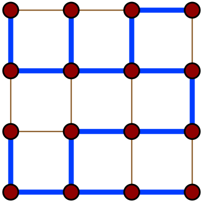
COMS 4771 Lecture 1

1. Course overview
2. Maximum likelihood estimation (review of some statistics)

Course overview



Minimum spanning tree

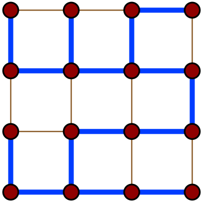
- **Input**: Graph G.
- **Output**: A minimum spanning tree in G.

Input/output relationship well-specified for all inputs.

Minimum spanning tree

- **Input**: Graph G.
- **Output**: A minimum spanning tree in G.

Input/output relationship well-specified for all inputs.

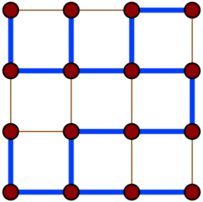


Bird species recognition

- **Input**: Image of a bird.
- **Output**: Species name of the bird.

Input/output relationship is difficult to specify.

Algorithmic problems



Minimum spanning tree

- **Input**: Graph G.
- **Output**: A minimum spanning tree in G.

Input/output relationship well-specified for all inputs.

Bird species recognition

- **Input**: Image of a bird.
- **Output**: Species name of the bird.

Input/output relationship is difficult to specify.

Machine learning: use examples of input/output pairs to *learn* the mapping

 $\mapsto$ "indigo bunting"

Machine learning in context

Perspective of intelligent systems

- **Goal**: robust system with "intelligent" behavior
  - **Often**: hard-coded solution too complex, not robust, sub-optimal
- How do we learn from past experiences to perform well in the future?

Machine learning in context

Perspective of intelligent systems

- **Goal**: robust system with "intelligent" behavior
  - **Often**: hard-coded solution too complex, not robust, sub-optimal
- How do we learn from past experiences to perform well in the future?

Perspective of algorithmic statistics

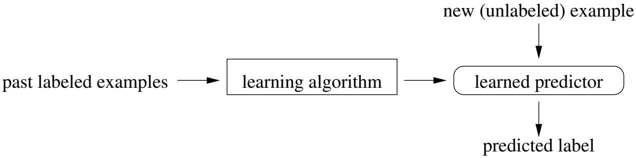
- **Goal**: statistical analysis of large, complex data sets
  - **Past**: ≤ 100 data points of two variables.
    Data collection and statistical analysis done by hand/eye.
  - **Now**: several million data and variables, collected by high-throughput automatic processes.
- How can we automate statistical analysis for modern applications?

Supervised learning

Abstract problem

- **Data**: labeled examples $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \in \mathcal{X} \times \mathcal{Y}$ from some population.

  $\mathcal{X}$ = **input (feature) space**; $\mathcal{Y}$ = **output (label, response) space**.
- **Underlying assumption**: there's a relatively simple function $f \colon \mathcal{X} \to \mathcal{Y}$ such that $f(x) \approx y$ for most (x, y) in the population.
- **Learning task**: using the labeled examples, construct $\hat{f}$ such that $\hat{f} \approx f$.
- **At test time**: use $\hat{f}$ to predict y for new *(and previously unseen)* x's.



Supervised learning: examples

- Spam filtering
  - $\mathcal{X}$ = e-mail messages
  - $\mathcal{Y} = \{\text{spam}, \text{not spam}\}$

Supervised learning: examples

- Spam filtering
  - $\mathcal{X}$ = e-mail messages
  - $\mathcal{Y} = \{\text{spam}, \text{not spam}\}$
- Optical character recognition
  - $\mathcal{X} = 32 \times 32$ pixel images
  - $\mathcal{Y} = \{\mathsf{A}, \mathsf{B}, \ldots, \mathsf{Z}, \mathsf{a}, \mathsf{b}, \ldots, \mathsf{z}, 0, 1, \ldots, 9\}$

