# Statistical Machine Learning

**Christoph Lampert**

IST AUSTRIA

*Institute of Science and Technology*

Spring Semester 2015/2016 // Lecture 2

# Decision Theory (for Supervised Learning Problems)

Goal:

- Understand existing algorithms
- Develop new algorithms with specific (optimal?) properties

For this, we'll have to rely on mathematics. Forget about the implementation, data etc... for now.

## Notation

We treat all quantities of interest as *random variables*:

- input: random variable, $X$, taking values $x \in \mathcal{X}$
  (we treat $\mathcal{X}$ as if it is continuous, but discrete works analogously)
- output: random variable, $Y$, taking values and $y \in \mathcal{Y}$.
- joint probability distribution/density $p(X = x, Y = y)$.
- we write $p(x, y)$ for of $p(X = x, Y = y)$,

  $p(y|x)$ instead of $p(Y = y|X = x)$, etc.

# Classification

First first look at classification, $\mathcal{Y} = \{1, \ldots, M\}$, or $\mathcal{Y} = \{-1, +1\}$.

**Question: What's the best classifier for a fully known problem?**

# Classification

First first look at classification, $\mathcal{Y} = \{1, \ldots, M\}$, or $\mathcal{Y} = \{-1, +1\}$.

**Question: What's the best classifier for a fully known problem?**

## Definition (Generalization error)

Let $c : \mathcal{X} \to \mathcal{Y}$ be a decision rule. The *generalization error*, $\mathcal{R}$, of $c$ is the probability of $c$ making a wrong prediction, i.e.

$$\mathcal{R}(c) := \Pr_{(x,y)\sim p(x,y)} \{c(x) \neq y\}.$$

# Classification

First first look at classification, $\mathcal{Y} = \{1, \ldots, M\}$, or $\mathcal{Y} = \{-1, +1\}$.

**Question: What's the best classifier for a fully known problem?**

## Definition (Generalization error)

Let $c : \mathcal{X} \to \mathcal{Y}$ be a decision rule. The *generalization error*, $\mathcal{R}$, of $c$ is the probability of $c$ making a wrong prediction, i.e.

$$\mathcal{R}(c) := \Pr_{(x,y)\sim p(x,y)} \{c(x) \neq y\}.$$

## Definition (Bayes Classifier, Bayes Risk)

The prediction rule that minimizes the generalization error, with

$$c^* := \operatorname*{\mathbf{argmin}}_{c:\mathcal{X}\to\mathcal{Y}} \mathcal{R}(c)$$

is called **Bayes classifier**. The value $\mathcal{R}(c_{Bayes})$ is called the **Bayes risk**.

**Lemma**

*The Bayes classifier has the decision rule*

$$c(x) := \operatorname*{\mathbf{argmax}}_{y \in \mathcal{Y}} p(y|x) \qquad \text{for any } x \in \mathcal{X}.$$

Proof. We show: no classifier has lower generalization error than the Bayes classifier...

In binary classification we can write $c^*$ in closed form:

**Lemma**

*For $\mathcal{Y} = \{-1, +1\}$, the Bayes classifier is given by*

$$c^*(x) = \text{sign}\,[\log \frac{p(x, +1)}{p(x, -1)}],$$

*as well as*

$$c^*(x) = \text{sign}\,[\log \frac{p(+1|x)}{p(-1|x)}].$$

Proof: Exercise...

# Should we use $c^*$ to decide for every problem?

- $c^*$ is optimal when trying to *minimize the number of wrong decision*.
- That's often a good strategy, but not always.

## Reminder

To evaluate a learning task, we use *loss function* $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$.
$\ell(y, \bar{y})$ is the loss incurred when predicting $\bar{y}$ if the correct answer is $y$.

# Should we use $c^*$ to decide for every problem?

- $c^*$ is optimal when trying to *minimize the number of wrong decision*.
- That's often a good strategy, but not always.

## Reminder

To evaluate a learning task, we use *loss function* $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$.
$\ell(y, \bar{y})$ is the loss incurred when predicting $\bar{y}$ if the correct answer is $y$.

## Example: Doctor's dilemma

*There's a shadow on the X-ray. Should you diagnose cancer?*

$x$: X-ray image. $y \in \{\texttt{yes}, \texttt{no}\}$: cancer

- $\ell(\texttt{yes}, \texttt{yes}) = 0$ (you did your job well)
- $\ell(\texttt{yes}, \texttt{no}) = 1000$ (the cancer gets worse, the patient could die)
  $\ell(\texttt{no}, \texttt{yes}) = 1$ (the patient is upset until further test are made)
  $\ell(\texttt{no}, \texttt{no}) = 0$ (you did your job well)

Common: one outcome is rare, but has high loss if mispredicted

Instead of minimizing the error probability, minimize the *expected loss*!

