Machine Learning

Fabrice Rossi

CEREMADE
Université Paris Dauphine

2021

Standard programming

Solving a task with a computer

- input and output definition
- algorithm design
- implementation

Examples

- jpeg image converter
- interactive 3D world
- computational fluid dynamics
- chess program
- spam filtering (with e.g. SpamAssassin)

Standard programming

Solving a task with a computer

- input and output definition
- algorithm design
- implementation

Examples

- jpeg image converter
- interactive 3D world
- computational fluid dynamics
- chess program
- spam filtering (with e.g. SpamAssassin)

100% human design

SpamAssassin

```
Return-Path: <lojhsouek@alite.ae>
Date: Thu, 5 Dec 2019 16:24:57 +0800 (CST)
From: jitao25526187006 <jitao25526187006@126.com>
Sender: lojhsouek <lojhsouek@alite.ae>
To: "fabrice.rossi" <fabrice.rossi@apiacoa.org>
Subject: Re: Newer dealers are cheaper due to the expansion of the
        new projector lamp
MIME-Version: 1.0
Content-Type: text/html; charset=UTF-8
Content-Transfer-Encoding: quoted-printable

<p style=3D"margin-top:2px;margin-bottom:2px;">This is the Selina of the HA=
ITTON projector lamp.</p><p style=3D"margin-top:2px;margin-bottom:2px;">goo=
d luck!</p><p style=3D"margin-top:2px;margin-bottom:2px;">Thank you for tak=
ing the time to read my email.</p><p style=3D"margin-top:2px;margin-bottom:=
2px;">If you need to buy a projector lamp , I look forward to your reply.</=
p><p style=3D"margin-top:2px;margin-bottom:2px;">We are the second professi=
onally produced projector lamp in China with 10 years of manufacturing expe=
rience,  </p><p style=3D"margin-top:2px;margin-bottom:2px;">Burner=E2=
=80=99s quality is well,Longer service life.</p><p style=3D"margin-top:2px;=
margin-bottom:2px;"><br/></p><p style=3D"margin-top:2px;margin-bottom:2px;"=
>I have known your company for a long time, even though we have not really =
cooperated.</p><p style=3D"margin-top:2px;margin-bottom:2px;">Maybe you hav=
e a fixed supplier now, it doesn&#39;t matter, if you need it, we will be y=
our second choice.</p><p style=3D"margin-top:2px;margin-bottom:2px;">In ord=
er to promote market expansion, we hope to offer new distributors a more fa=
vorable price than last year.</p><p style=3D"margin-top:2px;margin-bottom:2=
px;">Are you interested in creating brilliance with us?</p><p style=3D"marg=
in-top:2px;margin-bottom:2px;">For more information, please reply to the em=
ail for more information.</p><p style=3D"margin-top:2px;margin-bottom:2px;"=
><br/></p><p style=3D"margin-top:2px;margin-bottom:2px;">If you have any ot=
her questions, please feel free to contact me.</p><p style=3D"margin-top:2p=
x;margin-bottom:2px;">wish you a happy life!</p><p style=3D"margin-top:2px;=
margin-bottom:2px;">Selina Lau</p><p style=3D"margin-top:2px;margin-bottom:=
2px;"><br/></p><p style=3D"margin-top: 2px; margin-bottom: 2px;"><br/></p>
```

SpamAssassin

```
Return-Path: <lojhsouek@alite.ae>
Delivered-To: fabrice.rossi@apiacoa.org
X-Spam-Flag: YES
X-Spam-Score: 6.297
X-Spam-Level: ******
X-Spam-Status: Yes, score=6.297 tagged_above=-15 required=4.9
              tests=[BAYES_50=0.8, FREEMAIL_FORGED_FROMDOMAIN=0.001,
              FREEMAIL_FROM=0.001, FROM_LOCAL_DIGITS=0.001, FROM_LOCAL_HEX=0.006,
              HEADER_FROM_DIFFERENT_DOMAINS=0.249, HTML_MESSAGE=0.001,
              HTML_MIME_NO_HTML_TAG=0.377, MIME_HTML_ONLY=0.723, RCVD_IN_PBL=3.335,
              RDNS_NONE=0.793, T_SPF_PERMERROR=0.01] autolearn=no autolearn_force=no
Date: Thu, 5 Dec 2019 16:24:57 +0800 (CST)
From: jitao25526187006 <jitao25526187006@126.com>
Sender: lojhsouek <lojhsouek@alite.ae>
To: "fabrice.rossi" <fabrice.rossi@apiacoa.org>
Subject: ***SPAM*** Re: Newer dealers are cheaper due to the expansion of the
        new projector lamp
MIME-Version: 1.0
Content-Type: text/html; charset=UTF-8
Content-Transfer-Encoding: quoted-printable

<p style=3D"margin-top:2px;margin-bottom:2px;">This is the Selina of the HA=
ITTON projector lamp.</p><p style=3D"margin-top:2px;margin-bottom:2px;">goo=
d luck!</p><p style=3D"margin-top:2px;margin-bottom:2px;">Thank you for tak=
ing the time to read my email.</p><p style=3D"margin-top:2px;margin-bottom:=
2px;">If you need to buy a projector lamp , I look forward to your reply.</=
p><p style=3D"margin-top:2px;margin-bottom:2px;">We are the second professi=
onally produced projector lamp in China with 10 years of manufacturing expe=
rience,  </p><p style=3D"margin-top:2px;margin-bottom:2px;">Burner=E2=
=80=99s quality is well,Longer service life.</p><p style=3D"margin-top:2px;=
margin-bottom:2px;"><br/></p><p style=3D"margin-top:2px;margin-bottom:2px;"=
>I have known your company for a long time, even though we have not really =
cooperated.</p><p style=3D"margin-top:2px;margin-bottom:2px;">Maybe you hav=
e a fixed supplier now, it doesn&#39;t matter, if you need it, we will be y=
our second choice.</p><p style=3D"margin-top:2px;margin-bottom:2px;">In ord=
er to promote market expansion, we hope to offer new distributors a more fa=
```

Rule examples

```
# PBL is the Policy Block List: https://www.spamhaus.org/pbl/
header RCVD_IN_PBL              eval:check_rbl('zen-lastexternal', 'zen.spamhaus.org.', '^127\.0\.0\.1[01]$')
describe RCVD_IN_PBL            Received via a relay in Spamhaus PBL
tflags RCVD_IN_PBL              net
reuse  RCVD_IN_PBL

meta RDNS_NONE      (__RDNS_NONE && !__CGATE_RCVD && !__DOMINO_RCVD)
describe RDNS_NONE  Delivered to internal network by a host with no rDNS

meta HTML_MIME_NO_HTML_TAG        MIME_HTML_ONLY && !__TAG_EXISTS_HTML
describe HTML_MIME_NO_HTML_TAG        HTML-only message, but there is no HTML tag

body DRUG_DOSAGE       m{[\d\.]+ *\$? *(?:[\\/]|per) *d.?o.?s.?e}i
describe DRUG_DOSAGE   Talks about price per dose

# jm: keep this case-sensitive, otherwise it FP's
body DRUG_ED_CAPS      /\b(?:CIALIS|LEVITRA|VIAGRA)/
describe DRUG_ED_CAPS  Mentions an E.D. drug


body DRUG_ED_SILD      /\bsildenafil\b/i
describe DRUG_ED_SILD  Talks about an E.D. drug using its chemical name

body DRUG_ED_GENERIC      /\bGeneric Viagra\b/
describe DRUG_ED_GENERIC Mentions Generic Viagra

body DRUG_ED_ONLINE      /\bviagra .{0,25}(?:express|online|overnight)/i
describe DRUG_ED_ONLINE  Fast Viagra Delivery
```

100% human design

Machine learning

Machine programming

- climbing one step in abstraction
- designing programs with a program
- output: a program that solves a task
- input?

