Unsupervised and Online Learning

Twitter API usecase

Sébastien Loustau, CEO @artfact

Nantes Machine Learning Meetup, 30 Mai 2016

Contents

Contents

Machine learning

$$x \longrightarrow \boxed{\texttt{nature}} \longrightarrow y$$

Machine learning

$$x \longrightarrow \boxed{\text{nature}} \longrightarrow y$$

- **prédire** la réponse y à partir de x,

Machine learning

$$x \longrightarrow \boxed{\text{nature}} \longrightarrow y$$

- **prédire** la réponse y à partir de x,
- **comprendre** le lien entre x et y.

Machine learning

$$x \longrightarrow \boxed{\text{nature}} \longrightarrow y$$

- **prédire** la réponse y à partir de x,
- **comprendre** le lien entre x et y.

$$x \longrightarrow \boxed{\text{algorithm}} \longrightarrow \widehat{y}$$

~~Machine~~ Supervised learning

$$x \longrightarrow \boxed{\text{nature}} \longrightarrow y$$

- **prédire** la réponse y à partir de x,
- **comprendre** le lien entre x et y.

$$x \longrightarrow \boxed{\text{algorithm}} \longrightarrow \widehat{y}$$

Today : unsupervised learning

$$\boxed{\text{nature}} \longrightarrow x$$

Today : unsupervised learning

$$\boxed{\text{nature}} \longrightarrow x$$

- décrire les observations x,

Today : unsupervised learning

$$\boxed{\text{nature}} \longrightarrow x$$

- décrire les observations x,
- réduire la dimension de x,

Today : unsupervised learning

$$\boxed{\text{nature}} \longrightarrow x$$

- décrire les observations x,
- réduire la dimension de x,
- grouper les observations x.

Today : unsupervised learning

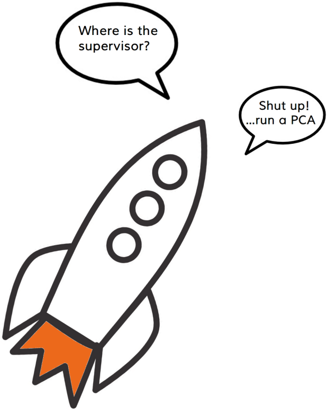
$$\boxed{\text{nature}} \longrightarrow x$$

- décrire les observations x,
- réduire la dimension de x,
- grouper les observations x.

Algorithms

PCA, k-means, spectral k-means, hierarchical clustering, Gaussian mixtures, Principal curves analysis, word2vect...

Qu'est-ce que ça change ?



UNSUPERVISED LEARNING

Supervised routine

Supervised routine

$$x \longrightarrow \boxed{\text{algorithm}} \longrightarrow \widehat{y}$$

Supervised routine

$$x \longrightarrow \boxed{\text{algorithm}} \longrightarrow \widehat{y}$$

- Build a model from a training sample $\{(X_i, Y_i), i = 1, \ldots, n\}$.

Supervised routine

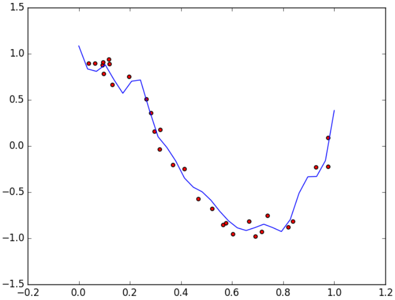
$$x \longrightarrow \boxed{\text{algorithm}} \longrightarrow \widehat{y}$$

- Build a model from a training sample $\{(X_i, Y_i), i = 1, \ldots, n\}$.
- Observe new x and predict $\hat{y}$ as above.

Supervised routine

$$x \longrightarrow \boxed{\text{algorithm}} \longrightarrow \widehat{y}$$

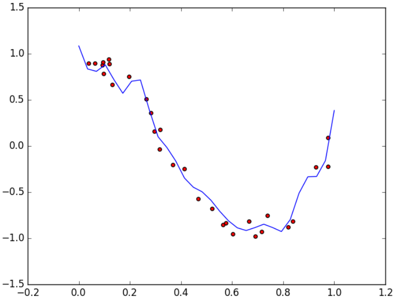
- Build a model from a training sample $\{(X_i, Y_i), i = 1, \ldots, n\}$.
- Observe new x and predict $\hat{y}$ as above.
- Problem : overfitting !



