Machine Learning

CISC 5800

Dr Daniel Leeds

What is machine learning

- Finding patterns in data
- Adapting program behavior

- Advertise a customer's favorite products
- Search the web to find pictures of dogs
- Change radio channel when user says "change channel"

Advertise a customer's favorite products

This summer, I had two meetings, one in Portland and one in Baltimore

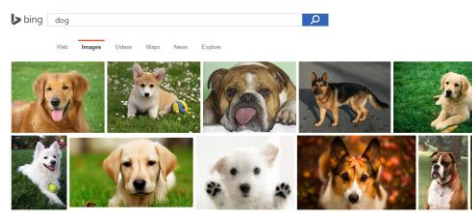
Today I get an e-mail from Priceline:

Search the web to find pictures of dogs

Filenames:
- Dog.jpg
- Puppy.bmp

Caption text

Pixel patterns



Change radio channel when user says "change channel"

- Distinguish user's voice from music
- Understand what user has said

What's covered in this class

- Theory: describing patterns in data
  - Probability
  - Linear algebra
  - Calculus/optimization

- Implementation: programming to find and react to patterns in data
  - Matlab
  - Data sets of text, speech, pictures, user actions, neural data…

Outline of topics

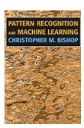
- Groundwork: probability, slopes, and programming
- Classification overview: Training, testing, and overfitting
- Discriminative and generative methods: Regression vs Naïve Bayes
- Classifier theory: Separability, information criteria
- Support vector machines: Slack variables and kernels
- Expectation-Maximization: Gaussian mixture models
- Dimensionality reduction: Principle Component Analysis
- Graphical models: Bayes nets, Hidden Markov model

What you need to do in this class

- Class attendance
- Assignments: homeworks (4) and final project
- Exams: midterm and final

Resources

- Office hours: Wednesday 3-4pm and by appointment
- Course web site: http://storm.cis.fordham.edu/leeds/cisc5800
- Fellow students
- Textbooks/online notes



- Matlab

Outline of topics

- Groundwork: probability, slopes, and programming
- Classification overview: Training, testing, and overfitting
- Discriminative and generative methods: Regression vs Naïve Bayes
- Classifier theory: Separability, information criteria
- Support vector machines: Slack variables and kernels
- Expectation-Maximization: Gaussian mixture models
- Dimensionality reduction: Principle Component Analysis
- Graphical models: Bayes nets, Hidden Markov model

Probability

What is the probability that a child likes chocolate?

The “frequentist” approach:
- Ask 100 children
- Count who likes chocolate
- Divide by number of children asked

Name	Chocolate?
Sarah	Yes
Melissa	Yes
Darren	No
Stacy	Yes
Brian	No

P(“child likes chocolate”) = $\frac{85}{100} = 0.85$

In short: P(C)=0.85 C=“child likes chocolate”

General probability properties

P(A) means “Probability that statement A is true”

- $0 \leq \text{Prob}(A) \leq 1$
- Prob(True)=1
- Prob(False)=0

Random variables

A variable can take on a value from a given set of values:
- {True, False}
- {Cat, Dog, Horse, Cow}
- {0,1,2,3,4,5,6,7}

A random variable holds each value with a given probability
To start, let us consider a **binary variable**
- P(LikesChocolate) = P(LikesChocolate=True) = 0.85

Complements

C="child likes chocolate"

P("child likes chocolate") = $\frac{85}{100} = 0.85$

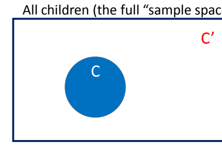
What is the probability that a child DOES NOT like chocolate?

Complement: C' = "child doesn't like chocolate"

P(C') =

In general: P(A') =

All children (the full "sample space")



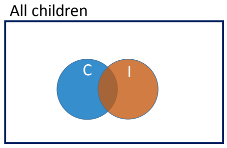
Addition rule

Prob(A or B) = ???