Machine learning

Machine programming

- climbing one step in abstraction
- designing programs with a program
- output: a program that solves a task
- input?

Learning from examples

- input: a *learning* set of pairs (input, output)
- output: a program g
- if $(\mathbf{x}, \mathbf{y})$ is in the set, g should output $\mathbf{y}$ if given $\mathbf{x}$ as input (i.e., $g(\mathbf{x}) = \mathbf{y}$)
- this is called supervised learning
- example: produce SpamAssassin using tagged emails!

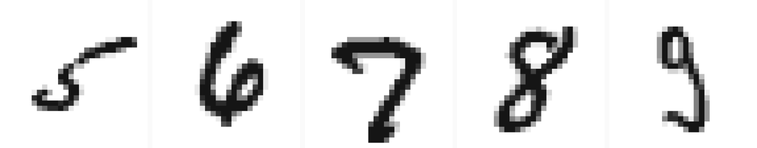
Example

MNIST data set

Example

X 

Y ? ? ? ? ?

X 

Y ? ? ? ? ?

MNIST data set

Examples

Pattern recognition

- finding specific patterns in signals
- "low" level
  - character recognition
  - whole image categorization
  - spoken word recognition
- higher level
  - licence plate recognition
  - face detection and recognition
  - speech analysis

Entity assessment

- scoring (a.k.a. probability estimation)
  - credit scoring
  - ad click
- pricing
  - market price for second hand objects
  - financial instrument pricing
- embedding
  - representing a non numerical entity by some numerical quantities
  - e.g. words by vectors

Minimal formal model

Data spaces

- $\mathcal{X}$: input space ($\mathbf{X} \in \mathcal{X}$)
- $\mathcal{Y}$: output space ($\mathbf{Y} \in \mathcal{Y}$)
  - if $|\mathcal{Y}| < \infty$: classification (and a given $\mathbf{Y}$ is a label or a class label)
  - if $\mathcal{Y} = \mathbb{R}$: regression

Programs

- mathematical version: a function g from $\mathcal{X}$ to $\mathcal{Y}$
- running the program: $g(\mathbf{X}) = \mathbf{Y}$
- preferred term: model

Machine learning program

- input: a data set $\mathcal{D} = \{(\mathbf{X}_i, \mathbf{Y}_i)\}_{1 \leq i \leq N}$ (or a data sequence)
- output: a function from $\mathcal{X}$ to $\mathcal{Y}$ (a model)

Is machine learning feasible?

Is machine learning feasible?

Yes!

Is machine learning feasible?

Yes!

- strong theoretical guarantees
- Probably Approximately Correct (PAC) framework
  - we look for g (the task solving program/model) in some class of models
  - is the class is not too complex, g can be recovered approximately with high probability given a reasonable amount (N) of input data
- Asymptotic framework
  - in addition the class of models can grow in complexity with the data size
  - then the best possible g (no restriction) can be reached asymptotically (almost surely in some cases): this is a consistency property

Is machine learning feasible?

Yes!

- strong theoretical guarantees
- Probably Approximately Correct (PAC) framework
  - we look for g (the task solving program/model) in some class of models
  - is the class is not too complex, g can be recovered approximately with high probability given a reasonable amount (N) of input data
- Asymptotic framework
  - in addition the class of models can grow in complexity with the data size
  - then the best possible g (no restriction) can be reached asymptotically (almost surely in some cases): this is a consistency property

How?

Neighbors

Additional hypothesis

- $\mathcal{X}$ is equipped with a dissimilarity d
- d is a dissimilarity on $\mathcal{X}$ iff:
  1. d is a function from $\mathcal{X} \times \mathcal{X}$ to $\mathbb{R}^+$
  2. $\forall \mathbf{X}, \mathbf{X}'$, $d(\mathbf{X}, \mathbf{X}') = d(\mathbf{X}', \mathbf{X})$
  3. $\forall \mathbf{X}, \mathbf{X}'$, $\mathbf{X} \neq \mathbf{X}' \Leftrightarrow d(\mathbf{X}, \mathbf{X}') > 0$

Neighbors

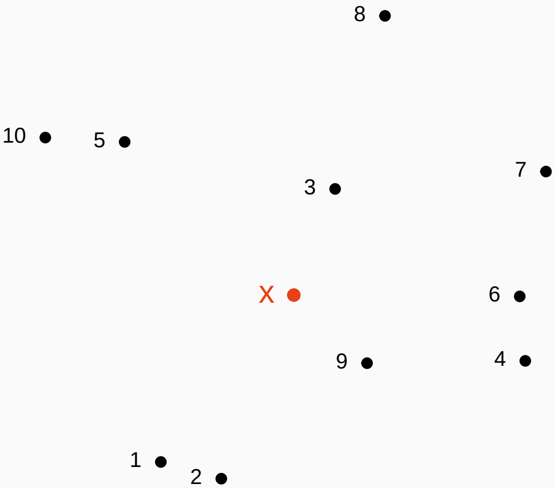
- $\mathcal{D} = ((\mathbf{X}_i, \mathbf{Y}_i))_{1 \leq i \leq N}$
- nn is the function from $\mathcal{X} \times \{1, \ldots, N\}$ to $\{1, \ldots, N\}$ such that

$$d(\mathbf{x}, \mathbf{X}_{nn(\mathbf{x},1)}) \leq d(\mathbf{x}, \mathbf{X}_{nn(\mathbf{x},2)}) \leq \cdots \leq d(\mathbf{x}, \mathbf{X}_{nn(\mathbf{x},N)}) \quad (1)$$

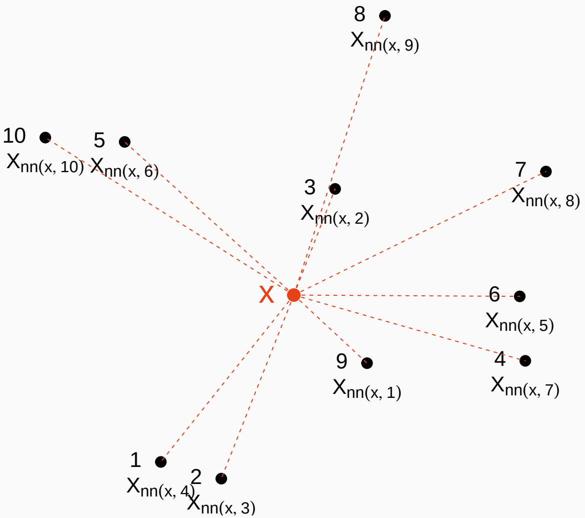
$$\text{if } d(\mathbf{x}, \mathbf{X}_{nn(\mathbf{x},k)}) = d(\mathbf{x}, \mathbf{X}_{nn(\mathbf{x},k+1)}) \text{ then } nn(\mathbf{x}, k) < nn(\mathbf{x}, k+1) \quad (2)$$

- denoted $nn_{\mathcal{D}}$ if needed

nn function

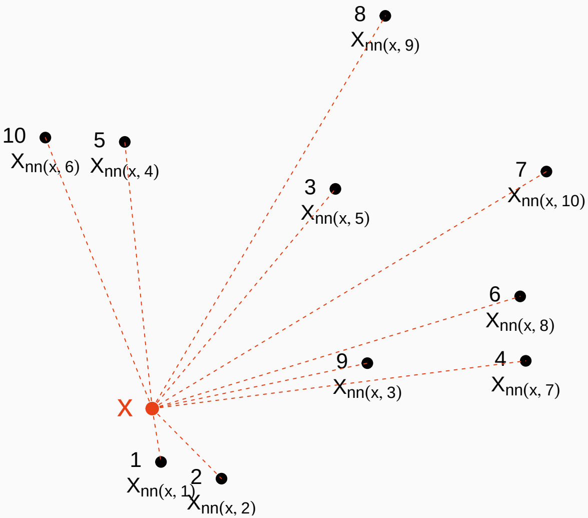


nn function



8
$X_{nn(x, 9)}$
10
$X_{nn(x, 10)}$
5
$X_{nn(x, 6)}$
3
$X_{nn(x, 2)}$
7
$X_{nn(x, 8)}$
X
6
$X_{nn(x, 5)}$
9
$X_{nn(x, 1)}$
4
$X_{nn(x, 7)}$
1
$X_{nn(x, 4)}$
2
$X_{nn(x, 3)}$

nn function


8
$X_{nn(x, 9)}$
10
$X_{nn(x, 6)}$
5
$X_{nn(x, 4)}$
3
$X_{nn(x, 5)}$
7
$X_{nn(x, 10)}$
6
$X_{nn(x, 8)}$
9
$X_{nn(x, 3)}$
4
$X_{nn(x, 7)}$
x
1
$X_{nn(x, 1)}$
2
$X_{nn(x, 2)}$