## Definition

The classifier of minimal expected $\ell$-risk is given by

$$c^*_\ell(x) := \mathbf{argmin}_{y \in \mathcal{Y}} \mathbb{E}_{\bar{y} \sim p(\bar{y}|x)} \ell(\, \bar{y}, y \,).$$

## Lemma

*For $\mathcal{Y} = \{-1, +1\}$, and $\ell(y, \bar{y})$ given by the table*

| $y \setminus \bar{y}$ | $-1$ | $+1$ |
|---|---|---|
| $-1$ | $a$ | $b$ |
| $+1$ | $c$ | $d$ |

*the risk w.r.t. $\ell$ is minimized by the decision rule*

$$c^*_\ell(x) = \mathrm{sign}[\quad \log \frac{p(x,+1)}{p(x,-1)} + \log \frac{c-d}{b-a} \quad ],$$

*or equivalently*

$$c^*_\ell(x) = \mathrm{sign}[\quad \log \frac{p(+1|x)}{p(-1|x)} + \log \frac{c-d}{b-a} \quad ].$$

Proof: Exercise.

## Observation

The *generalization error* is the *risk* for 0/1-loss, i.e. $\ell(y, y') = [\![y \neq y']\!]$.

**Question: What's the best classifier for a fully known problem?**

**Question answered. We have identified the optimal classifiers!**

# Learning from Data

In the real world, $p(x, y)$ is unknown, but we have a training set $\mathcal{D}$.
There's at least 3 approaches:

## Definition

Given a training set $\mathcal{D}$, we call it

- a **generative probabilistic approach**:
  if we use $\mathcal{D}$ to build a model $\hat{p}(x, y)$ of $p(x, y)$, and then define

$$c(x) := \operatorname*{\mathbf{argmax}}_{y \in \mathcal{Y}} \hat{p}(x, y) \quad \text{or} \quad c_\ell(x) := \operatorname*{\mathbf{argmin}}_{y \in \mathcal{Y}} \mathbb{E}_{\bar{y} \sim \hat{p}(x, \bar{y})} \ell(\, \bar{y}, y \,).$$

- a **discriminative probabilistic approach**:
  if we use $\mathcal{D}$ to build a model $\hat{p}(y|x)$ of $p(y|x)$ and define

$$c(x) := \operatorname*{\mathbf{argmax}}_{y \in \mathcal{Y}} \hat{p}(y|x) \quad \text{or} \quad c_\ell(x) := \operatorname*{\mathbf{argmin}}_{y \in \mathcal{Y}} \mathbb{E}_{\bar{y} \sim \hat{p}(\bar{y}|x)} \ell(\, \bar{y}, y \,).$$

- a **decision theoretic approach**: if we use $\mathcal{D}$ to directly seach for a classifier $c$ in a hypothesis class $\mathcal{H}$.

## Setting

We are given

- a **training set** of examples $\mathcal{D} = \{(x^1, y^1), \ldots, (x^n, y^n)\}$,
  (note: rather a multi-set, elements can occur more than once)

Assumption:

- $\mathcal{D}$ are *independent and identically distributed (i.i.d.)* samples from the unknown distribution $p(x, y)$.

Shorthand notation,

- $\mathcal{D}^X := \{x^1, \ldots, x^n\}$, input part of $\mathcal{D}$ ,
- $\mathcal{D}^Y := \{y^1, \ldots, y^n\}$, output part of $\mathcal{D}$,
- $\mathcal{D}_y := \{(x^i, y^i) \in \mathcal{D} : y^i = y\}$, all examples of label $y$.

**Let's use $\mathcal{D}$ to form an estimate of $p(x, y)$.**

## Definition

There's (at least) three approaches:

- **parametric estimate**:
  - fix a model class $p(x, y; \theta)$,
  - estimate parameters $\hat{\theta}$ such that $p(x, y; \hat{\theta}) \approx p(x, y)$.
  - the size of $\theta$ is independent of how large $\mathcal{D}$ is
- **non-parametric estimate**:
  - estimate any $\hat{p}(x, y) \approx p(x, y)$
  - the number of parameters/complexity of $\hat{p}(x, y)$ can grow with $|\mathcal{D}|$
- hybrids of the two

# Generative Probabilistic Models: Multinomial

If $\mathcal{X}$ and $\mathcal{Y}$ are *finite*, we can represent any $p(x, y)$ as a table of values.

To simplify notation, we look at arbitrary $z \in \mathcal{Z}$ (think: $z = (x, y)$):

**Definition (Empirical estimate)**

Let $z^1, \ldots, z^n$ be samples from $p(z)$, then we call

$$\hat{p}_n(z) := \frac{1}{n} \sum_{i=1}^{n} [\![ z^i = z ]\!]$$

the empirical estimate of $p(z)$.

**Theorem (Convergence of the empirical estimate)**

*Let $z^1, z^2, \ldots$ be i.i.d. samples from $p(z)$. For every possible value $z \in \mathcal{Z}$*

$$\Pr\left\{ \lim_{n\to\infty} \hat{p}_n(z) \;=\; p(z) \right\} = 1.$$

**Proof.**

Every textbook on statistics: *law of large numbers* (strong version). □

# The curse of dimensionality

**Setting:**
Let $\mathcal{Z} = \mathcal{Z}_1 \times \cdots \times \mathcal{Z}_d$, i.e. data decomposes into $d$ non-trivial "features", "attributes", or "dimensions". Let $m_j := |\mathcal{Z}_j| \geq 2$ for $j = 1, \ldots, d$.