Name	Chocolate?	Ice cream?
Sarah	Yes	No
Melissa	Yes	Yes
Darren	No	No
Stacy	Yes	Yes
Brian	No	Yes

C="child likes chocolate"
I="child likes ice cream"

All children



Joint and marginal probabilities

Corrected slide

Across 100 children:
- 55 like chocolate AND ice cream
- 30 like chocolate but not ice cream
- 5 like ice cream but not chocolate
- 10 don't like chocolate nor ice cream

Prob(I) =
Prob(C) =
Prob(I,C)

Conditional probability

Corrected slide

Across 100 children:
- 55 like chocolate AND ice cream P(C,I)
- 30 like chocolate but not ice cream P(C,I')
- 5 like ice cream but not chocolate P(C',I)
- 10 don't like chocolate nor ice cream P(C',I')

Also, **Multiplication Rule:**

P(A,B) = P(A|B) P(B)

P(A,B):Probability A and B are both true

- Prob(C|I) : Probability child likes chocolate given s/he likes ice cream

$$P(C|I) = \frac{P(C,I)}{P(I)} = \frac{P(C,I)}{P(C,I)+P(C',I)}$$

Independence

If the truth value of B does not affect the truth value of A:
- P(A|B) = P(A)

Equivalently
- P(A,B) = P(A) P(B)

Multi-valued random variables

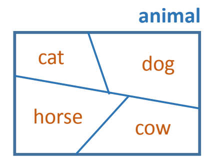
A random variable can hold more than two values, each with a given probability

- P(Animal=Cat)=0.5
- P(Animal=Dog)=0.3
- P(Animal=Horse)=0.1
- P(Animal=Cow)=0.1

Probability rules: multi-valued variables

For a given variable A:

- P($A = a_i$ and $A = a_j$) = 0 if $i \neq j$
- $\sum_i P(A = a_i) = 1$
- $P(A = a_i) = \sum_j P(A = a_i, B = b_j)$



Bayes rule

$$\mathrm{P(A|B)} = \frac{P(B|A)P(A)}{P(B)}$$

Terminology:

- P(A|B) is the “posterior probability”
- P(B|A) is the “likelihood”
- P(A) is the “prior probability”

We will spend (much) more time with Bayes rule in following lectures

Continuous random variables

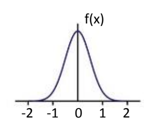
A random variable can take on a continuous range of values

- From 0 to 1
- From 0 to ∞
- From $-\infty$ to ∞

Probability expressed through a
“probability density function” **f(x)**

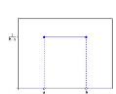
$P(A \epsilon [a,b]) = \int_a^b f(x)dx$

“Probability A has value between i and j is area under the curve of f between i and j

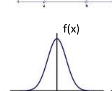


Common probability distributions

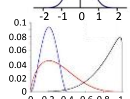
- Uniform: $f_{uniform}(x) = \begin{cases} \frac{1}{b-a} & if\ a \leq x \leq b \\ 0 & otherwise \end{cases}$



- Gaussian: $f_{gauss}(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$

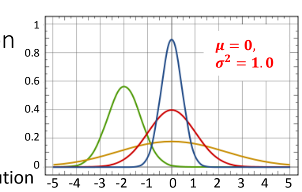


- Beta: $f_{beta}(x) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{\mathrm{B}(\alpha,\beta)}$



The Gaussian function

$f_{gauss}(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$



- Mean μ – center of distribution
- Standard deviation σ – width of distribution

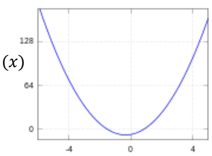
- Which color is μ=-2, σ^2=0.5? Which color is μ=0, σ^2=0.2?

