2. Statistics

Alex Smola
Yahoo! Research and ANU

http://alex.smola.org/teaching/berkeley2012
Stat 260 SP 12
2. Statistics

Essential tools for data analysis
Statistics

- Probabilities
  - Bayes rule, Dependence, independence, conditional probabilities
  - Priors, Naive Bayes classifier
- Tail bounds
  - Chernoff, Hoeffding, Chebyshev, Gaussian
  - A/B testing
- Kernel density estimation
  - Parzen windows, Nearest neighbors, Watson-Nadaraya estimator
- Exponential families
  - Gaussian, multinomial, Poisson
  - Conjugate distributions and smoothing, integrating out
JELLY BEANS CAUSE ACNE!

SCIENTISTS! INVESTIGATE!

But we're playing Minecraft. ...Fine.

WE FOUND NO LINK BETWEEN JELLY BEANS AND ACNE (p > 0.05).

THAT SETTLES THAT.

I HEAR IT'S ONLY A CERTAIN COLOR THAT CAUSES IT.

SCIENTISTS!

But Minecraft!
WE FOUND NO LINK BETWEEN PURPLE JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN BROWN JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN PINK JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN BLUE JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN TEAL JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN SALMON JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN RED JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN TURQUOISE JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN MAGENTA JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN YELLOW JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN GREY JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN TAN JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN CYAN JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN GREEN JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN MAUVE JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN BEIGE JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN LILAC JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN BLACK JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN PEACH JELLY BEANS AND ACNE (P > 0.05).

WE FOUND NO LINK BETWEEN ORANGE JELLY BEANS AND ACNE (P > 0.05).

WHOA!
GREEN JELLY BEANS LINKED TO ACNE!

95% CONFIDENCE

ONLY 5% CHANCE OF COINCIDENCE!
2.1 Probabilities
Probability

- Space of events $X$
  - server working; slow response; server broken
- income of the user (e.g. $95,000)
- query text for search (e.g. “statistics tutorial”)
- Probability axioms (Kolmogorov)
  \[
  \Pr(X) \in [0, 1], \quad \Pr(\mathcal{X}) = 1 \\
  \Pr(\bigcup_i X_i) = \sum_i \Pr(X_i) \text{ if } X_i \cap X_j = \emptyset
  \]
- Example queries
  - $P(\text{server working}) = 0.999$
  - $P(90,000 < \text{income} < 100,000) = 0.1$
All events
Venn Diagram

All events

$X \cap X'$
$Pr(X \cup X') = Pr(X) + Pr(X') - Pr(X \cap X')$
(In)dependence

- Independence \( \Pr(x, y) = \Pr(x) \cdot \Pr(y) \)
- Login behavior of two users (approximately)
- Disk crash in different colos (approximately)
(In)dependence

- **Independence** \( \Pr(x, y) = \Pr(x) \cdot \Pr(y) \)
- Login behavior of two users (approximately)
- Disk crash in different colos (approximately)

- **Dependent events**
  - Emails \( \Pr(x, y) \neq \Pr(x) \cdot \Pr(y) \)
  - Queries
  - News stream / Buzz / Tweets
  - IM communication
  - Russian Roulette
(In)dependence

- Independence \[ \Pr(x, y) = \Pr(x) \cdot \Pr(y) \]
- Login behavior of two users (approximately)
- Disk crash in different colos (approximately)

- Dependent events
  - Emails \[ \Pr(x, y) \neq \Pr(x) \cdot \Pr(y) \]
  - Queries
  - News stream / Buzz / Tweets
  - IM communication
  - Russian Roulette

Everywhere!
A Graphical Model

\[ p(\text{spam, mail}) = p(\text{spam}) \ p(\text{mail} | \text{spam}) \]
Bayes Rule

- **Joint Probability**
  \[
  \Pr(X, Y) = \Pr(X|Y) \Pr(Y) = \Pr(Y|X) \Pr(X)
  \]

- **Bayes Rule**
  \[
  \Pr(X|Y) = \frac{\Pr(Y|X) \cdot \Pr(X)}{\Pr(Y)}
  \]

- **Hypothesis testing**
- **Reverse conditioning**
AIDS test (Bayes rule)

- Data
  - Approximately 0.1% are infected
  - Test detects all infections
  - Test reports positive for 1% healthy people
  - Probability of having AIDS if test is positive
AIDS test (Bayes rule)

• **Data**
  • Approximately **0.1%** are infected
  • Test detects **all** infections
  • Test reports positive for **1%** healthy people

• **Probability of having AIDS if test is positive**

\[
Pr(a = 1|t) = \frac{Pr(t|a = 1) \cdot Pr(a = 1)}{Pr(t)} = \frac{Pr(t|a = 1) \cdot Pr(a = 1)}{Pr(t|a = 1) \cdot Pr(a = 1) + Pr(t|a = 0) \cdot Pr(a = 0)}
\]

\[
= \frac{1 \cdot 0.001}{1 \cdot 0.001 + 0.01 \cdot 0.999} = 0.091
\]
Improving the diagnosis
Improving the diagnosis

- Use a follow-up test
- Test 2 reports positive for 90% infections
- Test 2 reports positive for 5% healthy people

\[
\frac{0.01 \cdot 0.05 \cdot 0.999}{1 \cdot 0.9 \cdot 0.001 + 0.01 \cdot 0.05 \cdot 0.999} = 0.357
\]
Improving the diagnosis

• Use a follow-up test
  • Test 2 reports positive for 90% infections
  • Test 2 reports positive for 5% healthy people

\[
\frac{0.01 \cdot 0.05 \cdot 0.999}{1 \cdot 0.9 \cdot 0.001 + 0.01 \cdot 0.05 \cdot 0.999} = 0.357
\]

• Why can’t we use Test 1 twice?
  Outcomes are not independent but tests 1 and 2 are conditionally independent
Improving the diagnosis

• Use a follow-up test
• Test 2 reports positive for 90% infections
• Test 2 reports positive for 5% healthy people

\[
\frac{0.01 \cdot 0.05 \cdot 0.999}{1 \cdot 0.9 \cdot 0.001 + 0.01 \cdot 0.05 \cdot 0.999} = 0.357
\]

• Why can’t we use Test 1 twice?
  Outcomes are not independent but tests 1 and 2 are conditionally independent

\[
p(t_1, t_2 | a) = p(t_1 | a) \cdot p(t_2 | a)
\]
Logarithms are good

- Floating point numbers
- Probabilities can be very small. In particular products of many probabilities. Underflow!
- Store data in mantissa, not exponent

\[ \pi = \log p \]

\[
\prod_i p_i \rightarrow \sum_i \pi_i \quad \sum_i p_i \rightarrow \max \pi + \log \sum_i \exp[\pi_i - \max \pi]
\]