Supervised learning: examples

- Spam filtering
  - $\mathcal{X} =$ e-mail messages
  - $\mathcal{Y} = \{\text{spam}, \text{not spam}\}$
- Optical character recognition
  - $\mathcal{X} = 32 \times 32$ pixel images
  - $\mathcal{Y} = \{\text{A}, \text{B}, \ldots, \text{Z}, \text{a}, \text{b}, \ldots, \text{z}, 0, 1, \ldots, 9\}$
- Online dating
  - $\mathcal{X} =$ user profiles $\times$ user profiles
  - $\mathcal{Y} = [0, 1]$

Supervised learning: examples

- Spam filtering
  - $\mathcal{X}$ = e-mail messages
  - $\mathcal{Y} = \{\text{spam}, \text{not spam}\}$
- Optical character recognition
  - $\mathcal{X} = 32 \times 32$ pixel images
  - $\mathcal{Y} = \{\mathsf{A}, \mathsf{B}, \ldots, \mathsf{Z}, \mathsf{a}, \mathsf{b}, \ldots, \mathsf{z}, 0, 1, \ldots, 9\}$
- Online dating
  - $\mathcal{X}$ = user profiles $\times$ user profiles
  - $\mathcal{Y} = [0, 1]$
- Machine translation
  - $\mathcal{X}$ = sequences of English words
  - $\mathcal{Y}$ = sequences of French words

Abstract problem

- **Data**: (unlabeled) examples $x_1, x_2, \ldots, x_n \in \mathcal{X}$ from some population.
- **Underlying assumption**: there's some interesting structure in the population to be discovered.
- **Learning task**: using the unlabeled examples, find the interesting structure.
- **Uses**: visualization/interpretation, pre-process data for downstream learning, . . .

Unsupervised learning

Abstract problem

- **Data**: (unlabeled) examples $x_1, x_2, \ldots, x_n \in \mathcal{X}$ from some population.
- **Underlying assumption**: there's some interesting structure in the population to be discovered.
- **Learning task**: using the unlabeled examples, find the interesting structure.
- **Uses**: visualization/interpretation, pre-process data for downstream learning, ...

Examples

- Discover sub-communities of individuals in a social network.
- Explain variability of market price movement using a few latent factors.
- Learn a useful representation of data that improves supervised learning.

What else is there with machine learning?

Advanced issues

- Structured output spaces
- Distributed learning
- Incomplete data
- Causal inference
- Privacy
- ...

Other models of learning

- Semi-supervised learning
- Active learning
- Online learning
- Reinforcement learning
- ...

Major application areas

- Natural language processing
- Speech recognition
- Computer vision
- Computational biology
- Information retrieval
- ...

Modes of study

- Mathematical analysis
- Cross-domain evaluations
- End-to-end application study
- ...

This course

`http://www.cs.columbia.edu/~djhsu/coms4771-s15/`

Topics

1. **Supervised learning**
   - Core issues of statistical machine learning
   - Algorithmic, statistical, and analytical tools
2. **Some topics in unsupervised learning**
   - Common statistical models
   - Frameworks for developing new models and algorithms

Coursework

1. Around five homework assignments (theory & programming): 40%
2. Two in-class exams (3/11, 5/4): 30% each
3. No late assignments accepted, no make-up exams

Prerequisites

Mathematical prerequisites

- Basic algorithms and data structures
- Linear algebra (e.g., vector spaces, orthogonality, spectral decomposition)
- Multivariate calculus (e.g., limits, Taylor expansion, gradients)
- Probability/statistics (e.g., random variables, expectation, LLN, MLE)

Computational prerequisites

- You should have regular access to and be able to program in MATLAB.

  MATLAB is available for download for SEAS students:
  `http://portal.seas.columbia.edu/matlab/`

Resources

`http://www.cs.columbia.edu/~djhsu/coms4771-s15/`

Course staff

- **Instructor**: Daniel Hsu
- **Teaching assistants**: Huaiyuan Cao, Angus Ding, Henrique Gubert, Siyao Li, Michael Yang
- **Office hours, course e-mail, online forum (Piazza)**: see course website
- Office hour attendance highly recommended.