K nearest neighbors

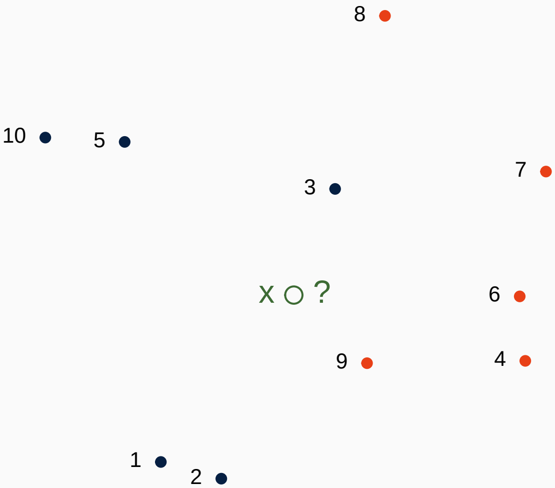
Finite output space

- when $|\mathcal{Y}| < \infty$
- given as input the data set $\mathcal{D} = ((\mathbf{X}_i, \mathbf{Y}_i))_{1 \leq i \leq N}$ and the parameter K, the K nearest neighbors (K-nn) machine learning program outputs g_{K-nn} defined by

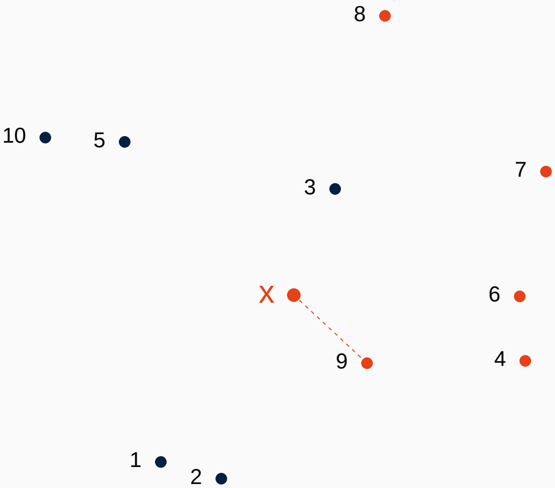
$$g_{K-nn}(\mathbf{x}) = \arg\max_{\mathbf{y} \in \mathcal{Y}} \left| \left\{ k \in \{1, \ldots, K\} \,|\, \mathbf{Y}_{nn_{\mathcal{D}}(\mathbf{x},k)} = \mathbf{y} \right\} \right|$$

- in simple terms: $g_{K-nn}(\mathbf{x})$ is the most common value of $\mathbf{y}$ in the examples that are the K closest ones to $\mathbf{x}$

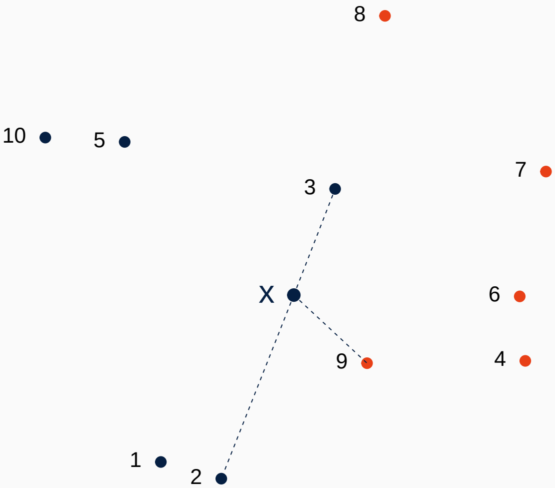
$g(x)$?



$g_{1-nn}(x)$



$g_{3-nn}(x)$



K nearest neighbors

Finite output space

- when $|\mathcal{Y}| < \infty$
- given as input the data set $\mathcal{D} = ((\mathbf{X}_i, \mathbf{Y}_i))_{1\leq i\leq N}$ and the parameter K, the K nearest neighbors (K-nn) machine learning program outputs g_{K-nn} defined by

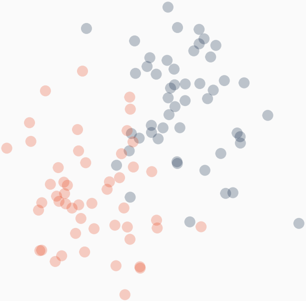
$$g_{K-nn}(\mathbf{x}) = \arg\max_{\mathbf{y}\in\mathcal{Y}} \left|\left\{k \in \{1,\ldots,K\} \,|\mathbf{Y}_{nn_{\mathcal{D}}(\mathbf{x},k)} = \mathbf{y}\right\}\right|$$

- in simple terms: $g_{K-nn}(\mathbf{x})$ is the most common value of $\mathbf{y}$ in the examples that are the K closest ones to $\mathbf{x}$

Consistency

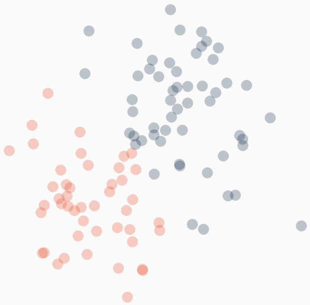
- if $\mathcal{X}$ is a finite dimensional Banach space and $|\mathcal{Y}| = 2$
- then the K-nn method is consistent if K depends on N, K_N, in such a way that $K_N \to \infty$ and $\frac{K_N}{N} \to 0$