Supervised routine

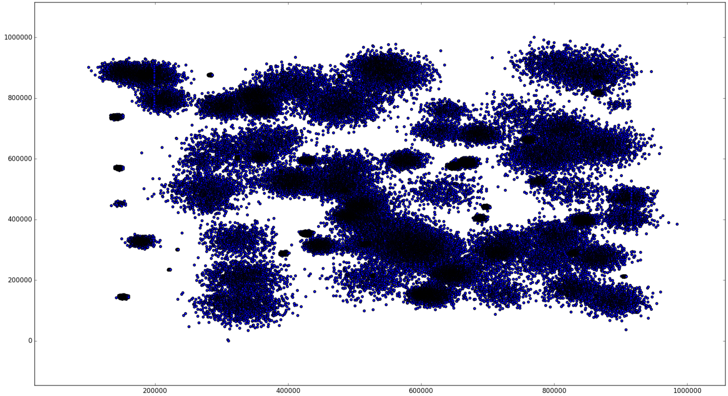
$$x \longrightarrow \boxed{\text{algorithm}} \longrightarrow \widehat{y}$$

- Build a model from a training sample $\{(X_i, Y_i), i = 1, \ldots, n\}$.
- Observe new x and predict $\hat{y}$ as above.
- Problem : overfitting !



- Solution : Training set + test set, Leave-One-Out, V-fold Cross validation.

Unsupervised : science or art ?



Statistical Learning vs Online Learning

Statistical Learning

We observe a training set $\mathcal{D}_n = \{(X_i, Y_i), i = 1, \ldots, n\}$.

Statistical Learning vs Online Learning

Statistical Learning

We observe a training set $\mathcal{D}_n = \{(X_i, Y_i), i = 1, \ldots, n\}$. New x arrives. We build a model/algorithm thanks to $\mathcal{D}_n$ and predict $\hat{y}$.

Statistical Learning vs Online Learning

Statistical Learning

We observe a training set $\mathcal{D}_n = \{(X_i, Y_i), i = 1, \ldots, n\}$. New x arrives. We build a model/algorithm thanks to $\mathcal{D}_n$ and predict $\hat{y}$.

Online Learning

Data arrives sequentially.

Statistical Learning vs Online Learning

Statistical Learning

We observe a training set $\mathcal{D}_n = \{(X_i, Y_i), i = 1, \ldots, n\}$. New x arrives. We build a model/algorithm thanks to $\mathcal{D}_n$ and predict $\hat{y}$.

Online Learning

Data arrives sequentially. At each time t, we want to make a decision based on past observations.

Statistical Learning vs Online Learning

Statistical Learning

We observe a training set $\mathcal{D}_n = \{(X_i, Y_i), i = 1, \ldots, n\}$. New x arrives. We build a model/algorithm thanks to $\mathcal{D}_n$ and predict $\hat{y}$.

Online Learning

Data arrives sequentially. At each time t, we want to make a decision based on past observations. **No assumption over the data mechanism**.

Game with expert advices

$$y = \begin{matrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{matrix}$$

Game with expert advices

$$y = \begin{matrix} \color{red}{0} \\ \color{red}{0} \\ \color{red}{1} \\ \color{red}{1} \\ \color{red}{1} \\ \color{red}{1} \\ \color{red}{0} \\ \color{red}{1} \\ \color{red}{0} \\ \color{red}{\vdots} \\ \color{red}{0} \end{matrix}, \text{experts} = \begin{matrix} 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & \cdots & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & \cdots & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & \cdots & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & \cdots & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & \cdots & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & \cdots & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & \cdots & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 1 & 1 & 0 & 0 & 0 & \cdots & 0 & 1 \end{matrix}$$

Game with expert advices

$$y = \begin{matrix} \color{red}{0} \\ \color{red}{0} \\ \color{red}{1} \\ \color{red}{1} \\ \color{red}{1} \\ \color{red}{1} \\ \color{red}{0} \\ \color{red}{1} \\ \color{red}{0} \\ \color{red}{\vdots} \\ \color{red}{0} \end{matrix}, \text{experts} = \begin{matrix} 1 & 0 & 0 & \color{red}{0} & 0 & \cdots & 0 & 1 \\ 0 & 1 & 0 & \color{red}{0} & 0 & \cdots & 1 & 1 \\ 0 & 1 & 1 & \color{red}{1} & 1 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \color{red}{1} & 0 & \cdots & 1 & 1 \\ 1 & 1 & 0 & \color{red}{1} & 1 & \cdots & 1 & 0 \\ 0 & 1 & 0 & \color{red}{1} & 1 & \cdots & 0 & 1 \\ 1 & 1 & 0 & \color{red}{0} & 0 & \cdots & 0 & 0 \\ 1 & 0 & 0 & \color{red}{1} & 0 & \cdots & 0 & 1 \\ 1 & 1 & 0 & \color{red}{0} & 0 & \cdots & 0 & 1 \\ \vdots & \vdots & \vdots & \color{red}{\vdots} & \vdots & \cdots & \vdots & \vdots \\ 1 & 1 & 0 & \color{red}{0} & 0 & \cdots & 0 & 1 \end{matrix}$$