- Known bug e.g. in Mahout Dirichlet clustering
Application: Naive Bayes
Naive Bayes Spam Filter
Naive Bayes Spam Filter

• **Key assumption**
  Words occur independently of each other given the label of the document

\[
p(w_1, \ldots, w_n | \text{spam}) = \prod_{i=1}^{n} p(w_i | \text{spam})
\]
Naive Bayes Spam Filter

- **Key assumption**
  Words occur independently of each other given the label of the document

\[
p(w_1, \ldots, w_n | \text{spam}) = \prod_{i=1}^{n} p(w_i | \text{spam})
\]

- **Spam classification via Bayes Rule**

\[
p(\text{spam} | w_1, \ldots, w_n) \propto p(\text{spam}) \prod_{i=1}^{n} p(w_i | \text{spam})
\]
Naive Bayes Spam Filter

• **Key assumption**
Words occur independently of each other given the label of the document

\[ p(w_1, \ldots, w_n | \text{spam}) = \prod_{i=1}^{n} p(w_i | \text{spam}) \]

• **Spam classification via Bayes Rule**

\[ p(\text{spam} | w_1, \ldots, w_n) \propto p(\text{spam}) \prod_{i=1}^{n} p(w_i | \text{spam}) \]

• **Parameter estimation**
Compute spam probability and word distributions for spam and ham
Equally likely phrases

- Get rich quick. Buy UCB stock.
- Buy Viagra. Make your UCB experience last longer.
- You deserve a PhD from UCB. We recognize your expertise.
Equally likely phrases

• Get rich quick. Buy UCB stock.
• Buy Viagra. Make your UCB experience last longer.
• You deserve a PhD from UCB. We recognize your expertise.

• Make your rich UCB PhD experience last longer.
A Graphical Model

spam

W_1

W_2

\ldots

W_n
A Graphical Model

\[ p(w_1, \ldots, w_n | \text{spam}) = \prod_{i=1}^{n} p(w_i | \text{spam}) \]
A Graphical Model

\[
p(w_1, \ldots, w_n | \text{spam}) = \prod_{i=1}^{n} p(w_i | \text{spam})
\]
A Graphical Model

$p(w_1, \ldots, w_n | \text{spam}) = \prod_{i=1}^{n} p(w_i | \text{spam})$
Naive Bayes Spam Filter

• Data
  • Emails (headers, body, metadata)
  • Labels (spam/ham)
    assume that users actually label all mails

• Processing capability
  • Billions of e-mails
  • 1000s of servers

• Need to estimate $p(y), p(x_i | y)$
  • Compute distribution of $x_i$ for every $y$
  • Compute distribution of $y$
Delivered-To: alex.smola@gmail.com
Received: by 10.216.47.73 with SMTP id s51cs361171web;
  Tue, 3 Jan 2012 14:17:53 -0800 (PST)
Received: by 10.213.17.145 with SMTP id s17mr2519891eba.147.1325629071725;
  Tue, 03 Jan 2012 14:17:51 -0800 (PST)
Return-Path: <alex+caf_=alex.smola@gmail.com@smola.org>
Received: from mail-ey0-f175.google.com (mail-ey0-f175.google.com [209.85.215.175])
  by mx.google.com with ESMTP id n4si29264232eef.57.2012.01.03.14.17.51
  (version=TLSv1/SSLv3 cipher=OTHER);
  Tue, 03 Jan 2012 14:17:51 -0800 (PST)
Received-SPF: neutral (google.com: 209.85.215.175 is neither permitted nor denied by best
  guess record for domain of alex+caf_=alex.smola@gmail.com@smola.org) client-ip=209.85.215.175;
Authentication-Results: mx.google.com; spf=neutral (google.com: domain of alex+caf_=alex.smola@gmail.com@smola.org
designates 209.85.215.175 as permitted sender) client-ip=209.85.215.175;
Received: by eaal1 with SMTP id l1so15092746eaa.6
  for <alex.smola@gmail.com>; Tue, 03 Jan 2012 14:17:51 -0800 (PST)
Received: by 10.205.135.18 with SMTP id ie18mr5325064bkc.72.1325629071362;
  Tue, 03 Jan 2012 14:17:51 -0800 (PST)
X-Forwarded-To: alex.smola@gmail.com
X-Forwarded-For: alex@smola.org
Delivered-To: alex@smola.org
Received: by 10.204.65.198 with SMTP id k6cs206093bktl;
  Tue, 3 Jan 2012 14:17:50 -0800 (PST)
Received: by 10.52.88.179 with SMTP id bh19mr10729402vdb.38.1325629067895;
  Tue, 03 Jan 2012 14:17:48 -0800 (PST)
Return-Path: <althoff.tim@googlemail.com>
Received: from mail-vx0-f179.google.com (mail-vx0-f179.google.com [209.85.220.179])
  by mx.google.com with ESMTPS id dt4si1176704vdb.93.2012.01.03.14.17.48
  (version=TLSv1/SSLv3 cipher=OTHER);
  Tue, 03 Jan 2012 14:17:48 -0800 (PST)
Received-SPF: pass (google.com: domain of althoff.tim@googlemail.com
designates 209.85.220.179 as permitted sender) client-ip=209.85.220.179;
Received: by vcbf13 with SMTP id f13so11295098vcb.10
  for <alex@smola.org>; Tue, 03 Jan 2012 14:17:48 -0800 (PST)
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed;
d=googlemail.com; s=gamma;
h=mime-version:sender:date:x-google-sender-auth:message-id:subject
  :from:to:content-type;
b=WCbdZ5sXac25dpH02Xcy0D0dts993hKwsAVXgPrFh0w=;
Received: by 10.220.108.81 with SMTP id e17mr24104004vcp.67.1325629067787;
  Tue, 03 Jan 2012 14:17:47 -0800 (PST)
Sender: althoff.tim@gmail.com
Subject: CS 281B. Advanced Topics in Learning and Decision Making
From: Tim Althoff <althoff@eecs.berkeley.edu>
Date: Tue, 3 Jan 2012 14:17:47 -0800
MIME-Version: 1.0
Content-Type: multipart/alternative; boundary=f46d043c7af4b07e8d04b5a7113a

this is a gross simplification
Recall - Map Reduce

- 1000s of (faulty) machines
- Lots of jobs are mostly embarrassingly parallel (except for a sorting/transpose phase)
- Functional programming origins
  - Map(key,value) processes each (key,value) pair and outputs a new (key,value) pair
  - Reduce(key,value) reduces all instances with same key to aggregate

