Core Info

- **Where**: Telecom ParisTech
- **Location**: Amphi Estaunié or B312
- **ECTS**: 5 ECTS
- **Volume**: 40h
- **When**: 12 weeks (including one week break for holidays + one week for exam)
- **Online**: All teaching materials on moodle: http://datascience-x-master-paris-saclay.fr/education/
- Students upload their projects / reports via moodle too.
- **All students **must** be registered on moodle.**
Who am I?

Robert M. Gower

- Assistant Prof at Telecom
- robert.gower@telecom-paristech.fr
- www.ens.fr/~rgower
- Research topics: Stochastic algorithms for optimization, numerical linear algebra, quasi-Newton methods and automatic differentiation (backpropagation).
Introduction to Optimization in Machine Learning

Robert M. Gower

Master 2 Data Science, Univ. Paris Saclay
Optimisation for Data Science
An Introduction to Supervised Learning
References for this class

Chapter 1
Understanding Machine Learning: From Theory to Algorithms

Convex Optimization
Pages 67 to 79
Is There a Cat in the Photo?

Yes

No
Is There a Cat in the Photo?

Yes
Is There a Cat in the Photo?

Yes
Is There a Cat in the Photo?

No
Is There a Cat in the Photo?

Yes
Is There a Cat in the Photo?

Find mapping \( h \) that assigns the “correct” target to each input

\[
h : x \in X \rightarrow y \in \mathbb{R}
\]
Labeled Data: The training set

\[
\begin{align*}
x^1 \{ & \quad y^1 = 1 \\
x^2 \{ & \quad y^2 = 1 \\
x^3 \{ & \quad y^3 = -1 \\
\cdots & \quad \cdots \\
x^n \{ & \quad y^n = 1
\end{align*}
\]
Labeled Data: The training set

$x^1 \{ x^1 \}$

$y^1 = 1$

$x^2 \{ x^2 \}$

$y^2 = 1$

$x^3 \{ x^3 \}$

$y^3 = -1$

$\cdots x^n \{ x^n \}$

$y^n = 1$

$y = -1$ means no/false
Labeled Data: The training set

\[ x^1 \{ \begin{array}{l} y^1 = 1 \end{array} \]  \quad x^2 \{ \begin{array}{l} y^2 = 1 \end{array} \]  \quad x^3 \{ \begin{array}{l} y^3 = -1 \end{array} \]  \quad \cdots \ x^n \{ \begin{array}{l} y^n = 1 \end{array} \]  

\[ y = -1 \text{ means no/false} \]  

Learning Algorithm
Labeled Data: The training set

\[ x^1 \{ \begin{array}{c} y^1 = 1 \end{array} \} \quad x^2 \{ \begin{array}{c} y^2 = 1 \end{array} \} \quad x^3 \{ \begin{array}{c} y^3 = -1 \end{array} \} \quad \cdots \quad x^n \{ \begin{array}{c} y^n = 1 \end{array} \} \]

\[ y = -1 \text{ means no/false} \]

Learning Algorithm

\[ h : x \in X \rightarrow y \in \mathbb{R} \]
Labeled Data: The training set

\[ x^1 \{ \begin{array}{c} x^1 \\ y^1 = 1 \end{array} \} \quad x^2 \{ \begin{array}{c} x^2 \\ y^2 = 1 \end{array} \} \quad x^3 \{ \begin{array}{c} x^3 \\ y^3 = -1 \end{array} \} \quad \cdots \quad x^n \{ \begin{array}{c} x^n \\ y^n = 1 \end{array} \} \]

\[ y = -1 \text{ means no/false} \]

\[ h : x \in X \rightarrow y \in \mathbb{R} \]

\[ h \left( \begin{array}{c} \text{dog} \end{array} \right) = -1 \]
Example: Linear Regression for Height

Labeled data  \( x \in \mathbb{R}^2, y \in \mathbb{R}_+ \)

<table>
<thead>
<tr>
<th>( x^1 )</th>
<th>( x^n )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sex</td>
<td>Sex</td>
</tr>
<tr>
<td>Male</td>
<td>Female</td>
</tr>
<tr>
<td>Age</td>
<td>Age</td>
</tr>
<tr>
<td>30</td>
<td>70</td>
</tr>
<tr>
<td>Height</td>
<td>Height</td>
</tr>
<tr>
<td>1,72 cm</td>
<td>1,52 cm</td>
</tr>
</tbody>
</table>
Example: Linear Regression for Height