Keep or Kill + Majority vote algorithm

$$
\begin{array}{cccccccccc}
 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
\color{red}{\vdots} & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
 & & & & & & \cdots & &
\end{array}
$$

Keep or Kill + Majority vote algorithm

$$
\begin{array}{cccccccc}
0 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
 & & & & & & \cdots & &
\end{array}
$$

Keep or Kill + Majority vote algorithm

$$
\begin{array}{ccccccccc}
0 & x & 0 & 0 & 0 & 0 & \cdots & 0 & x \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
 & & & & & & \cdots & &
\end{array}
$$

Keep or Kill + Majority vote algorithm

$$
\begin{array}{ccccccccc}
\color{red}{0} & x & 0 & 0 & 0 & 0 & \cdots & 0 & x \\
 & x & 1 & 0 & 0 & 0 & \cdots & 1 & x \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
\color{red}{\vdots} & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
 & & & & & & \cdots & &
\end{array}
$$

Keep or Kill + Majority vote algorithm

$$
\begin{array}{cccccccccc}
0 & & x & 0 & 0 & 0 & 0 & \cdots & 0 & x \\
0 & & x & 1 & 0 & 0 & 0 & \cdots & 1 & x \\
 & & & & & & & \cdots & & \\
 & & & & & & & \cdots & & \\
 & & & & & & & \cdots & & \\
 & & & & & & & \cdots & & \\
 & & & & & & & \cdots & & \\
 & & & & & & & \cdots & & \\
 & & & & & & & \cdots & & \\
\vdots & & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
 & & & & & & & \cdots & &
\end{array}
$$

Keep or Kill + Majority vote algorithm

$$\begin{array}{cccccccccc}
\color{red}{0} & x & 0 & 0 & 0 & 0 & \cdots & 0 & x \\
\color{red}{0} & x & x & 0 & 0 & 0 & \cdots & x & x \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
\color{red}{\vdots} & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
 & & & & & & \cdots & &
\end{array}$$

Keep or Kill + Majority vote algorithm

$$
\begin{array}{cccccccccc}
0 & x & 0 & 0 & 0 & 0 & \cdots & 0 & x \\
0 & x & x & 0 & 0 & 0 & \cdots & x & x \\
 & x & x & 1 & 1 & 1 & \cdots & x & x \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
 & & & & & & \cdots & &
\end{array}
$$

Keep or Kill + Majority vote algorithm

$$
\begin{array}{ccccccccc}
\color{red}{0} & x & 0 & 0 & 0 & 0 & \cdots & 0 & x \\
\color{red}{0} & x & x & 0 & 0 & 0 & \cdots & x & x \\
\color{red}{1} & x & x & 1 & 1 & 1 & \cdots & x & x \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
\color{red}{\vdots} & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
 & & & & & & \cdots & &
\end{array}
$$

Keep or Kill + Majority vote algorithm

$$
\begin{array}{cccccccccc}
0 & & x & 0 & 0 & 0 & 0 & \cdots & 0 & x \\
0 & & x & x & 0 & 0 & 0 & \cdots & x & x \\
1 & & x & x & 1 & 1 & 1 & \cdots & x & x \\
 & & x & x & 0 & 1 & 0 & \cdots & x & x \\
 & & & & & & & \cdots & & \\
 & & & & & & & \cdots & & \\
 & & & & & & & \cdots & & \\
 & & & & & & & \cdots & & \\
 & & & & & & & \cdots & & \\
\vdots & & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
 & & & & & & & \cdots & &
\end{array}
$$

Keep or Kill + Majority vote algorithm

$$
\begin{array}{ccccccccc}
\color{red}{0} & x & 0 & 0 & 0 & 0 & \cdots & 0 & x \\
\color{red}{0} & x & x & 0 & 0 & 0 & \cdots & x & x \\
\color{red}{1} & x & x & 1 & 1 & 1 & \cdots & x & x \\
\color{red}{1} & x & x & 0 & 1 & 0 & \cdots & x & x \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
\color{red}{\vdots} & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
 & & & & & & \cdots & &
\end{array}
$$