## Lemma

*The number of samples needed to estimate $\hat{p}(z)$* ***grows exponentially in*** *$d$ (unless we made additional assumptions).*

## Proof.

$\hat{p}(z)$ has $|\mathcal{Z}| = \prod_{j=1}^{d} m_j \geq 2^d$ entries. Without further assumptions, each entry can be set arbitrarily, independently, except for the one constraint that they must sum to $1$. Each sample influences only one bin, so we need at least $2^d - 1$ samples (in practice, many times that, of course). □

**Example (Dating agency table)**

| TRAINING | eyes | height | handsome | sex | soccer | date? |
|---|---|---|---|---|---|---|
| Apu | blue | tall | yes | male | no | yes |
| Bernice | brown | short | yes | female | no | no |
| ⋮ | | | | | | |
| Itchy | brown | short | no | male | yes | yes |

Could we estimate $p(x, y)$ here?

- $|\mathcal{X} \times \mathcal{Y}| = 96$, $p(x, y)$ has $95$ free parameters
- We have $9$ samples.
- **Most possible combinations we have never seen!**

**Example (Dating agency table)**

| TRAINING | eyes | height | handsome | sex | soccer | date? |
|---|---|---|---|---|---|---|
| Apu | blue | tall | yes | male | no | yes |
| Bernice | brown | short | yes | female | no | no |
| ⋮ | | | | | | |
| Itchy | brown | short | no | male | yes | yes |

Could we estimate $p(x, y)$ here?

- $|\mathcal{X} \times \mathcal{Y}| = 96$, $p(x, y)$ has $95$ free parameters
- We have $9$ samples.
- **Most possible combinations we have never seen!**

Bayes classifier from $\hat{p}(x, y)$: $\quad c(x) := \mathbf{argmax}_{y \in \mathcal{Y}} \hat{p}(x, y)$

- $\hat{p}(\text{Apu}, \texttt{yes}) = \frac{1}{9}, \quad \hat{p}(\text{Apu}, \texttt{no}) = 0, \quad \rightarrow \quad c(\text{Apu}) = \texttt{yes},$
- $\hat{p}(\text{Jimbo}, \texttt{yes}) = 0, \quad \hat{p}(\text{Jimbo}, \texttt{no}) = 0, \quad \rightarrow \quad c(\text{Jimbo}) = ???,$

No clue about previously unseen patterns $\rightarrow$ very little generalization

# Naive Bayes Model

## Definition

Let $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_d$. The *Naive Bayes (NB)* estimate of $p(x, y)$ is

$$\hat{p}_{\mathsf{NB}}(x, y) := \hat{p}(y) \prod_{j=1}^{d} \hat{p}_j(x_j|y),$$

where

- $\hat{p}(y)$ is an estimate of $p(y)$,
- $\hat{p}_j(x_j|y)$ are estimates of $p(x_j|y)$ for every $j = 1, \ldots, d$.

# Naive Bayes Model

## Definition

Let $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_d$. The *Naive Bayes (NB)* estimate of $p(x, y)$ is

$$\hat{p}_{\mathsf{NB}}(x, y) := \hat{p}(y) \prod_{j=1}^{d} \hat{p}_j(x_j|y),$$

where

- $\hat{p}(y)$ is an estimate of $p(y)$,
- $\hat{p}_j(x_j|y)$ are estimates of $p(x_j|y)$ for every $j = 1, \ldots, d$.

## Lemma

*The number of free parameters in $p_{NB}(x, y)$ grows linear with $d$.*

## Proof.

$p_{\mathsf{NB}}(x, y)$ has $|\mathcal{Y}|[1 + \sum_{j=1}^{d}(m_j - 1)] - 1$ degrees of freedom. □

# Naive Bayes Classifier

## Definition

The *Naive Bayes* classifier is given by

$$c(x) := \underset{y \in \mathcal{Y}}{\mathbf{argmax}}\, \hat{p}_{NB}(x, y)$$

A Naive Bayes classifier needs much fewer examples for 'training' than one based on a full probability table.

# Naive Bayes Classifier

## Definition

The *Naive Bayes* classifier is given by

$$c(x) := \operatorname*{\mathbf{argmax}}_{y \in \mathcal{Y}} \hat{p}_{NB}(x, y)$$

A Naive Bayes classifier needs much fewer examples for 'training' than one based on a full probability table.

## Remark

Even for $n \to \infty$, we likely won't have $\hat{p}_{\text{NB}}(x, y) \not\to p(x, y)$!

So, most likely, **the NB model is wrong** as a density estimate.
But that doesn't mean it doesn't work for making decisions!
In fact, NB is *very successful*, e.g. in Spam filtering (text classification).

# Naive Bayes Classifier

## Definition

The *Naive Bayes* classifier is given by

$$c(x) := \underset{y \in \mathcal{Y}}{\mathbf{argmax}}\, \hat{p}_{NB}(x, y)$$

A Naive Bayes classifier needs much fewer examples for 'training' than one based on a full probability table.

## Remark

Even for $n \to \infty$, we likely won't have $\hat{p}_{\text{NB}}(x, y) \not\to p(x, y)$!

So, most likely, **the NB model is wrong** as a density estimate.
But that doesn't mean it doesn't work for making decisions!
In fact, NB is *very successful*, e.g. in Spam filtering (text classification).