Labeled data \( x \in \mathbb{R}^2, y \in \mathbb{R}_+ \)

<table>
<thead>
<tr>
<th>( x^1_1 )</th>
<th>( x^1_2 )</th>
<th>( y^1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sex</td>
<td>Male</td>
<td>Height</td>
</tr>
<tr>
<td>Age</td>
<td>30</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( x^n_1 )</th>
<th>( x^n_2 )</th>
<th>( y^n )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sex</td>
<td>Female</td>
<td>Height</td>
</tr>
<tr>
<td>Age</td>
<td>70</td>
<td></td>
</tr>
</tbody>
</table>

Example Hypothesis: Linear Model

\[
 h_w(x_1, x_2) = w_0 + x_1 w_1 + x_2 w_2 \quad x_0 = 1 \quad \langle w, x \rangle
\]
Example: Linear Regression for Height

Labeled data \( x \in \mathbb{R}^2, y \in \mathbb{R}_+ \)

<table>
<thead>
<tr>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( y )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sex</td>
<td>Male</td>
<td></td>
</tr>
<tr>
<td>Age</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>Height</td>
<td>1,72 cm</td>
<td></td>
</tr>
</tbody>
</table>

\( x_1 \) \( x_2 \) \( y \) \( x_1 \) \( x_2 \) \( y \)

Example Hypothesis: Linear Model

\[ h_w(x_1, x_2) = w_0 + x_1 w_1 + x_2 w_2 \]

Example Training Problem:

\[
\min_{w \in \mathbb{R}^3} \frac{1}{n} \sum_{i=1}^{n} \left( h_w(x_1^i, x_2^i) - y^i \right)^2
\]
Linear Regression for Height

Height vs. Age

Graph showing the relationship between Height and Age with data points plotted.
Linear Regression for Height

The Training Algorithm

\[
\min_{w \in \mathbb{R}^3} \frac{1}{n} \sum_{i=1}^{n} \left( h_w(x_1^i, x_2^i) - y^i \right)^2
\]
Linear Regression for Height

The Training Algorithm

\[ h_w(x_1, x_2) \]

Other options aside from linear?

The Training Algorithm

\[
\min_{w \in \mathbb{R}^3} \frac{1}{n} \sum_{i=1}^{n} (h_w(x_1^i, x_2^i) - y^i)^2
\]
Parametrizing the Hypothesis

**Linear:**

$$h_w(x) = \sum_{i=0}^{d} w_i x_i$$

**Polynomial:**

$$h_w(x) = \sum_{i,j=0}^{d} w_{ij} x_i x_j$$

**Neural Net:**

- **Unit $v_1$:**
  $$v_1 = \text{sign}(w_{11}x_1 + w_{12}x_2)$$
- **Unit $v_4$:**
  $$v_4 = \frac{1}{1 + \exp(w_{41}x_1 + w_{42}x_2)}$$
Loss Functions

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} (h_w(x^i) - y^i)^2
\]

Why a Squared Loss?
Loss Functions

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} (h_w(x^i) - y^i)^2
\]

Why a Squared Loss?

Let \( y_h := h_w(x) \)

Loss Functions

\[
\ell : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}_+
\]

\[
(y_h, y) \rightarrow \ell(y_h, y)
\]

The Training Problem

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell(h_w(x^i), y^i)
\]
**Loss Functions**

Why a Squared Loss?

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} (h_w(x^i) - y^i)^2
\]

Let \( y_h := h_w(x) \)

Typically a convex function

\[
\ell : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}_+
\]

\((y_h, y) \rightarrow \ell(y_h, y)\)

The Training Problem

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell \left( h_w(x^i), y^i \right)
\]
Choosing the Loss Function

Let $y_h := h_w(x)$

Quadratic Loss  \[ \ell(y_h, y) = (y_h - y)^2 \]

Binary Loss  \[ \ell(y_h, y) = \begin{cases} 0 & \text{if } y_h = y \\ 1 & \text{if } y_h \neq y \end{cases} \]

Hinge Loss  \[ \ell(y_h, y) = \max\{0, 1 - y_h y\} \]
Choosing the Loss Function

Let \( y_h := h_w(x) \)