Keep or Kill + Majority vote algorithm

$$
\begin{array}{ccccccccc}
0 & x & 0 & 0 & 0 & 0 & \cdots & 0 & x \\
0 & x & x & 0 & 0 & 0 & \cdots & x & x \\
1 & x & x & 1 & 1 & 1 & \cdots & x & x \\
1 & x & x & x & 1 & x & \cdots & x & x \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
 & & & & & & \cdots & &
\end{array}
$$

Keep or Kill + Majority vote algorithm

$$
\begin{array}{ccccccccc}
0 & x & 0 & 0 & 0 & 0 & \cdots & 0 & x \\
0 & x & x & 0 & 0 & 0 & \cdots & x & x \\
1 & x & x & 1 & 1 & 1 & \cdots & x & x \\
1 & x & x & x & 1 & x & \cdots & x & x \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
 & & & & & & \cdots & & \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
 & & & & & & \cdots & &
\end{array}
$$

Learn from your mistakes phenomenon

Learn from your mistakes phenomenon

Let W_k = `survivors after k mistakes`.

Learn from your mistakes phenomenon

Let $W_k =$ `survivors after k mistakes`. Then:

$$W_k \leq \frac{W_{k-1}}{2}$$

Learn from your mistakes phenomenon

Let W_k = `survivors after k mistakes`. Then:

$$W_k \leq \frac{W_{k-1}}{2} \leq \frac{W_{k-2}}{4}$$

Learn from your mistakes phenomenon

Let $W_k = \texttt{survivors after k mistakes}$. Then:

$$W_k \leq \frac{W_{k-1}}{2} \leq \frac{W_{k-2}}{4} \leq \ldots \leq \frac{W_0}{2^k}$$

Learn from your mistakes phenomenon

Let $W_k = \texttt{survivors after k mistakes}$. Then:

$$W_k \leq \frac{W_{k-1}}{2} \leq \frac{W_{k-2}}{4} \leq \ldots \leq \frac{W_0}{2^k} = 2^{-k} N.$$

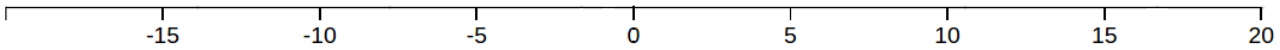
Learn from your mistakes phenomenon

Let $W_k = \texttt{survivors after k mistakes}$. Then:

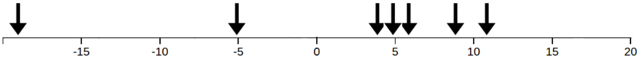
$$W_k \leq \frac{W_{k-1}}{2} \leq \frac{W_{k-2}}{4} \leq \ldots \leq \frac{W_0}{2^k} = 2^{-k} N.$$

We end up with $k \leq C \log N$!

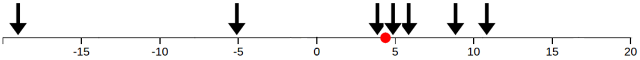
General case



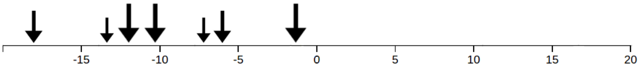
General case



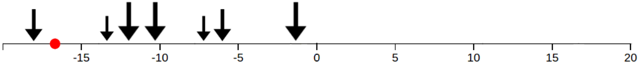
General case



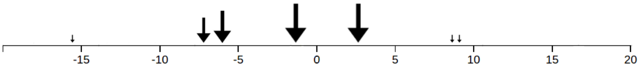
General case



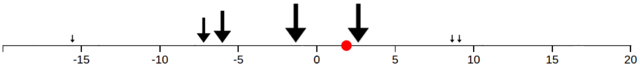
General case



General case



General case



Mathematical formalism

- $(y_t)_{t=1}^{T}$, $y_t \in \mathbb{R}$ a sequence of inputs,

Mathematical formalism

- $(y_t)_{t=1}^T$, $y_t \in \mathbb{R}$ a sequence of inputs,
- $\{p_{k,t}, k = 1, \dots, N\}_{t=1}^T$ expert advices.

Mathematical formalism

- $(y_t)_{t=1}^{T}$, $y_t \in \mathbb{R}$ a sequence of inputs,
- $\{p_{k,t}, k = 1, \ldots, N\}_{t=1}^{T}$ expert advices.

Game protocol

$\forall t = 1, \ldots, T$:

1. Observe $p_{k,t}$, $k = 1, \ldots, N$.

Mathematical formalism

- $(y_t)_{t=1}^T$, $y_t \in \mathbb{R}$ a sequence of inputs,
- $\{p_{k,t}, k = 1, \dots, N\}_{t=1}^T$ expert advices.