*"All models are wrong, but some are useful." (George E. P. Box, 1979)*

## Parametric models for finite domains

Both models we saw so far are *parametric*:

For finite $z \in \mathcal{Z}$, $p(z)$ is *multinomial* distribution:

- $|\mathcal{Z}|$ parameters: $\theta_z$ for $z \in \mathcal{Z}$ with $p(Z = z) = \theta_z$
- parameters fulfill
  - $\theta_z \geq 0$
  - $\sum_z \theta_z = 1$

Similar for Naive Bayes model:

- $\hat{p}(y)$ is multinomial for $y \in \mathcal{Y}$, parameter $\theta_y \in \mathbb{R}^{|\mathcal{Y}|}$,
  - $\hat{p}(y) = \theta_y$ with $\theta_y \geq 0$, $\sum_{y \in \mathcal{Y}} \theta_y = 1$,
- $\hat{p}(x_j|y)$ is multinomial for $x_j \in \mathcal{X}_j$, parameters $\theta^j_{x_j}$
  - $\hat{p}(x_j|y) = \theta^y_{x_j}$ with $\theta^y_{x_j} \geq 0$, $\sum_{x_j \in \mathcal{X}_j} \theta^y_{x_j} = 1$, for all $y \in \mathcal{Y}$

We set parameters as $\theta_z = \frac{1}{n} \sum_{i=1}^{n} [\![ z^i = z ]\!]$? Why?

Let $\hat{p}(z;\theta)$ be a parametric model with parameter $\theta \in \Theta$.
Let $\mathcal{D} = \{z^1, \ldots, z^n\}$ be i.i.d. samples from $p(z)$.

## Definition (Parameter estimation)

There's (at least) two main approaches to set $\theta$:

**Maximum Likelihood (ML) Estimation:**
Which parameter value makes it most likely that we observed $\mathcal{D}$?

$$\theta_{ML} = \operatorname*{\mathbf{argmax}}_{\theta\in\Theta} \; p(z^1, \ldots, z^n; \theta) \;=\; \operatorname*{\mathbf{argmax}}_{\theta\in\Theta} \prod_i p(z^i; \theta)$$

**Bayesian Parameter Estimation:**
Treat $\theta$ as a random variable itself. What's its most likely value given $\mathcal{D}$?

$$\begin{aligned}\theta_{Bayes} &= \operatorname*{\mathbf{argmax}}_{\theta\in\Theta} \; p(\theta \mid z^1, \ldots, z^n) \\ &= \operatorname*{\mathbf{argmax}}_{\theta\in\Theta} \; p(\theta)p(z^1, \ldots, z^n|\,\theta) \;=\; \operatorname*{\mathbf{argmax}}_{\theta\in\Theta} \; p(\theta)\prod_i p(z^i;\,\theta)\end{aligned}$$

where $p(\theta)$ is a *prior* distribution over the possible parameter values.

# Parameter Estimation: Blackboard

## Remark

In practice, one almost always uses the log-likelihood, which gives the same $\theta$ (since $\log$ is a monotonous function):

$$\theta_{ML} = \underset{\theta \in \Theta}{\mathbf{argmax}} \log \prod_{i=1}^{n} \hat{p}(x^i; \theta) = \underset{\theta \in \Theta}{\mathbf{argmax}} \sum_{i=1}^{n} \log \hat{p}(x^i; \theta)$$

and

$$\begin{aligned} \theta_{Bayes} &= \underset{\theta \in \Theta}{\mathbf{argmax}} \quad \log \left[ p(\theta) \prod_i p(z^i; \theta) \right] \\ &= \underset{\theta \in \Theta}{\mathbf{argmax}} \quad \log p(\theta) + \sum_i \log p(z^i; \theta) \end{aligned}$$

Example on blackboard.

# Laplace smoothing

## Definition (Laplace smoothing)

Let $z^1, \ldots, z^n$ be i.i.d. samples from $p(z)$. For $\alpha \geq 0$ we call

$$\hat{p}_n(z) := \frac{1}{n + |\mathcal{Z}|\alpha}(\alpha + \sum_{i=1}^{n} [\![z^i = z]\!]) \tag{1}$$

the *smoothed empirical estimate* of $p(z)$ (with smoothing parameter $\alpha$).

Bayesian interpretation:

- Bayesian estimate of parameters $\theta_z$ of a multinomial distribution
- Prior on $\theta$: symmetric Dirichlet distribution with parameter $\alpha$

$$p(\theta) = \frac{1}{B(\alpha)} \prod_{z=1}^{|\mathcal{Z}|} (\theta_z)^{\alpha - 1} \text{ with } B(\alpha) = \frac{\Gamma(\alpha)^{|\mathcal{Z}|}}{\Gamma(\alpha|\mathcal{Z}|)}$$

Laplace's "rule of succession": $\alpha = 1$. More common: $\alpha < 1$, e.g. $\frac{1}{2}$.

# Continuous Data

If $\mathcal{X}$ is continuous, $p(x, y)$ is a strange object, mixing continuous and discrete. Instead of modeling $p(x, y)$, we decompose it:

## Definition

Let $p(x, y) = p(x|y)p(y)$.

- $p(y)$ are called **class priors**,
- $p(x|y)$, for $y \in \mathcal{Y}$, are called **class conditional densities**.

## Remark

$p(y)$ is a discrete probability distribution over $|\mathcal{Y}|$ possible values, i.e.

- $p(y) \geq 0$ for all $y \in \mathcal{Y}$, and $\sum_y p(y) = 1$.