Example



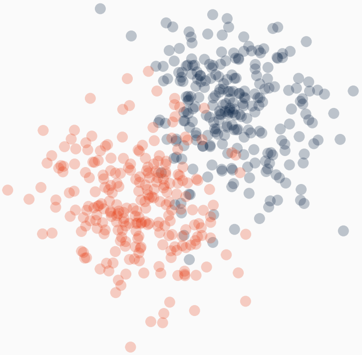
N=100

Example



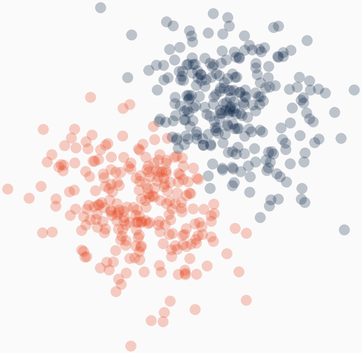
N=100, K=11

Example



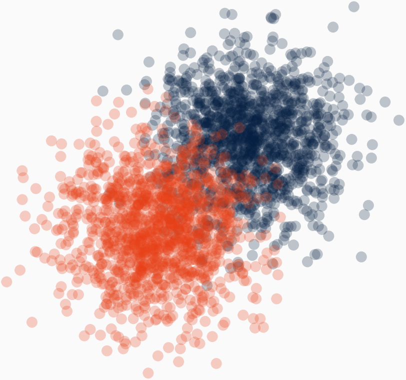
N=500

Example



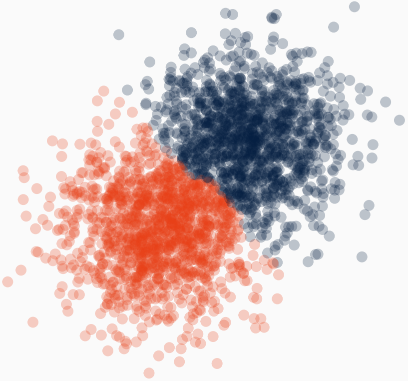
N=500, K=23

Example



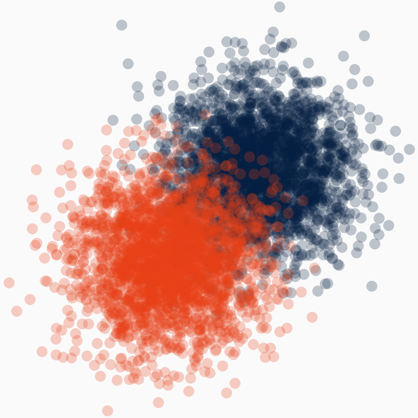
N=2500

Example



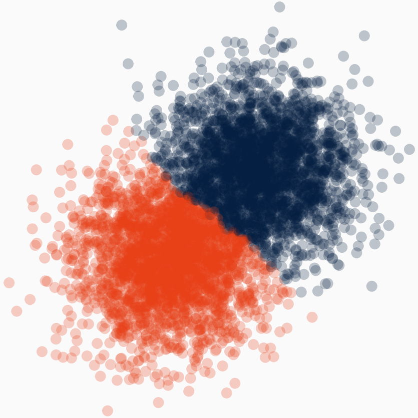
N=2500, K=51

Example



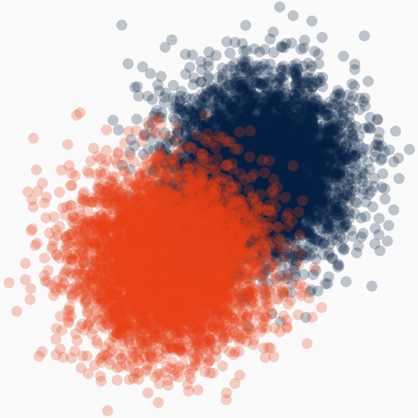
N=5000

Example



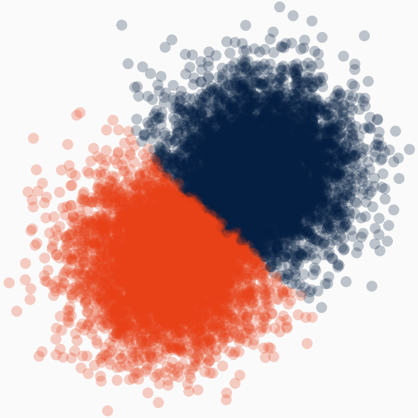
N=5000, K=71

Example



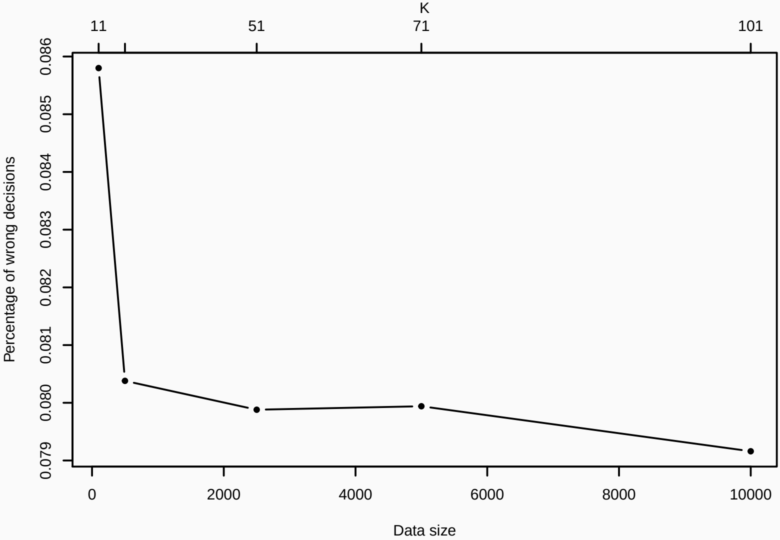
N=10000

Example



N=10000, K=101

Example


K
11
51
71
101
0.086
0.085
0.084
0.083
0.082
0.081
0.080
0.079
Percentage of wrong decisions
0
2000
4000
6000
8000
10000
Data size

In practice?

Algorithm choice

- numerous ML algorithms
- with parameters (e.g., K for the K-nn method)
- How to chose the “best” model?

Efficiency

- computational efficiency
- data efficiency

And many other issues...

Machine learning vs AI

Artificial Intelligence

Artificial Intelligence (AI) is intelligence displayed by machines, in contrast with the natural intelligence (NI) displayed by humans and other animals.

Wikipedia AI page

Machine learning

- is about learning:
  - this is only a small part of intelligence!
  - a data set is needed: it is produced by humans (limited autonomy)
- ML is only a tool that might be useful (in the future!) to build real AI
- beware of syllogisms: what *can* be solved with human intelligence does not always *need* intelligence to be solved

Outline

Introduction

Loss and risk

Judging a ML algorithm

Supervised learning

- input: $\mathcal{D} = \{(\mathbf{X}_i, \mathbf{Y}_i)\}_{1 \leq i \leq N}$
- output: $g : \mathcal{X} \to \mathcal{Y}$
- "ideally" we would like that $\forall i, g(\mathbf{X}_i) = \mathbf{Y}_i$

Weakening the goal

- $\forall i, g(\mathbf{X}_i) = \mathbf{Y}_i$ is too strong
  - limited knowledge
  - intrinsic randomness
- approximate answers, i.e. $\forall i, g(\mathbf{X}_i) \simeq \mathbf{Y}_i$

Quality of an output

Loss function

A loss function l is

- a function from $\mathcal{Y} \times \mathcal{Y}$ to $\mathbb{R}^+$
- such that $\forall \mathbf{Y} \in \mathcal{Y}, \quad l(\mathbf{Y}, \mathbf{Y}) = 0$

Interpretation

$l(g(\mathbf{X}), \mathbf{Y})$ measures the loss incurred by the user of a model g when the true value $\mathbf{Y}$ is replaced by the value $g(\mathbf{X})$.

Weakening the goal

- $\forall i, g(\mathbf{X}_i) = \mathbf{Y}_i$ is replaced by
- $\forall i, l(g(\mathbf{X}_i), \mathbf{Y}_i)$ should be as small as possible

Examples

$\mathcal{Y} = \mathbb{R}$ (Regression)

- $l_2(p, t) = (p - t)^2$
- $l_1(p, t) = |p - t|$
- $l_{APE}(p, t) = \frac{|p-t|}{|t|}$

$|\mathcal{Y}| < \infty$ (Classification)

- $l_b(p, t) = \mathbf{1}_{p \neq t}$
- general case when $\mathcal{Y} = \{y_1, y_2\}$

$l(p, t)$	$t = y_1$	$t = y_2$
$p = y_1$	0	$l(y_1, y_2)$
$p = y_2$	$l(y_2, y_1)$	0

asymmetric costs are important in practice (think SPAM versus non SPAM)

Quality of a model

From local to global

- loss functions work at a local scale: what happens for *one* input
- we need a *global* assessment of a model g: how will the model behave if deployed?