Game protocol

$\forall t = 1, \dots, T$:

1. Observe $p_{k,t}$, $k = 1, \dots, N$.
2. Predict $\hat{y}_t$.

Mathematical formalism

- $(y_t)_{t=1}^T$, $y_t \in \mathbb{R}$ a sequence of inputs,
- $\{p_{k,t}, k = 1, \ldots, N\}_{t=1}^T$ expert advices.

Game protocol

$\forall t = 1, \ldots, T$:

1. Observe $p_{k,t}$, $k = 1, \ldots, N$.
2. Predict $\hat{y}_t$.
3. Observe y_t and pay
   - $\ell(\hat{y}_t, y_t)$ for your algorithm,
   - $\ell(p_{k,t}, y_t)$ for expert number k.

Mathematical formalism

- $(y_t)_{t=1}^T$, $y_t \in \mathbb{R}$ a sequence of inputs,
- $\{p_{k,t}, k = 1, \dots, N\}_{t=1}^T$ expert advices.

Game protocol

$\forall t = 1, \dots, T$:

1. Observe $p_{k,t}$, $k = 1, \dots, N$.
2. Predict $\hat{y}_t$.
3. Observe y_t and pay
   - $\ell(\hat{y}_t, y_t)$ for your algorithm,
   - $\ell(p_{k,t}, y_t)$ for expert number k.

Popular loss functions include $\ell(\hat{y}, y) = (\hat{y} - y)^2$, $|\hat{y} - y|$, $(1 - \hat{y}y)_+$.

General result

In the general case, we want to control the "regret" :

$$\sum_{t=1}^{T} (\hat{y}_t - y_t)^2 - \sum_{t=1}^{T} (y_t^\star - y_t)^2,$$

where $(y_t^\star)$ is the best expert.

General result

In the general case, we want to control the "regret" :

$$\sum_{t=1}^{T} (\hat{y}_t - y_t)^2 - \sum_{t=1}^{T} (y_t^\star - y_t)^2,$$

where $(y_t^\star)$ is the best expert.

Cesa-Bianci & Lugosi, 2006

If $\ell(\cdot, z)$ is convex and $[0,1]$-bounded :

$$\sum_{t=1}^{T} \ell(\hat{y}_t, y_t) - \min_{k=1,\dots N} \sum_{t=1}^{T} \ell(p_{k,t}, y_t) \leq \frac{\log N}{\lambda} + \frac{\lambda T}{8},$$

General result

In the general case, we want to control the "regret" :

$$\sum_{t=1}^{T} (\hat{y}_t - y_t)^2 - \sum_{t=1}^{T} (y_t^\star - y_t)^2,$$

where $(y_t^\star)$ is the best expert.

Cesa-Bianci & Lugosi, 2006

If $\ell(\cdot, z)$ is convex and $[0,1]$-bounded :

$$\sum_{t=1}^{T} \ell(\hat{y}_t, y_t) - \min_{k=1,\dots N} \sum_{t=1}^{T} \ell(p_{k,t}, y_t) \leq \frac{\log N}{\lambda} + \frac{\lambda T}{8},$$

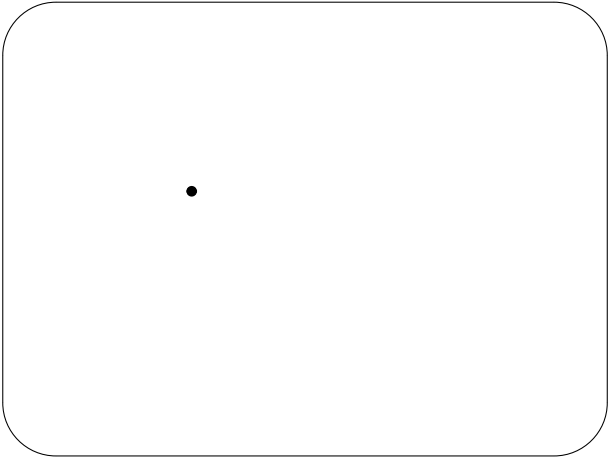
where :

$$\hat{y}_t = \sum_{k=1}^{N} \frac{e^{-\lambda \sum_{u=1}^{t-1} \ell(p_{k,u}, y_u)}}{W_{t-1}} p_{k,t}, \ \forall t = 1, \dots, T.$$

