Introduction au Machine learning et à la classification supervisée

Agathe Guilloux

Introduction

Outline I

Organisation du cours

- Programme
  - Jour 1 : big data / data science + notebook “ds_with_python”
  - Jour 2 et 3 : algorithmes de classification + “classification_gro”
  - Jour 4 : text mining + “notebook_imdb”
- Evaluation
  - par équipe de 3, analyser le jeu de données “amazon reviews” https://www.kaggle.com/bittlingmayer/amazonreviews
  - soutenances de 15 minutes avec une présentation le 26/09.

Big data / Data Science

Vocabulaire et buzz words

- Statistique
- Intelligence artificielle (AI)
- Machine Learning (ML)
- Big Data
- Data Science
- Deep Learning (DL)
- et ensuite ??

Wikipedia

Big data

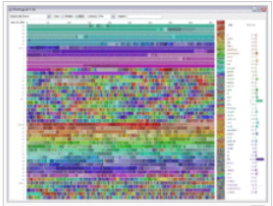
From Wikipedia, the free encyclopedia

This article is about large collections of data. For the band, see Big Data (band).

Big data[1][2] is the term for a collection of data sets so large and complex that it becomes difficult to process using on-hand database management tools or traditional data processing applications. The challenges include capture, curation, storage,[3] search, sharing, transfer, analysis[4] and visualization. The trend to larger data sets is due to the additional information derivable from analysis of a single large set of related data, as compared to separate smaller sets with the same total amount of data, allowing correlations to be found to "spot business trends, determine quality of research, prevent diseases, link legal citations, combat crime, and determine real-time roadway traffic conditions."[5][6][7]

As of 2012, limits on the size of data sets that are feasible to process in a reasonable amount of time were on the order of exabytes of data.[8] Scientists regularly encounter limitations due to large data sets in many areas, including meteorology, genomics,[9] connectomics, complex physics simulations,[10] and biological and environmental research.[11] The limitations also affect Internet search, finance and business informatics. Data sets grow in size in part because they are increasingly being gathered by ubiquitous information-sensing mobile devices, aerial sensory technologies (remote sensing), software logs, cameras, microphones, radio-frequency identification readers, and wireless sensor networks.[12][13] The world's technological per-capita capacity to store information has roughly doubled every 40 months since the 1980s;[14] as of 2012, every day 2.5 exabytes (2.5×10^{18}) of data were created.[15] The challenge for large enterprises is determining who should own big data initiatives that straddle the entire organization.[16]

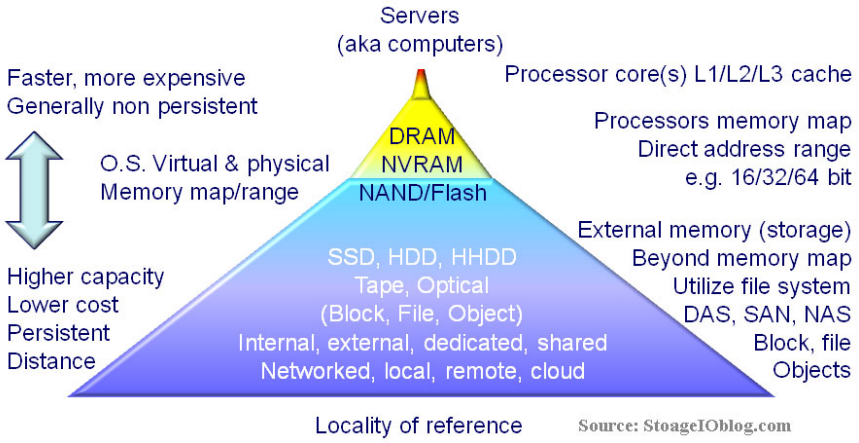
Big data is difficult to work with using most relational database management systems and desktop statistics and visualization packages, requiring instead "massively parallel software running on tens, hundreds, or even thousands of servers".[17] What is considered "big data" varies depending on the capabilities of the organization managing the set, and on the capabilities of the applications that are traditionally used to process and analyze the data set in its domain. "For some organizations, facing hundreds of gigabytes of data for the first time may trigger a need to reconsider data management options. For others, it may take tens or hundreds of terabytes before data size becomes a significant consideration."[18]

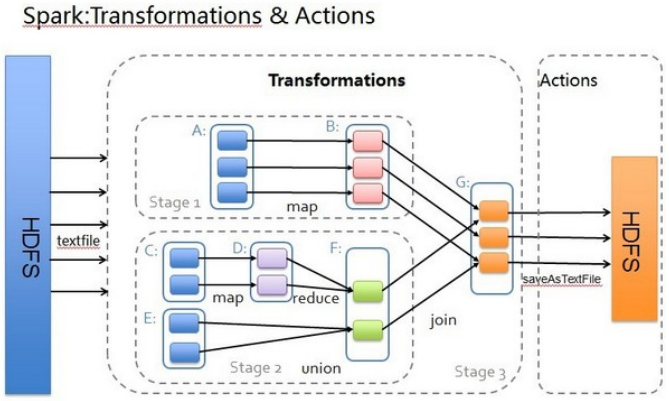


A visualization created by IBM of Wikipedia edits. At multiple terabytes in size, the text and images of Wikipedia are a classic example of big data.

- **Le Big data** est un terme désignant des ensembles de données si importants et complexes qu'il devient difficile de les analyser en utilisant des applications de traitement de données traditionnelles.
- **La Statistique** est l'étude de la collecte, de l'analyse, de l'interprétation, de la présentation et de l'organisation des données.
- **L'Intelligence artificielle** est définie comme l'étude d'*agents intelligents*: tout appareil qui perçoit son environnement et prend des mesures qui maximisent ses chances de réussir.
- **Le Machine learning** ou apprentissage statistique est un champ d'étude de l'intelligence artificielle qui se fonde sur des approches statistiques pour donner aux ordinateurs la capacité d' « apprendre » à partir de données.
- **La Data science/ Science des données** est l'étude de l'extraction généralisable de connaissances à partir de données, mais le mot clé est science !