For any fixed $y \in \mathcal{Y}$, $p(x|y)$ is a probability density, i.e.

- $p(x|y) \geq 0$ for all $x \in \mathcal{X}$, and $\int_x p(x|y)\ \mathrm{dx} = 1$.

# Gaussian density estimation

Most popular parametric model for continuous data is **Gaussian**:

## Definition (Gaussian Density Parameter Estimation)

For $x \in \mathbb{R}^d$, let $\hat{p}(x|y;\mu,\Sigma) = \mathcal{G}(x,\mu_y,\Sigma_y)$ with

$$\mathcal{G}(x,\mu,\Sigma) = \frac{1}{\sqrt{(2\pi)^d \det \Sigma_y}} \exp(-\frac{1}{2}(x-\mu_y)^\top \Sigma_y^{-1}(x-\mu_y)).$$

Given a set $\mathcal{D} = \{(x^1,y^1),\ldots,(x^n,y^n)\}$, we estimate all $\mu_y$ and $\Sigma_y$ for $y \in \mathcal{Y}$ using the classical formulas:

$$\mu_y = \frac{1}{n_y} \sum_{\{i:y^i=y\}} x^i \qquad \Sigma_y = \frac{1}{n_y} \sum_{\{i:y^i=y\}} (x^i-\mu_y)(x^i-\mu_y)^\top \tag{2}$$

Remark: Alternatively, we can assume a fixed $\Sigma_y$ and estimate only $\mu_y$, or estimate a single $\Sigma$ for all classes, or set $\Sigma_y = \sigma_y Id$ and estimate $\sigma$, etc.

**Example (Gaussian Model of Height Distribution)**

We observe the following situation:

- $X$: height of a person in cm, $Y = \{\texttt{(male}, \texttt{female}\}$.
- $\mathcal{D} = \{(181, \texttt{m}), (165, \texttt{f}), (161, \texttt{f}), (172, \texttt{m}) , (175, \texttt{m}), (178, \texttt{f})\}$.

**Example (Gaussian Model of Height Distribution)**

We observe the following situation:

- $X$: height of a person in cm, $\quad Y = \{\texttt{(male}, \texttt{female}\}$.
- $\mathcal{D} = \{(181, \texttt{m}),\ (165, \texttt{f}),\ (161, \texttt{f}),\ (172, \texttt{m})\ ,(175, \texttt{m}),\ (178, \texttt{f})\}$.

$\mathcal{X} = \mathbb{R}^1$, so $\hat{p}(x|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp(-\frac{1}{2\sigma_y^2}(x - \mu_y)^2)$.

$$\mu_{\texttt{m}} = \frac{1}{3}(181 + 172 + 175) = 176 \qquad \sigma^2_{\texttt{m}} = \frac{1}{3}(5^2 + 4^2 + 1^2) = 14$$

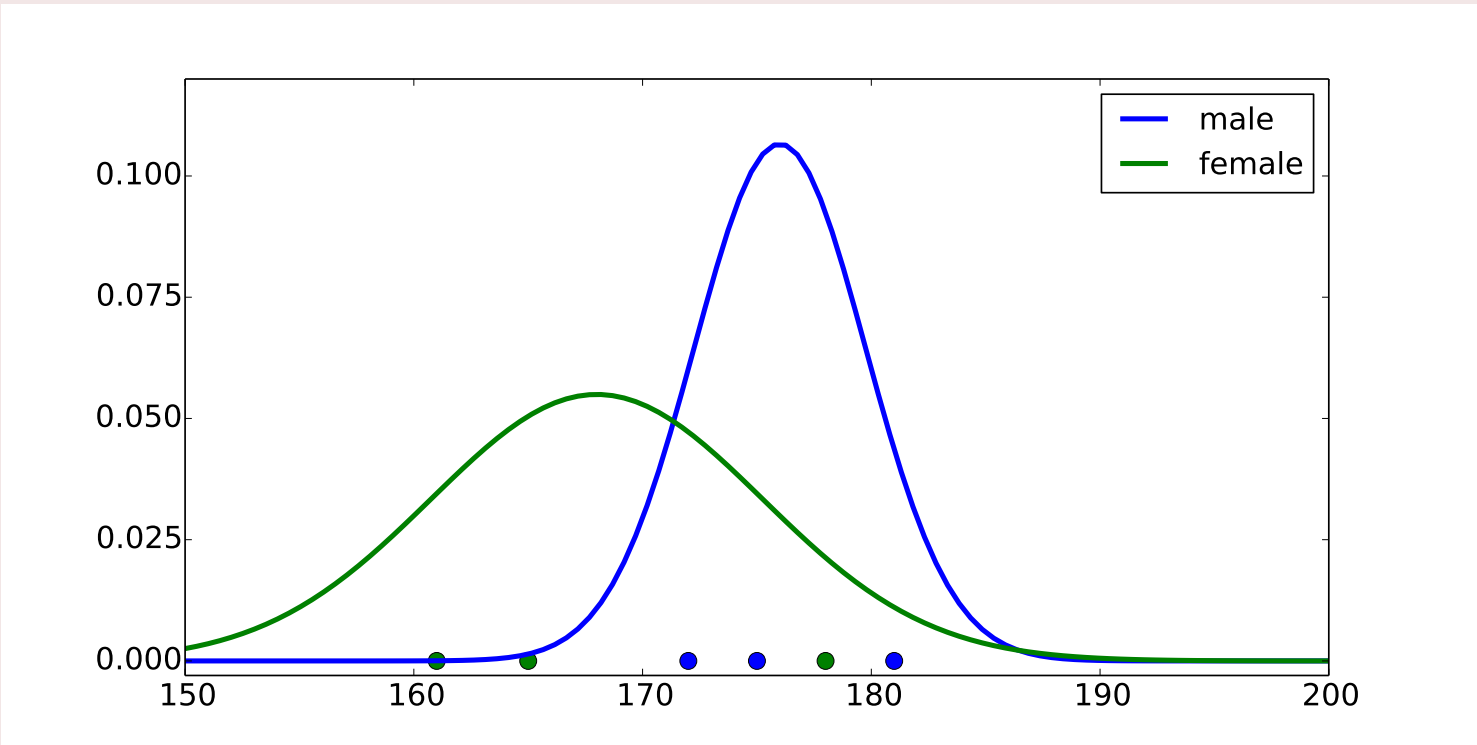
$$\mu_{\texttt{f}} = \frac{1}{3}(161 + 165 + 178) = 168 \qquad \sigma^2_{\texttt{f}} = \frac{1}{3}(7^2 + 3^2 + 10^2) \approx 52.7$$