- $N(\mu_1, \sigma_1^2) + N(\mu_2, \sigma_2^2) = N(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2)$

Calculus: finding the slope of a function

What is the minimum value of: $f(x)=x^2-5x+6$

Find value of x where slope is 0



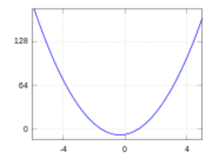
General rules: slope of f(x): $\frac{d}{dx}f(x) = f'(x)$

- $\frac{d}{dx}x^a = ax^{a-1}$
- $\frac{d}{dx}kf(x) = kf'(x)$
- $\frac{d}{dx}[f(x) + g(x)] = f'(x) + g'(x)$

25

Calculus: finding the slope of a function

What is the minimum value of: $f(x)=x^2-5x+6$



- f'(x)=
- What is the slope at x=5?
- What is the slope at x=-5?

- What value of x gives slope of 0?

26

More on derivatives: $\frac{d}{dx}f(x) = f'(x)$

- $\frac{d}{dx}f(w) = 0$ -- w is not related to x, so derivative is 0
- $\frac{d}{dx}\left(f(g(x))\right)$=$g'(x) \cdot f'(g(x))$
- $\frac{d}{dx}\log x = \frac{1}{x}$
- $\frac{d}{dx}e^x = e^x$

27

Programming in Matlab: Data types

- Numbers: -8.5, 0, 94
- Characters: 'j', '#', 'K' - always surrounded by single quotes
- Groups of numbers/characters – placed in between [ ]
  - [5 10 12; 3 -4 12; -6 0 0] - spaces/commas separate columns, semi-colons separate rows
  - 'hi robot', ['h' 'i' ' ' 'robot'] - a collection of characters can be grouped inside a set of single quotes

28

Matrix indexing

- Start counting at 1

matrix1=[4 8 12; 6 3 0; -2 -7 -12];

matrix1(2,3) -> 0

- Last row/column can also be designated by keyword "end"

matrix1(1,end) -> 12

- Colon indicates counting up by increment
  - [2:10] -> [2 3 4 5 6 7 8 9 10]
  - [3:4:19] -> [3 7 11 15 19]

matrix1(2,1:3) -> [6 3 0]

29

Vector/matrix functions

vec1=[9, 3, 5, 7]; matrix2=[4.5 -3.2; 2.2 0; -4.4 -3];

- mean mean(vec1) -> 6
- min min(vec1) -> 3
- max max(vec1) -> ?
- std std(vec1) -> 2.58
- length length(vec1) -> ?
- size size(matrix2) -> [3 2];

30

Extra syntax notes

- Semicolons suppress output of computations:

```
> a=4+5
a =
   9
> b=6+7;
>
```

- % starts a comment for the line (like // in C++)
- .* , ./ , .^ performs element-wise arithmetic

```
>c=[2 3 4]./[2 1 2]
>c =
    [1   3   1]
>
```


Variables

- who, whos – list variables in environment
- Comparisons:
  - Like C++: ==, <, >, <=, >=
  - Not like C++: not ~, and &, or |
- Conditions:
  - if(…),   end;
- Loops:
  - while(…),   end;
  - for x=a:b,   end;

Data: .mat files

- **save** filename variableNames

- **load** filename

- Confirm correct directories:
  - pwd – show directory (**p**rint **w**orking **d**irectory)
  - cd – **c**hange **d**irectory
  - ls – **l**i**s**t files in directory

Define new functions: .m files

- Begin file with function header:

function output = function_name(input)

statement1;
statement2;
⋮

- Can allow multiple inputs/outputs

function [output1, output2] = function_name(input1, input2, input3)

Linear algebra: data features

- Vector – list of numbers: each number describes a data **feature**

- Matrix – list of lists of numbers: features for each data point

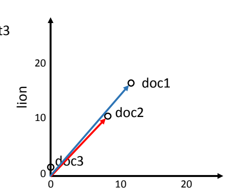
	Document 1	Document 2	Document 3
Wolf	12	8	0
Lion	16	10	2
Monkey	14	11	1
Broker	0	1	14
Analyst	1	0	10
Dividend	1	1	12
⋮	⋮	⋮	⋮

