

COMS 4771 Lecture 1

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

ADMINISTRIVIA

THIS COURSE

<http://www.satyenkale.com/coms4771/>

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. Four homework assignments (theory & programming): 24%
2. Kaggle-style prediction competition: 26%
3. Two in-class exams (3/3, 4/28): 25% each
4. 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/>

<http://www.satyenkale.com/coms4771/>

Course staff

- ▶ **Instructor:** Satyen Kale
- ▶ **Teaching assistants:** Gaurang Sadekar, Patanjali Vakhulabharanam, Robert Ying, Xiyue Yu
- ▶ **Office hours, 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], “A Course in Machine Learning” [CML], “Understanding Machine Learning” [UML] (available free online; see course website)

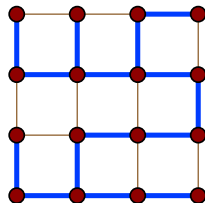
MACHINE LEARNING OVERVIEW

ALGORITHMIC PROBLEMS

Minimum spanning tree

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

Input/output relationship well-specified for all inputs.

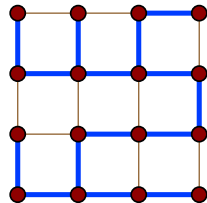


ALGORITHMIC PROBLEMS

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.

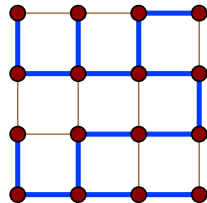


ALGORITHMIC PROBLEMS

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"

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 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?

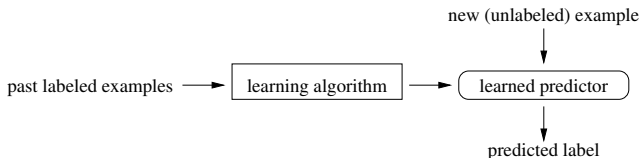
Perspective of algorithmic statistics

- ▶ **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 $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n) \in \mathcal{X} \times \mathcal{Y}$ from some population.
 $\mathcal{X}$ = input (feature) space; $\mathcal{Y}$ = output (label, response) space.
- **Underlying assumption:** there's a relatively simple function $f: \mathcal{X} \rightarrow \mathcal{Y}$ 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
- ▶ $\mathcal{Y} = \{A, B, \dots, Z, a, b, \dots, z, 0, 1, \dots, 9\}$

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
- ▶ $\mathcal{Y} = \{A, B, \dots, Z, a, b, \dots, z, 0, 1, \dots, 9\}$

- ▶ 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
 - ▶ $\mathcal{Y} = \{A, B, \dots, Z, a, b, \dots, z, 0, 1, \dots, 9\}$
- ▶ Online dating
 - ▶ $\mathcal{X} = \text{user profiles} \times \text{user profiles}$
 - ▶ $\mathcal{Y} = [0, 1]$
- ▶ Machine translation
 - ▶ $\mathcal{X}$ = sequences of English words
 - ▶ $\mathcal{Y}$ = sequences of French words

Abstract problem

- ▶ **Data:** (unlabeled) examples $x_1, x_2, \dots, x_n \in \mathcal{X}$ 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 $x_1, x_2, \dots, x_n \in \mathcal{X}$ 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
- ▶ ...

MAXIMUM LIKELIHOOD ESTIMATION

EXAMPLE: BINARY PREDICTION

- ▶ On a sequence of 10 days, observe weather:
1, 0, 1, 1, 0, 0, 1, 0, 1, 0
1 = rain, 0 = no rain

EXAMPLE: BINARY PREDICTION

- ▶ On a sequence of 10 days, observe weather:
1, 0, 1, 1, 0, 0, 1, 0, 1, 0
1 = rain, 0 = no rain
- ▶ Goal: understand the distribution of the weather

EXAMPLE: BINARY PREDICTION

- ▶ On a sequence of 10 days, observe weather:
1, 0, 1, 1, 0, 0, 1, 0, 1, 0
1 = rain, 0 = no rain
- ▶ Goal: understand the distribution of the weather
- ▶ Modeling assumption: rain on each day with a fixed probability $p \in [0, 1]$
- ▶ **Bernoulli distribution with parameter p :** " $X \sim \text{Bern}(p)$ " means X is a $\{0, 1\}$ -valued random variable with $\Pr[X = 1] = p$.