## Example (Gaussian Model of Height Distribution)

We observe the following situation:

- $X$: height of a person in cm, $\quad Y = \{\texttt{(male, female}\}$.
- $\mathcal{D} = \{(181, \texttt{m}),\ (165, \texttt{f}),\ (161, \texttt{f}),\ (172, \texttt{m})\ ,(175, \texttt{m}),\ (178, \texttt{f})\}$.

$\mathcal{X} = \mathbb{R}^1$, so $\hat{p}(x|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp(-\frac{1}{2\sigma_y^2}(x - \mu_y)^2)$.



**Lemma**

*The classical expressions for estimating $\mu_y$ and $\Sigma_y$ for a Gaussian are the maximum likelihood estimates for the parameters of $\hat{p}(x|y;\mu,\sigma)$.*

**Lemma**

*The classical expressions for estimating $\mu_y$ and $\Sigma_y$ for a Gaussian are the maximum likelihood estimates for the parameters of $\hat{p}(x|y;\mu,\sigma)$.*

**Proof**. With $\mathcal{G}(x;\mu,\Sigma) = \frac{1}{(2\pi \det \Sigma)^{d/2}} \exp\{-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)\}$, solve $\mu_{ML} = \mathbf{argmax}_\mu \mathcal{L}(\mu)$ for $\mathcal{L}(\mu) = \log \sum_{i=1}^n \log \mathcal{G}(x^i;\mu,\Sigma)$.

$$\mathcal{L}(\mu) = \frac{1}{2}\sum_{i=1}^n (x^i-\mu)^\top \Sigma^{-1}(x^i-\mu) - \frac{d}{2}\log 2\pi - \frac{d}{2}\log\det\Sigma$$

$$\nabla_\mu L(\mu,\Sigma) = \sum_{i=1}^n \Sigma^{-1}(x^i-\mu) \;=\; \Sigma^{-1}\sum_{i=1}^n (x^i-\mu)$$

$$H_\mu L(\mu,\Sigma) = -\Sigma^{-1} \preccurlyeq 0$$

$$\mu_{ML} = \frac{1}{n}\sum_{i=1}^n x^i \;\Rightarrow\; \nabla_\mu L(\mu_{ML},\Sigma) = 0 \;\Rightarrow\; \text{maximum of } \mathcal{L}$$

$\Sigma_{ML}$ analogously, but requires some matrix derivatives.

# Classification based on Gaussian models

Let $\hat{p}(x|y;\mu_y,\Sigma_y) = \frac{1}{\sqrt{(2\pi)^d \det \Sigma_y}} \exp(-\frac{1}{2}(x-\mu_y)^\top \Sigma_y^{-1}(x-\mu_y))$.
How to make decisions?

General Bayes classifier:

$$c(x) = \mathop{\mathbf{argmax}}_{y\in\mathcal{Y}} \{ \frac{\hat{p}(y)}{\sqrt{(2\pi)^d \det \Sigma_y}} \exp(-\frac{1}{2}(x-\mu_y)^\top \Sigma_y^{-1}(x-\mu_y))\}$$

For two classes, $\mathcal{Y} = \{+1,-1\}$:

$$\begin{aligned} c(x) &= \text{sign}\,[\log \frac{p(x,+1)}{p(x,-1)}] \\ &= \text{sign}\,\Big[(x-\mu_{-1})^\top(\Sigma_{-1})^{-1}(x-\mu_{-1}) \\ &\qquad - (x-\mu_{+1})^\top(\Sigma_{+1})^{-1}(x-\mu_{+1}) - \log\frac{\det\Sigma_{+1}}{\det\Sigma_{-1}}\Big] \end{aligned}$$