Quality of a model

From local to global

- loss functions work at a local scale: what happens for *one* input
- we need a *global* assessment of a model g: how will the model behave if deployed?
- expected loss

Quality of a model

From local to global

- loss functions work at a local scale: what happens for *one* input
- we need a *global* assessment of a model g: how will the model behave if deployed?
- expected loss

Empirical risk

- simple aggregation of local losses: average loss
- the empirical risk of a model g on a data set $\mathcal{D} = \{(\mathbf{X}_i, \mathbf{Y}_i)\}_{1 \leq i \leq N}$ for a loss function l is

$$\widehat{R}_l(g, \mathcal{D}) = \frac{1}{N} \sum_{i=1}^{N} l(g(\mathbf{X}_i), \mathbf{Y}_i) = \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x},\mathbf{y}) \in \mathcal{D}} l(g(\mathbf{x}), \mathbf{y})$$

- a good model has a low empirical risk

Data science intermezzo

Loss functions

- should not be chosen lightly
- have complex consequences, for instance
  - asymmetric losses can be seen as example weighting
  - APE loss induces underestimation
- a good data scientist knows how to translate objectives into loss functions

Empirical risk

- is only an average: does not rule out extreme behavior
- in particular, the actual loss can strongly vary with the "location" of $\mathbf{x}$ in $\mathcal{X}$
- reporting only the empirical risk is not sufficient!

Confusion matrix and co.

Confusion matrix

- when $\mathcal{Y}$ is finite, one should report a confusion matrix $\widehat{C}(\mathcal{D})$ with entries

$$\widehat{C}_{\mathbf{y},\mathbf{y}'} = |\{i \in \{1,\ldots,N\} \mid g(\mathbf{X}_i) = \mathbf{y} \text{ and } \mathbf{Y}_i = \mathbf{y}'\}|$$

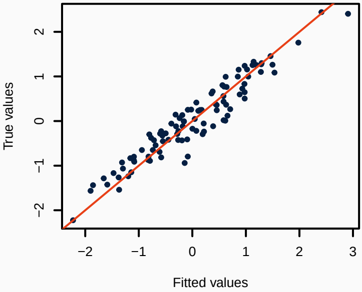
- $\widehat{C}_{\mathbf{y},\mathbf{y}'}$ counts the number of times g outputs $\mathbf{y}$ while it should output $\mathbf{y}'$
- transposed conventions have been used

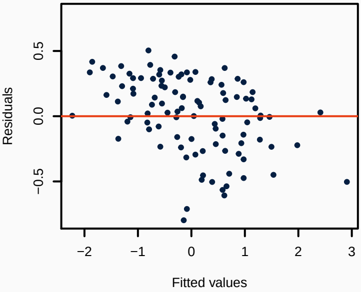
Positive and negative

- when $\mathcal{Y} = \{-1, 1\}$
- true positive: $g(\mathbf{X}_i) = \mathbf{Y}_i = 1$
- false negative: $g(\mathbf{X}_i) = -\mathbf{Y}_i = -1$
- etc.

Diagnostic plots

- when $\mathcal{Y} = \mathbb{R}$
- originally for linear regression (standard statistical model)
- some are useful for general regression models:
  - scatter plot of $\mathbf{Y}_i$ as a function of $g(\mathbf{X}_i)$
  - scatter plot of $\mathbf{Y}_i - g(\mathbf{X}_i)$ as a function of $g(\mathbf{X}_i)$ (residual plot)





Generalization

New data

- assume g is such as $\widehat{R}_l(g, \mathcal{D})$ is small
- what can we expect on a new data set $\widehat{R}_l(g, \mathcal{D}')$?
- generalization performances
- if g is learned on $\mathcal{D}$ and $\widehat{R}_l(g, \mathcal{D}) \ll \widehat{R}_l(g, \mathcal{D}')$, g is overfitting

Generalization

New data

- assume g is such as $\widehat{R}_l(g, \mathcal{D})$ is small
- what can we expect on a new data set $\widehat{R}_l(g, \mathcal{D}')$?
- generalization performances
- if g is learned on $\mathcal{D}$ and $\widehat{R}_l(g, \mathcal{D}) \ll \widehat{R}_l(g, \mathcal{D}')$, g is overfitting

Mathematical model

- stationary behavior: $\mathcal{D} \simeq \mathcal{D}'$
- hypotheses:
  - observations are random variables with values in $\mathcal{X} \times \mathcal{Y}$
  - they are distributed according to a fixed and unknown distribution D
  - observations are independent

Risk

Data set revisited

- $\mathcal{D} = ((\mathbf{X}_i, \mathbf{Y}_i))_{1 \leq i \leq N}$
- $(\mathbf{X}_i, \mathbf{Y}_i) \sim D$
- $\mathcal{D} \sim D^N$ (product distribution)

Risk of a model

- The risk of g for the loss function l is

$$R_l(g) = \mathbb{E}_{(\mathbf{X},\mathbf{Y}) \sim D}(l(g(\mathbf{X}), \mathbf{Y}))$$

- we should write $R_l(g, D)$
- the empirical risk $\widehat{R}_l(g, \mathcal{D})$ is a random variable
- if g is fixed or independent from $\mathcal{D}$, then $\widehat{R}_l(g, \mathcal{D}) \xrightarrow[|\mathcal{D}| \to \infty]{a.s.} R_l(g)$ (strong law of large numbers)

Formal definition of ML

Supervised learning

- input: a data set $\mathcal{D} = ((\mathbf{X}_i, \mathbf{Y}_i))_{1\leq i\leq N}$ with $\mathcal{D} \sim D^N$ and a loss function l
- output: a function $g_{\mathcal{D}} : \mathcal{X} \to \mathcal{Y}$
- goal: ensure that $R_l(g_{\mathcal{D}})$ is as small as possible
- best risk

$$R_l^* = \inf_{g:\mathcal{X}\to\mathcal{Y}} R_l(g)$$

Consistency

a machine learning algorithm is universally (i.e. for any D)

- consistent if $\mathbb{E}_{\mathcal{D}\sim D^N}(R_l(g_{\mathcal{D}})) \xrightarrow[N\to\infty]{} R_l^*$
- strongly consistent if $R_l(g_{\mathcal{D}}) \xrightarrow[N\to\infty]{a.s.} R_l^*$

ML and statistical models

Statistical models

- specification for D (in general parametric)
- estimation with maximum likelihood
- numerous variants (especially Bayesian approaches)

Very different philosophies

- Machine Learning
  - performance oriented
  - universal consistency
  - limited post learning tuning, exploration, interpretation, etc.
- Statistical models
  - strong hypotheses
  - bad behavior under wrong specification
  - very rich framework for post estimation exploitation

ML and statistical models

Statistical models

- specification for D (in general parametric)
- estimation with maximum likelihood
- numerous variants (especially Bayesian approaches)

Very different philosophies

- Machine Learning
  - performance oriented
  - universal consistency
  - limited post learning tuning, exploration, interpretation, etc.
- Statistical models
  - strong hypotheses
  - bad behavior under wrong specification
  - very rich framework for post estimation exploitation
- but many links!

Empirical risk minimization

A simple idea

- $R_l(g)$ cannot be computed as D is unknown
- but if $g \perp\!\!\!\perp \mathcal{D}$, $\widehat{R}_l(g, \mathcal{D}) \xrightarrow[|\mathcal{D}|\rightarrow\infty]{a.s.} R_l(g)$
- let's replace $R_l(g)$ by $\widehat{R}_l(g, \mathcal{D})$!

