Week 8: Naive Bayes Classification

Data Science Certificate Program

Ryerson University

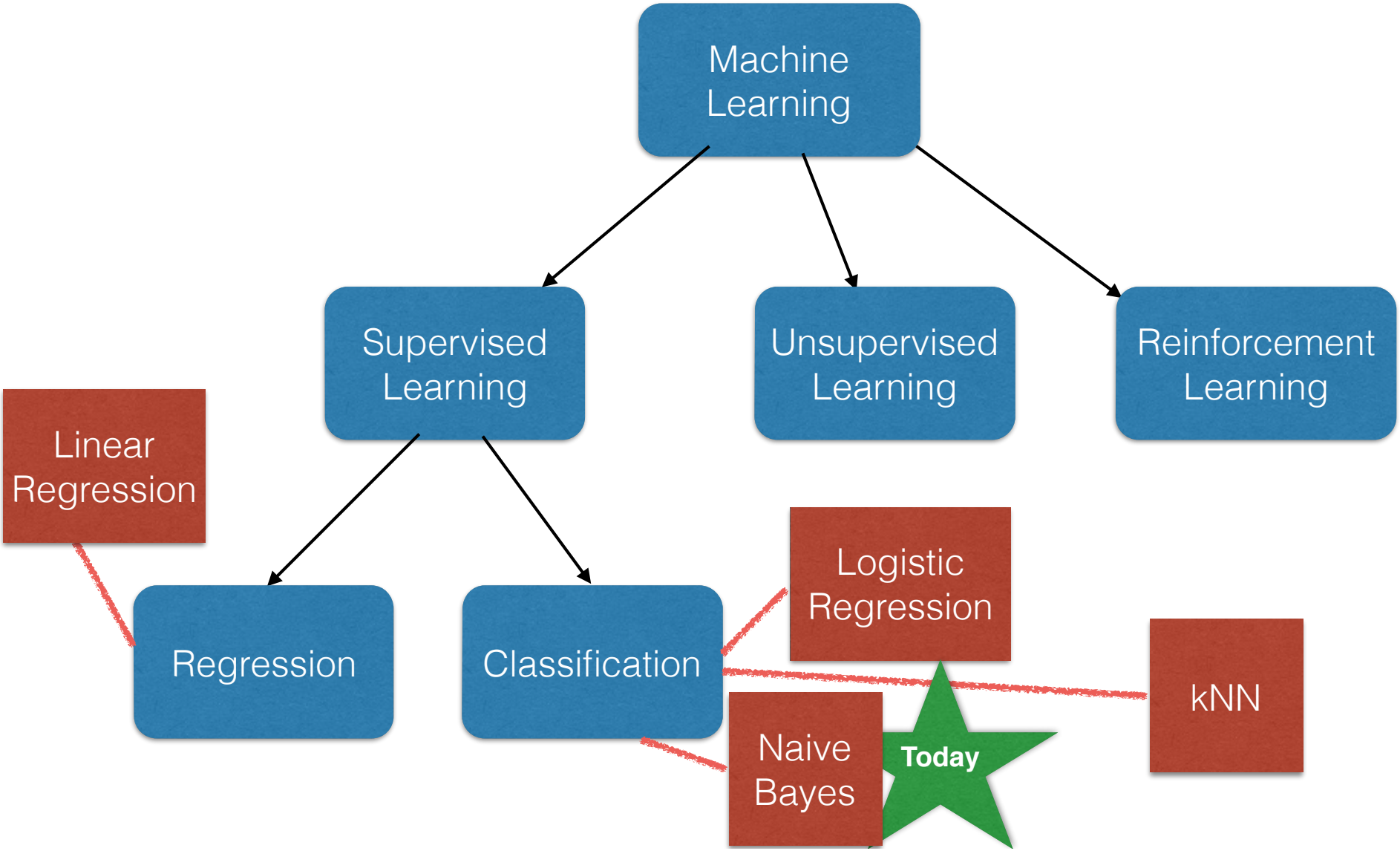
Bora Caglayan

12 Mar, 2015

Outline

- Bayes Rule
- Joint Probabilities and Independence Assumption
- Naive Bayes Model
- Naive Bayes for Spam Classification
- Introduction to Graphical Models
- Text Mining with R

Machine Learning Problems



Bayes Rule

Prior Probability

Likelihood of the evidence 'E' if the Hypothesis 'H' is true

$$P(H|E) = \frac{P(H) * P(E|H)}{P(E)}$$

Posterior Probability of 'H' given the evidence

Priori probability that the evidence itself is true

Bayes Rule

- What you know about the hypothesis after the data D arrive is what you knew before about the hypothesis and what the data D told you.

- D : 35 year old customer with an income of $50,000 PA
- h : Hypothesis that our customer will buy our computer

P(h/D) : Probability that customer D will buy our computer given that we know his age and income
P(h) : Probability that any customer will buy our computer regardless of age (Prior Probability)
P(D/h) : Probability that the customer is 35 yrs old and earns $50,000, given that he has bought our computer (Posterior Probability)
P(D) : Probability that a person from our set of customers is 35 yrs old and earns $50,000

$$P(h/D) = \frac{P(D/h)\,P(h)}{P(D)}$$

Joint Probabilities and Independence

Joint probabilities can be written as: $P(x_1, x_2..., x_n)$

$$P(x, y) = P(x|y) * P(x)$$

x and y are independent if: $P(x, y) = P(x) * P(y)$

Outcome of x do not depend on the outcome of y.

