Application of Artificial Intelligence

Opportunities and limitations through life & Earth sciences examples

Clovis Galiez

Grenoble
Statistiques pour les sciences du Vivant et de l'Homme

March 24, 2021

Disclaimer

- You should form teams of 4 persons on Teide.
- Answer the questions in the template at `https://clovisg.github.io/teaching/asdia/ctd1/quote.tar` and post-it on teide.
- You can use the following Discord server `https://discord.gg/RsajWEdgvv`, I'll be present to answer live questions during the lecture slots. Do not hesitate to post your understandings and mis-understandings out of the time slots, I won't judge it, I'll only judge your involvment and curiosity.
- You can send me emails (`clovis.galiez@grenoble-inp.fr`) for specific questions, and I'll answer publicly on the riot channel.
- Slides will be posted on `https://clovisg.github.io` at the end of each session

Goals

- Have a critical understanding of the place of AI in society
- Discover and practice machine learning (ML) techniques
  - Linear regression
  - Logistic regression
- Experiment some limitations
  - Curse of dimensionality
  - Hidden overfitting
  - Sampling bias
- Towards autonomy with ML techniques
  - Design experiments
  - Organize the data
  - Evaluate performances

Today's outline

- AI? What for?
- Glance on the applications in these series
  - Microbiome and metagenomics
- Curse of dimensionality
- Regularization

AI? What is it? What for?

Scope of these series: machine learning

AI includes a lot of domains (e.g. logic or statistics) with different goals (e.g. prediction, description of a system) and techniques (e.g. rule inference, neural networks).

80's expert systems

Modern artificial intelligence is mainly based on **data science**.

We will focus on the *data science* part of artificial intelligence : **machine learning**.

Some machine learning methods

What machine learning tool you already know?

Some machine learning methods

What machine learning tool you already know?

For classification tasks:

- Linear Discriminant Analysis (LDA)
- Logistic regression
- Support Vector Machine (SVM)
- Artificial neural networks

For regression tasks:

- Linear regression
- Regressive artificial neural networks

Controversies

In the media:

- + AI solve all problems: ecology, unemployment, etc.
- - AI is dangerous: "big data is watching you"
- - AI is not fair: biases

Interesting article about biases in The Consversation.

In the scientific community:

- + AI solves everything: you can predict anything if you have the data
- - AI does not explain anything: it's only black boxes

AI and CO_2

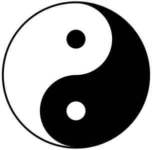
AI can consumes a lot of electrical energy, having a strong environmental impact. Here are some figures showing the equivalent CO2 emission for creating some famous AI models for natural language processing:

Model	Hardware	Power (W)	Hours	kWh·PUE	CO_2e	Cloud compute cost
$\text{Transformer}_{base}$	P100x8	1415.78	12	27	26	$41–$140
Transformer_{big}	P100x8	1515.43	84	201	192	$289–$981
ELMo	P100x3	517.66	336	275	262	$433–$1472
BERT_{base}	V100x64	12,041.51	79	1507	1438	$3751–$12,571
BERT_{base}	TPUv2x16	—	96	—	—	$2074–$6912
NAS	P100x8	1515.43	274,120	656,347	626,155	$942,973–$3,201,722
NAS	TPUv2x1	—	32,623	—	—	$44,055–$146,848
GPT-2	TPUv3x32	—	168	—	—	$12,902–$43,008

Table 3: Estimated cost of training a model in terms of CO_2 emissions (lbs) and cloud compute cost (USD).[7] Power and carbon footprint are omitted for TPUs due to lack of public information on power draw for this hardware.

[Strubell et al. https://arxiv.org/pdf/1906.02243.pdf]

Be fair



What is the right place of AI?

Assignment 1 - Manichean views

Find two AI applications (can be softwares, proof of concepts, etc.), one you would characterize as good, one as bad. Write a one-page assignment to explain why.
Try to think in particular what would be the **societal impacts** if the examples you chose were generalized in the world.