Materials

- **Lecture slides**: posted on course website
- **Textbooks**: readings from "The Elements of Statistical Learning" [ESL] and "A Course in Machine Learning" [CML] (both available free online; see course website)

Maximum likelihood estimation

Statistical models

- A **model** $\mathcal{P} = \{P_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \mathcal{T}\}$ is a set of probability distributions over $\mathcal{X}$ indexed by a **parameter space** $\mathcal{T}$.
- Often, we use models with a fixed number of parameters (e.g., $\mathcal{T} \subset \mathbb{R}^d$); these are called **parametric models**.
- Also, often deal with models where each $P_{\boldsymbol{\theta}}$ has a **density function** $p(\,\cdot\,;\boldsymbol{\theta})\colon \mathcal{X} \to \mathbb{R}_+$.

Statistical models

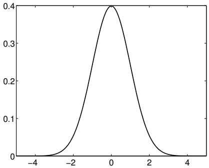
- A **model** $\mathcal{P} = \{P_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \mathcal{T}\}$ is a set of probability distributions over $\mathcal{X}$ indexed by a **parameter space** $\mathcal{T}$.
- Often, we use models with a fixed number of parameters (e.g., $\mathcal{T} \subset \mathbb{R}^d$); these are called **parametric models**.
- Also, often deal with models where each $P_{\boldsymbol{\theta}}$ has a **density function** $p(\,\cdot\,;\boldsymbol{\theta})\colon \mathcal{X} \to \mathbb{R}_+$.

Parameter estimation

- Given data, choose parameter $\boldsymbol{\theta} \in \mathcal{T}$ such that $P_{\boldsymbol{\theta}}$ "fits the data well".
- Use chosen $P_{\boldsymbol{\theta}}$ to make inferences or draw conclusions (e.g., use in supervised learning to build a predictor).
- Our main tool will be **maximum likelihood estimation**.

Gaussian density in one dimension ($\mathcal{X} = \mathbb{R}$)

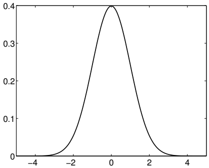
$$p(x; \mu, \sigma) := \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$



- μ = mean, σ^2 = variance
- $\frac{x-\mu}{\sigma}$ = deviation from mean in units of σ.

Gaussian density in one dimension ($\mathcal{X} = \mathbb{R}$)

$$p(x; \mu, \sigma) := \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

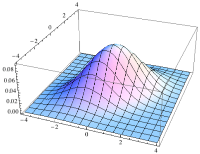


- μ = mean, σ^2 = variance
- $\dfrac{x-\mu}{\sigma}$ = deviation from mean in units of σ.

Gaussian density in d dimensions ($\mathcal{X} = \mathbb{R}^d$)

In the density function, the quadratic function

$$-\frac{(x-\mu)^2}{2\sigma^2} \quad = \quad -\frac{1}{2}(x-\mu)(\sigma^2)^{-1}(x-\mu)$$



is replaced by a multivariate quadratic form:

$$p(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) := \frac{1}{\sqrt{(2\pi)^d \det(\boldsymbol{\Sigma})}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

Maximum likelihood estimation

Setting

- **Given**: data $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n \in \mathcal{X}$; parametric model $\mathcal{P} = \{P_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \mathcal{T}\}$.
- **The i.i.d. assumption**: assume $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ are independent and identically distributed according to the same probability distribution.
- **Likelihood of $\boldsymbol{\theta}$ given data**: (under the i.i.d. assumption)

$$\prod_{i=1}^{n} p(\boldsymbol{x}_i; \boldsymbol{\theta}),$$

the probability mass (or density) of the data, as given by $P_{\boldsymbol{\theta}}$.