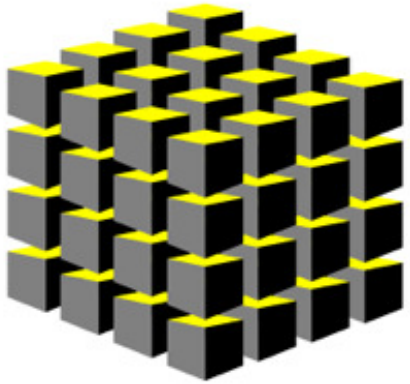
Example 1: rock music popularity, global mean temperature
Example 2: planet positions, probability of winning lottery

Counter Example: age, maximum heart rate …

Joint Probabilities and Independence

- Assume all x_i are discrete with $|x_i| = k$. If N is large, a naive table representation is HUGE: k^N entries

Example: $p(x_1, x_2, x_3)$ with $|x_i| = 4$

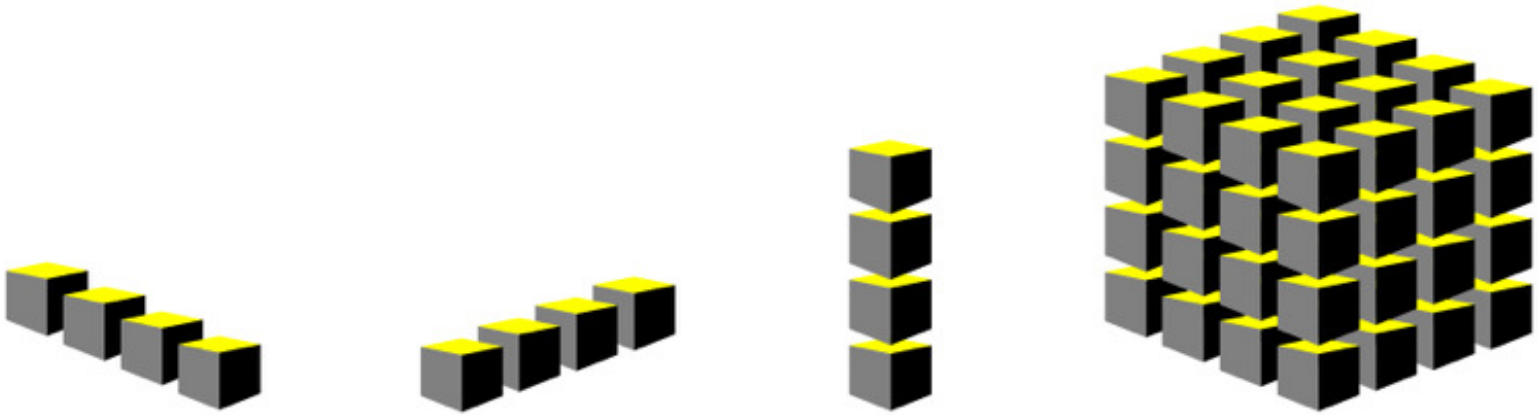


Each cell is a positive number s.t. $\sum_{x_1,x_2,x_3} p(x_1, x_2, x_3) = 1$

- We need efficient data structures to represent joint distributions $p(x_1, x_2, \ldots, x_N)$

Joint Probabilities and Independence

- Assume $p(x_1, x_2, \ldots, x_N) = \prod_k p(x_k)$.

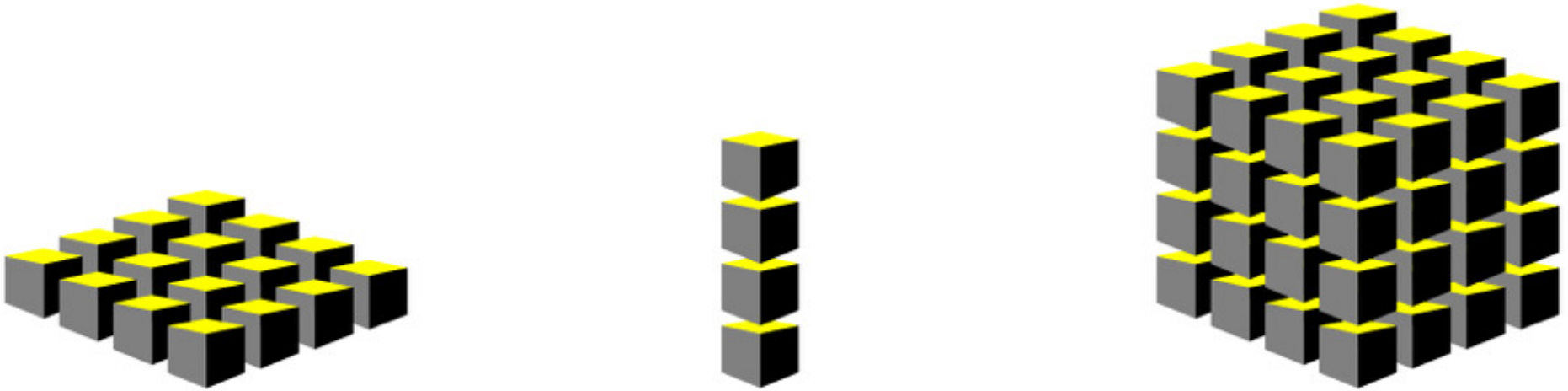


$p(x_1) \times p(x_2) \times p(x_3) = p(x_1, x_2, x_3)$

We need to store 4×3 numbers instead of 4^3 !

- However, complete independence is too restrictive and not very useful.