Example - extremely naive wordcount
- Map(docID, document) for each document emit many (wordID, count) pairs
- Reduce(wordID, count) sum over all counts for given wordID and emit (wordID, aggregate)

from Ramakrishnan, Sakrejda, Canon, DoE 2011
Recall - Map Reduce

- 1000s of (faulty) machines
- Lots of jobs are mostly embarrassingly parallel (except for a sorting/transpose phase)
- Functional programming origins
  - Map(key, value)
    processes each (key, value) pair and outputs a new (key, value) pair
  - Reduce(key, value)
    reduces all instances with same key to aggregate
- Example - extremely naive wordcount
  - Map(docID, document)
    for each document emit many (wordID, count) pairs
  - Reduce(wordID, count)
    sum over all counts for given wordID and emit (wordID, aggregate)
Naive Bayes Classifier

- Two classes (spam/ham)
- Binary features (e.g. presence of $$\$$, viagra)
- Simplistic Algorithm
  - Count occurrences of feature for spam/ham
  - Count number of spam/ham mails

\[
p(x_i = \text{TRUE}|y) = \frac{n(i, y)}{n(y)} \quad \text{and} \quad p(y) = \frac{n(y)}{n}
\]

\[
p(y|x) \propto \frac{n(y)}{n} \prod_{i:x_i=\text{TRUE}} \frac{n(i, y)}{n(y)} \prod_{i:x_i=\text{FALSE}} \frac{n(y) - n(i, y)}{n(y)}
\]
Naive NaiveBayes Classifier

\[ p(y|x) \propto \frac{n(y)}{n} \prod_{i:x_i=\text{TRUE}} \frac{n(i,y)}{n(y)} \prod_{i:x_i=\text{FALSE}} \frac{n(y) - n(i,y)}{n(y)} \]

What if \( n(i,y) = 0 \)?

What if \( n(i,y) = n(y) \)?
p(y|x) \propto \frac{n(y)}{n} \prod_{i:x_i=TRUE} \frac{n(i,y)}{n(y)} \prod_{i:x_i=FALSE} \frac{n(y) - n(i,y)}{n(y)}

what if n(i,y)=0?

what if n(i,y)=n(y)?
Simple Algorithm

- For each document \((x,y)\) do
  - Aggregate label counts given \(y\)
  - For each feature \(x_i\) in \(x\) do
    - Aggregate statistic for \((x_i, y)\) for each \(y\)
  - For \(y\) estimate distribution \(p(y)\)
  - For each \((x_i, y)\) pair do
    Estimate distribution \(p(x_i | y)\), e.g. Parzen Windows, Exponential family (Gauss, Laplace, Poisson, ...), Mixture

- Given new instance compute

\[
p(y|x) \propto p(y) \prod_j p(x_j | y)
\]
Simple Algorithm

- For each document \((x,y)\) do
  - Aggregate label counts given \(y\) pass over all data
  - For each feature \(x_i\) in \(x\) do
    - Aggregate statistic for \((x_i,y)\) for each \(y\)
  - For \(y\) estimate distribution \(p(y)\)
  - For each \((x_i,y)\) pair do
    Estimate distribution \(p(x_i | y)\), e.g. Parzen Windows, Exponential family (Gauss, Laplace, Poisson, ...), Mixture
- Given new instance compute
  \[
p(y|x) \propto p(y) \prod_j p(x_j | y)
  \]
MapReduce Algorithm

- Map(document (x, y))
  - For each mapper for each feature $x_i$ in x do
    - Aggregate statistic for $(x_i, y)$ for each y
    - Send statistics (key = $(x_i, y)$, value = counts) to reducer

- Reduce($x_i$, y)
  - Aggregate over all messages from mappers
  - Estimate distribution $p(x_i | y)$, e.g. Parzen Windows, Exponential family (Gauss, Laplace, Poisson, ...), Mixture
  - Send coordinate-wise model to global storage

- Given new instance compute

$$p(y | x) \propto p(y) \prod_{j} p(x_j | y)$$
MapReduce Algorithm

- Map(document (x,y))
  - For each mapper for each feature \( x_i \) in \( x \) do
    - Aggregate statistic for \( (x_i, y) \) for each \( y \)
    - Send statistics (key = \( (x_i, y) \), value = counts) to reducer

- Reduce\( (x_i, y) \)
  - Aggregate over all messages from mappers
  - Estimate distribution \( p(x_i | y) \), e.g. Parzen Windows, Exponential family (Gauss, Laplace, Poisson, ...), Mixture
  - Send coordinate-wise model to global storage

- Given new instance compute

\[
p(y|x) \propto p(y) \prod_j p(x_j | y)
\]
Estimating Probabilities
Binomial Distribution

• Two outcomes (head, tail); (0,1)
• Data likelihood
  \[ p(X; \pi) = \pi^{n_1} (1 - \pi)^{n_0} \]
• Maximum Likelihood Estimation
  • Constrained optimization problem
  • Incorporate constraint via
  • Taking derivatives yields
  \[ \theta = \log \frac{n_1}{n_0 + n_1} \iff p(x = 1) = \frac{n_1}{n_0 + n_1} \]
... in detail ...

\[ p(X; \theta) = \prod_{i=1}^{n} p(x_i; \theta) = \prod_{i=1}^{n} \frac{e^{\theta x_i}}{1 + e^{\theta}} \]

\[ \implies \log p(X; \theta) = \theta \sum_{i=1}^{n} x_i - n \log [1 + e^{\theta}] \]

\[ \implies \partial_{\theta} \log p(X; \theta) = \sum_{i=1}^{n} x_i - n \frac{e^{\theta}}{1 + e^{\theta}} \]

\[ \iff \frac{1}{n} \sum_{i=1}^{n} x_i = \frac{e^{\theta}}{1 + e^{\theta}} = p(x = 1) \]
\[ p(X; \theta) = \prod_{i=1}^{n} p(x_i; \theta) = \prod_{i=1}^{n} \frac{e^{\theta x_i}}{1 + e^{\theta}} \]

\[ \implies \log p(X; \theta) = \theta \sum_{i=1}^{n} x_i - n \log [1 + e^{\theta}] \]

\[ \implies \partial_\theta \log p(X; \theta) = \sum_{i=1}^{n} x_i - n \frac{e^{\theta}}{1 + e^{\theta}} \]

\[ \frac{1}{n} \sum_{i=1}^{n} x_i = \frac{e^{\theta}}{1 + e^{\theta}} = p(x = 1) \]

empirical probability of \( x=1 \)
Discrete Distribution

• n outcomes (e.g. USA, Canada, India, UK, NZ)
• Data likelihood

\[ p(X; \pi) = \prod_i \pi_i^{n_i} \]