EXAMPLE: BINARY PREDICTION

- ▶ On a sequence of 10 days, observe weather:
1, 0, 1, 1, 0, 0, 1, 0, 1, 0
1 = rain, 0 = no rain
- ▶ Goal: understand the distribution of the weather
- ▶ Modeling assumption: rain on each day with a fixed probability $p \in [0, 1]$
- ▶ **Bernoulli distribution with parameter p :** " $X \sim \text{Bern}(p)$ " means X is a $\{0, 1\}$ -valued random variable with $\Pr[X = 1] = p$.
- ▶ How to estimate the probability of rain?
- ▶ Which Bernoulli distribution models data best?
(a) $\text{Bern}(0.1)$ (b) $\text{Bern}(0.5)$ (c) $\text{Bern}(0.8)$ (d) $\text{Bern}(1)$

MAXIMUM LIKELIHOOD ESTIMATION FOR BERNOULLI

- Data: 1, 0, 1, 1, 0, 0, 1, 0, 1, 0

MAXIMUM LIKELIHOOD ESTIMATION FOR BERNOULLI

- ▶ Data: 1, 0, 1, 1, 0, 0, 1, 0, 1, 0
- ▶ If data from $\text{Bern}(p)$,

$$\textit{likelihood of data} = p^5(1 - p)^5$$

MAXIMUM LIKELIHOOD ESTIMATION FOR BERNOULLI

- ▶ Data: 1, 0, 1, 1, 0, 0, 1, 0, 1, 0
- ▶ If data from $\text{Bern}(p)$,

$$\text{likelihood of data} = p^5(1-p)^5$$

- ▶ **Maximum Likelihood Estimate** for p is then

$$\arg \max_{p \in [0,1]} p^5(1-p)^5.$$

MAXIMUM LIKELIHOOD ESTIMATION FOR BERNOULLI

- ▶ Data: 1, 0, 1, 1, 0, 0, 1, 0, 1, 0
- ▶ If data from $\text{Bern}(p)$,

$$\text{likelihood of data} = p^5(1-p)^5$$

- ▶ **Maximum Likelihood Estimate** for p is then

$$\arg \max_{p \in [0,1]} p^5(1-p)^5.$$

- ▶ To compute maximum, take derivative, set to 0.

Maximum at $p = 0.5$

MAXIMUM LIKELIHOOD ESTIMATION FOR BERNOULLI

- ▶ Data: 1, 0, 1, 1, 0, 0, 1, 0, 1, 0
- ▶ If data from $\text{Bern}(p)$,

$$\text{likelihood of data} = p^5(1-p)^5$$

- ▶ **Maximum Likelihood Estimate** for p is then

$$\arg \max_{p \in [0,1]} p^5(1-p)^5.$$

- ▶ To compute maximum, take derivative, set to 0.

$$\text{Maximum at } p = 0.5$$

- ▶ Generalize: data $x_1, x_2, \dots, x_n \in \{0, 1\}$
- ▶ MLE of the Bernoulli parameter p is

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

GENERAL SETUP

- ▶ Data $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n \in \mathcal{X}$

GENERAL SETUP

- ▶ Data $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n \in \mathcal{X}$
- ▶ A **model** $\mathcal{P} = \{P_\theta : \theta \in \Theta\}$ is a set of probability distributions over $\mathcal{X}$ indexed by a **parameter space** Θ .

GENERAL SETUP

- ▶ Data $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n \in \mathcal{X}$
- ▶ A **model** $\mathcal{P} = \{P_\theta : \theta \in \Theta\}$ is a set of probability distributions over $\mathcal{X}$ indexed by a **parameter space** Θ .
- ▶ Example: Bernoulli model.
 - ▶ Parameter set $\Theta = [0, 1]$
 - ▶ For any $\theta \in \Theta$, $P_\theta = \text{Bern}(\theta)$.

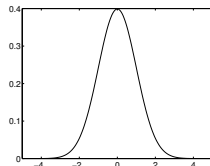
GENERAL SETUP

- ▶ Data $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n \in \mathcal{X}$
- ▶ A **model** $\mathcal{P} = \{P_\theta : \theta \in \Theta\}$ is a set of probability distributions over $\mathcal{X}$ indexed by a **parameter space** Θ .
- ▶ Example: Bernoulli model.
 - ▶ Parameter set $\Theta = [0, 1]$
 - ▶ For any $\theta \in \Theta$, $P_\theta = \text{Bern}(\theta)$.
- ▶ Also, often deal with models where each P_θ has a **density function** $p(\cdot; \theta) : \mathcal{X} \rightarrow \mathbb{R}_+$.