Joint Probabilities and Independence



$$p(x_1, x_2) \quad \times \quad p(x_3) \quad = \quad p(x_1, x_2, x_3)$$

We need to store $4^2 + 4$ numbers instead of 4^3.

- Still some variables are independent from rest. We will make conditional independence assumptions instead.

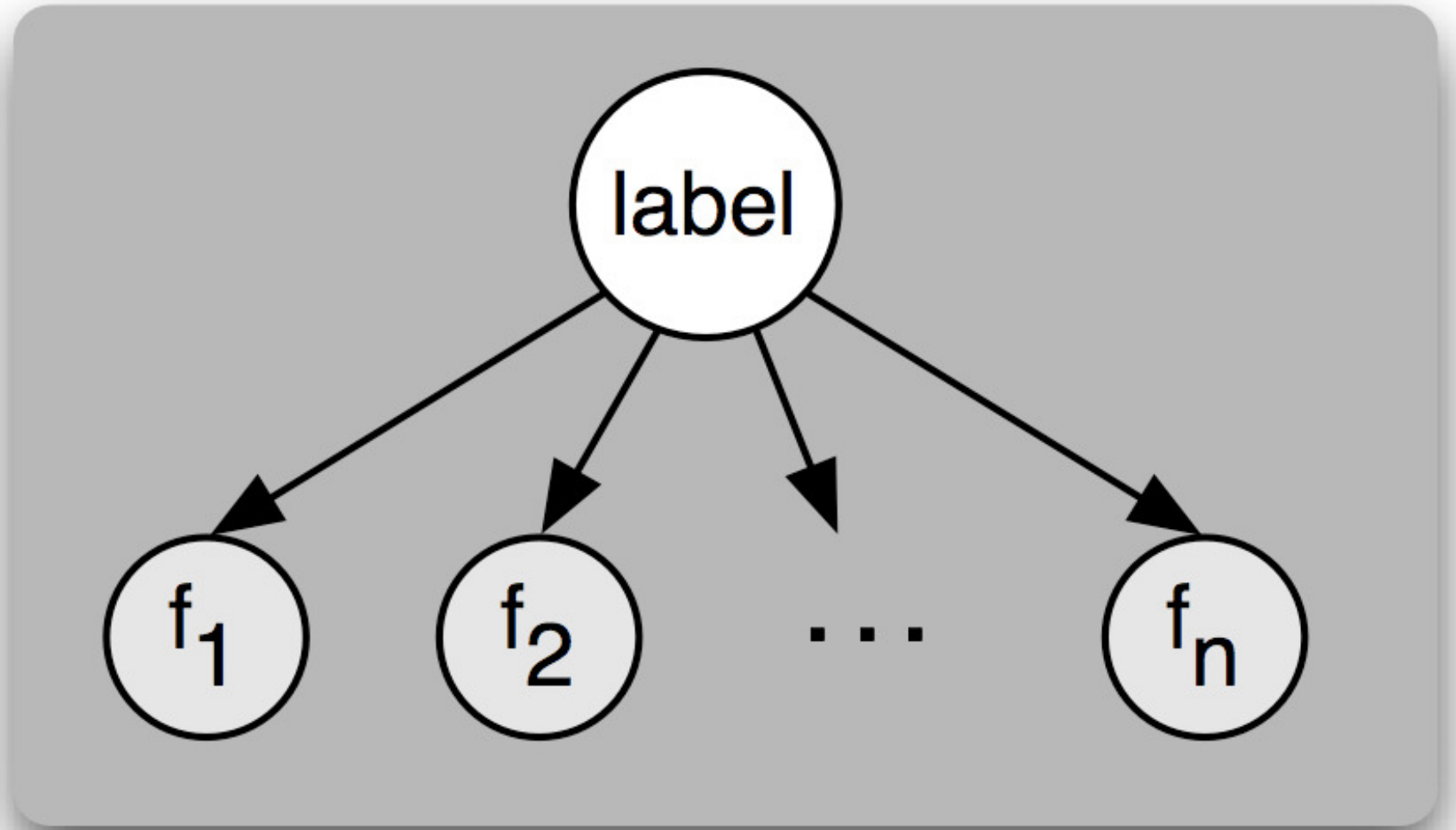
Naive Bayes Model

- Naive Bayes is the simplest Bayesian model for classification.
- **Assumption:** All of the input variables are independent of each other. The output label is dependent on all of the input variables.

$$P(y|(X_1, X_2...)) = \frac{1}{C} * P(X_1|y) * P(X_2|y) * P(y)...$$

$$C = P(X_1) * P(X_2)... \textit{ (not important)}$$

Naive Bayes Model



Naive Bayes Model

- Naive Bayes learning phase:
  - Find conditional probability tables for each input variable pair and the output variable using the training data.
- Evaluation phase:
  - Predict the classes of the test instances.