We will now give a little bit of background about microbes, which will be our main application troughout this lecture.

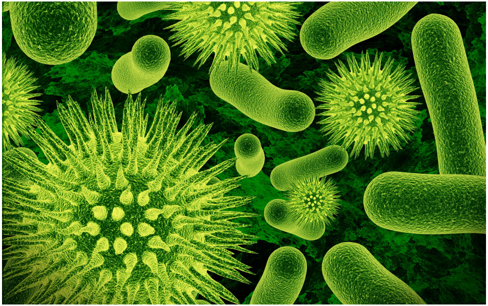
Machine learning for microbial bioinformatics

The microbial world

They are everywhere... they work hard 24h a day... they fight against each other... and they collaborate.

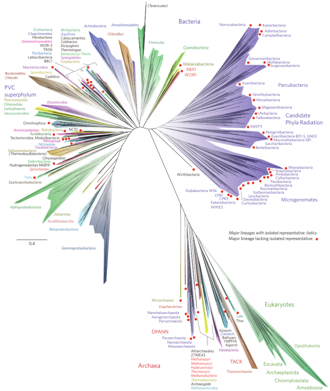
The microbial world

They are everywhere... they work hard 24h a day... they fight against each other... and they collaborate.



There are very diverse in terms of morphology, mechanisms, and genetics: bacteria, fungus, viruses, picoeukaryotes, etc.

Not a fixed knowledge: **we still continue to discover new branches of life**:



[Hug et al. 2016]

The Candidate Phyla Radiation (top right, in purple) has been discovered in 2016!

Microbiome importance in biogeochemical cycles

Nitrogen cycle [Canfield et al., Science 2010]

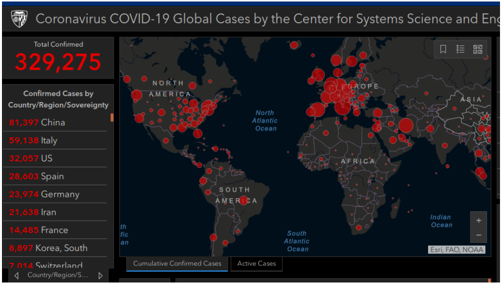
CO_2 turnover: viruses kill 20% of the living biomass in the ocean every day! [Suttle, Nat. Microbiol. 2007]

Microbiome importance in human health

The bright side:

Health status highly correlated with the diversity of the gut microbiome [Valdes et al. 2018]

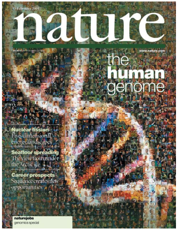
The dark side:



Covid-19

The human gut microbiome

2000's
Human genome

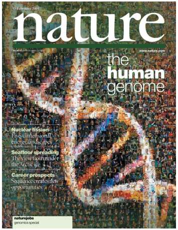


2010's
Gut metagenomes

$\approx$ 20k protein-coding genes

The human gut microbiome

2000's
Human genome



2010's
Gut metagenomes

$\approx$ 20k protein-coding genes $\xrightarrow{\times 100}$ $\approx$ 2M protein-coding genes

Human gut microbiome is rich! What microbes do there is absolutely neccesary to keep alive!

Gut microbiota and higher order diseases

- Autism
  spectrum disorder (ASD), but the underlying mechanisms are unknown. Many studies have shown alterations in the composition of the fecal flora and metabolic products of the gut microbiome in patients with ASD. The gut microbiota influences brain development and behaviors through the neuroendocrine, neuroimmune and autonomic nervous systems. In addition, an abnormal gut microbiota is associated with several diseases,
  [Li et al. Front. in Cell. Neur. 2017]
- Type II diabetes (50 microbial genes → AUC ROC 0.81)
  [Qin et al. *Nature* 2012]
- Parkinson's differential abundance of gut microbial species
  [Heintz-Buschart et al. *Mov. Disord.* 2018]