• Maximum Likelihood Estimation

• Constrained optimization problem ... or ...
• Incorporate constraint via

\[ p(x; \theta) = \frac{\exp \theta_{x}}{\sum_{x'} \exp \theta_{x'}} \]

• Taking derivatives yields

\[ \theta_i = \log \frac{n_i}{\sum_j n_j} \iff p(x = i) = \frac{n_i}{\sum_j n_j} \]
Tossing a Dice

12

24

60

120
Tossing a Dice

12

24

60

120
Key Questions

- Do empirical averages converge?
- Probabilities
- Means / moments
- Rate of convergence and limit distribution
- Worst case guarantees
- Using prior knowledge

... drug testing, semiconductor fabs
... computational advertising
... user interface design...
2.2 Tail Bounds
Expectations

- Random variable \( x \) with probability measure

- Expected value of \( f(x) \)

\[
E[f(x)] = \int f(x) \, dp(x)
\]

- Special case - discrete probability mass

\[
Pr \{x = c\} = E[\{x = c\}] = \int \{x = c\} \, dp(x)
\]

(same trick works for intervals)

- Draw \( x_i \) identically and independently from \( p \)

- Empirical average

\[
E_{\text{emp}}[f(x)] = \frac{1}{n} \sum_{i=1}^{n} f(x_i) \quad \text{and} \quad Pr_{\text{emp}} \{x = c\} = \frac{1}{n} \sum_{i=1}^{n} \{x_i = c\}
\]
Deviations

• Gambler rolls dice 100 times

\[ \hat{P}(X = 6) = \frac{1}{n} \sum_{i=1}^{n} \{x_i = 6\} \]

• ‘6’ only occurs 11 times. Fair number is 16.7

**IS THE DICE TAINTED?**

• Probability of seeing ‘6’ at most 11 times

\[ \Pr(X \leq 11) = \sum_{i=0}^{11} p(i) = \sum_{i=0}^{11} \binom{100}{i} \left( \frac{1}{6} \right)^i \left( \frac{5}{6} \right)^{100-i} \approx 7.0\% \]

It’s probably OK … can we develop general theory?
Deviation

- Gambler rolls dice 100 times

\[ \hat{P}(X = 6) = \frac{1}{n} \sum_{i=1}^{n} \{x_i = 6\} \]

- ‘6’ only occurs 11 times. Fair number is 16.7

IS THE DICE TAINTED?

- Probability of seeing ‘6’ at most 11 times

\[ \Pr(X \leq 11) = \sum_{i=0}^{11} p(i) = \sum_{i=0}^{11} \left(\begin{array}{c}
100 \\
i
\end{array}\right) \left[\frac{1}{6}\right]^{i} \left[\frac{5}{6}\right]^{100-i} \approx 7.0\% \]

It’s probably OK … can we develop general theory?

ad campaign working
new page layout better
drug working
Empirical average for a dice

how quickly does it converge?
Law of Large Numbers

- Random variables $x_i$ with mean $\mu = \mathbb{E}[x_i]$
- Empirical average $\hat{\mu}_n := \frac{1}{n} \sum_{i=1}^{n} x_i$

**Weak Law of Large Numbers**

$$\lim_{n \to \infty} \Pr (|\hat{\mu}_n - \mu| \leq \epsilon) = 1 \text{ for any } \epsilon > 0$$

**Strong Law of Large Numbers**

$$\Pr \left( \lim_{n \to \infty} \hat{\mu}_n = \mu \right) = 1$$

this means convergence in probability
Empirical average for a dice

Upper and lower bounds are $\mu \pm \sqrt{\text{Var}(x)/n}$

This is an example of the central limit theorem
Central Limit Theorem

• Independent random variables $x_i$ with mean $\mu_i$ and standard deviation $\sigma_i$

• The random variable

$$z_n := \left[ n \sum_{i=1}^{n} \sigma_i^2 \right]^{-\frac{1}{2}} \left[ n \sum_{i=1}^{n} x_i - n \mu_i \right]$$

converges to a Normal Distribution $\mathcal{N}(0, 1)$
Central Limit Theorem

• Independent random variables $x_i$ with mean $\mu_i$ and standard deviation $\sigma_i$

• The random variable

\[
z_n := \left[ \sum_{i=1}^{n} \sigma_i^2 \right]^{-\frac{1}{2}} \left[ \sum_{i=1}^{n} x_i - \mu_i \right]
\]

converges to a Normal Distribution $\mathcal{N}(0, 1)$

• Special case - IID random variables & average

\[
\frac{\sqrt{n}}{\sigma} \left[ \frac{1}{n} \sum_{i=1}^{n} x_i - \mu \right] \to \mathcal{N}(0, 1)
\]

$O \left( n^{-\frac{1}{2}} \right)$ convergence
Slutsky’s Theorem

- Continuous mapping theorem
  - \( X_i \) and \( Y_i \) sequences of random variables
  - \( X_i \) has as its limit the random variable \( X \)
  - \( Y_i \) has as its limit the constant \( c \)
  - \( g(x,y) \) is a continuous function for all \( g(x,c) \)

- \( g(X_i, Y_i) \) converges in distribution to \( g(X,c) \)
Delta Method

- Random variable $X_i$ convergent to $b$
  
  $a_n^{-2}(X_n - b) \to \mathcal{N}(0, \Sigma)$ with $a_n^2 \to 0$ for $n \to \infty$

- $g$ is a continuously differentiable function for $b$

- Then $g(X_i)$ inherits convergence properties
  
  $a_n^{-2} (g(X_n) - g(b)) \to \mathcal{N}(0, [\nabla_x g(b)] \Sigma [\nabla_x g(b)]^\top)$

- Proof: use Taylor expansion for $g(X_n) - g(b)$
  
  $a_n^{-2} [g(X_n) - g(b)] = [\nabla_x g(\xi_n)]^\top a_n^{-2}(X_n - b)$

- $g(\xi_n)$ is on line segment $[X_n, b]$

- By Slutsky’s theorem it converges to $g(b)$

- Hence $g(X_i)$ is asymptotically normal
Tools for the proof
Fourier Transform

• Fourier transform relations

\[ F[f](\omega) := (2\pi)^{-\frac{d}{2}} \int_{\mathbb{R}^n} f(x) \exp(-i \langle \omega, x \rangle) dx \]

\[ F^{-1}[g](x) := (2\pi)^{-\frac{d}{2}} \int_{\mathbb{R}^n} g(\omega) \exp(i \langle \omega, x \rangle) d\omega. \]