Maximum likelihood estimation

Setting

- **Given**: data $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n \in \mathcal{X}$; parametric model $\mathcal{P} = \{P_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \mathcal{T}\}$.
- **The i.i.d. assumption**: assume $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ are independent and identically distributed according to the same probability distribution.
- **Likelihood of $\boldsymbol{\theta}$ given data**: (under the i.i.d. assumption)

$$\prod_{i=1}^{n} p(\boldsymbol{x}_i; \boldsymbol{\theta}),$$

  the probability mass (or density) of the data, as given by $P_{\boldsymbol{\theta}}$.

Maximum likelihood estimator

The **maximum likelihood estimator** (MLE) for the model $\mathcal{P}$ is

$$\boldsymbol{\theta}_{\mathsf{ML}} := \arg\max_{\boldsymbol{\theta} \in \mathcal{T}} \prod_{i=1}^{n} p(\boldsymbol{x}_i; \boldsymbol{\theta})$$

i.e., the parameter $\boldsymbol{\theta} \in \mathcal{T}$ whose likelihood is highest given the data.

Logarithm trick

Recall: logarithms turn products into sums

$$\log\left(\prod_{i=1}^{n} f_i\right) = \sum_{i=1}^{n} \log(f_i)$$

Logarithms and maxima

The logarithm is monotonically increasing on $\mathbb{R}_{++}$.

Consequence: Application of $\log$ does not change the *location* of a maximum or minimum:

$\max_y \log(g(y)) \neq \max_y g(y)$	The *value* changes (in general).
$\boxed{\arg\max_y \log(g(y)) = \arg\max_y g(y)}$	The *location* does not change.

it's important to understand this!!!

MLE: MAXIMALITY CRITERION

Likelihood and logarithm trick

$$\begin{aligned}
\boldsymbol{\theta}_{\mathsf{ML}} &= \underset{\boldsymbol{\theta} \in \mathcal{T}}{\arg\max} \prod_{i=1}^{n} p(\boldsymbol{x}_i; \boldsymbol{\theta}) \\
&= \underset{\boldsymbol{\theta} \in \mathcal{T}}{\arg\max} \log \left(\prod_{i=1}^{n} p(\boldsymbol{x}_i; \boldsymbol{\theta}) \right) \\
&= \underset{\boldsymbol{\theta} \in \mathcal{T}}{\arg\max} \sum_{i=1}^{n} \log p(\boldsymbol{x}_i; \boldsymbol{\theta})
\end{aligned}$$

Maximality criterion

Assuming $\boldsymbol{\theta}$ is unconstrained, the log-likelihood maximizer must satisfy

$$\mathbf{0} = \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{x}_i; \boldsymbol{\theta})$$

For some models, can analytically find unique solution $\boldsymbol{\theta} \in \mathcal{T}$.

Example: Gaussian mean MLE

Model: multivariate Gaussians with fixed covariance

The model $\mathcal{P}$ is the set of all Gaussian densities on $\mathbb{R}^d$ with *fixed* covariance matrix $\boldsymbol{\Sigma}$:

$$\mathcal{P} = \left\{ g(\,.\,; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \,:\, \boldsymbol{\mu} \in \mathbb{R}^d \right\}$$

where g is the Gaussian density function. The parameter space is $\mathcal{T} = \mathbb{R}^d$.

MLE equation

Solve the following equation (from the maximality criterion) for $\boldsymbol{\mu}$:

$$\sum_{i=1}^{n} \nabla_{\boldsymbol{\mu}} \log g(\boldsymbol{x}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \mathbf{0}.$$

Example: Gaussian mean MLE

$$
\begin{aligned}
\mathbf{0} &= \sum_{i=1}^{n} \nabla_{\boldsymbol{\mu}} \left[\log \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}) \right) \right] \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\mu}} \left[\log \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}|}} - \frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}) \right] \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\mu}} \left[-\frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}) \right] = -\sum_{i=1}^{n} \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu})
\end{aligned}
$$

Example: Gaussian mean MLE

$$\begin{aligned}
\mathbf{0} &= \sum_{i=1}^{n} \nabla_{\boldsymbol{\mu}} \left[\log \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}) \right) \right] \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\mu}} \left[\log \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}|}} - \frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}) \right] \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\mu}} \left[-\frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}) \right] = -\sum_{i=1}^{n} \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu})
\end{aligned}$$