Naive Bayes Model

- Example: Play Tennis

PlayTennis: training examples

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Naive Bayes Model

- Learning Phase P(X|Y)

Outlook	Play=*Yes*	Play=*No*
Sunny	2/9	3/5
Overcast	4/9	0/5
Rain	3/9	2/5

Temperature	Play=*Yes*	Play=*No*
Hot	2/9	2/5
Mild	4/9	2/5
Cool	3/9	1/5

Humidity	Play=*Yes*	Play=No
High	3/9	4/5
Normal	6/9	1/5

Wind	Play=*Yes*	Play=*No*
Strong	3/9	3/5
Weak	6/9	2/5

$P(\text{Play}=Yes) = 9/14$ $P(\text{Play}=No) = 5/14$

Naive Bayes Model

- Test Phase
  - Given a new instance, predict its label

    $\mathbf{x}'$=(Outlook=*Sunny*, Temperature=*Cool*, Humidity=*High*, Wind=*Strong*)
  - Look up tables achieved in the learning phrase

P(Outlook=*Sunny*|Play=*Yes*) = 2/9

P(Temperature=*Cool*|Play=*Yes*) = 3/9

P(Huminity=*High*|Play=*Yes*) = 3/9

P(Wind=*Strong*|Play=*Yes*) = 3/9

P(Play=*Yes*) = 9/14

P(Outlook=*Sunny*|Play=*No*) = 3/5

P(Temperature=*Cool*|Play==*No*) = 1/5

P(Huminity=*High*|Play=*No*) = 4/5

P(Wind=*Strong*|Play=*No*) = 3/5

P(Play=*No*) = 5/14

P(*Yes*|$\mathbf{x}'$): [P(*Sunny*|Yes)P(*Cool*|*Yes*)P(*High*|Yes)P(*Strong*|*Yes*)]P(Play=*Yes*) = 0.0053

P(*No*|$\mathbf{x}'$): [P(*Sunny*|No) P(*Cool*|No)P(*High*|*No*)P(*Strong*|*No*)]P(Play=*No*) = 0.0206

Given the fact P(*Yes*|$\mathbf{x}'$) < P(*No*|$\mathbf{x}'$), we label $\mathbf{x}'$ to be "*No*".

Naive Bayes Model

- Naïve Bayes: the conditional independence assumption
  - Training is very easy and fast; just requiring considering each attribute in each class separately
  - Test is straightforward; just looking up tables or calculating conditional probabilities with estimated distributions
- A popular generative model
  - Performance competitive to most of state-of-the-art classifiers even in presence of violating independence assumption
  - Many successful applications, e.g., spam mail filtering
  - A good candidate of a base learner in ensemble learning
  - Apart from classification, naïve Bayes can do more…

Summary

Naive Bayes model

- **Definition:** Classification using the conditional probability tables of the variables.
- **Exploratory analysis:** Look for high correlations.
- **Preprocessing:** Changing scales, *feature selection, feature extraction*
- **Algorithms:** Conditional probability estimations
- **Experiment: k-fold cross validation, random split**
- **Performance criteria:** recall, precision, confusion matrix
- **Advantages:** Even if there is some interdependence in the input the models may run surprisingly well.
- **Disadvantage:** Model may be too simple for some problems.

Text Classification Problem

INPUT

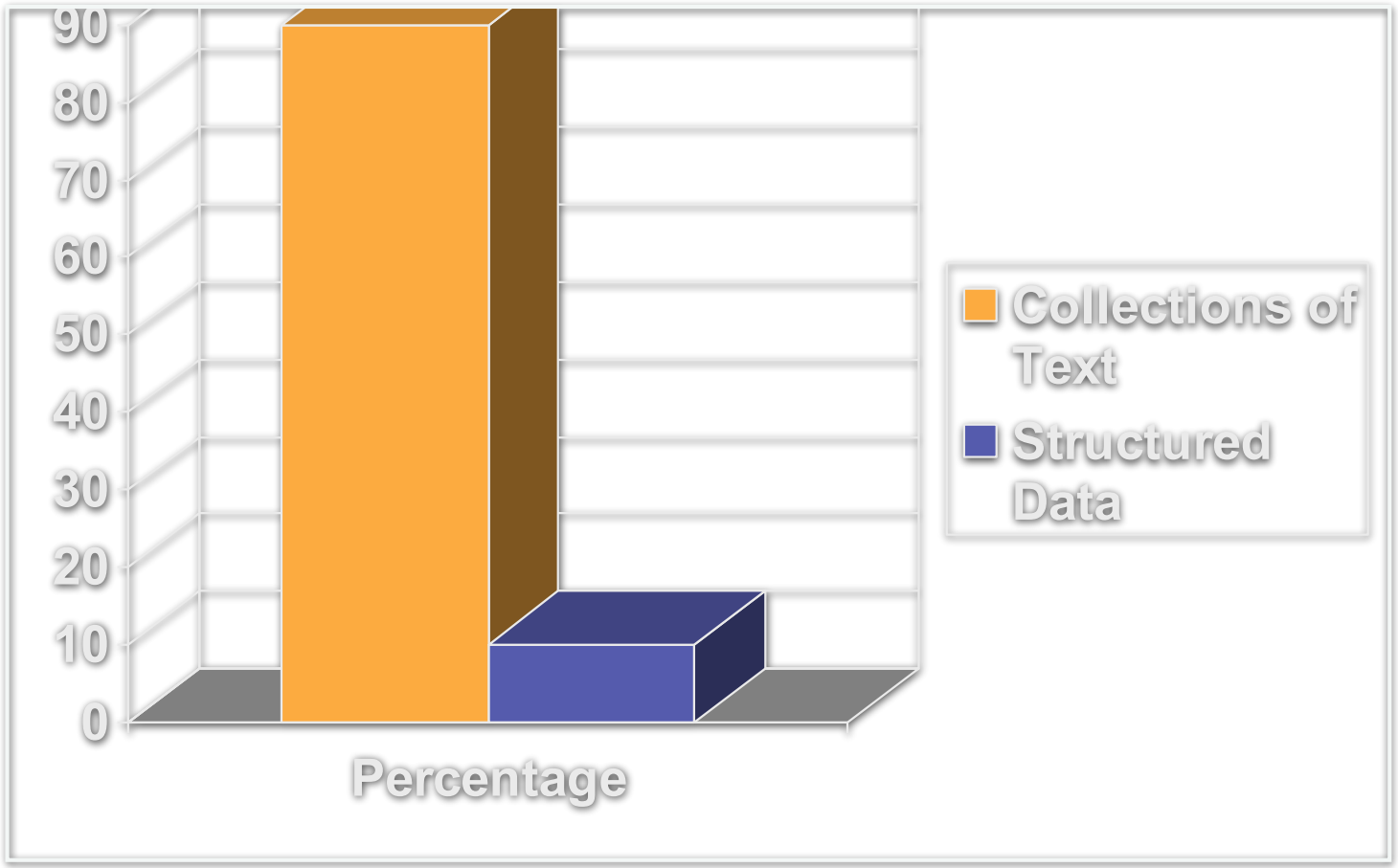
- A document *d*
- A set of classes C= {C1, C2 … CM}