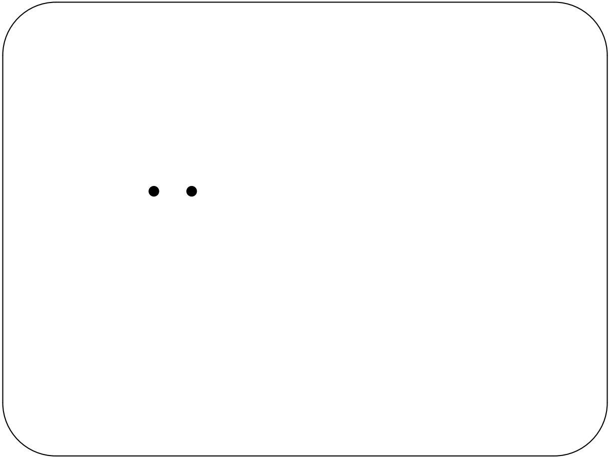
Contents

The problem of Online Clustering

The problem of Online Clustering

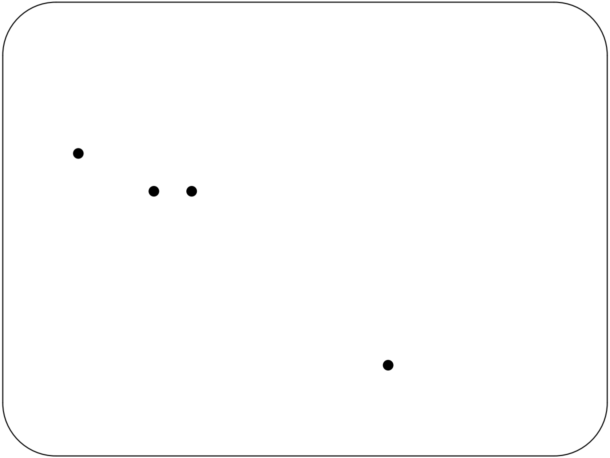


The problem of Online Clustering

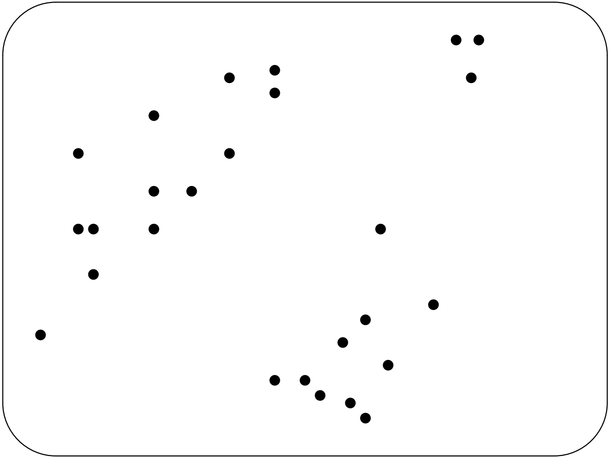


The problem of Online Clustering

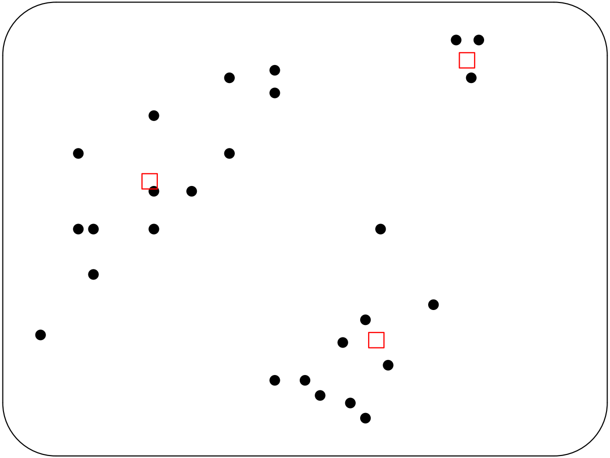
The problem of Online Clustering



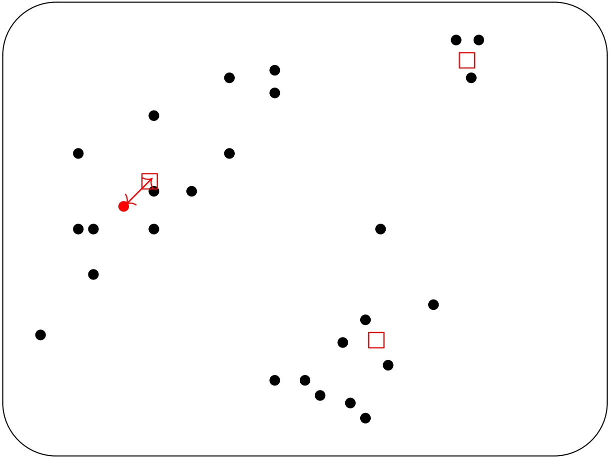
The problem of Online Clustering



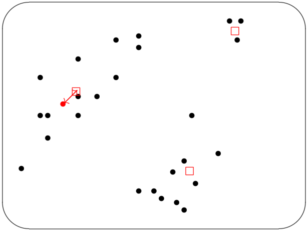
The problem of Online Clustering



The problem of Online Clustering

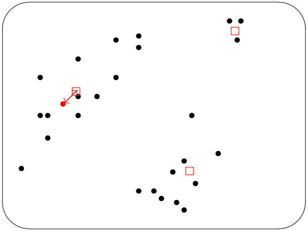


The problem of Online Clustering



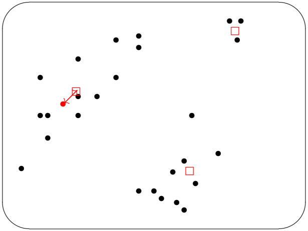
- Principle : clustering as prediction !

The problem of Online Clustering



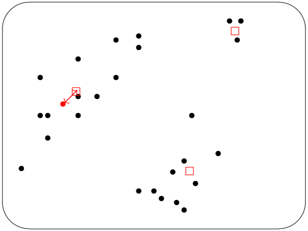
- Principle : clustering as prediction !
- Sparsity assumption : points are grouped into s clusters.

The problem of Online Clustering



- Principle : clustering as prediction !
- Sparsity assumption : points are grouped into s clusters.
- Use PAC-Bayesian regularization to choose the number of clusters.

The problem of Online Clustering

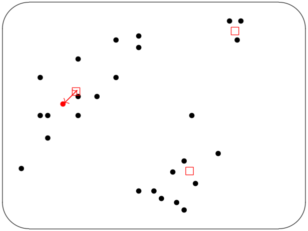


- Principle : clustering as prediction !
- Sparsity assumption : points are grouped into s clusters.
- Use PAC-Bayesian regularization to choose the number of clusters.

We prove new kind of sparsity regret bounds:

$$\sum_{t=1}^{T} \ell(\hat{\mathbf{c}}_t, x_t) - \inf_{\mathbf{c} \in \mathbb{R}^{dp}} \left\{ \sum_{t=1}^{T} \ell(\mathbf{c}, x_t) + \lambda |\mathbf{c}|_0 \right\},$$

The problem of Online Clustering



- Principle : clustering as prediction !