Gut microbiota and higher order diseases

- Autism
  spectrum disorder (ASD), but the underlying mechanisms are unknown. Many studies have shown alterations in the composition of the fecal flora and metabolic products of the gut microbiome in patients with ASD. The gut microbiota influences brain development and behaviors through the neuroendocrine, neuroimmune and autonomic nervous systems. In addition, an abnormal gut microbiota is associated with several diseases,
  [Li et al. Front. in Cell. Neur. 2017]
- Type II diabetes (50 microbial genes $\rightarrow$ AUC ROC 0.81)
  [Qin et al. *Nature* 2012]
- Parkinson's differential abundance of gut microbial species
  [Heintz-Buschart et al. *Mov. Disord.* 2018]

Can we associate the presence of microbes to a phenotype?

You may ask yourself

What all of this has to do with machine learning?!

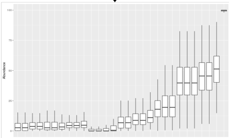
Metagenomics: the (*very*) big picture

sample

sequencing

metagenomic

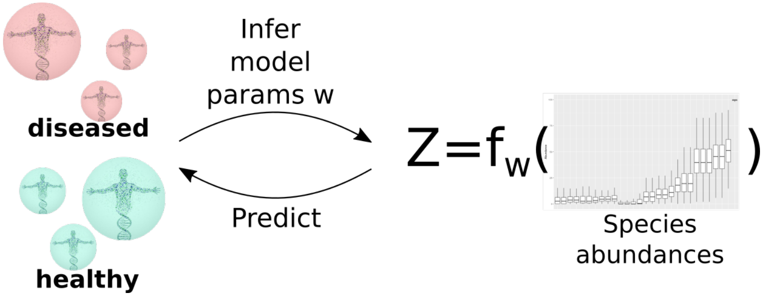
metagenome



Abundance of species

MWAS: metagenome-wide association studies

We can build models to predict diseases from microbial abundances, a process known as MWAS:



MWAS as a classification problem

Let:

- $\vec{X}$ be an M-dimensional random vector of abundance of species,
- and Z binary (0/1) random variable describing the disease state of a human.

Define a predictor $f : \mathbb{R}_+^M \to [0,1]$ such that it minimizes a *loss* on a training set $(\vec{x}_1, z_1), ..., (\vec{x}_N, z_N)$:

MWAS as a classification problem

Let:

- $\vec{X}$ be an M-dimensional random vector of abundance of species,
- and Z binary (0/1) random variable describing the disease state of a human.

Define a predictor $f : \mathbb{R}_+^M \rightarrow [0, 1]$ such that it minimizes a *loss* on a training set $(\vec{x}_1, z_1), ..., (\vec{x}_N, z_N)$:

$$\min_f - \sum_{i=1}^{N} z_i . \log f(\vec{x}_i) + (1 - z_i) . \log(1 - f(\vec{x}_i))$$

Goal of the next sessions:

Can we diagnosis **Inflammatory Bowel Disease** or predict IBD through the structure of the gut microbial community?

Techniques involved: logistic regression, lasso regularization.

ML traps:
I. The curse of dimensionality

A model predicts unknown outcomes

≈ Definition

We will define a model as a function depending on parameters $\vec{\theta}$ and features $\vec{x}$ describing a target variable $\vec{y}$.

The role of **machine learning** is

A model predicts unknown outcomes

≈ Definition

We will define a model as a function depending on parameters $\vec{\theta}$ and features $\vec{x}$ describing a target variable $\vec{y}$.

The role of **machine learning** is to **infer** the parameters $\vec{\theta}$ from a **training** set $\{(\vec{x}, \vec{y})_i, i \in 1, ..N\}$ of known relations in order to have $f(\vec{x}_i) \approx \vec{y}_i$.

A model predicts unknown outcomes

≈ Definition

We will define a model as a function depending on parameters $\vec{\theta}$ and features $\vec{x}$ describing a target variable $\vec{y}$.