OUTPUT

- Predicted class of a given document.

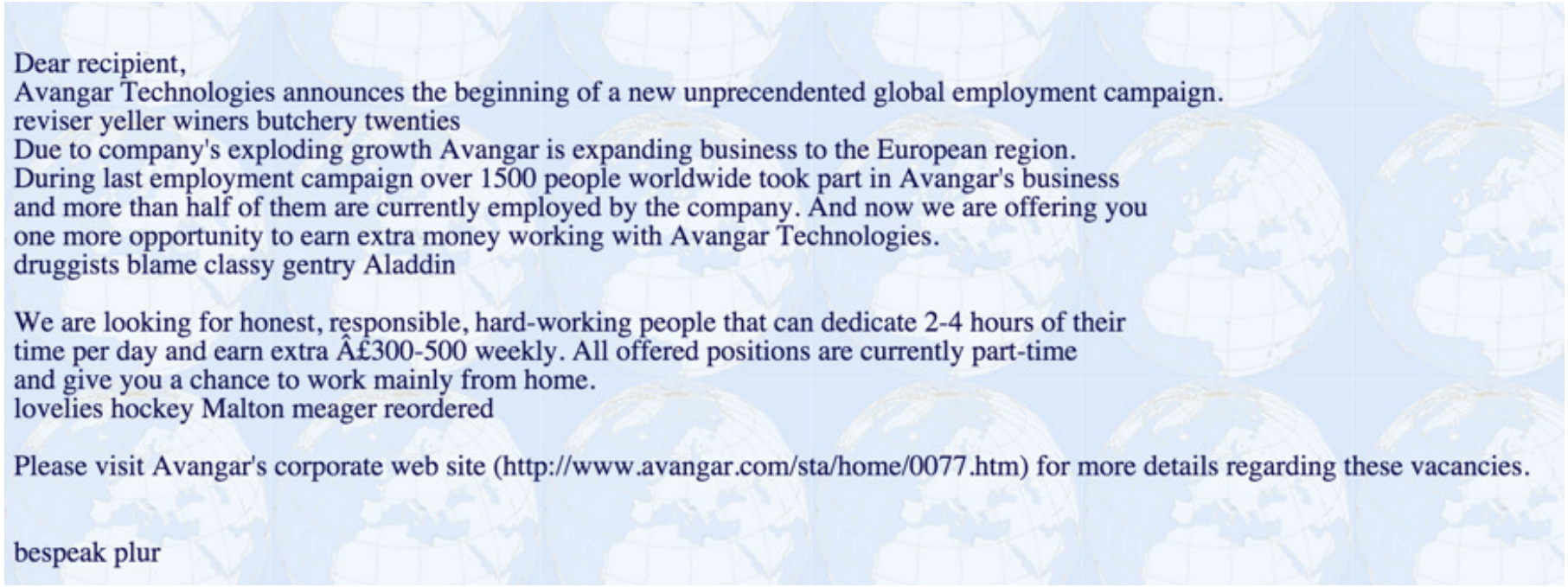
Text Classification Problem

Most of the text data is not structured.



Spam Classification with Naive Bayes

A spam example:



Dear recipient,
Avangar Technologies announces the beginning of a new unprecendented global employment campaign.
reviser yeller winers butchery twenties
Due to company's exploding growth Avangar is expanding business to the European region.
During last employment campaign over 1500 people worldwide took part in Avangar's business
and more than half of them are currently employed by the company. And now we are offering you
one more opportunity to earn extra money working with Avangar Technologies.
druggists blame classy gentry Aladdin

We are looking for honest, responsible, hard-working people that can dedicate 2-4 hours of their
time per day and earn extra Â£300-500 weekly. All offered positions are currently part-time
and give you a chance to work mainly from home.
lovelies hockey Malton meager reordered

Please visit Avangar's corporate web site (http://www.avangar.com/sta/home/0077.htm) for more details regarding these vacancies.

bespeak plur

Spam Classification with Naive Bayes

A non-spam example:

Dear Paul

It was very nice meeting you last week during the conference - I hope you're feeling better! I enjoyed both the conference and the introduction to Liverpool's nightlife.

Many thanks also for the suggestions you made for my presentation, and for the publicity material. These will be really useful - I'm sure the sale of the new encyclopaedias will increase as a result.

If you come to Delhi please let me know. It's a great city too, and I'd love to have the chance of being your guide as you were mine in Liverpool.