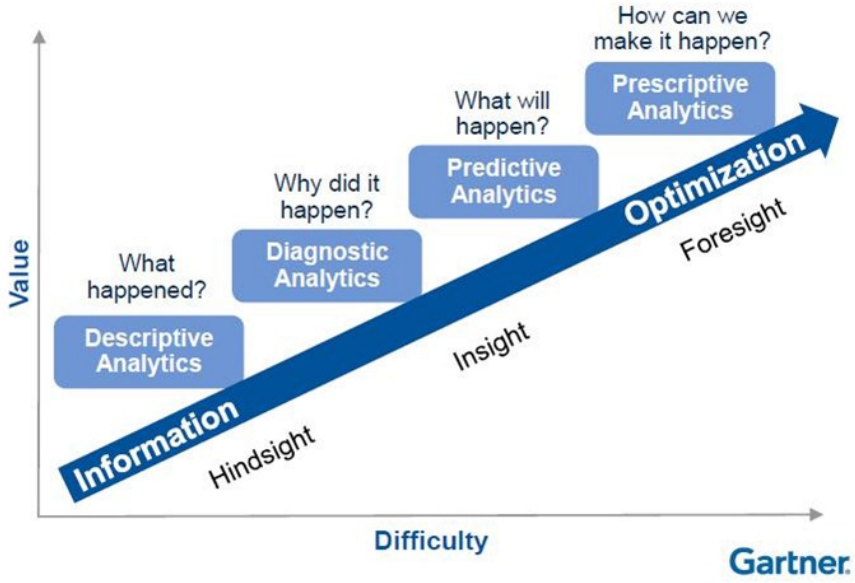
Wikipedia





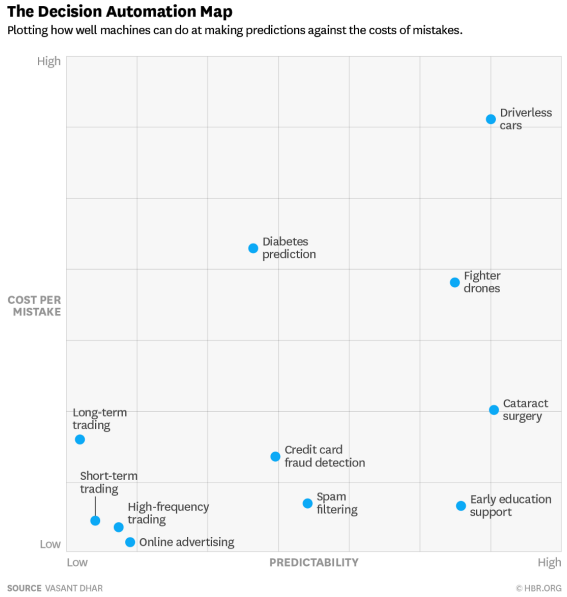
- Le Big data est un terme désignant des ensembles de données si importants et complexes qu'il devient difficile de les analyser en utilisant des applications de traitement de données traditionnelles.
- **La Statistique** est l'étude de la collecte, de l'analyse, de l'interprétation, de la présentation et de l'organisation des données.
- **L'Intelligence artificielle** est définie comme l'étude d'*agents intelligents*: tout appareil qui perçoit son environnement et prend des mesures qui maximisent ses chances de réussir.
- **Le Machine learning** ou apprentissage statistique est un champ d'étude de l'intelligence artificielle qui se fonde sur des approches statistiques pour donner aux ordinateurs la capacité d' « apprendre » à partir de données.
- **La Data science/ Science des données** est l'étude de l'extraction généralisable de connaissances à partir de données, mais le mot clé est science !

Wikipedia



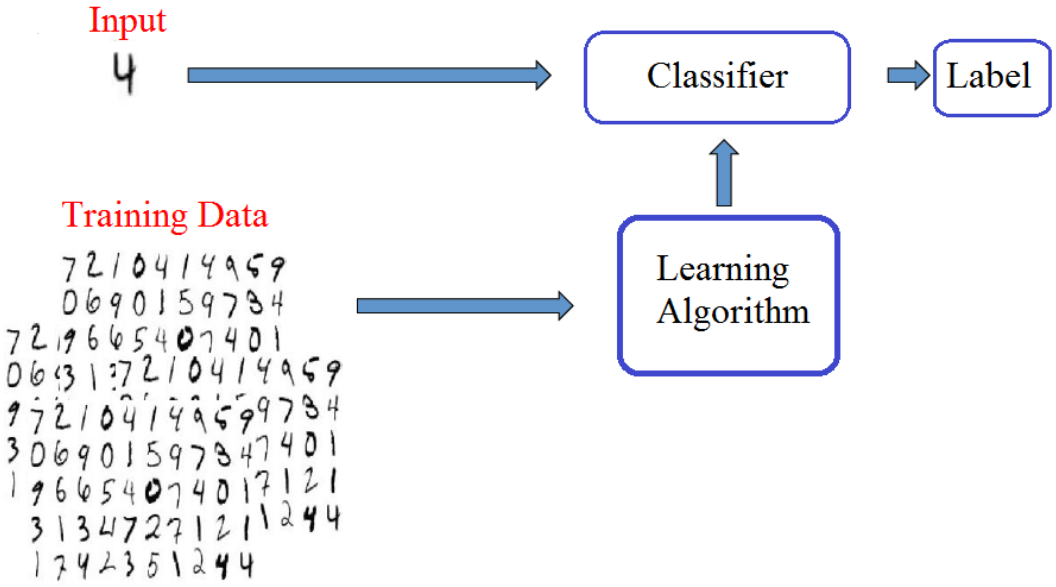
- **Le Big data** est un terme désignant des ensembles de données si importants et complexes qu'il devient difficile de les analyser en utilisant des applications de traitement de données traditionnelles.
- La Statistique est l'étude de la collecte, de l'analyse, de l'interprétation, de la présentation et de l'organisation des données.
- **L'Intelligence artificielle** est définie comme l'étude d'*agents intelligents*: tout appareil qui perçoit son environnement et prend des mesures qui maximisent ses chances de réussir.
- **Le Machine learning** ou apprentissage statistique est un champ d'étude de l'intelligence artificielle qui se fonde sur des approches statistiques pour donner aux ordinateurs la capacité d' « apprendre » à partir de données.
- **La Data science/ Science des données** est l'étude de l'extraction généralisable de connaissances à partir de données, mais le mot clé est science !

Wikipedia



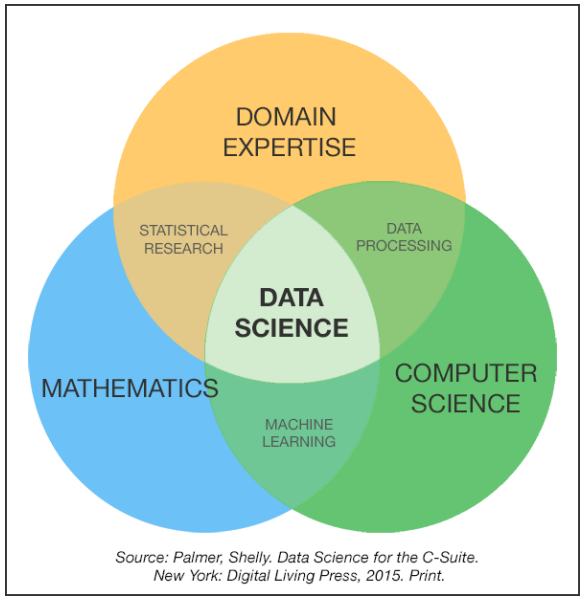
- **Le Big data** est un terme désignant des ensembles de données si importants et complexes qu'il devient difficile de les analyser en utilisant des applications de traitement de données traditionnelles.
- **La Statistique** est l'étude de la collecte, de l'analyse, de l'interprétation, de la présentation et de l'organisation des données.
- L'Intelligence artificielle est définie comme l'étude d'*agents intelligents*: tout appareil qui perçoit son environnement et prend des mesures qui maximisent ses chances de réussir.
- **Le Machine learning** ou apprentissage statistique est un champ d'étude de l'intelligence artificielle qui se fonde sur des approches statistiques pour donner aux ordinateurs la capacité d' « apprendre » à partir de données.
- **La Data science/ Science des données** est l'étude de l'extraction généralisable de connaissances à partir de données, mais le mot clé est science !

Wikipedia



- **Le Big data** est un terme désignant des ensembles de données si importants et complexes qu'il devient difficile de les analyser en utilisant des applications de traitement de données traditionnelles.
- **La Statistique** est l'étude de la collecte, de l'analyse, de l'interprétation, de la présentation et de l'organisation des données.
- **L'Intelligence artificielle** est définie comme l'étude d'*agents intelligents*: tout appareil qui perçoit son environnement et prend des mesures qui maximisent ses chances de réussir.
- Le Machine learning ou apprentissage statistique est un champ d'étude de l'intelligence artificielle qui se fonde sur des approches statistiques pour donner aux ordinateurs la capacité d' « apprendre » à partir de données.
- **La Data science/ Science des données** est l'étude de l'extraction généralisable de connaissances à partir de données, mais le mot clé est science !