• Useful identities
  • Identity

\[ F^{-1} \circ F = F \circ F^{-1} = \text{Id} \]

• Derivative

\[ F[\partial_x f] = -i\omega F[f] \]

• Convolution (also holds for inverse transform)

\[ F[f \circ g] = (2\pi)^{\frac{d}{2}} F[f] \cdot F[g] \]
The Characteristic Function Method

• Characteristic function
  \[ \phi_X(\omega) := F^{-1}[p(x)] = \int \exp(i \langle \omega, x \rangle) dp(x) \]

• For \( X \) and \( Y \) independent we have
  • Joint distribution is convolution
    \[ p_{X+Y}(z) = \int p_X(z - y) p_Y(y) dy = p_X \circ p_Y \]
  • Characteristic function is product
    \[ \phi_{X+Y}(\omega) = \phi_X(\omega) \cdot \phi_Y(\omega) \]

• Proof - plug in definition of Fourier transform

• Characteristic function is unique
Proof - Weak law of large numbers

- Require that expectation exists
- Taylor expansion of exponential
  \[ \exp(iwx) = 1 + i \langle w, x \rangle + o(|w|) \]
  and hence \( \phi_X(\omega) = 1 + iwE_X[x] + o(|w|). \)
  (need to assume that we can bound the tail)
- Average of random variables
  \[ \phi_{\hat{\mu}_m}(\omega) = \left( 1 + \frac{i}{m} w\mu + o(m^{-1}|w|) \right)^m \]
- Limit is constant distribution
  \[ \phi_{\hat{\mu}_m}(\omega) \to \exp i\omega\mu = 1 + i\omega\mu + \ldots \]
Warning

• Moments may not always exist
• Cauchy distribution

\[ p(x) = \frac{1}{\pi} \frac{1}{1 + x^2} \]

• For the mean to exist the following integral would have to converge

\[ \int |x| dp(x) \geq \frac{2}{\pi} \int_{1}^{\infty} \frac{x}{1 + x^2} dx \geq \frac{1}{\pi} \int_{1}^{\infty} \frac{1}{x} dx = \infty \]
Proof - Central limit theorem

• Require that second order moment exists (we assume they’re all identical WLOG)

• Characteristic function

\[
\exp(iwx) = 1 + iwx - \frac{1}{2}w^2x^2 + o(|w|^2)
\]

and hence \(\phi_X(\omega) = 1 + iwE_X[x] - \frac{1}{2}w^2\text{var}_X[x] + o(|w|^2)\)

• Subtract out mean (centering)

\[
z_n := \left[\sum_{i=1}^{n} \sigma_i^2\right]^{-\frac{1}{2}} \left[\sum_{i=1}^{n} x_i - \mu_i\right]
\]

\[
\phi_{Z_m}(\omega) = \left(1 - \frac{1}{2m}w^2 + o(m^{-1}|w|^2)\right)^m \rightarrow \exp\left(-\frac{1}{2}w^2\right) \text{ for } m \rightarrow \infty
\]

This is the FT of a Normal Distribution
Central Limit Theorem in Practice

unscaled

scaled
Finite sample tail bounds
Simple tail bounds

• Gauss Markov inequality
  Random variable X with mean \( \mu \)
  \[
  \Pr(X \geq \epsilon) \leq \frac{\mu}{\epsilon}
  \]
  Proof - decompose expectation
  \[
  \Pr(X \geq \epsilon) = \int_{\epsilon}^{\infty} dp(x) \leq \int_{\epsilon}^{\infty} \frac{x}{\epsilon} dp(x) \leq \epsilon^{-1} \int_{0}^{\infty} x dp(x) = \frac{\mu}{\epsilon}.
  \]

• Chebyshev inequality
  Random variable X with mean \( \mu \) and variance \( \sigma^2 \)
  \[
  \Pr(|\hat{\mu}_m - \mu| > \epsilon) \leq \sigma^2 m^{-1} \epsilon^{-2}
  \]
  or equivalently \( \epsilon \leq \sigma / \sqrt{m} \delta \)
  Proof - applying Gauss-Markov to \( Y = (X - \mu)^2 \) with confidence \( \epsilon^2 \) yields the result.
Gauss-Markov

\[ \epsilon \leq \frac{\mu}{\delta} \]
Scales properly in \( \mu \) but expensive in \( \delta \)

Chebyshev

\[ \epsilon \leq \frac{\sigma}{\sqrt{m\delta}} \]
Proper scaling in \( \sigma \) but still bad in \( \delta \)

Can we get logarithmic scaling in \( \delta \)?
Chernoff bound

- KL-divergence variant of Chernoff bound
  \[ K(p, q) = p \log \frac{p}{q} + (1 - p) \log \frac{1 - p}{1 - q} \]

- \( n \) independent tosses from biased coin with \( p \)

\[
\Pr \left\{ \sum_i x_i \geq nq \right\} \leq \exp \left( -nK(q, p) \right) \leq \exp \left( -2n(p - q)^2 \right)
\]

- **Proof**  
  w.l.o.g. \( q > p \) and set \( k \geq qn \)

\[
\frac{\Pr \left\{ \sum_i x_i = k \mid q \right\}}{\Pr \left\{ \sum_i x_i = k \mid p \right\}} = \frac{q^k(1 - q)^{n-k}}{p^k(1 - p)^{n-k}} \geq \frac{q^{qn}(1 - q)^{n-qn}}{p^{qn}(1 - p)^{n-qn}} = \exp \left( nK(q, p) \right)
\]

\[
\sum_{k \geq nq} \Pr \left\{ \sum_i x_i = k \mid p \right\} \leq \sum_{k \geq nq} \Pr \left\{ \sum_i x_i = k \mid q \right\} \exp(-nK(q, p)) \leq \exp(-nK(q, p))
\]
McDiarmid Inequality

- Independent random variables $X_i$
- Function $f : \mathcal{X}^m \to \mathbb{R}$

Deviation from expected value

$$\Pr \left( |f(x_1, \ldots, x_m) - \mathbb{E}_{X_1, \ldots, X_m} [f(x_1, \ldots, x_m)]| > \epsilon \right) \leq 2 \exp \left( -2\epsilon^2 C^{-2} \right)$$

Here $C$ is given by $C^2 = \sum_{i=1}^{m} c_i^2$ where

$$|f(x_1, \ldots, x_i, \ldots, x_m) - f(x_1, \ldots, x'_i, \ldots, x_m)| \leq c_i$$