# of word occurrences

Feature space

- Each data feature defines a dimension in space

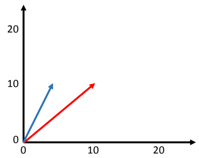
	Document1	Document2	Document3
Wolf	12	8	0
Lion	16	10	2
Monkey	14	11	1
Broker	0	1	14
Analyst	1	0	10
Dividend	1	1	12
⋮	⋮	⋮	⋮



The dot product

The dot product compares two vectors:

- $\boldsymbol{a} = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}, \boldsymbol{b} = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}$ $\quad \boldsymbol{a} \cdot \boldsymbol{b} = \sum_{i=1}^{n} a_i b_i \;= \boldsymbol{a}^T \boldsymbol{b}$



$$\begin{bmatrix} \mathbf{5} \\ \mathbf{10} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{10} \\ \mathbf{10} \end{bmatrix} = \mathbf{5 \times 10 + 10 \times 10}$$

$$= \mathbf{50 + 100 = 150}$$

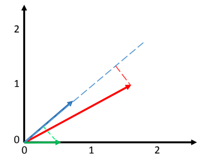
37

The dot product, continued $\boldsymbol{a} \cdot \boldsymbol{b} = \sum_{i=1}^{n} a_i b_i$

Magnitude of a vector is the sum of the squares of the elements

$$|\boldsymbol{a}| = \sqrt{\sum_i a_i^2}$$

If $\boldsymbol{a}$ has unit magnitude, $\boldsymbol{a} \cdot \boldsymbol{b}$ is the "projection" of $\boldsymbol{b}$ onto $\boldsymbol{a}$



$$\begin{bmatrix} 0.71 \\ 0.71 \end{bmatrix} \cdot \begin{bmatrix} 1.5 \\ 1 \end{bmatrix} = .71 \times 1.5 + .71 \times 1$$

$$\approx 1.07 + .71 = 1.78$$

$$\begin{bmatrix} 0.71 \\ 0.71 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0.5 \end{bmatrix} = .71 \times 0 + .71 \times 0.5$$

$$\approx 0 + .35 = 0.35$$

38

Multiplication

"scalar" means single numeric value (not a multi-element matrix)

- Scalar × matrix: Multiply each element of the matrix by the scalar value

$$c \begin{bmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{bmatrix} = \begin{bmatrix} c\,a_{11} & \cdots & c\,a_{1m} \\ \vdots & \ddots & \vdots \\ c\,a_{n1} & \cdots & c\,a_{nm} \end{bmatrix}$$

- Matrix × column vector: dot product of each row with vector

$$\begin{bmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{bmatrix} \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix} =$$

$$\begin{bmatrix} -\boldsymbol{a}_1- \\ \vdots \\ -\boldsymbol{a}_n- \end{bmatrix} \qquad \boldsymbol{b}$$

39

Multiplication

- Matrix × matrix: Compute dot product of each left row and right column

$$\begin{bmatrix} -\boldsymbol{a}_1- \\ \vdots \\ -\boldsymbol{a}_n- \end{bmatrix} \begin{bmatrix} | & & | \\ \boldsymbol{b}_1 & \cdots & \boldsymbol{b}_m \\ | & & | \end{bmatrix} = \begin{bmatrix} \boldsymbol{a}_1 \cdot \boldsymbol{b}_1 & \cdots & \boldsymbol{a}_1 \cdot \boldsymbol{b}_m \\ \vdots & \ddots & \vdots \\ \boldsymbol{a}_n \cdot \boldsymbol{b}_1 & \cdots & \boldsymbol{a}_n \cdot \boldsymbol{b}_m \end{bmatrix}$$

NB: Matrix dimensions need to be compatible for valid multiplication – number of rows of left matrix (**A**) = number of columns of right matrix (**B**)

40