Wikipedia



- **Le Big data** est un terme désignant des ensembles de données si importants et complexes qu'il devient difficile de les analyser en utilisant des applications de traitement de données traditionnelles.
- **La Statistique** est l'étude de la collecte, de l'analyse, de l'interprétation, de la présentation et de l'organisation des données.
- **L'Intelligence artificielle** est définie comme l'étude d'*agents intelligents*: tout appareil qui perçoit son environnement et prend des mesures qui maximisent ses chances de réussir.
- **Le Machine learning** ou apprentissage statistique est un champ d'étude de l'intelligence artificielle qui se fonde sur des approches statistiques pour donner aux ordinateurs la capacité d' « apprendre » à partir de données.
- La Data science/ Science des données est l'étude de l'extraction généralisable de connaissances à partir de données, mais le mot clé est science !

Les influences majeures

Quatre influences majeures agissent aujourd'hui:

- La théorie formelle de la statistique
- L'accélération du développement des ordinateurs
- Le défi, dans de nombreux domaines, de corpus de données toujours plus grands
- L'accent mis sur la quantification dans une variété toujours plus large de disciplines

Les influences majeures - Tukey (1962)

Quatre influences majeures agissent aujourd'hui:

- La théorie formelle de la statistique
- L'accélération du développement des ordinateurs
- Le défi, dans de nombreux domaines, de corpus de données toujours plus grands
- L'accent mis sur la quantification dans une variété toujours plus large de disciplines

- Il parlait de l'analyse de données.
- Datamining, Machine learning , Big Data, AI ...

Faire de la science des données

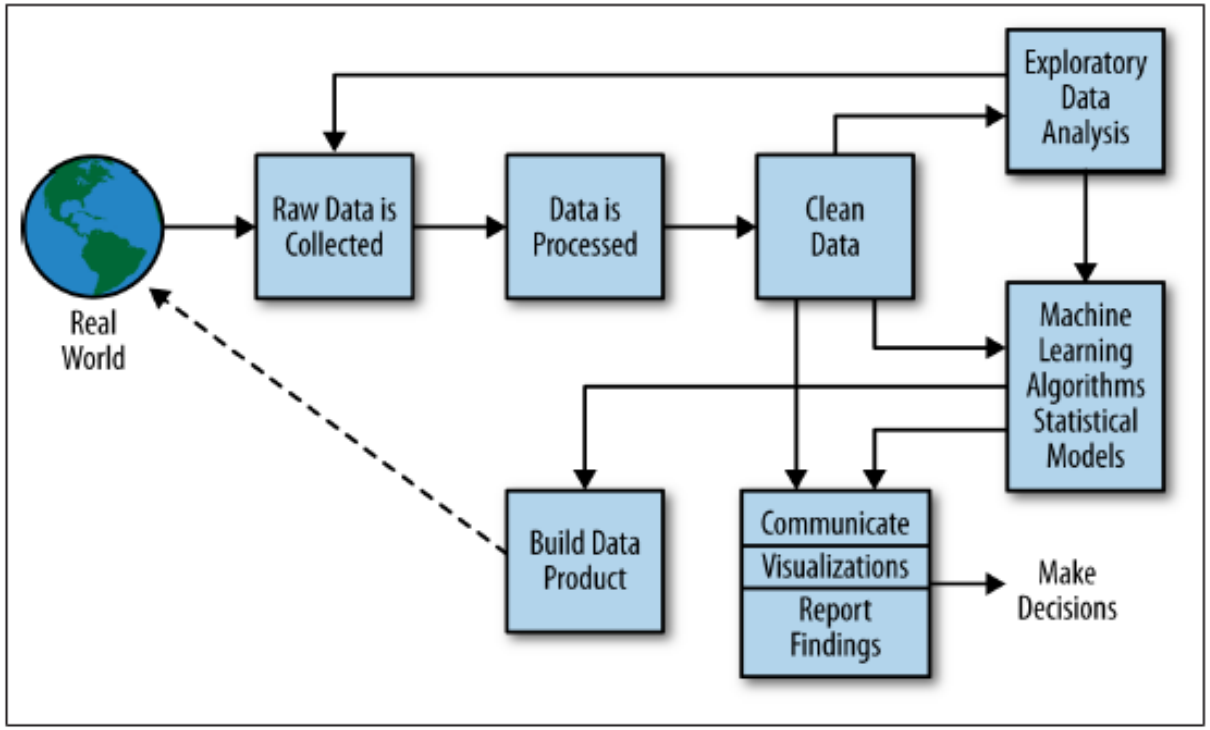
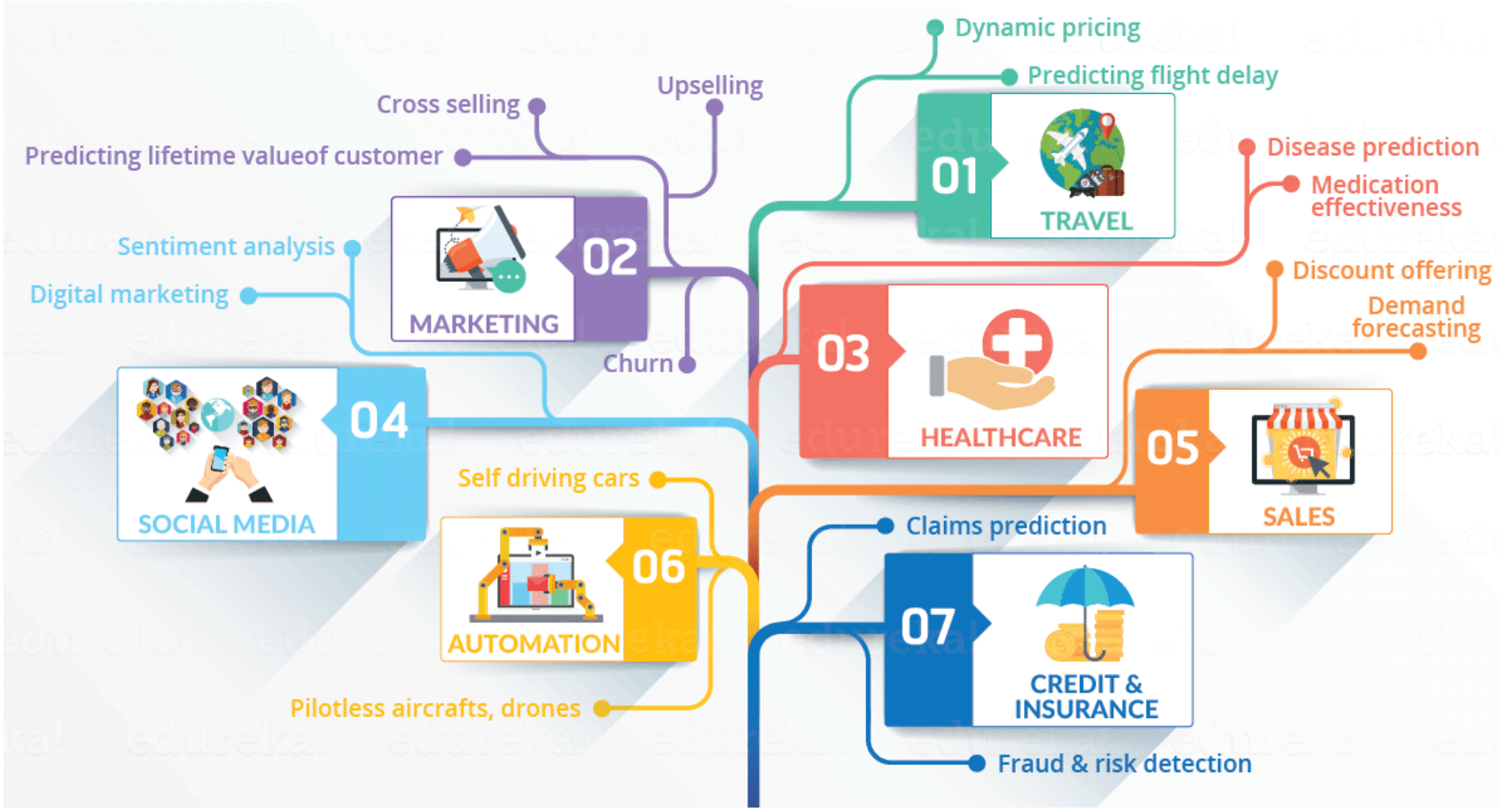