- Hoeffding’s theorem
  $f$ is average and $X_i$ have bounded range $c$

$$\Pr \left( |\hat{\mu}_m - \mu| > \epsilon \right) \leq 2 \exp \left( -\frac{2m\epsilon^2}{c^2} \right).$$
Scaling behavior

- Hoeffding

\[ \delta := \Pr (|\hat{\mu}_m - \mu| > \epsilon) \leq 2 \exp \left( -\frac{2m\epsilon^2}{c^2} \right) \]

\[ \Rightarrow \log \delta / 2 \leq -\frac{2m\epsilon^2}{c^2} \]

\[ \Rightarrow \epsilon \leq c \sqrt{\frac{\log 2 - \log \delta}{2m}} \]

This helps when we need to combine several tail bounds since we only pay logarithmically in terms of their combination.
More tail bounds

- Higher order moments
- Bernstein inequality (needs variance bound)

\[
\Pr (\mu_m - \mu \geq \epsilon) \leq \exp \left( -\frac{t^2/2}{\sum_i \mathbb{E}[X_i^2] + Mt/3} \right)
\]

here \( M \) upper-bounds the random variables \( X_i \)

- Proof via Gauss-Markov inequality applied to exponential sums (hence exp. inequality)
- See also Azuma, Bennett, Chernoff, ...

- Absolute / relative error bounds
- Bounds for (weakly) dependent random variables
Tail bounds in practice
A/B testing

• Two possible webpage layouts
• Which layout is better?

• Experiment
  • Half of the users see A
  • The other half sees design B

• How many trials do we need to decide which page attracts more clicks?

Assume that the probabilities are $p(A) = 0.1$ and $p(B) = 0.11$ respectively and that $p(A)$ is known.
Need to bound for a deviation of 0.01
Mean is \( p(B) = 0.11 \) (we don’t know this yet)
Want failure probability of 5%

If we have no prior knowledge, we can only bound the variance by \( \sigma^2 = 0.25 \)

\[
m \leq \frac{\sigma^2}{\epsilon^2 \delta} = \frac{0.25}{0.01^2 \cdot 0.05} = 50,000
\]

If we know that the click probability is at most 0.15 we can bound the variance at \( 0.15 \times 0.85 = 0.1275 \). This requires at most 25,500 users.
Hoeffding’s bound

- Random variable has bounded range [0, 1] (click or no click), hence $c=1$
- Solve Hoeffding’s inequality for $m$

$$m \leq -\frac{c^2 \log \delta/2}{2\epsilon^2} = -\frac{1 \cdot \log 0.025}{2 \cdot 0.01^2} < 18,445$$

This is slightly better than Chebyshev.
Normal Approximation
(Central Limit Theorem)

- Use asymptotic normality
- Gaussian interval containing 0.95 probability

\[
\frac{1}{2\pi\sigma^2} \int_{\mu - \epsilon}^{\mu + \epsilon} \exp \left( -\frac{(x - \mu)^2}{2\sigma^2} \right) dx = 0.95
\]

is given by \( \epsilon = 2.96\sigma \).

- Use variance bound of 0.1275 (see Chebyshev)

\[
m \leq \frac{2.96^2 \sigma^2}{\epsilon^2} = \frac{2.96^2 \cdot 0.1275}{0.01^2} \leq 11,172
\]

Same rate as Hoeffding bound!
Better bounds by bounding the variance.
Beyond

• Many different layouts?
• Combinatorial strategy to generate them (aka the Thai Restaurant process)
• What if it depends on the user / time of day
• Stateful user (e.g. query keywords in search)
• What if we have a good prior of the response (rather than variance bound)?

• Explore/exploit/reinforcement learning/control (more details at the end of this class)
2.3 Kernel Density Estimation
Density Estimation

• For discrete bins (e.g. male/female; English/French/German/Spanish/Chinese) we get good uniform convergence:

• Applying the union bound and Hoeffding:

\[
\Pr \left( \sup_{a \in A} |\hat{p}(a) - p(a)| \geq \epsilon \right) \leq \sum_{a \in A} \Pr \left( |\hat{p}(a) - p(a)| \geq \epsilon \right) \\
\leq 2|A| \exp \left( -2m\epsilon^2 \right)
\]

• Solving for error probability:

\[
\frac{\delta}{2|A|} \leq \exp(-m\epsilon^2) \implies \epsilon \leq \sqrt{\frac{\log 2|A| - \log \delta}{2m}}
\]

good news
• Continuous domain = infinite number of bins
• Curse of dimensionality
  • 10 bins on [0, 1] is probably good
  • $10^{10}$ bins on $[0, 1]^{10}$ requires high accuracy in estimate: probability mass per cell also decreases by $10^{10}$. 
Bin Counting
Bin Counting
Parzen Windows

- Naive approach
  Use empirical density (delta distributions)
  \[ p_{\text{emp}}(x) = \frac{1}{m} \sum_{i=1}^{m} \delta_{x_i}(x) \]

- This breaks if we see slightly different instances

- Kernel density estimate
  Smear out empirical density with a nonnegative smoothing kernel \( k_x(x') \) satisfying
  \[ \int_{\mathcal{X}} k_x(x') dx' = 1 \text{ for all } x \]
• **Density estimate**

\[
p_{\text{emp}}(x) = \frac{1}{m} \sum_{i=1}^{m} \delta_{x_i}(x)
\]

\[
\hat{p}(x) = \frac{1}{m} \sum_{i=1}^{m} k_{x_i}(x)
\]

• **Smoothing kernels**

- **Gauss**: \((2\pi)^{-\frac{1}{2}} e^{-\frac{1}{2}x^2}\)
- **Laplace**: \(\frac{1}{2} e^{-|x|}\)
- **Epanechnikov**: \(\frac{3}{4} \max(0, 1 - x^2)\)
- **Uniform**: \(\frac{1}{2} \chi([-1,1])(x)\)
Size matters

Fig. 2.5. Left: a naive density estimate given a sample of the weight of 18 persons. Right: the underlying weight distribution.

Fig. 2.6. Parzen windows density estimate associated with the 18 observations of the Figure above. From left to right: Gaussian kernel density estimate with kernel of width 0.3, 1, 3, and 10 respectively.

Fig. 2.7. Some kernels for Parzen windows density estimation. From left to right: Gaussian kernel, Laplace kernel, Epanechikov kernel, and uniform density.

Moreover, there is the issue of choosing a suitable kernel function. The fact that a large variety of them exists might suggest that this is a crucial issue. In practice, this turns out not to be the case and instead, the choice of a suitable kernel width is much more vital for good estimates. In other words, size matters, shape is secondary.