ERM algorithm

- choose a class of functions $\mathcal{G}$ from $\mathcal{X}$ to $\mathcal{Y}$
- define
$$g_{ERM,\mathcal{D}} = \arg\min_{g \in \mathcal{G}} \widehat{R}_l(g, \mathcal{D})$$
- machine learning as an optimization problem

Empirical risk and learning

Empirical risk can be misleading

- for the 1-nn, in general:

$$\widehat{R}_l(g_{1-nn}, \mathcal{D}) = 0$$

  if g_{1-nn} has been constructed on $\mathcal{D}$
- indeed if all the $\mathbf{X}_i$ are distinct, $nn(\mathbf{X}_i, 1) = i$ and thus $g_{1-nn}(\mathbf{X}_i) = \mathbf{Y}_i$
- unrealistic value for $R_l(g_{1-nn})$

Empirical risk and learning

Empirical risk can be misleading

- for the 1-nn, in general:

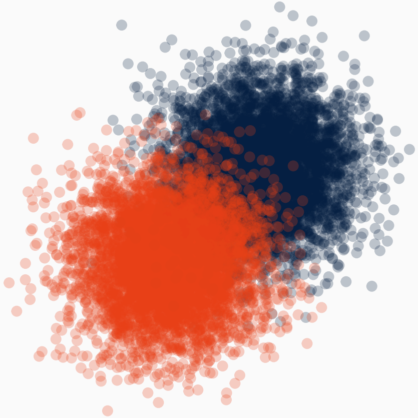
$$\widehat{R}_l(g_{1-nn}, \mathcal{D}) = 0$$

  if g_{1-nn} has been constructed on $\mathcal{D}$
- indeed if all the $\mathbf{X}_i$ are distinct, $nn(\mathbf{X}_i, 1) = i$ and thus $g_{1-nn}(\mathbf{X}_i) = \mathbf{Y}_i$
- unrealistic value for $R_l(g_{1-nn})$

Source of the problem

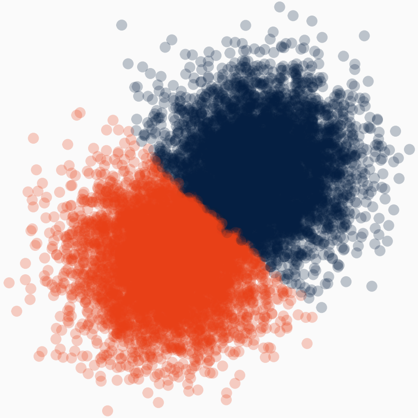
- the strong law of large numbers applies when the random variables are independent
- the $(l(g_{\mathcal{D}}(\mathbf{X}_i), \mathbf{Y}_i))_{1\leq i\leq N}$ are dependent variables!

Example



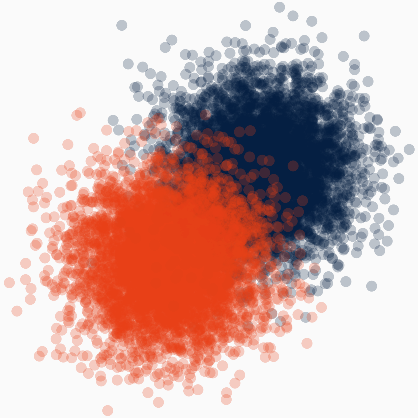
N=10000

Example



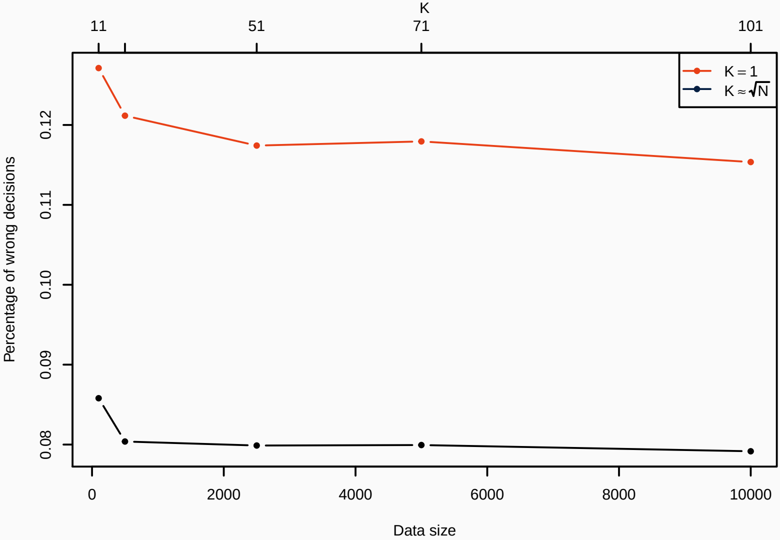
N=10000, K=101

Example



N=10000, K=1

Example



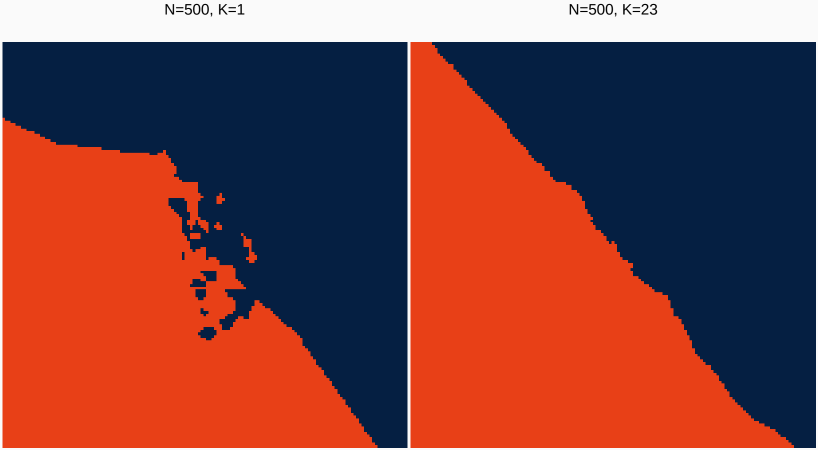
K
11
51
71
101
K = 1
K ≈ √N
Percentage of wrong decisions
0.12
0.11
0.10
0.09
0.08
0
2000
4000
6000
8000
10000
Data size

Decision functions

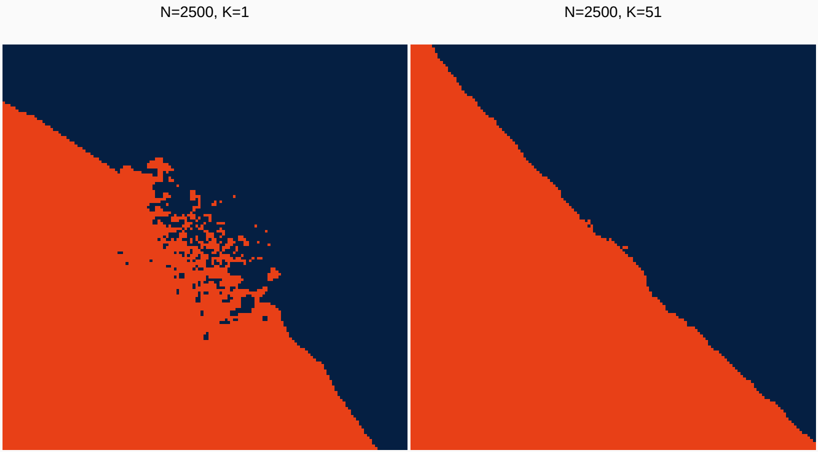
N=100, K=1

N=100, K=11

Decision functions



Decision functions



Decision functions

N=5000, K=1

N=5000, K=71

Decision functions

N=10000, K=1

N=10000, K=101

Remark

Theorethical guarantees for the 1-nn

- under simple hypotheses for a binary classification problem with $l(p,v) = \mathbf{1}_{p \neq v}$ and defining $R_1^* = \lim_{N\to\infty} R_l(1-nn_{\mathcal{D}_N})$ we have

$$R_l^* \leq R_1^* \leq 2R_l^*(1-R_l^*)$$

- for $R_l^* = 0$ (no noise case): $R_1^* = 0$!
- with M classes

$$R_l^* \leq R_1^* \leq R_l^* \left(2 - \frac{M}{M-1} R_l^*\right)$$

- overfitting is a complex problem

Methodology

Does ERM work?

- Yes!

Does ERM work?

- Yes!
- but one needs to control the complexity of the class of models $\mathcal{G}$
- this is a form of regularization: one cannot look for the model in an arbitrary class of models
- this will be addressed latter in the course

Methodology

Does ERM work?