Figure 2-2. The data science process

Exemples de cas d'usage

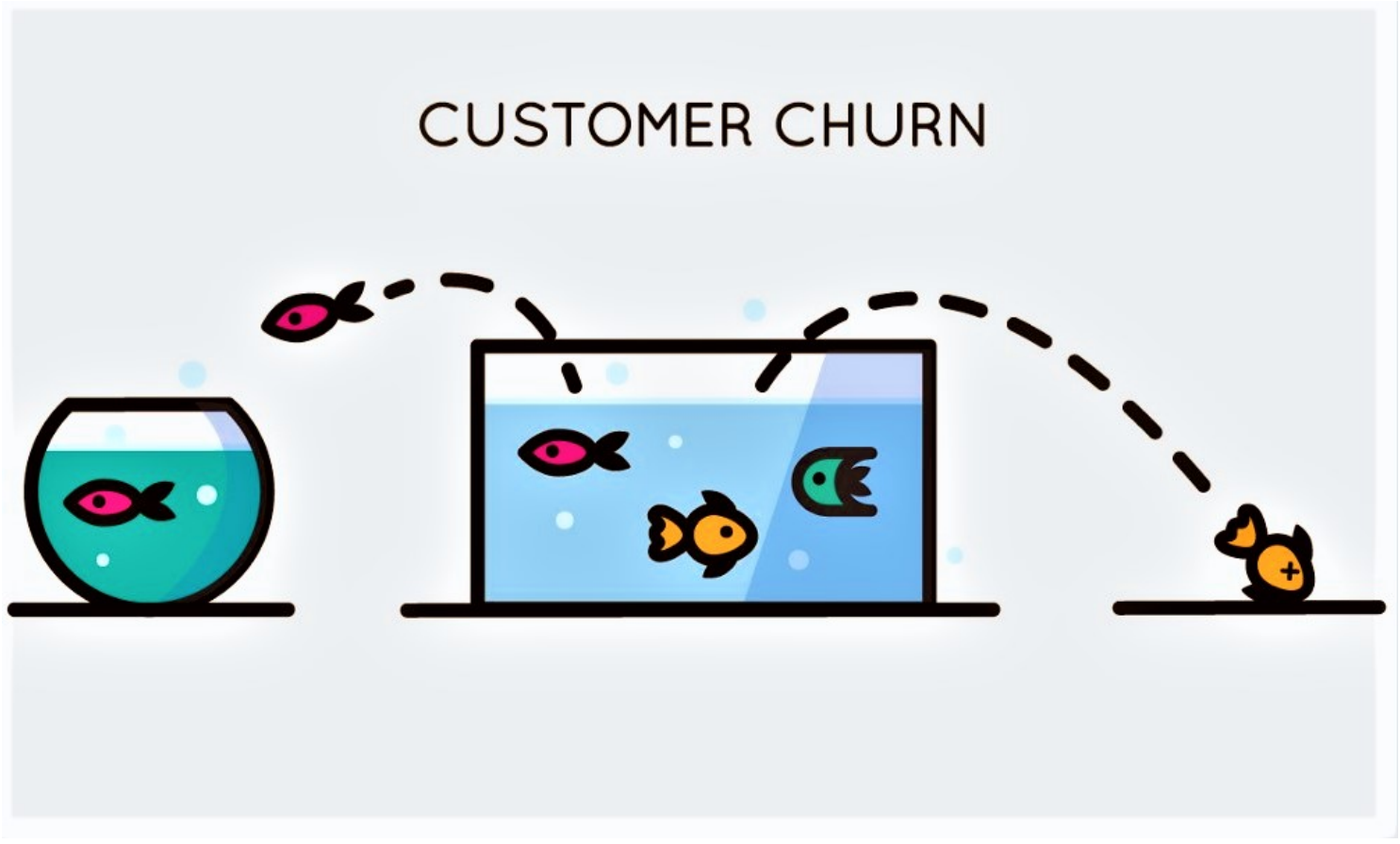
Big Data/DS : où et pourquoi ?



Cas d'usage en marketing

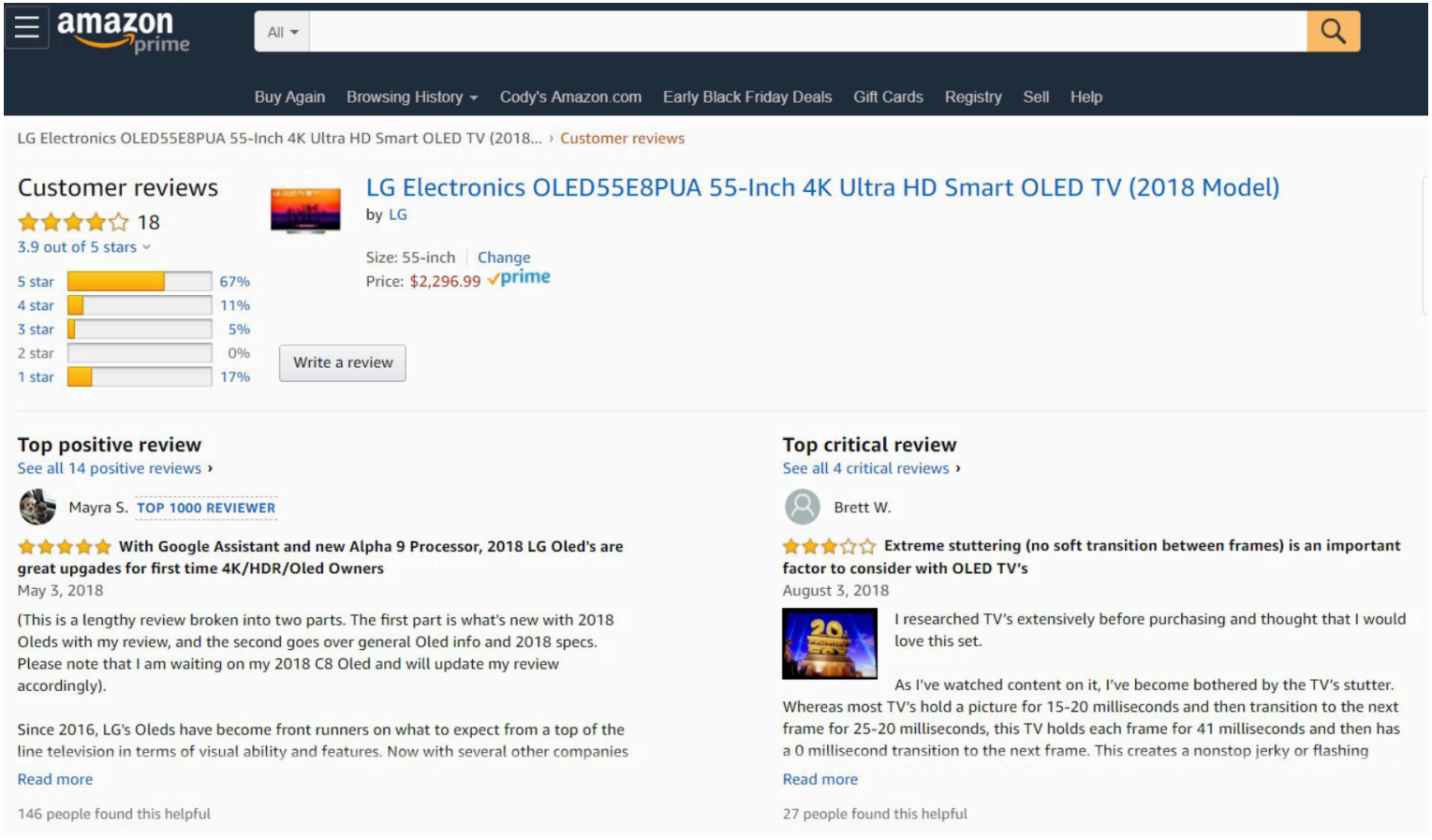
- Prédiction du churn
- Marketing personnalisés et segmentation des clients
- "Sentiment analysis" des clients
- Recommandation