The problem is that we do not know which kernel width is best for the data. If the problem is one-dimensional, we might hope to be able to eyeball the size of $r$. Obviously, in higher dimensions this approach fails. A second
• **Kernel width**

\[ k_{x_i}(x) = r^{-d} h \left( \frac{x - x_i}{r} \right) \]

• Too narrow overfits

• Too wide smoothes with constant distribution

• **How to choose?**
Smoothing

Gaussian Kernel with width $\sigma = 1$
Smoothing

Laplacian Kernel with width $\lambda = 1$
Smoothing

Laplacian Kernel with width $\lambda = 10$
Capacity Control
Capacity control

- Need automatic mechanism to select scale
- Overfitting
  - Maximum likelihood will lead to $r=0$
    (smoothing kernels peak at instances)
  - This is (typically) a set of measure 0.
- Validation set
  Set aside data just for calibrating $r$
- Leave-one-out estimation
  Estimate likelihood using all but one instance
- Alternatives: use a prior on $r$; convergence analysis
Capacity Control

- **Validation set**

\[
\log \hat{p}(X') = \sum_{x' \in X'} \log \hat{p}(x') = \sum_{x' \in X'} \log \sum_{x \in X} k \left( \frac{x-x'}{r} \right) - |X'| [d \log r + \log |X|]
\]

- **Leave-one-out crossvalidation**

\[
\hat{p}_{X \setminus \{x\}}(x) = \frac{1}{m-1} \sum_{x' \in X \setminus \{x\}} r^{-d}k \left( \frac{x' - x}{r} \right)
\]

\[
= \frac{m}{m-1} \left[ \hat{p}(x) - m^{-1} r^{-d} k(0) \right]
\]

\[
\Rightarrow L[X] = m \log m / (m - 1) + \sum_{x \in X} \log \left[ \hat{p}(x) - m^{-1} r^{-d} k(0) \right]
\]
Leave-one-out estimate
Optimal estimate

Laplacian Kernel with width optimal $\lambda$
Silverman’s rule
Silverman’s rule

- Chicken and egg problem
  - Want wide kernel for low density region
  - Want narrow kernel where we have much data
- Need density estimate to estimate density
- Simple hack
  Use average distance from k nearest neighbors

\[ r_i = \frac{r}{k} \sum_{x \in \text{NN}(x_i, k)} \|x_i - x\| \]
true density
non adaptive estimate
adaptive estimate
Watson-Nadaraya estimator
Weighted smoother

• Problem
  Given pairs \((x_i, y_i)\) estimate \(y | x\) for new \(x\)

• Idea
  Use distance weighted average of \(y_i\)
  \[
  \hat{y}(x) = \sum_i y_i \frac{k_{x_i}(x)}{\sum_j k_{x_j}(x)} = \frac{\sum_i y_i k_{x_i}(x)}{\sum_j k_{x_j}(x)}
  \]
Watson-Nadaraya Classifier
Watson-Nadaraya regression estimate
k-Nearest Neighbors

- Further simplification
  - Same weight for all nearest neighbors
  - Same number of neighbors everywhere
- Classification
  Use majority rule to estimate label
- Regression
  Use average for label
2.4 Exponential Families
Exponential Families
Exponential Families

- Density function

\[ p(x; \theta) = \exp \left( \langle \phi(x), \theta \rangle - g(\theta) \right) \]

where \( g(\theta) = \log \sum_{x'} \exp \left( \langle \phi(x'), \theta \rangle \right) \)
Exponential Families

• **Density function**

\[ p(x; \theta) = \exp (\langle \phi(x), \theta \rangle - g(\theta)) \]

where \( g(\theta) = \log \sum_{x'} \exp (\langle \phi(x'), \theta \rangle) \)

• **Log partition function generates cumulants**

\[ \partial_\theta g(\theta) = \mathbf{E} [\phi(x)] \]
\[ \partial_\theta^2 g(\theta) = \text{Var} [\phi(x)] \]
Exponential Families

• Density function

\[ p(x; \theta) = \exp (\langle \phi(x), \theta \rangle - g(\theta)) \]

where \( g(\theta) = \log \sum_{x'} \exp (\langle \phi(x'), \theta \rangle) \)

• Log partition function generates cumulants

\[ \partial_\theta g(\theta) = \mathbb{E} [\phi(x)] \]

\[ \partial^2_\theta g(\theta) = \text{Var} [\phi(x)] \]

• \( g \) is convex (second derivative is p.s.d.)
Examples

• Binomial Distribution
  \( \phi(x) = x \)

• Discrete Distribution
  \( \phi(x) = e_x \)
  \( (e_x \text{ is unit vector for } x) \)

• Gaussian
  \( \phi(x) = \left( x, \frac{1}{2}xx^\top \right) \)

• Poisson (counting measure \( 1/x! \))
  \( \phi(x) = x \)

• Dirichlet, Beta, Gamma, Wishart, …
Normal Distribution

\[ \phi_{\mu, \sigma^2}(x) \]

- \( \mu = 0, \quad \sigma^2 = 0.2, \) (blue)
- \( \mu = 0, \quad \sigma^2 = 1.0, \) (red)
- \( \mu = 0, \quad \sigma^2 = 5.0, \) (orange)
- \( \mu = -2, \quad \sigma^2 = 0.5, \) (green)
Poisson Distribution

\[ p(x; \lambda) = \frac{\lambda^x e^{-\lambda}}{x!} \]
Beta Distribution

\[ p(x; \alpha, \beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha, \beta)} \]
Dirichlet Distribution

... this is a distribution over distributions ...
Maximum Likelihood
Maximum Likelihood

- Negative log-likelihood

\[- \log p(X; \theta) = \sum_{i=1}^{n} g(\theta) - \langle \phi(x_i), \theta \rangle\]
Maximum Likelihood

• Negative log-likelihood

\[ - \log p(X; \theta) = \sum_{i=1}^{n} g(\theta) - \langle \phi(x_i), \theta \rangle \]

• Taking derivatives

\[ -\partial_\theta \log p(X; \theta) = m \left[ \mathbb{E}[\phi(x)] - \frac{1}{m} \sum_{i=1}^{n} \phi(x_i) \right] \]

We pick the parameter such that the distribution matches the empirical average.
Conjugate Priors