# Gaussian Mixture Models (GMMs)

More flexibility by modeling each class as a **Mixture of Gaussians**

$$\hat{p}(x|y;\pi,\vec{\mu},\vec{\Sigma}) = \sum_{k=1}^{K} \pi_k \, \mathcal{G}(x;\mu_k,\Sigma_k) \quad \text{with } \pi_k \geq 0 \text{ and } \sum_{k=1}^{K} \pi_k = 1.$$

# Gaussian Mixture Models (GMMs)

More flexibility by modeling each class as a **Mixture of Gaussians**

$$\hat{p}(x|y;\pi,\vec{\mu},\vec{\Sigma}) = \sum_{k=1}^{K} \pi_k \, \mathcal{G}(x;\mu_k,\Sigma_k) \quad \text{with } \pi_k \geq 0 \text{ and } \sum_{k=1}^{K} \pi_k = 1.$$

No closed form for MLE parameters, but popular iterative algorithm:

## Expectation-Maximization (EM) algorithm for GMMs

**input** $x^1, \ldots, x^n$, $K$

init $\pi, \vec{\mu}, \vec{\Sigma}$

**repeat**

$\hat{\gamma}_{ik} = \pi_k \mathcal{G}(x^i;\mu_k,\Sigma_k), \quad \gamma_{ik} = \hat{\gamma}_{ik}/(\sum_j \hat{\gamma}_{ij})$ E-step

$\pi_k = \frac{1}{n}\sum_{i=1}^{n} \gamma_{ik}$

$\mu_k = \frac{1}{n\pi_k}\sum \gamma_{ik} x^i$ M-step(s)

$\Sigma_k = \frac{1}{n\pi_k}\sum_i \gamma_{ik}(x^i-\mu_k)(x^i-\mu_k)^\top$

**until** convergence

**output** $\pi, \vec{\mu}, \vec{\Sigma}$

**Definition**

Let $K_h(x) : \mathcal{X} \to \mathbb{R}$ be a (fixed) kernel function, where $h$ is a *bandwidth* parameter. Then

$$\hat{p}(x|y) := \frac{1}{|\{y_i = y\}|} \sum_{\{i:y_i=y\}} K_h(x - x^i)$$

is called a *kernel density estimate (KDE)* of $p(x|y)$.

Alternative name: *Parzen windows estimate*.

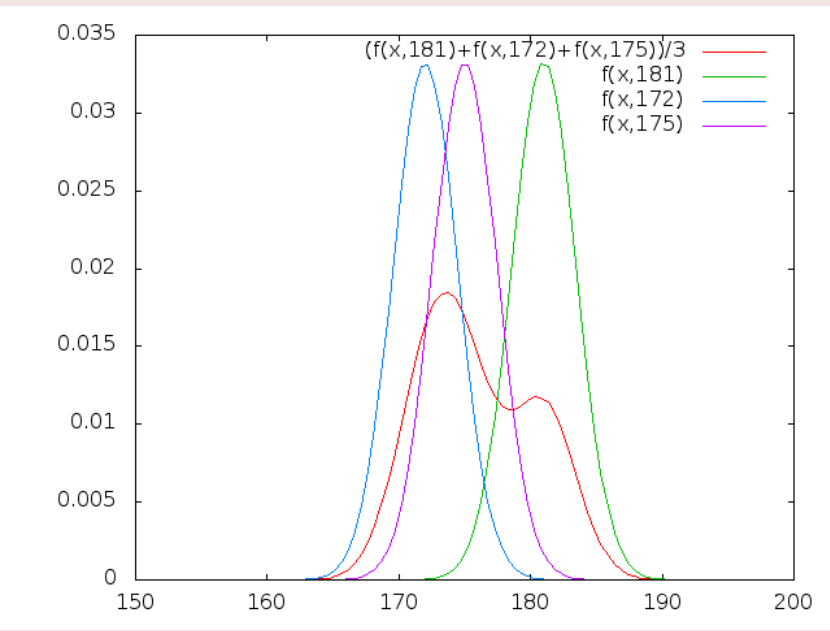
Kernel density estimates are *non-parametric*. The number of terms grows with the number of examples.

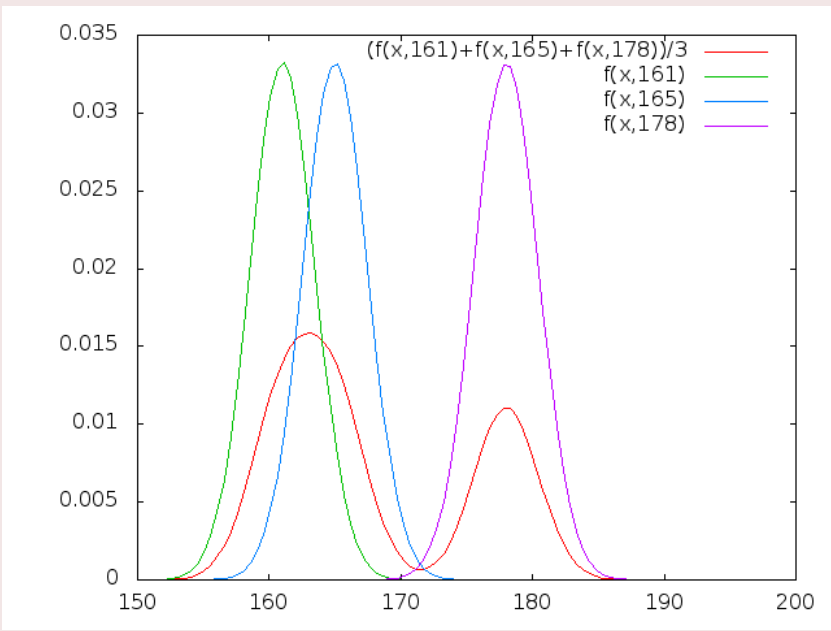
# Example: Kernel density estimate

## Example

- $X$: height of a person in cm, $Y = \{\texttt{(male, female}\}$.
- $\mathcal{D} = \{(181, \texttt{m}), (165, \texttt{f}), (161, \texttt{f}), (172, \texttt{m}), (175, \texttt{m}), (178, \texttt{f})\}$.