Churn/attrition

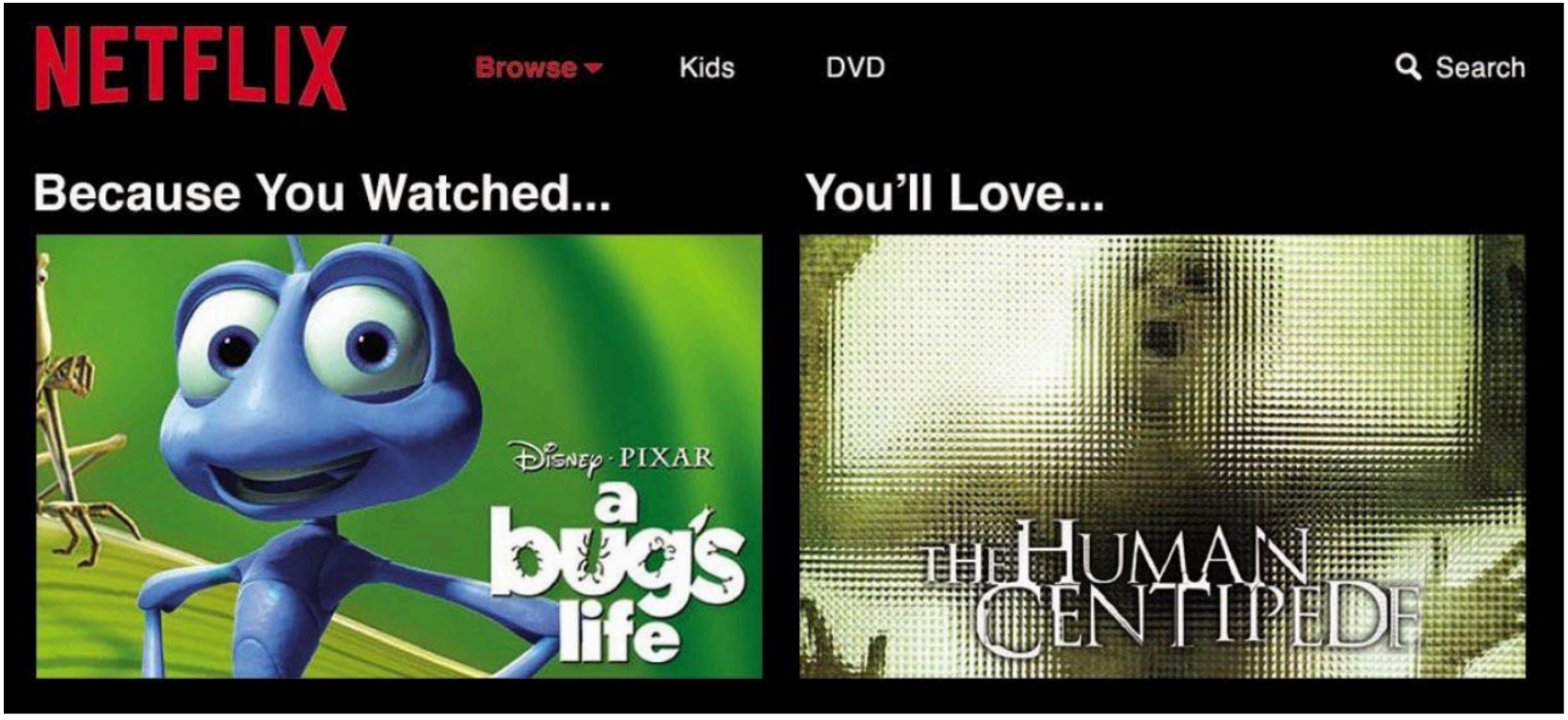


Segmentation des clients

Sentiment analysis



Recommandation



Because You Watched...You'll Love... — What Problem Does Movie Recommendation Help Solve?

Machine Learning/Apprentissage statistique

Une définition du Machine Learning

par Tom Mitchell (`http://www.cs.cmu.edu/~tom/`)

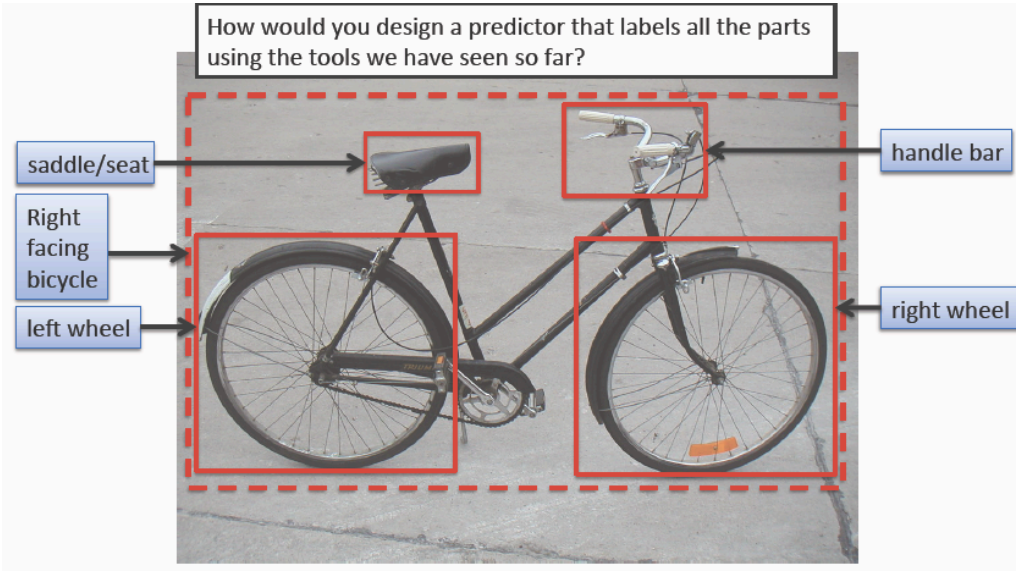
Un programme informatique est réputé apprendre (learn) d'une expérience E pour certaines classes de tâches T et une mesure de performance P, si ses performances aux tâches T, mesurée par P, s'améliorent avec l'expérience.