- Unless we have lots of data estimates are weak
- Usually we have an idea of what to expect
  \[ p(\theta|X) \propto p(X|\theta) \cdot p(\theta) \]
  we might even have ‘seen’ such data before
- Solution: add ‘fake’ observations
  \[ p(\theta) \propto p(X_{\text{fake}}|\theta) \text{ hence } p(\theta|X) \propto p(X|\theta)p(X_{\text{fake}}|\theta) = p(X \cup X_{\text{fake}}|\theta) \]
- Inference (generalized Laplace smoothing)
  \[
  \frac{1}{n} \sum_{i=1}^{n} \phi(x_i) \rightarrow \frac{1}{n+m} \sum_{i=1}^{n} \phi(x_i) + \frac{m}{n+m} \mu_0 
  \]
  fake count
  fake mean
Example: Gaussian Estimation

• Sufficient statistics: $x, x^2$

• Mean and variance given by

$\mu = \mathbb{E}[x]$ and $\sigma^2 = \mathbb{E}[x^2] - \mathbb{E}[x]^2$

• Maximum Likelihood Estimate

$\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} x_i$ and $\sigma^2 = \frac{1}{n} \sum_{i=1}^{n} x_i^2 - \hat{\mu}^2$

• Maximum a Posteriori Estimate

$\hat{\mu} = \frac{1}{n + n_0} \sum_{i=1}^{n} x_i$ and $\sigma^2 = \frac{1}{n + n_0} \sum_{i=1}^{n} x_i^2 + \frac{n_0}{n + n_0} 1 - \hat{\mu}^2$
Collapsing

• Conjugate priors

\[ p(\theta) \propto p(X_{\text{fake}} | \theta) \]

Hence we know how to compute normalization

• Prediction

\[ p(x | X) = \int p(x | \theta)p(\theta | X) d\theta \]

\[ \propto \int p(x | \theta)p(X | \theta)p(X_{\text{fake}} | \theta) d\theta \]

\[ = \int p(\{x\} \cup X \cup X_{\text{fake}} | \theta) d\theta \]

look up closed form expansions

Beta, binomial
Dirichlet, multinomial
Gamma, Poisson
Wishart, Gauss

http://en.wikipedia.org/wiki/Exponential_family
Conjugate Prior in action

\[ p(x = i) = \frac{n_i}{n} \rightarrow p(x = i) = \frac{n_i + m_i}{n + m} \]

<table>
<thead>
<tr>
<th>Outcome</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Counts</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>MLE</td>
<td>0.15</td>
<td>0.30</td>
<td>0.10</td>
<td>0.05</td>
<td>0.20</td>
<td>0.20</td>
</tr>
<tr>
<td>MAP ((m_0 = 6))</td>
<td>0.15</td>
<td>0.27</td>
<td>0.12</td>
<td>0.08</td>
<td>0.19</td>
<td>0.19</td>
</tr>
<tr>
<td>MAP ((m_0 = 100))</td>
<td>0.16</td>
<td>0.19</td>
<td>0.16</td>
<td>0.15</td>
<td>0.17</td>
<td>0.17</td>
</tr>
</tbody>
</table>

\(m_i = m \cdot [\mu_0]_i\)
Conjugate Prior in action

• Discrete Distribution

\[ p(x = i) = \frac{n_i}{n} \rightarrow p(x = i) = \frac{n_i + m_i}{n + m} \]

• Tossing a dice

<table>
<thead>
<tr>
<th>Outcome</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Counts</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>MLE</td>
<td>0.15</td>
<td>0.30</td>
<td>0.10</td>
<td>0.05</td>
<td>0.20</td>
<td>0.20</td>
</tr>
<tr>
<td>MAP (m_0 = 6)</td>
<td>0.15</td>
<td>0.27</td>
<td>0.12</td>
<td>0.08</td>
<td>0.19</td>
<td>0.19</td>
</tr>
<tr>
<td>MAP (m_0 = 100)</td>
<td>0.16</td>
<td>0.19</td>
<td>0.16</td>
<td>0.15</td>
<td>0.17</td>
<td>0.17</td>
</tr>
</tbody>
</table>

\(m_i = m \cdot \left[ \mu_0 \right]_i\)
Conjugate Prior in action

- **Discrete Distribution**
  
  \[ p(x = i) = \frac{n_i}{n} \quad \rightarrow \quad p(x = i) = \frac{n_i + m_i}{n + m} \]

- **Tossing a dice**

<table>
<thead>
<tr>
<th>Outcome</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Counts</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>MLE</td>
<td>0.15</td>
<td>0.30</td>
<td>0.10</td>
<td>0.05</td>
<td>0.20</td>
<td>0.20</td>
</tr>
<tr>
<td>MAP ((m_0 = 6))</td>
<td>0.15</td>
<td>0.27</td>
<td>0.12</td>
<td>0.08</td>
<td>0.19</td>
<td>0.19</td>
</tr>
<tr>
<td>MAP ((m_0 = 100))</td>
<td>0.16</td>
<td>0.19</td>
<td>0.16</td>
<td>0.15</td>
<td>0.17</td>
<td>0.17</td>
</tr>
</tbody>
</table>

- **Rule of thumb**
  
  need 10 data points (or prior) per parameter

\[ m_i = m \cdot [\mu_0]_i \]
Honest dice

MLE

MAP
Tainted dice

MLE

MAP
Priors (part deux)

- **Parameter smoothing**
  \[ p(\theta) \propto \exp(-\lambda \|\theta\|_1) \text{ or } p(\theta) \propto \exp(-\lambda \|\theta\|_2^2) \]

- **Posterior**
  \[
p(\theta|x) \propto \prod_{i=1}^{m} p(x_i|\theta)p(\theta) \\
  \propto \exp \left( \sum_{i=1}^{m} \langle \phi(x_i), \theta \rangle - mg(\theta) - \frac{1}{2\sigma^2} \|\theta\|_2^2 \right)\]

- **Convex optimization problem (MAP estimation)**
  \[
  \text{minimize } g(\theta) - \left\langle \frac{1}{m} \sum_{i=1}^{m} \phi(x_i), \theta \right\rangle + \frac{1}{2m\sigma^2} \|\theta\|_2^2
  \]
Statistics

• Probabilities
  • Bayes rule, Dependence, independence, conditional probabilities
  • Priors, Naive Bayes classifier

• Tail bounds
  • Chernoff, Hoeffding, Chebyshev, Gaussian
  • A/B testing

• Kernel density estimation
  • Parzen windows, Nearest neighbors, Watson-Nadaraya estimator

• Exponential families
  • Gaussian, multinomial, Poisson
  • Conjugate distributions and smoothing, integrating out
Further reading

• Manuscript (book chapters 1 and 2)
  http://alex.smola.org/teaching/berkeley2012/slides/chapter1_2.pdf