The role of **machine learning** is to **infer** the parameters $\vec{\theta}$ from a **training** set $\{(\vec{x}, \vec{y})_i, i \in 1, ..N\}$ of known relations in order to have $f(\vec{x}_i) \approx \vec{y}_i$.

The **high hope** is that $f(\vec{x}) \approx \vec{y}$ for yet unknown $\vec{x}, \vec{y}$ couples.

A model predicts unknown outcomes

≈ Definition

We will define a model as a function depending on parameters $\vec{\theta}$ and features $\vec{x}$ describing a target variable $\vec{y}$.

The role of **machine learning** is to **infer** the parameters $\vec{\theta}$ from a **training** set $\{(\vec{x}, \vec{y})_i, i \in 1, ..N\}$ of known relations in order to have $f(\vec{x}_i) \approx \vec{y}_i$.

The **high hope** is that $f(\vec{x}) \approx \vec{y}$ for yet unknown $\vec{x}, \vec{y}$ couples.

We can check that on a training set, but will it generalize?

Overfitting

One of the main source of overfitting can be **model hyperparametrization**.

Overfitting

One of the main source of overfitting can be **model hyperparametrization**.

Exercise

Suppose you have a model with one binary parameter θ. Given the input, how many outputs can your model describe?

Overfitting

One of the main source of overfitting can be **model hyperparametrization**.

Exercise

Suppose you have a model with one binary parameter θ. Given the input, how many outputs can your model describe?
Suppose you have a model with N binary parameters θ_i. Given the input, how many outputs can your model describe?

Overfitting

One of the main source of overfitting can be **model hyperparametrization**.

Exercise

Suppose you have a model with one binary parameter θ. Given the input, how many outputs can your model describe?
Suppose you have a model with N binary parameters θ_i. Given the input, how many outputs can your model describe?

Important

It means that with many parameters, it can be easy to get very accurate predictions on the training set... But it won't necessarily generalize well!

Example: polynomial regression

Suppose you measure the fuel stream Y and the car speed x.
How could you simply model the dependency between x and Y?

Example: polynomial regression

Suppose you measure the fuel stream Y and the car speed x.
How could you simply model the dependency between x and Y?

$$Y = \beta_0 + \beta_1 . x + \epsilon \text{ with } \epsilon \sim \mathcal{N}(0, \sigma^2)$$

Example: polynomial regression

Suppose you measure the fuel stream Y and the car speed x.
How could you simply model the dependency between x and Y?

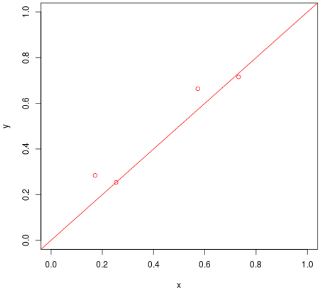
$$Y = \sum_{i=0}^{3} \beta_i x^i + \epsilon \text{ with } \epsilon \sim \mathcal{N}(0, \sigma^2).$$

Fitting the parameters

Your neighbor, gives you her home-made measurements. You, computer scientist, you fit the parameters of your model.

Fitting the parameters

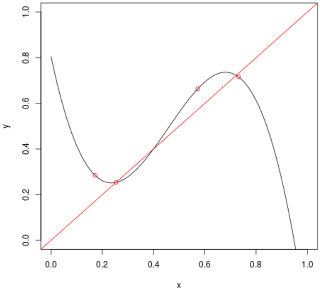
Your neighbor, gives you her home-made measurements. You, computer scientist, you fit the parameters of your model.



$$N = 4$$

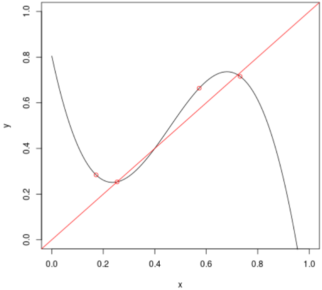
Fitting the parameters

Your neighbor, gives you her home-made measurements. You, computer scientist, you fit the parameters of your model.