Un robot qui apprend

Un robot doté d'un ensemble de capteurs et d'un algorithme d'apprentissage en ligne

- **Tâche**: jouer au football
- **Performance**: score
- **Expérience**:
  - environnement actuel
  - jeux passés

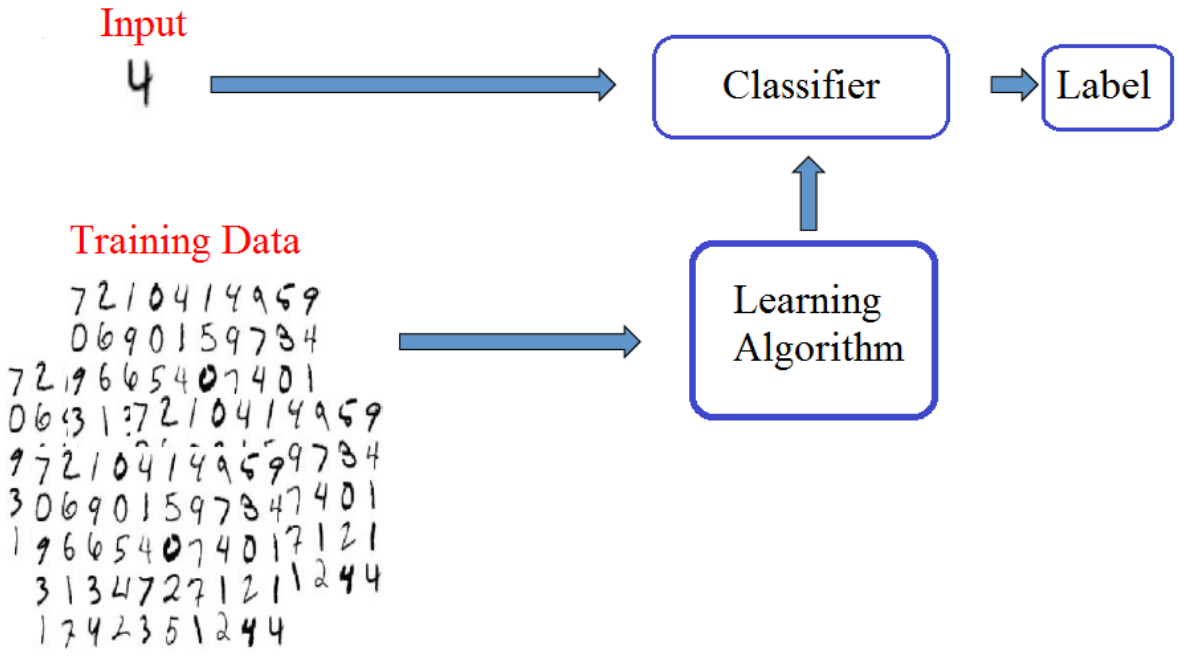
Reconnaissance d'objets dans une image



Un algorithme de détection/reconnaissance

- **Tâche** : dire si un objet est présent ou non dans l'image
- **Performance** : nombre d'erreurs
- **Expérience** : ensemble d'images '"labelisées" précédemment vues

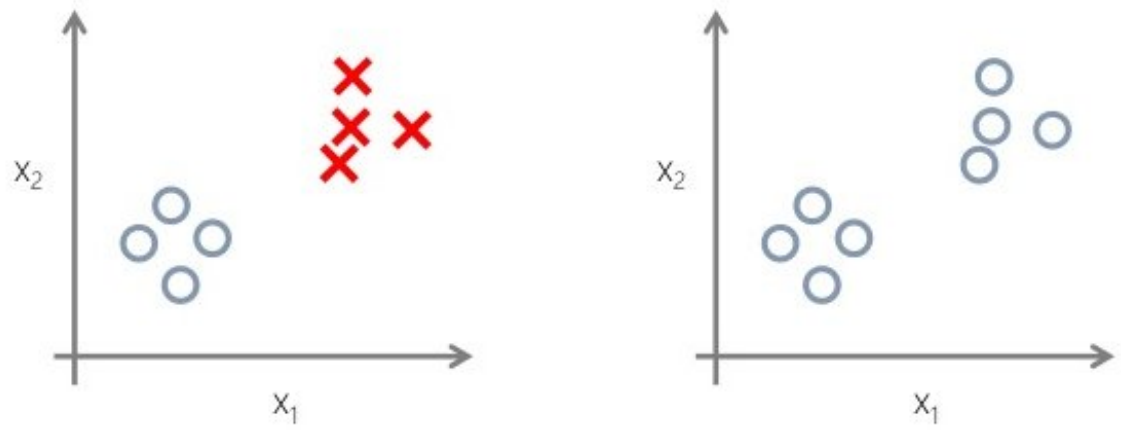
Machine Learning



Tom Mitchell (http://www.cs.cmu.edu/~tom/)

Un programme informatique est réputé apprendre (learn) d'une expérience E pour certaines classes de tâches T et une mesure de performance P, si ses performances aux tâches T, mesurée par P, s'améliorent avec l'expérience.