**Quadratic Loss** \[ \ell(y_h, y) = (y_h - y)^2 \]

**Binary Loss** \[ \ell(y_h, y) = \begin{cases} 0 & \text{if } y_h = y \\ 1 & \text{if } y_h \neq y \end{cases} \]

**Hinge Loss** \[ \ell(y_h, y) = \max\{0, 1 - y_h y\} \]
Choosing the Loss Function

Let \( y_h := h_w(x) \)

**Quadratic Loss**
\[
\ell(y_h, y) = (y_h - y)^2
\]

**Binary Loss**
\[
\ell(y_h, y) = \begin{cases} 
0 & \text{if } y_h = y \\
1 & \text{if } y_h \neq y 
\end{cases}
\]

**Hinge Loss**
\[
\ell(y_h, y) = \max\{0, 1 - y_hy\}
\]

**EXE:** Plot the binary and hinge loss function in when \( y = -1 \).
Loss Functions

Is a notion of Loss enough?

What happens when we do not have enough data?
Loss Functions

The Training Problem

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell (h_w(x^i), y^i)$$

Is a notion of Loss enough?

What happens when we do not have enough data?
Overfitting and Model Complexity

Fitting 1\textsuperscript{st} order polynomial

\[ h_w = \langle w, x \rangle \]

\[ w^* = \arg \min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} (h_w(x^i) - y^i)^2 \]
Overfitting and Model Complexity

Fitting 1\textsuperscript{st} order polynomial

\[ h_w = w_0 + w_1 x + w_2 x^2 \]

\[ w^* = \arg \min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} (h_w(x^i) - y^i)^2 \]
Overfitting and Model Complexity

Fitting 3rd order polynomial

$$h_w = \sum_{i=0}^{3} w_i x^i$$

$$w^* = \arg\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} (h_w(x^i) - y^i)^2$$
Overfitting and Model Complexity

Fitting 9\(^{th}\) order polynomial

\[ h_w = \sum_{i=0}^{9} w_i x^i \]

\[ w^* = \arg\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \left( h_w(x^i) - y^i \right)^2 \]
Regularization

Regularizer Functions
\[ R : \mathbb{R}^d \rightarrow \mathbb{R}_+ \]
\[ w \rightarrow R(w) \]

General Training Problem
\[ \min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell (h_w(x^i), y^i) + \lambda R(w) \]
Regularization

Regularizer Functions

\[ R : \mathbb{R}^d \rightarrow \mathbb{R}_+ \]

\[ w \rightarrow R(w) \]

General Training Problem

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell \left( h_w(x^i), y^i \right) + \lambda R(w)
\]

Goodness of fit, fidelity term ... etc
**Regularization**

**Regularizer Functions**

\[ R : \mathbb{R}^d \to \mathbb{R}_+ \]

\[ w \to R(w) \]

**General Training Problem**

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell(h_w(x^i), y^i) + \lambda R(w)
\]

- Goodness of fit, fidelity term ...etc
- Penlizes complexity
Regularization

**Regularizer Functions**

\[ R : \mathbb{R}^d \rightarrow \mathbb{R}_+ \]

\[ w \rightarrow R(w) \]

Controls tradeoff between fit and complexity

**General Training Problem**

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell (h_w(x^i), y^i) + \lambda R(w)
\]

Goodness of fit, fidelity term ...etc

Penlizes complexity
Regularization

**Regularizer Functions**

\[ R : \mathbb{R}^d \rightarrow \mathbb{R}_+ \]

\[ w \rightarrow R(w) \]

Controls tradeoff between fit and complexity

**General Training Problem**

\[ \min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell \left( h_w(x^i), y^i \right) + \lambda R(w) \]

- Goodness of fit, fidelity term ... etc
- Penalizes complexity

**Exe:**

\[ R(w) = \|w\|_2^2, \ |w|_1, \ |w|_p, \ \text{other norms} \ldots \]
Overfitting and Model Complexity

Fitting $k^{\text{th}}$ order polynomial

\[ h_w = \sum_{i=0}^{k} w_i x^i \]

\[ w^* = \arg \min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \left( h_w(x^i) - y^i \right)^2 + \lambda \| w \|_2^2 \]
Overfitting and Model Complexity