$N = 4$

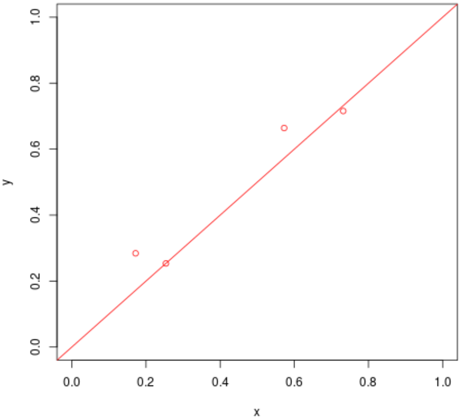
Fitting the parameters



What is the problem here? How to solve it?

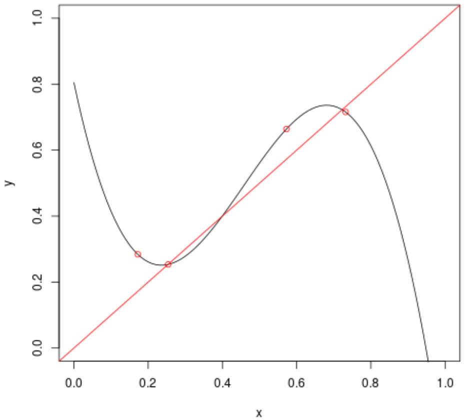
Adding more data helps!

$$N = 4$$



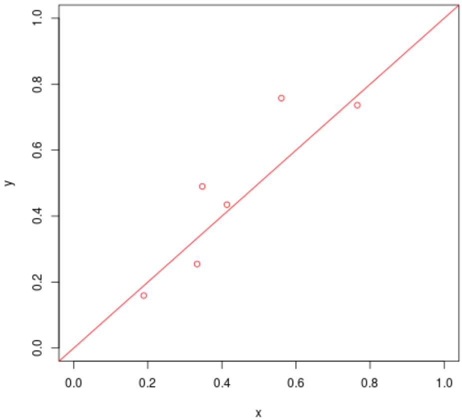
Adding more data helps!

$$N = 4$$



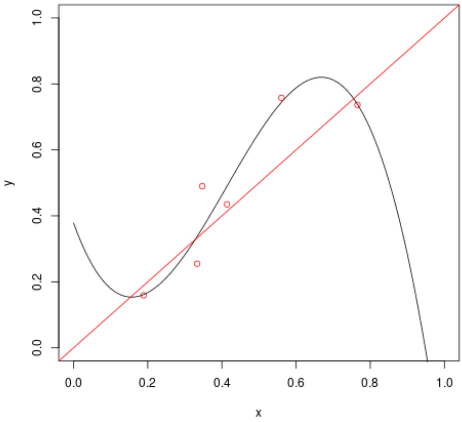
Adding more data helps!

$$N = 6$$



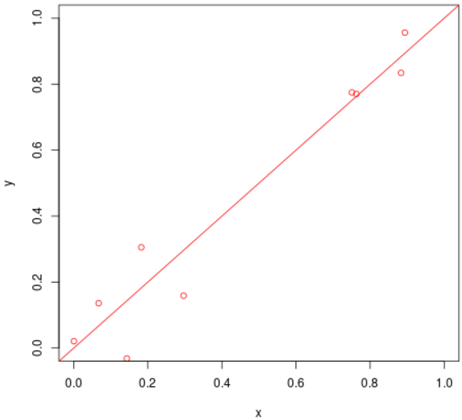
Adding more data helps!

$$N = 6$$



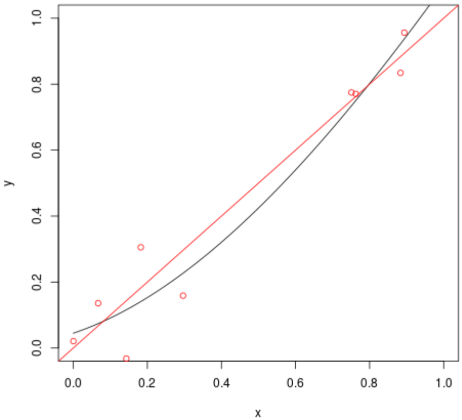
Adding more data helps!