- Sparsity assumption : points are grouped into s clusters.
- Use PAC-Bayesian regularization to choose the number of clusters.

We prove new kind of sparsity regret bounds:

$$\sum_{t=1}^{T} \ell(\hat{\mathbf{c}}_t, x_t) - \inf_{\mathbf{c} \in \mathbb{R}^{dp}} \left\{ \sum_{t=1}^{T} \ell(\mathbf{c}, x_t) + \lambda |\mathbf{c}|_0 \right\},$$

where $|\mathbf{c}|_0 = \mathrm{card}\{j = 1, \ldots, p : c_j \neq 0_{\mathbb{R}^d}\}$ and

$$\ell(\mathbf{c}, x) = \min_{j=1,\ldots,p} \|c_j - x\|_2^2.$$

Contents

Natural Langage Processing

```
A Federal Judge May Have Just Handed Democrats Victory
By Giving Trump The Worst News Ever
https://t.co/fuNnDwLJNL
```

Natural Langage Processing

```
A Federal Judge May Have Just Handed Democrats Victory
By Giving Trump The Worst News Ever
https://t.co/fuNnDwLJNL
@joshbranson:  Fine, I agree with Trump on one thing:
Billy Joel is awesome.  https://t.co/EB8KmsJzW6
```

Natural Langage Processing

```
A Federal Judge May Have Just Handed Democrats Victory
By Giving Trump The Worst News Ever
https://t.co/fuNnDwLJNL
@joshbranson:  Fine, I agree with Trump on one thing:
Billy Joel is awesome.  https://t.co/EB8KmsJzW6
@realDonaldTrump Trump:  Evil supervillan!
https://t.co/Xswmh39EYa
```

Natural Langage Processing

```
A Federal Judge May Have Just Handed Democrats Victory
By Giving Trump The Worst News Ever
https://t.co/fuNnDwLJNL
@joshbranson:  Fine, I agree with Trump on one thing:
Billy Joel is awesome.  https://t.co/EB8KmsJzW6
@realDonaldTrump Trump:  Evil supervillan!
https://t.co/Xswmh39EYa
```

Meaning ? Sentiment ? From words to vectors ?