- Yes!
- but one needs to control the complexity of the class of models $\mathcal{G}$
- this is a form of regularization: one cannot look for the model in an arbitrary class of models
- this will be addressed latter in the course

Basic element for a solution

- apply the ML method to a data set $\mathcal{D}$, the training set
- evaluate its risk on another independent data set $\mathcal{D}'$
- in summary: $\widehat{R}_l(g_{\mathcal{D}}, \mathcal{D}')$

Example

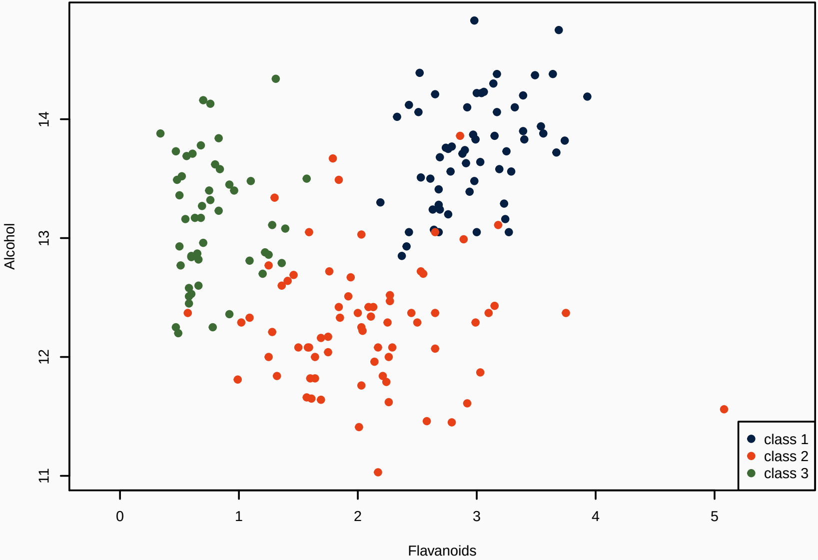
Data and loss

- chemical analysis of wines derived from three cultivars ($\mathcal{Y} = \{1, 2, 3\}$)
- 178 observations with 2 variables ($\mathcal{X} = \mathbb{R}^2$)
- $l(a, b) = \mathbf{1}_{a \neq b}$

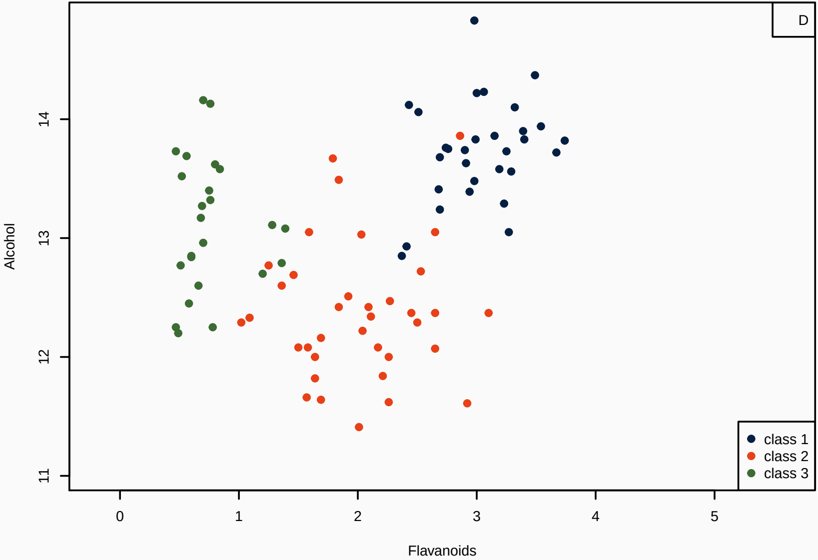
K-nn model

- use half the data for the training set $\mathcal{D}$
- use the other half for empirical risk evaluation $\mathcal{D}'$