Supervisé et non-supervisé



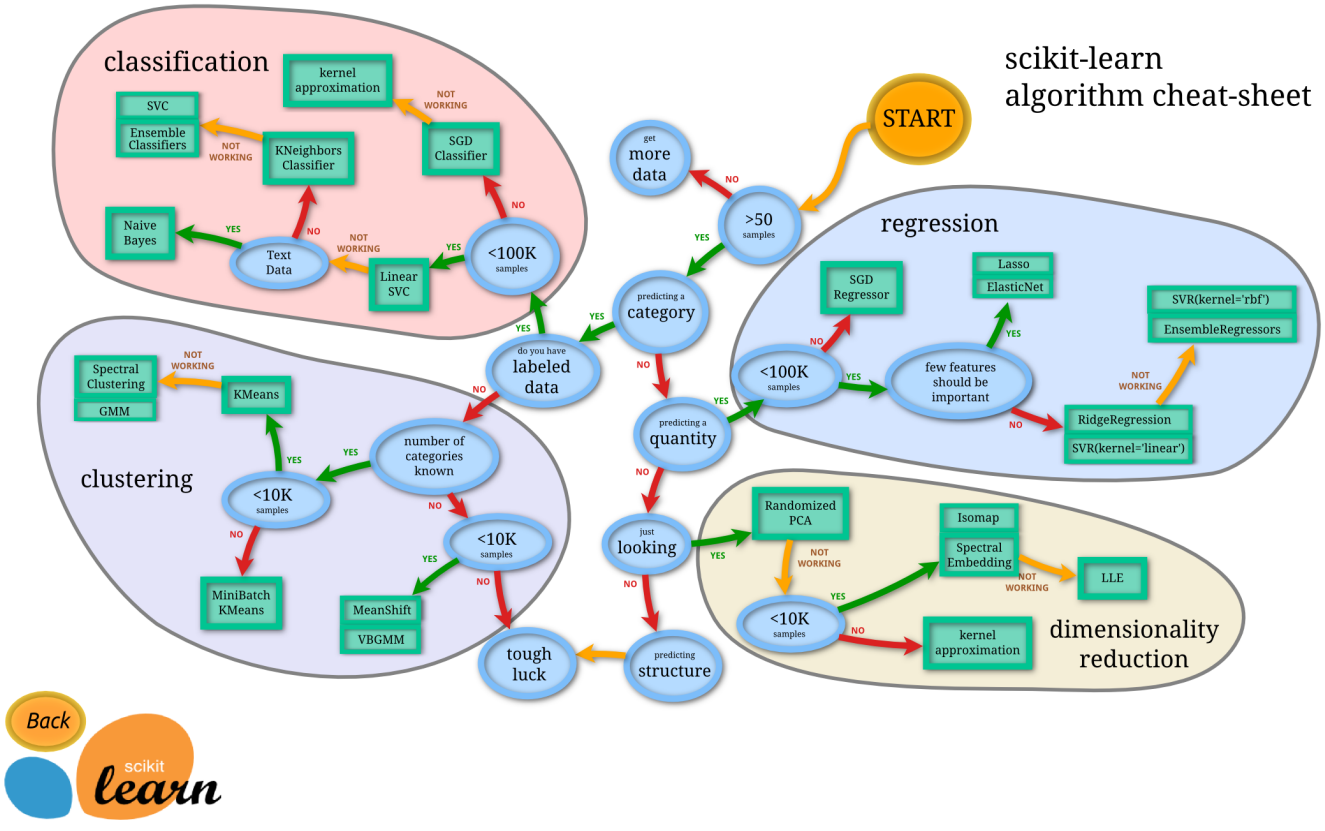
Apprentissage supervisé

- Objectif : apprendre une fonction f prédisant une variable Y à partir de features $\mathbf{X}$.
- Données : ensemble d'apprentissage $(\mathbf{X}_i, Y_i)$

Apprentissage non-supervisé

- Objectif: découvrir une structure au sein d'un ensemble d'individus $(\mathbf{X}_i)$.
- Data: Learning set $(\mathbf{X}_i)$

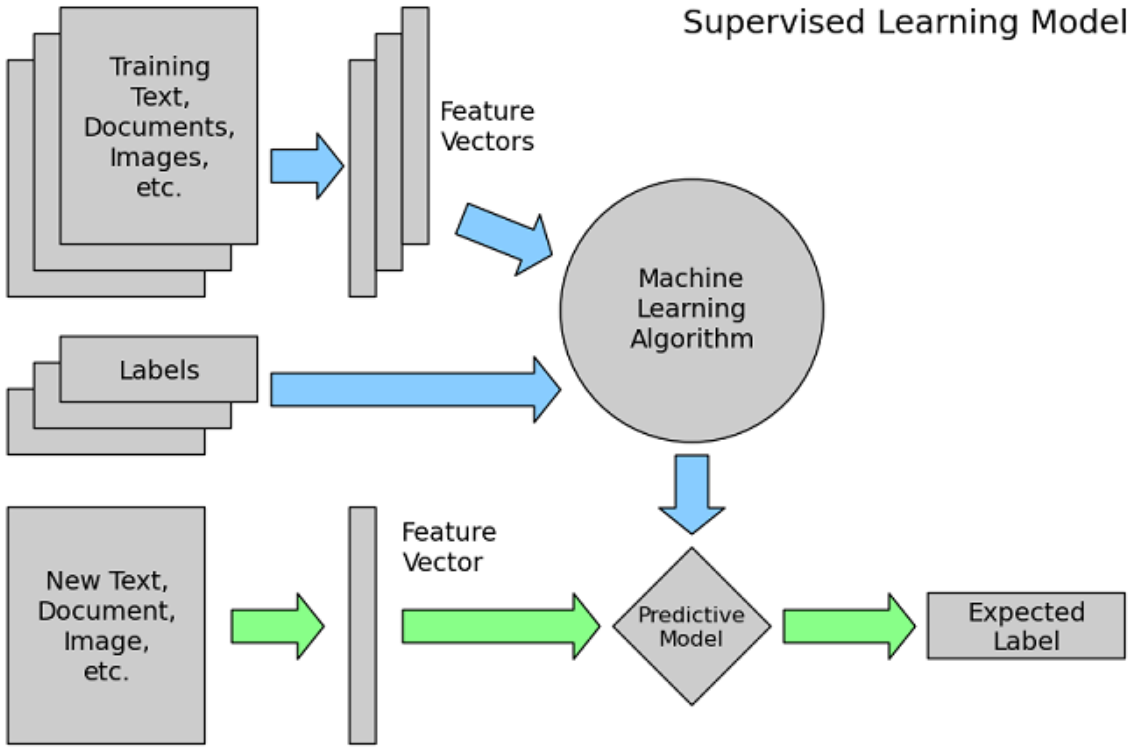
Machine Learning



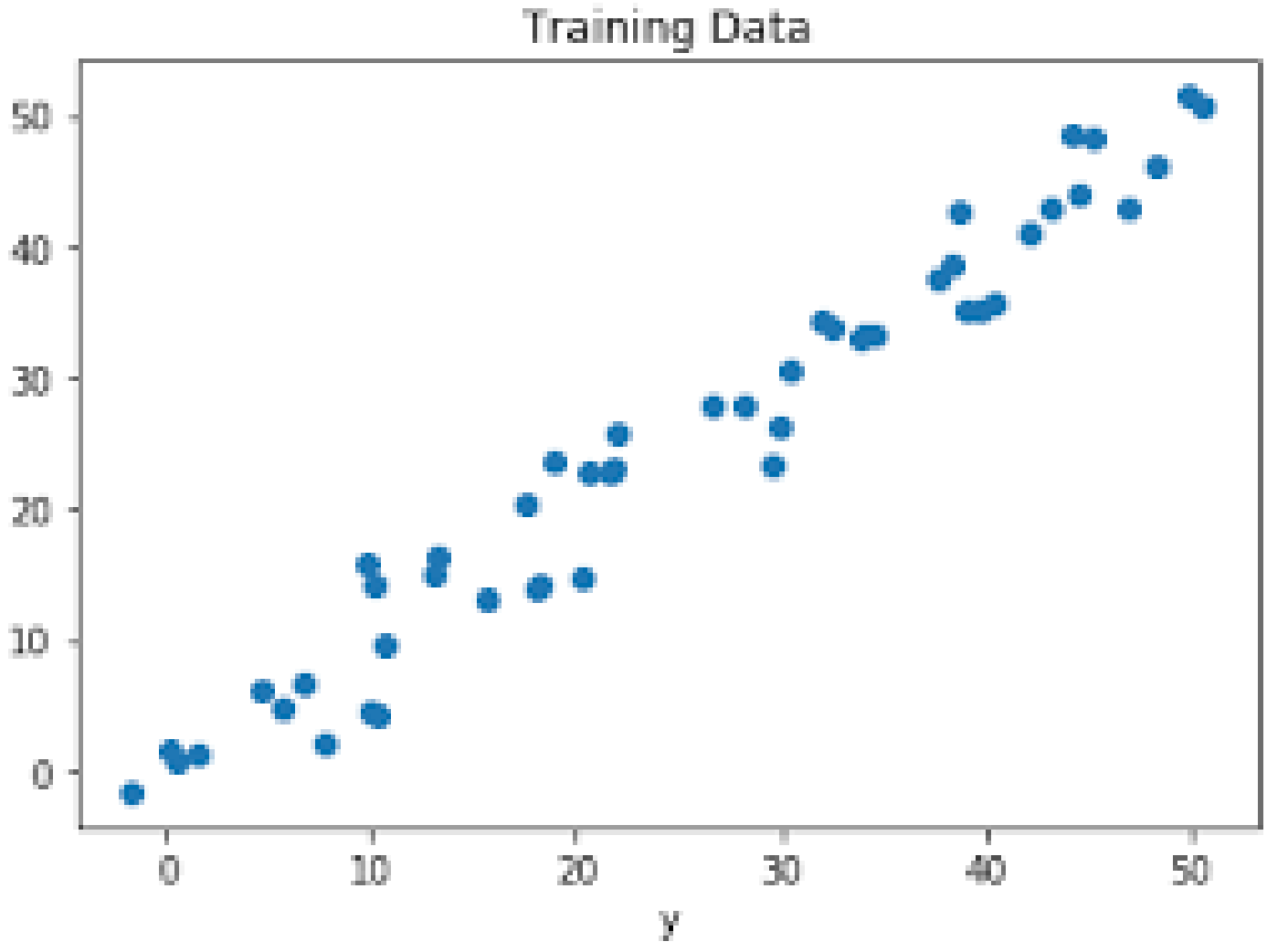
Méthodes pour le ML

- Grand catalogue de méthodes,
- Besoin de définir la performance,
- Design des features...