Best wishes
Mary
Sales Department

Spam Classification with Naive Bayes

- Step 1: Extraction of Features
  - Candidate features can be n-grams, stems.
  - Ignoring stop words (and, the etc.)
  - NLP libraries can identify (US, USA, states etc.)
  - Term frequency - inverse document frequency is a common method to identify important features.

$$\text{tf-idf}(t) = \text{tf}(t, d) \times \text{idf}(t)$$

Spam Classification with Naive Bayes

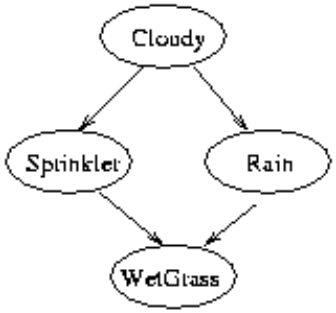
- Step 2: Design Experiment
  - k-fold cross validation etc.
- Step 3: Learn the Conditional Probabilities

$$P(Great|Spam)$$
$$P(Incredible|Spam)$$
$$\ldots$$

- Step 4: Evaluate the algorithm.

Introduction to Graphical Models

P(C=F)	P(C=T)
0.5	0.5



C	P(S=F)	P(S=T)
F	0.5	0.5
T	0.9	0.1

C	P(R=F)	P(R=T)
F	0.8	0.2
T	0.2	0.8

S	R	P(W=F)	P(W=T)
F	F	1.0	0.0
T	F	0.1	0.9
F	T	0.1	0.9
T	T	0.01	0.99

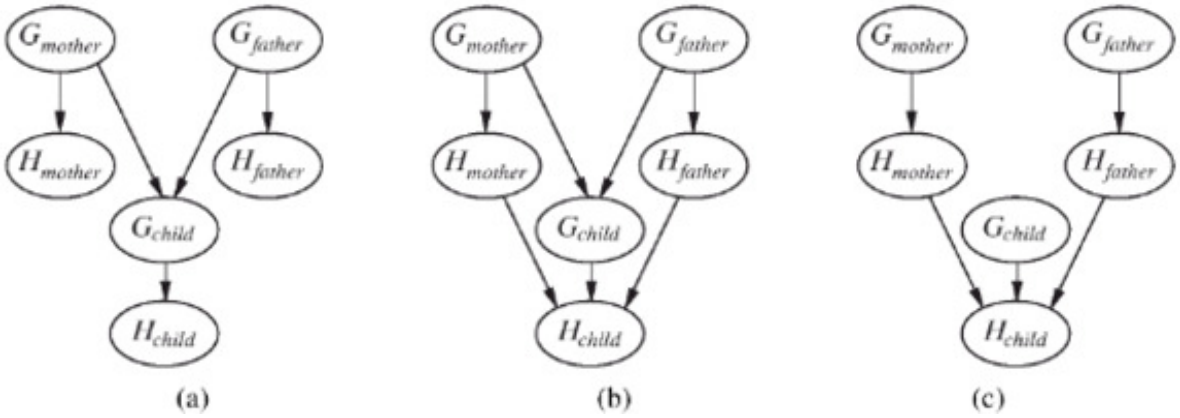
$$P(C, S, R, W) = P(C) * P(S|C) * P(R|C, S) * P(W|C, S, R)$$

$$\Pr(S = 1|W = 1) = \frac{\Pr(S = 1, W = 1)}{\Pr(W = 1)} = \frac{\sum_{c,r} \Pr(C = c, S = 1, R = r, W = 1)}{\Pr(W = 1)} = 0.2781/0.6471 = 0.430$$

Introduction to Graphical Models

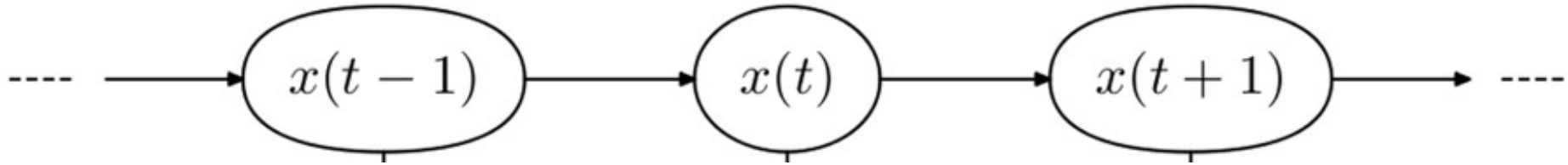
- Directed edges (arrows) represent dependence relationships. A—>B shows that A is dependent on B.

- Construction of graphical models changes the learning problem dramatically. (H is the left handedness of a person, G is the left handedness gene.)