For $K_h(x) = \frac{1}{\sqrt{2\pi h^2}} \exp(-\frac{1}{h^2}\|x\|^2)$ (Gaussian with bandwidth $h$):



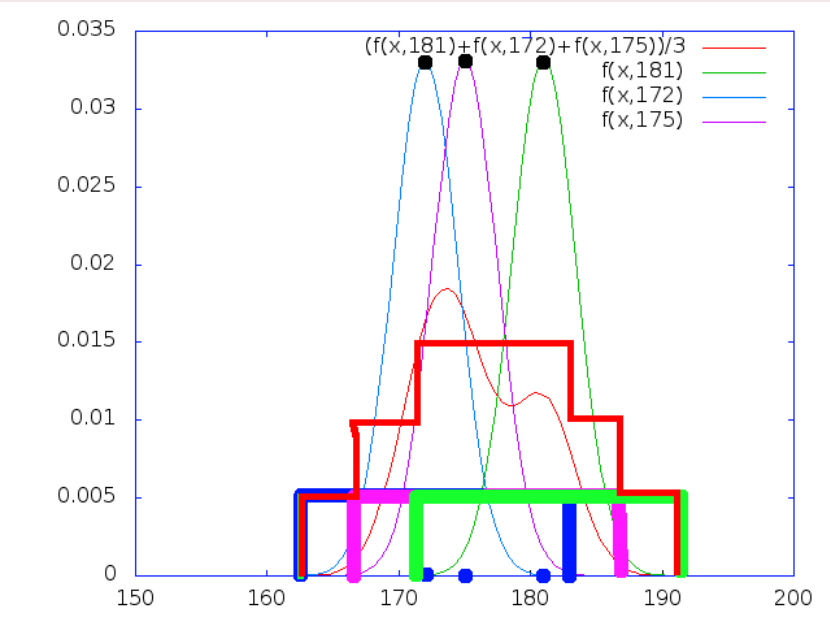



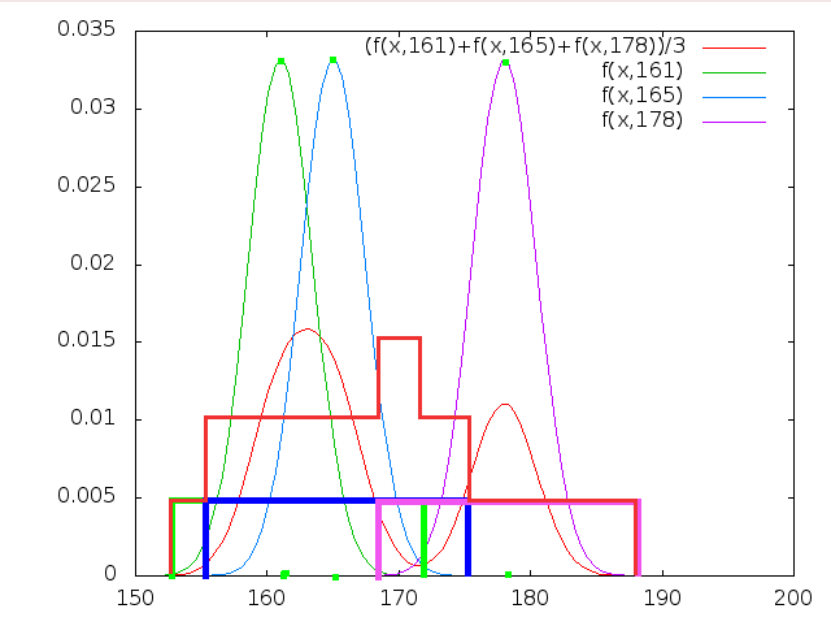
# Example: Kernel density estimate

## Example

- $X$: height of a person in cm, $Y = \{\texttt{male}, \texttt{female}\}$.
- $\mathcal{D} = \{(181, \texttt{m}), (165, \texttt{f}), (161, \texttt{f}), (172, \texttt{m}), (175, \texttt{m}), (178, \texttt{f})\}$.

For $K_h(x) = \frac{1}{2h}[\![|x| < h]\!]$ (Box kernel):



# Summary: Generative Models

For generative models, one uses the available data to estimate $p(x, y)$

- either directly, or
- through the decomposition $p(x, y) = p(x|y)p(y)$

Generative models are popular in the natural sciences because they

- model all information in the data
- reflect the data generation process

But: the suffer from **curse of dimensionality**!

- one either needs a *lot* of data,
- or, one must hae strong additional assumptions,
- or one must resort to a simple (usually wrong) model.