pi_{c} \times (\text{# of data points labeled as } \text{c})
\]

and

\[
\sum_{n,k} z_{nk} \log \theta_{y_n,k} = \sum_{c} \sum_{n:y_n=c} \sum_{k} z_{nk} \log \theta_{ck} = \sum_{c} \sum_{n:y_n=c,k} z_{nk} \log \theta_{ck}
\]

The later implies \(\{\theta_{ck}, k = 1, 2, \cdots, K\}\) and \(\{\theta_{c'k}, k = 1, 2, \cdots, K\}\) can be estimated independently.
Estimating $\{\pi_c\}$

We want to maximize

$$\sum_c \log \pi_c \times (\# \text{ of data points labeled as } c)$$

Intuition

- Similar to roll a dice (or flip a coin): each side of the dice shows up with a probability of $\pi_c$ (total C sides)
- And we have total N trials of rolling this dice

Solution

$$\pi^*_c = \frac{\# \text{ of data points labeled as } c}{N}$$
Estimating \( \{\theta_{ck}, k = 1, 2, \cdots, K\} \)

We want to maximize

\[
\sum_{n: y_n = c, k} z_{nk} \log \theta_{ck}
\]

Intuition

- Similar to roll a dice with color \( c \): each side of the dice shows up with a probability of \( \theta_{ck} \) (total \( K \) slides)
- And we have total \( \sum_{n: y_n = c, k} z_{nk} \) trials.

Solution

\[
\theta^*_{ck} = \frac{\# \text{of side-} k \text{ shows up in data points labeled as } c}{\# \text{of all slides in data points labeled as } c}
\]
Translating back to our problem of detecting spam emails

- Collect a lot of ham and spam emails as training examples
- Estimate the “bias”
  
  \[
  p(\text{ham}) = \frac{\#\text{ of ham emails}}{\#\text{ of emails}}, \quad p(\text{spam}) = \frac{\#\text{ of spam emails}}{\#\text{ of emails}}
  \]

- Estimate the weights (i.e., \(p(\text{dollar}|\text{ham})\) etc)
  
  \[
  p(\text{funny word}|\text{ham}) = \frac{\#\text{ of funny word in ham emails}}{\#\text{ of words in ham emails}} \tag{9}
  \]
  
  \[
  p(\text{funny word}|\text{spam}) = \frac{\#\text{ of funny word in spam emails}}{\#\text{ of words in spam emails}} \tag{10}
  \]
Classification rule

Given an unlabeled data point \( x = \{ z_k, k = 1, 2, \cdots, K \} \), label it with

\[
y^* = \arg \max_{c \in [C]} P(y = c | x) = \arg \max_{c \in [C]} P(y = c) P(x | y = c)
\]

\[
= \arg \max_{c \in [C]} \log \pi_c + \sum_k z_k \log \theta_{ck}
\]
A short derivation of the maximum likelihood estimation

**The steps are similar to the ones in Math Review**

To maximize

\[
\sum_{n:y_n=c} z_{nk} \log \theta_{ck}
\]

We use the Lagrangian multiplier

\[
\sum_{n:y_n=c,k} z_{nk} \log \theta_{ck} + \lambda \left( \sum_k \theta_{ck} - 1 \right)
\]

Taking derivatives with respect to \( \theta_{ck} \) and then find the stationary point

\[
\sum_{n:y_n=c} \frac{z_{nk}}{\theta_{ck}} + \lambda = 0 \rightarrow \theta_{ck} = -\frac{1}{\lambda} \sum_{n:y_n=c,k} z_{nk}
\]

Apply the constraint that \( \sum_k \theta_{ck} = 1 \),

\[
\theta_{ck} = \frac{\sum_{n:y_n=c,k} z_{nk}}{\sum_k \sum_{n:y_n=c} z_{nk}}
\]
Summary

You should know or be able to

- What naive Bayes model is
  - write down the joint distribution
  - explain the conditional independence assumption implied by the model
  - explain how this model can be used to distinguish spam from ham emails

- Be able to go through the short derivation for parameter estimation
  - The model illustrated here is called discrete Naive Bayes
  - Your homework asks you to apply the same principle to Gaussian naive Bayes
  - The derivation is very similar – except there you need to estimate Gaussian continuous random variables (instead of estimating discrete random variables like rolling a dice)

- think about another classification task that this model might be useful
To enhance your understanding

**write a personalized spam email detector yourself**

- Collect from your own email inbox, 500 samples of spam and good emails (the more, the merrier)
- Create a training (400 samples), validation (50 samples) and test dataset (50 samples)
- Estimate Naive Bayes model parameters for distinguishing ham and spam emails
- Apply the model to classify test dataset (you will use validation dataset later)
- Report your results on Discussion forum and post your questions of doing this experiment

*This recipe is not 100% bullet-proof. You will discover practical issues. Working on those issues will improve your understanding of the algorithm and its practice.*
Moving forward

Examine the classification rule for naive Bayes

\[ y^* = \arg \max_c \log \pi_c + \sum_k z_k \log \theta_{ck} \]

For binary classification problem, this is just to determine the label basing on

\[ \log \pi_1 + \sum_k z_k \log \theta_{1k} - \left( \log \pi_2 + \sum_k z_k \log \theta_{2k} \right) \]

This is just a linear function of the features \( \{z_k\} \)

\[ w_0 + \sum_k z_k w_k \]

where we “absorb” \( w_0 = \log \pi_1 - \log \pi_2 \) and \( w_k = \log \theta_{1k} - \log \theta_{2k} \).
Naive Bayes is a linear classifier

Fundamentally, what really matters in deciding decision boundary is

$$w_0 + \sum_k z_k w_k$$

This motivates many new methods. One of them is logistic regression, to be discussed in next lecture.