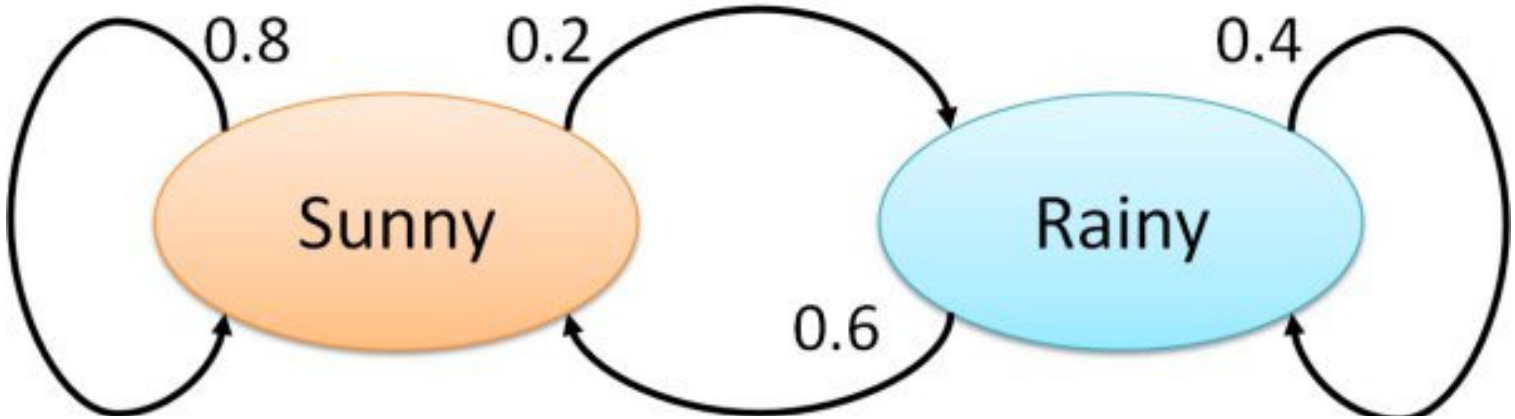
Introduction to Graphical Models

- Markov Chain: The probability of the new event is only dependent on the probability of the last event.



$$P(X_t|X_{t-1}) = \frac{P(X_{t-1}|X_t) * X(t)}{P(X_{t-1})}$$

$$P(X_t|(X_{t-1}, X_{t-2}...)) = P(X_t|X_{t-1}) * P(X_{t-2})...$$



Introduction to Graphical Models

- Markov Chains Converge over time.

$$\mathbf{P}^1 = \begin{array}{c} \\ \text{Rain} \\ \text{Nice} \\ \text{Snow} \end{array} \begin{array}{c} \begin{array}{ccc} \text{Rain} & \text{Nice} & \text{Snow} \end{array} \\ \begin{pmatrix} .500 & .250 & .250 \\ .500 & .000 & .500 \\ .250 & .250 & .500 \end{pmatrix} \end{array}$$

$$\mathbf{P}^2 = \begin{array}{c} \\ \text{Rain} \\ \text{Nice} \\ \text{Snow} \end{array} \begin{array}{c} \begin{array}{ccc} \text{Rain} & \text{Nice} & \text{Snow} \end{array} \\ \begin{pmatrix} .438 & .188 & .375 \\ .375 & .250 & .375 \\ .375 & .188 & .438 \end{pmatrix} \end{array}$$

$$\mathbf{P}^3 = \begin{array}{c} \\ \text{Rain} \\ \text{Nice} \\ \text{Snow} \end{array} \begin{array}{c} \begin{array}{ccc} \text{Rain} & \text{Nice} & \text{Snow} \end{array} \\ \begin{pmatrix} .406 & .203 & .391 \\ .406 & .188 & .406 \\ .391 & .203 & .406 \end{pmatrix} \end{array}$$

$$\mathbf{P}^4 = \begin{array}{c} \\ \text{Rain} \\ \text{Nice} \\ \text{Snow} \end{array} \begin{array}{c} \begin{array}{ccc} \text{Rain} & \text{Nice} & \text{Snow} \end{array} \\ \begin{pmatrix} .402 & .199 & .398 \\ .398 & .203 & .398 \\ .398 & .199 & .402 \end{pmatrix} \end{array}$$

$$\mathbf{P}^5 = \begin{array}{c} \\ \text{Rain} \\ \text{Nice} \\ \text{Snow} \end{array} \begin{array}{c} \begin{array}{ccc} \text{Rain} & \text{Nice} & \text{Snow} \end{array} \\ \begin{pmatrix} .400 & .200 & .399 \\ .400 & .199 & .400 \\ .399 & .200 & .400 \end{pmatrix} \end{array}$$

$$\mathbf{P}^6 = \begin{array}{c} \\ \text{Rain} \\ \text{Nice} \\ \text{Snow} \end{array} \begin{array}{c} \begin{array}{ccc} \text{Rain} & \text{Nice} & \text{Snow} \end{array} \\ \begin{pmatrix} .400 & .200 & .400 \\ .400 & .200 & .400 \\ .400 & .200 & .400 \end{pmatrix} \end{array}$$

References

- Spam dataset: https://archive.ics.uci.edu/ml/datasets/Spambase
- Comparison of machine learning methods for spam classification: http://airccse.org/journal/jcsit/0211ijcsit12.pdf
- Introduction to text mining: http://www.cs.waikato.ac.nz/~ihw/papers/04-IHW-Textmining.pdf
- A commercial text mining library: http://www.opencalais.com/

Week 8 Application Part

Preparation

```
library("tm") # text mining package for R
library("RCurl") # get text from the web
library("klaR") # implementation of naive bayes in R
library("caret") # experimental design
```

Importing and Analyze Text Data

```
data("crude")
# strip white space
crude <- tm_map(crude, stripWhitespace)
# remove stop words
crude <- tm_map(crude, removeWords, stopwords("english"))
# stem words
crude <- tm_map(crude, stemDocument)
```

Finding Frequent terms

```
dtm <- DocumentTermMatrix(crude)
# at least 10 occurrence
findFreqTerms(dtm,10)
```

```
 [1] "accord"   "barrel"   "bpd"      "crude"    "dlrs"     "fut
```

Finding associations

Finding associations

```
findAssocs(dtm, "opec", 0.8)
```

```
         opec
oil      0.88
emerg    0.87
15.8     0.86
analyst  0.85
meet     0.85
said     0.84
want     0.82
abil     0.80
```