$$N = 10$$



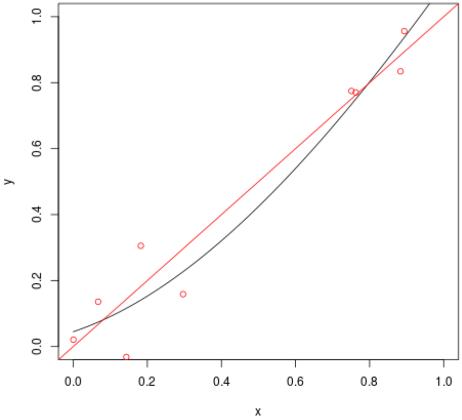
Adding more data helps!

$$N = 10$$



Adding more data helps!

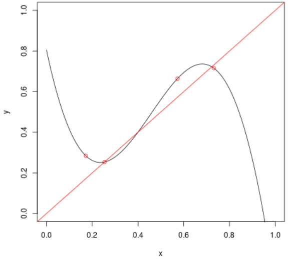
$$N = 10$$



What shall we do if we cannot get more data points?

Toward regularization

What is making you deeply think that this model is wrong?

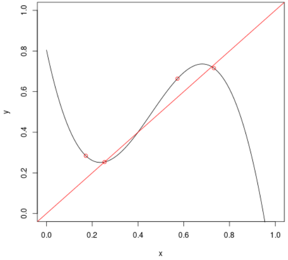


Maximum *likelihood* coefficients:

β_0	β_1	β_2	β_3
5.169	-54.388	155.755	-114.487

Toward regularization

What is making you deeply think that this model is wrong?



Maximum *likelihood* coefficients:

β_0	β_1	β_2	β_3
5.169	-54.388	155.755	-114.487

Some range of values for the parameters are unrealistic!

Regularization

The idea of regularization

Definition (well...)

Regularization is a set of methods for avoiding "unrealistic zones" in your parameter space.

Along the tutorials we will use:

- Ridge penalization (avoids high values of parameters)
- Lasso penalization (favor not using some parameters)

Other types of regularization (for Neural Networks in particular) include:

- Gaussian noise
- Dropout (favor independence in the responsibilities of the parameters)

Prior distributions

In the bayesian world, probabilities represent the degree of knowledge.

Prior distributions

In the bayesian world, probabilities represent the degree of knowledge.
So we can integrate *a priori* knowledge in our model :)

Prior distributions

In the bayesian world, probabilities represent the degree of knowledge.
So we can integrate *a priori* knowledge in our model :)

We consider $\beta_0, ...\beta_3$ as random variables (i.e. quantity having uncertainties).

We *model them*, for example with normal distributions centered on likely values (e.g. $\mu_0 = 0.1$, $\mu_1 = ...$) with some likely variability (e.g. $\eta_0 = 0.005$, etc.).

Prior distributions

In the bayesian world, probabilities represent the degree of knowledge.
So we can integrate *a priori* knowledge in our model :)

We consider $\beta_0, ...\beta_3$ as random variables (i.e. quantity having uncertainties).

We *model them*, for example with normal distributions centered on likely values (e.g. $\mu_0 = 0.1$, $\mu_1 = ...$) with some likely variability (e.g. $\eta_0 = 0.005$, etc.).

The model becomes:

$$\epsilon \sim \mathcal{N}(0, \sigma^2)$$
$$\beta_i \sim \mathcal{N}(\mu_i, \eta_i^2)$$
$$Y = \sum \beta_i . x^i + \epsilon$$

Prior distributions

In the bayesian world, probabilities represent the degree of knowledge.
So we can integrate *a priori* knowledge in our model :)

We consider $\beta_0, ...\beta_3$ as random variables (i.e. quantity having uncertainties).