For $\lambda$ big enough, the solution is a 2$^{nd}$ order polynomial

Fitting $k^{th}$ order polynomial

$$h_w = \sum_{i=0}^{k} w_i x^i$$

$$w^* = \arg \min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} (h_w(x^i) - y^i)^2 + \lambda \|w\|_2^2$$
Exe: Ridge Regression

Linear hypothesis
\[ h_w(x) = \langle w, x \rangle \]

L2 regularizer
\[ R(w) = \|w\|_2^2 \]

L2 loss
\[ \ell(y_h, y) = (y_h - y)^2 \]

Ridge Regression
\[ \min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} (y^i - \langle w, x^i \rangle)^2 + \lambda \|w\|_2^2 \]
Exe: Support Vector Machines

Linear hypothesis

\[ h_w(x) = \langle w, x \rangle \]

L2 regularizer

\[ R(w) = ||w||_2^2 \]

Hinge loss

\[ \ell(y_h, y) = \max\{0, 1 - y_h y\} \]

SVM with soft margin

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \max\{0, 1 - y^i \langle w, x^i \rangle\} + \lambda ||w||_2^2
\]
Exe: Logistic Regression

Linear hypothesis
\[ h_w(x) = \langle w, x \rangle \]

L2 regularizer
\[ R(w) = \|w\|_2^2 \]

Logistic loss
\[ \ell(y_h, y) = \ln(1 + e^{-yy_h}) \]

Logistic Regression
\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ln(1 + e^{-y_i \langle w, x^i \rangle}) + \lambda \|w\|_2^2
\]
The Machine Learners Job

(1) Get the labeled data: \((x^1, y^1), \ldots, (x^n, y^n)\)
The Machine Learners Job

(1) Get the labeled data: \((x^1, y^1), \ldots, (x^n, y^n)\)

(2) Choose a parametrization for hypothesis: \(h_w(x)\)
The Machine Learners Job

(1) Get the labeled data: \((x^1, y^1), \ldots, (x^n, y^n)\)

(2) Choose a parametrization for hypothesis: \(h_w(x)\)

(3) Choose a loss function: \(\ell(h_w(x), y) \geq 0\)
The Machine Learners Job

(1) Get the labeled data: \((x^1, y^1), \ldots, (x^n, y^n)\)

(2) Choose a parametrization for hypothesis: \(h_w(x)\)

(3) Choose a loss function: \(\ell(h_w(x), y) \geq 0\)

(4) Solve the *training problem*:

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell (h_w(x^i), y^i) + \lambda R(w)
\]
The Machine Learners Job

(1) Get the labeled data: \((x^1, y^1), \ldots, (x^n, y^n)\)

(2) Choose a parametrization for hypothesis: \(h_w(x)\)

(3) Choose a loss function: \(\ell(h_w(x), y) \geq 0\)

(4) Solve the training problem:

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell(h_w(x^i), y^i) + \lambda R(w)
\]

(5) Test and cross-validate. If fail, go back a few steps
The Machine Learners Job

(1) Get the labeled data: \((x^1, y^1), \ldots, (x^n, y^n)\)

(2) Choose a parametrization for hypothesis: \(h_w(x)\)

(3) Choose a loss function: \(\ell(h_w(x), y) \geq 0\)

(4) Solve the training problem:

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \ell \left(h_w(x^i), y^i\right) + \lambda R(w)
\]

(5) Test and cross-validate. If fail, go back a few steps
The Statistical Learning Problem: The hard truth

Do we really care if the loss \( \ell(h_w(x^i), y^i) \) is small on the *known* labelled data pairs \((x^i, y^i)\)? **Nope**

We really want to have a small loss on new unlabelled Observations!

Assume data sampled \((x, y) \sim \mathcal{D}\) where \(\mathcal{D}\) is an unknown distribution
The Statistical Learning Problem: The hard truth

The statistical learning problem:
Minimize the expected loss over an unknown expectation

\[
\min_{w \in \mathbb{R}^d} \mathbb{E}_{(x,y) \sim D} [\ell (h_w(x), y)]
\]

Variance of sample mean:

\[
\left| \mathbb{E}_{(x,y) \sim D} [\ell (h_w(x), y)] - \frac{1}{n} \sum_{i=1}^{n} \ell (h_w(x_i), y_i) \right| = O \left( \frac{1}{n} \right)
\]