Finding tf

```
dtm <- DocumentTermMatrix(crude,
        control = list(weighting = weightTfIdf))
# three important terms for the first document
sort(as.matrix(dtm)[1,], decreasing=T)[1:3]
```

```
   diamond     reduct      today
0.13941704 0.13941704 0.08828921
```

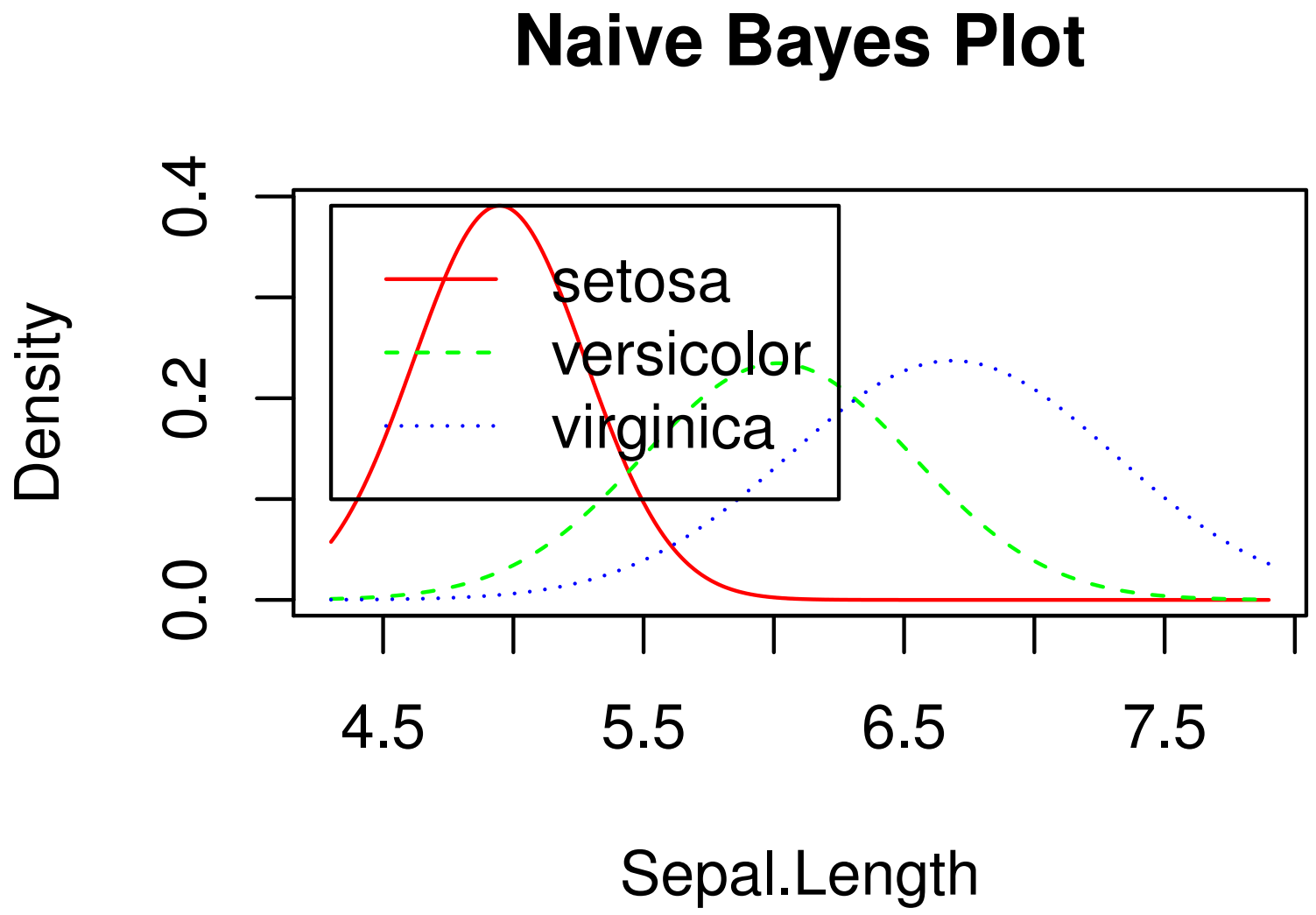
Naive Bayes Implementation

```
rn_train <- sample(nrow(iris),
                   floor(nrow(iris)*0.7))
train <- iris[rn_train,]
test <- iris[-rn_train,]
model <- NaiveBayes(Species~., data=train)
```

Naive Bayes Implementation

Show posterior probability densities

```
plot(model)
```



Naive Bayes Plot

Naive Bayes Implementation

```
# make predictions
predictions <- predict(model, test)
# summarize results
confusionMatrix(predictions$class, predictions$class)
```

```
Confusion Matrix and Statistics

            Reference
Prediction   setosa versicolor virginica
  setosa         16          0         0
  versicolor      0         19         0
  virginica       0          0        10

Overall Statistics

               Accuracy : 1
                 95% CI : (0.9213, 1)
    No Information Rate : 0.4222
```

Lab Preparation

```
library("RWeka") # rweka (embedded Weka software)

diabetes <- read.arff(system.file("arff", "diabetes.arff",
                                  package = "RWeka"))
data("acq")
```

Lab Problems:

1. Process acq text by stemming and create a document term matrix.
2. Show 3 important terms for **each** document using tf .
3. Run Naive Bayes using diabetes dataset.
4. Show posterior probability densities of diabetes.
5. (optional) Find and use the naive bayes functionality of weka for the same dataset.