GENERAL SETUP

- ▶ Data $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n \in \mathcal{X}$
- ▶ A **model** $\mathcal{P} = \{P_\theta : \theta \in \Theta\}$ is a set of probability distributions over $\mathcal{X}$ indexed by a **parameter space** Θ .
- ▶ Example: Bernoulli model.
 - ▶ Parameter set $\Theta = [0, 1]$
 - ▶ For any $\theta \in \Theta$, $P_\theta = \text{Bern}(\theta)$.
- ▶ Also, often deal with models where each P_θ has a **density function** $p(\cdot; \theta) : \mathcal{X} \rightarrow \mathbb{R}_+$.
- ▶ Example: Gaussian density in one dimension ($\mathcal{X} = \mathbb{R}$)

$$\Theta = \mathbb{R} \times \mathbb{R}_{++}, \text{ parameter } \theta = (\mu, \sigma)$$
$$p(x; \mu, \sigma) := \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$



MAXIMUM LIKELIHOOD ESTIMATION

Setting

- ▶ **Given:** data $x_1, x_2, \dots, x_n \in \mathcal{X}$; parametric model $\mathcal{P} = \{P_\theta : \theta \in \Theta\}$.
- ▶ **The i.i.d. assumption:** assume $x_1, x_2, \dots, x_n$ are independent and identically distributed according to the same probability distribution.
- ▶ **Likelihood of θ given data:** (under the i.i.d. assumption)

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

the probability mass (or density) of the data, as given by P_θ .

MAXIMUM LIKELIHOOD ESTIMATION

Setting

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

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

the probability mass (or density) of the data, as given by P_θ .

Maximum likelihood estimator

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

$$\theta_{\text{ML}} := \arg \max_{\theta \in \Theta} \prod_{i=1}^n p(\mathbf{x}_i; \theta)$$

i.e., the parameter $\theta \in \Theta$ 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).

$$\arg \max_y \log(g(y)) = \arg \max_y g(y)$$

The *location* does not change.

MLE: MAXIMALITY CRITERION

Likelihood and logarithm trick

$$\begin{aligned}\theta_{\text{ML}} &= \arg \max_{\theta \in \Theta} \prod_{i=1}^n p(\mathbf{x}_i; \theta) \\ &= \arg \max_{\theta \in \Theta} \log \left(\prod_{i=1}^n p(\mathbf{x}_i; \theta) \right) \\ &= \arg \max_{\theta \in \Theta} \sum_{i=1}^n \log p(\mathbf{x}_i; \theta)\end{aligned}$$

Maximality criterion

Assuming θ is unconstrained, the log-likelihood maximizer must satisfy

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

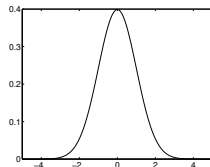
For some models, can analytically find unique solution $\theta \in \Theta$.

GAUSSIAN DISTRIBUTION

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

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

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

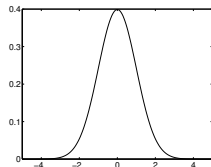


GAUSSIAN DISTRIBUTION

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

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

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



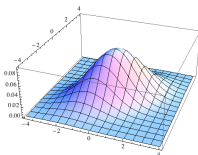
Gaussian density in d dimensions ($\mathcal{X} = \mathbb{R}^d$)

In the density function, the quadratic function

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

is replaced by a multivariate quadratic form:

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



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 Σ :

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

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

MLE equation

Solve the following equation (from the maximality criterion) for μ :

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

GAUSSIAN MLE

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

GAUSSIAN MLE

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

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

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

GAUSSIAN MLE

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

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

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

Conclusion

The maximum likelihood estimator of the Gaussian mean parameter is

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

EXAMPLE: GAUSSIAN WITH UNKNOWN COVARIANCE

Model: multivariate Gaussians

The model $\mathcal{P}$ is now

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

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

EXAMPLE: GAUSSIAN WITH UNKNOWN COVARIANCE

Model: multivariate Gaussians

The model $\mathcal{P}$ is now

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

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

ML approach

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

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

EXAMPLE: GAUSSIAN WITH UNKNOWN COVARIANCE

Model: multivariate Gaussians

The model $\mathcal{P}$ is now

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

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

ML approach

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

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

Solution

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

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

- ▶ **Maximum Likelihood Estimation:** a principled way to find the best distribution in a model: choose distribution that most likely generated the data
- ▶ MLE for Bernoulli, Gaussian can be computed in closed form.