We *model them*, for example with normal distributions centered on likely values (e.g. $\mu_0 = 0.1$, $\mu_1 = ...$) with some likely variability (e.g. $\eta_0 = 0.005$, etc.).

The model becomes:

$$\epsilon \sim \mathcal{N}(0, \sigma^2)$$
$$\beta_i \sim \mathcal{N}(\mu_i, \eta_i^2)$$
$$Y = \sum \beta_i.x^i + \epsilon$$

What is "random" here?

Prior distributions

In the bayesian world, probabilities represent the degree of knowledge.
So we can integrate *a priori* knowledge in our model :)

We consider $\beta_0, ...\beta_3$ as random variables (i.e. quantity having uncertainties).

We *model them*, for example with normal distributions centered on likely values (e.g. $\mu_0 = 0.1$, $\mu_1 = ...$) with some likely variability (e.g. $\eta_0 = 0.005$, etc.).

The model becomes:

$$\epsilon \sim \mathcal{N}(0, \sigma^2)$$
$$\beta_i \sim \mathcal{N}(\mu_i, \eta_i^2)$$
$$Y = \sum \beta_i . x^i + \epsilon$$

What is "random" here?

The β_i are model **parameters** (inferred from the training data).
The μ_i and η_i are **hyperparameters** (not inferred from the training).

Worked out example

Consider a simple model:

$$\epsilon \sim \mathcal{N}(0, \sigma^2)$$
$$\beta \sim \mathcal{N}(5, \eta^2)$$
$$Y = \beta x + \epsilon$$

Exercise

1. Write the likelihood of β for observing $(y_1, x_1), ...(y_N, x_N)$. Deduce for which β it reaches its maximum.
2. For which β is the *posterior* probability distribution $p(\beta | Y_1 = y_1, ...Y_N = y_N) = \frac{p(Y_1=y_1,...Y_N=y_N|\beta).p(\beta)}{p(Y_1=y_1,...Y_N=y_N)}$ maximal?
3. Interpret what is the effect of putting a prior distribution on the β.

Toward ridge regularization

Consider the linear model $Y = \sum \vec{\beta}.\vec{x_i} + \epsilon$.

Exercise

1. Show that the maximum likelihood solution is the same as the solution of the following optimization problem:

$$\min_{\vec{\beta}} \sum_{i=0}^{N} (y_i - \vec{\beta}.\vec{x_i})^2$$

Toward ridge regularization

Consider the linear model $Y = \sum \vec{\beta}.\vec{x_i} + \epsilon$.

Exercise

1. Show that the maximum likelihood solution is the same as the solution of the following optimization problem:

$$\min_{\vec{\beta}} \sum_{i=0}^{N} (y_i - \vec{\beta}.\vec{x_i})^2$$

2. Show that maximizing a posteriori probability with a Gaussian prior centered on zero on β is the same as solving the following optimization problem:

$$\min_{\vec{\beta}} \sum_{i=0}^{N} (y_i - \vec{\beta}.\vec{x_i})^2 + \lambda ||\vec{\beta}||_2^2$$

Toward ridge regularization

Consider the linear model $Y = \sum \vec{\beta}.\vec{x_i} + \epsilon$.

Exercise

1. Show that the maximum likelihood solution is the same as the solution of the following optimization problem:

$$\min_{\vec{\beta}} \sum_{i=0}^{N} (y_i - \vec{\beta}.\vec{x_i})^2$$

2. Show that maximizing a posteriori probability with a Gaussian prior centered on zero on β is the same as solving the following optimization problem:

$$\min_{\vec{\beta}} \sum_{i=0}^{N} (y_i - \vec{\beta}.\vec{x_i})^2 + \lambda ||\vec{\beta}||_2^2$$

This is called **ridge regularization**. What is it enforcing?

Toward ridge regularization

Consider the linear model $Y = \sum \vec{\beta}.\vec{x_i} + \epsilon$.