Vectorization

The main challenge of NLP is vectorization of words.

Vectorization

The main challenge of NLP is vectorization of words.

- **One hot representation** : each word is a vector with one 1 and A LOT of zeroes :

Vectorization

The main challenge of NLP is vectorization of words.

- **One hot representation** : each word is a vector with one 1 and A LOT of zeroes :
  - $\texttt{hotel} = (0, 0, 0, 1, 0, 0, \ldots, 0)$
  - $\texttt{Budapest} = (0, 0, 1, 0, 0, 0, \ldots, 0)$

Vectorization

The main challenge of NLP is vectorization of words.

- **One hot representation** : each word is a vector with one 1 and A LOT of zeroes :
  - `hotel` $= (0, 0, 0, 1, 0, 0, \ldots, 0)$
  - `Budapest` $= (0, 0, 1, 0, 0, 0, \ldots, 0)$
- Problem since $\langle \mathrm{hotel}, \mathrm{motel} \rangle = 0$.

Vectorization

The main challenge of NLP is vectorization of words.

- **One hot representation** : each word is a vector with one 1 and A LOT of zeroes :
  - `hotel` $= (0, 0, 0, 1, 0, 0, \dots, 0)$
  - `Budapest` $= (0, 0, 1, 0, 0, 0, \dots, 0)$
- Problem since $\langle \text{hotel}, \text{motel} \rangle = 0$.
- Gigantic dimension !

Capture syntactic and semantic

Represent a word by means of its neighbors.

Capture syntactic and semantic

 Represent a word by means of its neighbors.

- Given a corpus:
  `I love swimming and dancing.  I love NLP.`

Capture syntactic and semantic

Represent a word by means of its neighbors.

- Given a corpus:
  `I love swimming and dancing.  I love NLP.`
- Choose a window-size.

Capture syntactic and semantic

Represent a word by means of its neighbors.

- Given a corpus:
  `I love swimming and dancing.  I love NLP.`
- Choose a window-size.
- Compute the **coocurence matrix**:

	I	love	dancing	swimming	NLP	and
I	2	2	1	1	1	0
love	2	2	1	1	1	1
dancing	1	1	1	1	0	1
swimming	1	1	1	1	0	1
NLP	1	1	0	0	0	1
and	0	1	1	1	0	1

Word2vect

- Other solution : directly learn low dimensional word vectors.

Word2vect

- Other solution : directly learn low dimensional word vectors.
- Idea : predict, given each word one at a time, the word to the left and the word to the right.

Word2vect

- Other solution : directly learn low dimensional word vectors.
- Idea : predict, given each word one at a time, the word to the left and the word to the right.
- Maximize the likelihood of any context word given the center word:

$$\max_{\theta} \frac{1}{T} \sum_{t=1}^{T} \sum_{-m \leq j \leq m} \log \mathbb{P}(w_{t+j} | w_t),$$

Word2vect

- Other solution : directly learn low dimensional word vectors.
- Idea : predict, given each word one at a time, the word to the left and the word to the right.
- Maximize the likelihood of any context word given the center word:

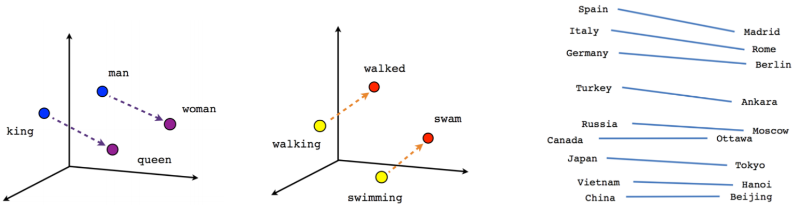
$$\max_{\theta} \frac{1}{T} \sum_{t=1}^{T} \sum_{-m \leq j \leq m} \log \mathbb{P}(w_{t+j} | w_t),$$

where

$$p(o|c) = \frac{\exp\left(\langle u_{\mathrm{o}}, v_{\mathrm{c}} \rangle\right)}{\sum_{w=1}^{W} \exp\left(\langle u_{\mathrm{w}}, v_{\mathrm{c}} \rangle\right)}.$$

- Optimization with Online Stochastic Gradient Descent.

Word2vect : linear relationships



Male-Female Verb tense Country-Capital