Multiplication by $(-\boldsymbol{\Sigma})$ on both sides gives

$$\mathbf{0} = \sum_{i=1}^{n} (\boldsymbol{x}_i - \boldsymbol{\mu}) \qquad \Longrightarrow \qquad \boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i$$

Example: Gaussian mean MLE

$$\begin{aligned}
\mathbf{0} &= \sum_{i=1}^{n} \nabla_{\boldsymbol{\mu}} \left[\log \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}) \right) \right] \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\mu}} \left[\log \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}|}} - \frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}) \right] \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\mu}} \left[-\frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}) \right] = -\sum_{i=1}^{n} \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu})
\end{aligned}$$

Multiplication by $(-\boldsymbol{\Sigma})$ on both sides gives

$$\mathbf{0} = \sum_{i=1}^{n} (\boldsymbol{x}_i - \boldsymbol{\mu}) \qquad \Longrightarrow \qquad \boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i$$

Conclusion

The maximum likelihood estimator of the Gaussian mean parameter is

$$\boldsymbol{\mu}_{\mathsf{ML}} := \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i.$$

Example: Gaussian with unknown covariance

Model: multivariate Gaussians

The model $\mathcal{P}$ is now

$$\mathcal{P} = \left\{ g(\,.\,; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \,:\, \boldsymbol{\mu} \in \mathbb{R}^d, \boldsymbol{\Sigma} \in \mathbb{S}_{++}^{d \times d} \right\}$$

where $\mathbb{S}_{++}^{d}$ is the set of symmetric positive definite $d \times d$ matrices. The parameter space is $\mathcal{T} = \mathbb{R}^d \times \mathbb{S}_{++}^{d \times d}$.

ML approach

Since we have just seen that the ML estimator of $\boldsymbol{\mu}$ does not depend on $\boldsymbol{\Sigma}$, we can compute $\boldsymbol{\mu}_{\mathsf{ML}}$ first. We then estimate $\boldsymbol{\Sigma}$ using the criterion

$$\sum_{i=1}^{n} \nabla_{\boldsymbol{\Sigma}} \log g(\boldsymbol{x}_i; \boldsymbol{\mu}_{\mathsf{ML}}, \boldsymbol{\Sigma}) = 0.$$

Solution

The ML estimator of $\boldsymbol{\Sigma}$ is

$$\boldsymbol{\Sigma}_{\mathsf{ML}} := \frac{1}{n} \sum_{i=1}^{n} (\boldsymbol{x}_i - \boldsymbol{\mu}_{\mathsf{ML}})(\boldsymbol{x}_i - \boldsymbol{\mu}_{\mathsf{ML}})^{\top}.$$

Bernoulli distribution

"$X \sim \mathrm{Bern}(p)$" means X is a $\{0,1\}$-valued random variable whose mean is p.

- $\Pr[X = 1] = p$, $\Pr[X = 0] = 1 - p$.
- Mean of X is p.
- Variance of X is $p(1-p)$.

Bernoulli likelihood

Likelihood of $p \in [0,1]$ given $x \in \{0,1\}$:

$$p^x (1-p)^{1-x}.$$

Example: Bernoulli MLE

Model: Bernoulli distributions

The model $\mathcal{P}$ is "all Bernoulli distributions".
The parameter space is $\mathcal{T} = [0, 1]$.

MLE equation

$$\sum_{i=1}^{n} \nabla_p \log\big(p^{x_i}(1-p)^{1-x_i}\big) \;=\; \sum_{i=1}^{n} \frac{x_i}{p} - \frac{1-x_i}{1-p} \;=\; 0.$$

(Question: what about $p = 0$ or $p = 1$?)

Solution

The maximum likelihood estimator of the Bernoulli parameter p is

$$p_{\mathsf{ML}} := \frac{1}{n} \sum_{i=1}^{n} x_i.$$