Exercise

1. Show that the maximum likelihood solution is the same as the solution of the following optimization problem:

$$\min_{\vec{\beta}} \sum_{i=0}^{N} (y_i - \vec{\beta}.\vec{x_i})^2$$

2. Show that maximizing a posteriori probability with a Gaussian prior centered on zero on β is the same as solving the following optimization problem:

$$\min_{\vec{\beta}} \sum_{i=0}^{N} (y_i - \vec{\beta}.\vec{x_i})^2 + \lambda ||\vec{\beta}||_2^2$$

This is called **ridge regularization**. What is it enforcing?
It tells the model **to avoid high values** for the parameters.

Further justification of ridge regularization

Having a model with N binary parameters θ_i. Given an input, the model can describe ? outputs.

Further justification of ridge regularization

Having a model with N binary parameters θ_i. Given an input, the model can describe 2^N outputs.

Further justification of ridge regularization

Having a model with N binary parameters θ_i. Given an input, the model can describe 2^N outputs.
Having a model with N parameters θ_i that live in $\{1, ..., K\}$. Given an input, the model can describe ? outputs.

Further justification of ridge regularization

Having a model with N binary parameters θ_i. Given an input, the model can describe 2^N outputs.
Having a model with N parameters θ_i that live in $\{1, ..., K\}$. Given an input, the model can describe K^N outputs.

Further justification of ridge regularization

Having a model with N binary parameters θ_i. Given an input, the model can describe 2^N outputs.
Having a model with N parameters θ_i that live in $\{1, ..., K\}$. Given an input, the model can describe K^N outputs.
How would you measure that for continuous parameters?

Further justification of ridge regularization

Having a model with N binary parameters θ_i. Given an input, the model can describe 2^N outputs.
Having a model with N parameters θ_i that live in $\{1, ..., K\}$. Given an input, the model can describe K^N outputs.
How would you measure that for continuous parameters?
With the volume:

$$V_N(r) = K_N.r^N$$

Further justification of ridge regularization

Having a model with N binary parameters θ_i. Given an input, the model can describe 2^N outputs.
Having a model with N parameters θ_i that live in $\{1, ..., K\}$. Given an input, the model can describe K^N outputs.
How would you measure that for continuous parameters?
With the volume:

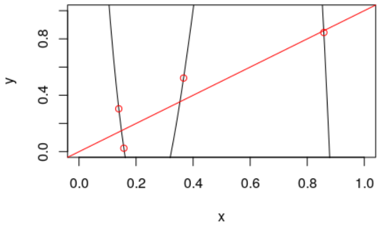
$$V_N(r) = K_N . r^N \xrightarrow[r \to \infty]{} \infty$$

Thus, there are "more" possible model outputs when parameters have high values.

Ridge regularization example

Let's come back to the model $Y = \sum_{i=0}^{3} \beta_i x^i + \epsilon$.

The maximum likelihood with 4 points will give a $\vec{\beta}$ fitting perfectly the points:



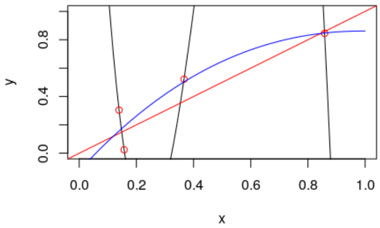
Maximum *likelihood* coefficients:

β_0	β_1	β_2	β_3
5.169	-54.388	155.755	-114.487

Ridge regularization example

Let's come back to the model $Y = \sum_{i=0}^{3} \beta_i x^i + \epsilon$.

With a prior $\mathcal{N}(0, \eta^2)$ the maximum a posteriori of the vector $\vec{\beta}$ corresponds to (blue curve):



Maximum *a posteriori* coefficients

β_0	β_1	β_2	β_3
-0.1279	2.2561	-1.5779	0.3180

Quizz

Overfitting depends on:

- Size of the training set
- Complexity of the problem
- The parametrization of the model
- The type of the model

Next week: Lasso regularization, logistic regression