Example

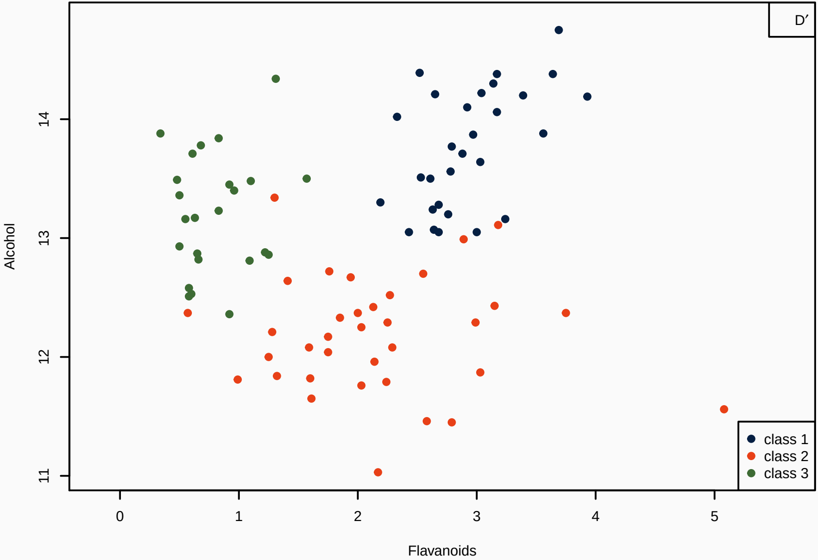


Example

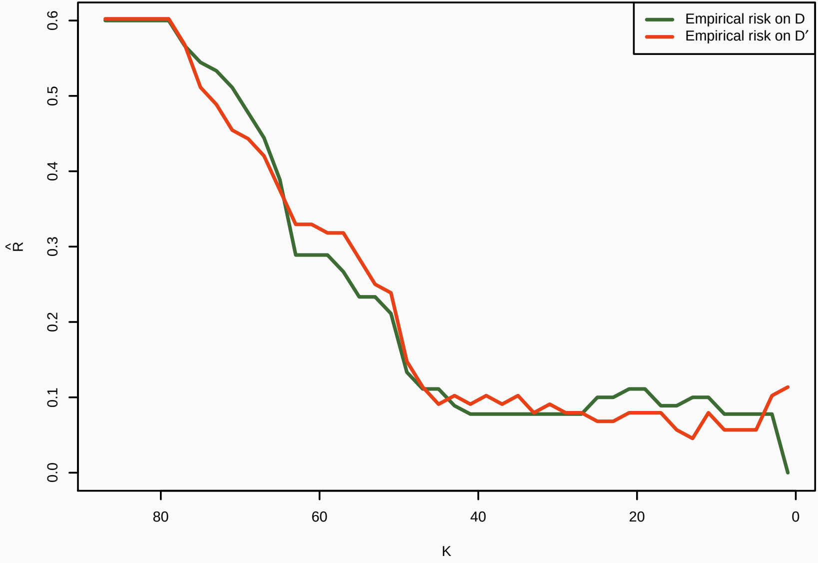


D
Alcohol
14
13
12
11
0
1
2
3
4
5
Flavanoids
class 1
class 2
class 3

Example

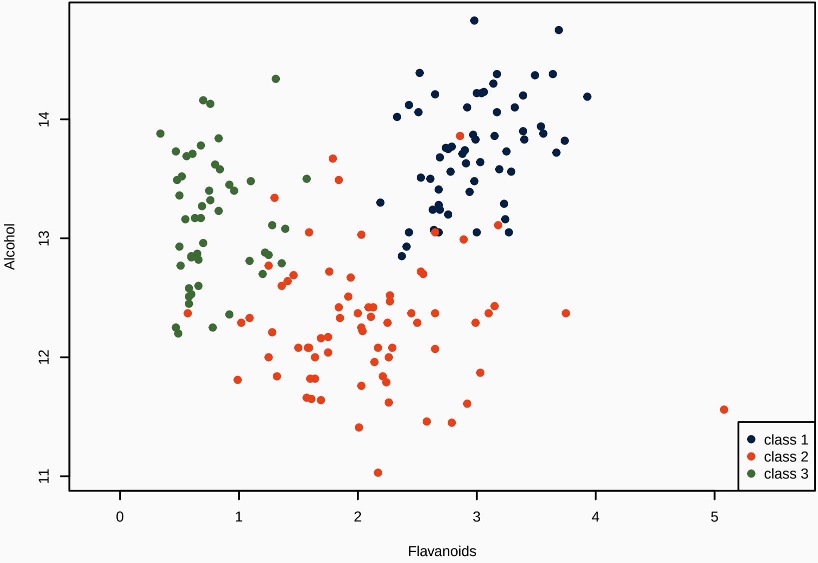

D′
Alcohol
14
13
12
11
0
1
2
3
4
5
Flavanoids
class 1
class 2
class 3

Results

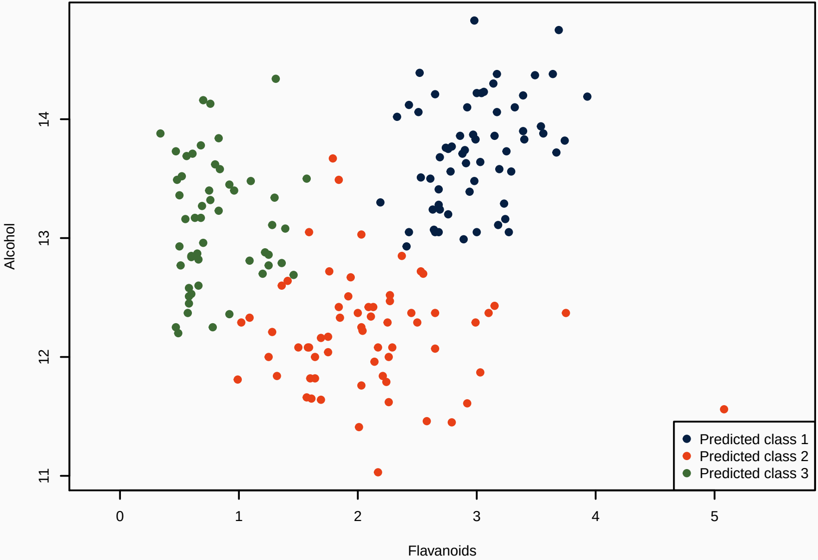


Empirical risk on D
Empirical risk on D′
0.0
0.1
0.2
0.3
0.4
0.5
0.6
$\hat{R}$
80
60
40
20
0
K

Outputs of the model



Outputs of the model



Basic general framework

1. split the data into $\mathcal{D}$ (training), $\mathcal{D}'$ (validation) and $\mathcal{D}''$ (test)
2. for each machine learning algorithm $\mathcal{A}$ under study
   2.1 for each value θ of the parameters of the algorithm
      2.1.1 compute the model using θ on $\mathcal{D}$, $g_{\mathcal{A},\theta,\mathcal{D}}$
      2.1.2 compute $\widehat{R}_l(g_{\mathcal{A},\theta,\mathcal{D}}, \mathcal{D}')$
3. chose the best algorithm with the best parameter, $\mathcal{A}^*$ and θ^* (according to $\widehat{R}_l(., \mathcal{D}')$)
4. compute the best model $g^* = g_{\mathcal{A}^*,\theta^*,\mathcal{D}\cup\mathcal{D}'}$
5. compute $\widehat{R}_l(g^*, \mathcal{D}'')$

Basic general framework

1. split the data into $\mathcal{D}$ (training), $\mathcal{D}'$ (validation) and $\mathcal{D}''$ (test)
2. for each machine learning algorithm $\mathcal{A}$ under study
   2.1 for each value θ of the parameters of the algorithm
      2.1.1 compute the model using θ on $\mathcal{D}$, $g_{\mathcal{A},\theta,\mathcal{D}}$
      2.1.2 compute $\widehat{R}_l(g_{\mathcal{A},\theta,\mathcal{D}}, \mathcal{D}')$
3. chose the best algorithm with the best parameter, $\mathcal{A}^*$ and θ^* (according to $\widehat{R}_l(., \mathcal{D}')$)
4. compute the best model $g^* = g_{\mathcal{A}^*,\theta^*,\mathcal{D}\cup\mathcal{D}'}$
5. compute $\widehat{R}_l(g^*, \mathcal{D}'')$

Goals of this course

- describe state-of-the-art alternative for the algorithms
- study some theoretical aspects, e.g. empirical risk minimization
- describe better frameworks

Other learning contexts

Variations over supervised learning

- different prediction spaces
  - data in $\mathcal{X} \times \mathcal{Y}$
  - model g from $\mathcal{X}$ to $\mathcal{Y}'$
  - loss function l from $\mathcal{Y}' \times \mathcal{Y}$ to $\mathbb{R}^+$
  - typically: scoring
- pairwise supervised learning
  - data in $\mathcal{X}$
  - relation on data points $r(\mathbf{X}, \mathbf{X}')$
  - model g from $\mathcal{X}$ to $\mathcal{Y}$
  - pairwise “loss” function $l(g(\mathbf{X}), g(\mathbf{X}'), r(\mathbf{X}, \mathbf{X}'))$
  - typically: learning to rank, representation learning

Other learning contexts

Unsupervised learning

- $\mathcal{D} = ((\mathbf{X}_i)_{1 \leq i \leq N})$
- no target value, no relation, nothing else!
- goal: "understanding" the data
- in practice, many concrete goals such as
  - finding clusters
  - finding frequent patterns
  - finding outliers
  - modeling the data distribution
  - etc.

Other learning contexts

Semi-supervised learning

- a data set $\mathcal{D} = ((\mathbf{X}_i, \mathbf{Y}_i))_{1 \leq i \leq N}$
- another data set $\mathcal{D}' = ((\mathbf{X}'_i)_{1 \leq i \leq N'})$
- supervised point of view
  - build a classical model using $\mathcal{D}$
  - use $\mathcal{D}'$ to get better results than those obtained with $\mathcal{D}$ only
- unsupervised point of view
  - build a clustering model using $\mathcal{D}'$
  - use $\mathcal{D}$ as constraints for the clustering
    - if $\mathbf{Y}_i = \mathbf{Y}_j$ then $\mathbf{X}_i$ and $\mathbf{X}_j$ must be in the same cluster
    - opposite constraints in the reverse situation

Other learning contexts

Reinforcement learning

- completely different context:
  - an agent and its environment with associated states
  - a set of actions the agent can take
  - probabilistic transitions: when the agent takes an action, the global state changes as a consequence, possibly in a stochastic way
  - immediate reward: the reward gained by taking an action
- goal: computing an optimal policy
  - a policy maps states to actions
  - the value of a policy is the expected total reward obtained by following it
  - “easy” if everything is known
- learning: discovering the optimal policy by performing actions

Difficulties

Data collection

- labeled data can be difficult to collect: questionable human labeling “farms”
- privacy (and GDPR compliance)

Bias

- biased data lead to biased models
- e.g. COMPAS recidivism algorithm

Interpretability

- trade-off between interpretable models (white box) and accurate models (black box)
- the best current models (from deep learning) are very opaque
- interpretability is requested by the GDPR (in a way)

Conclusion

Machine Learning

Discipline that designs algorithms to build programs that solve complex tasks using examples of successful completion of those tasks

Building blocks

- data
  - task dependent form (e.g. (input, output) pairs for supervised learning)
  - being able to compare entities is mandatory
- quality measures: how “good” is the solution to the given task?
- optimization techniques

Licence

Version

Last git commit: 2021-01-19
By: Fabrice Rossi (Fabrice.Rossi@apiacoa.org)
Git hash: a623238c82efeb5372d8b821e0e946cfd8c918cc

Changelog

- December 2019:
  - new examples: spam, MNIST, artificial data set for knn
  - practical application examples
  - graphical illustration of k-nearest neighbors
  - more learning paradigm
  - difficulties
  - summary and conclusion
- January 2018: initial version