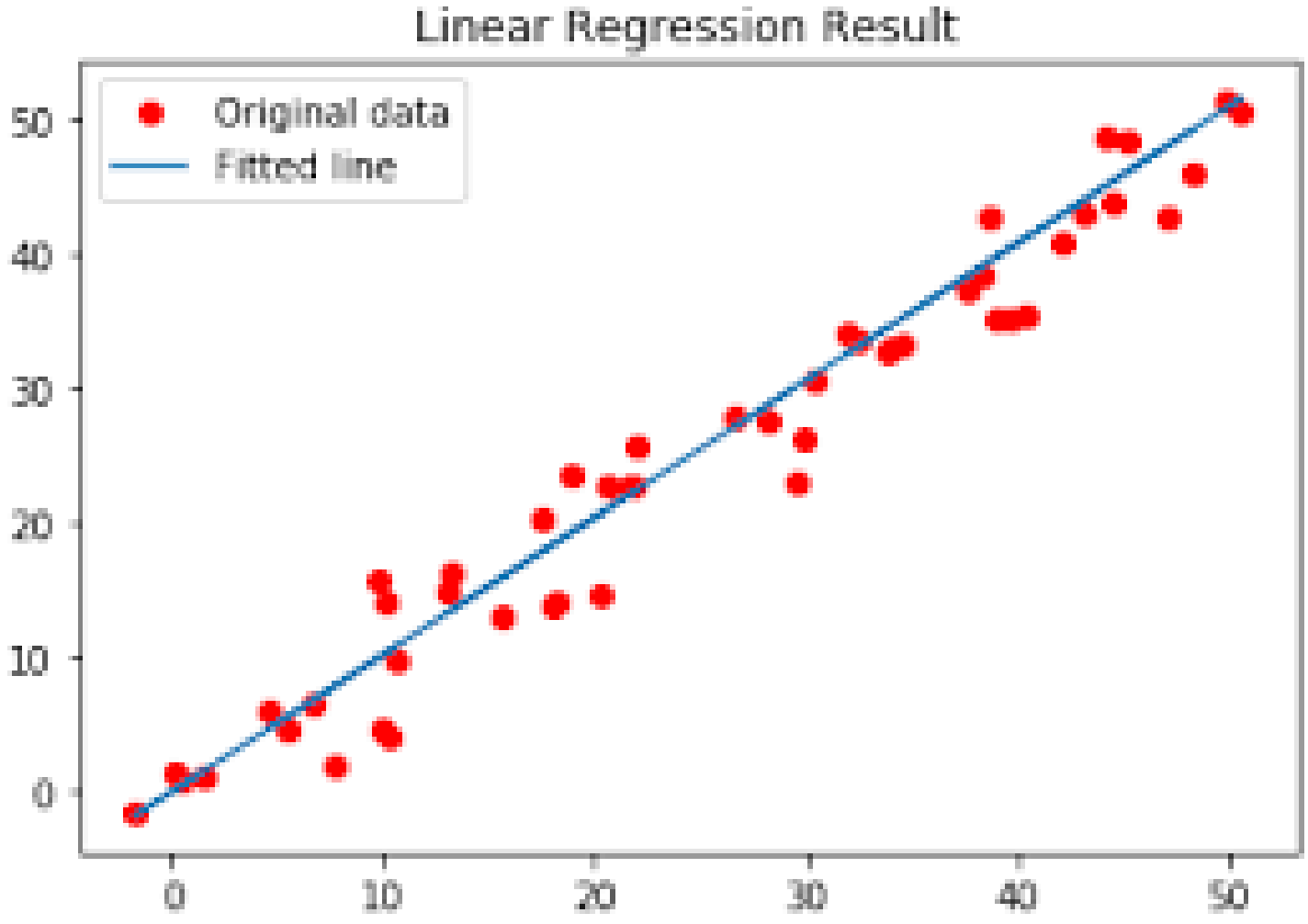
Apprentissage supervisé



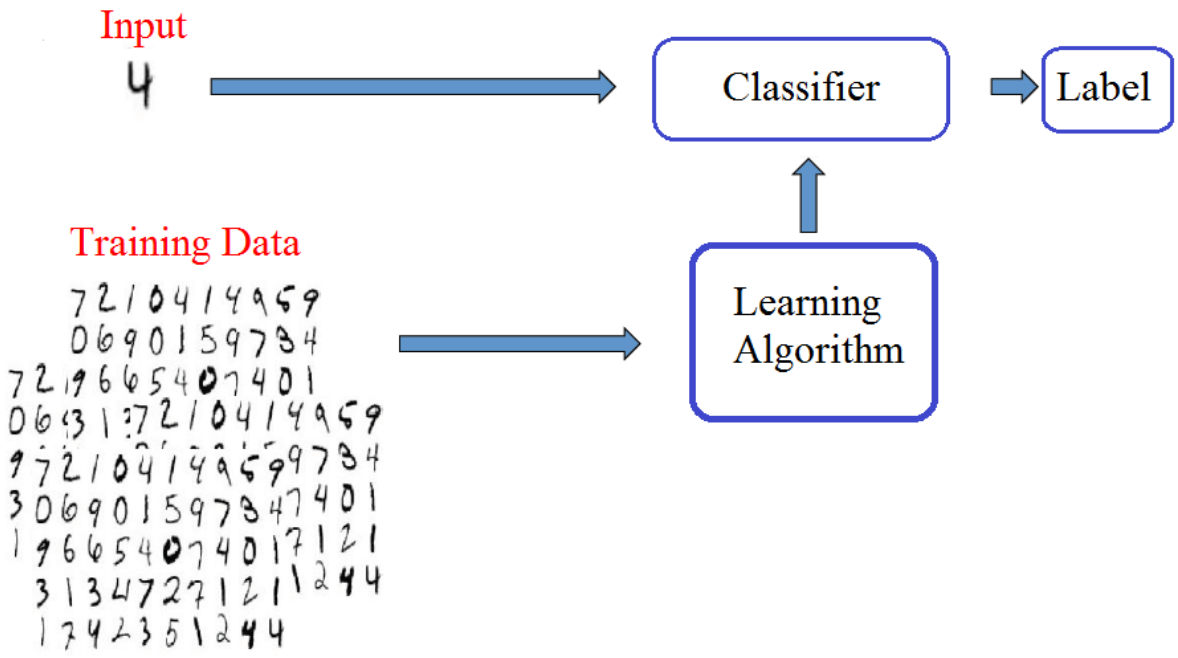
Régression I



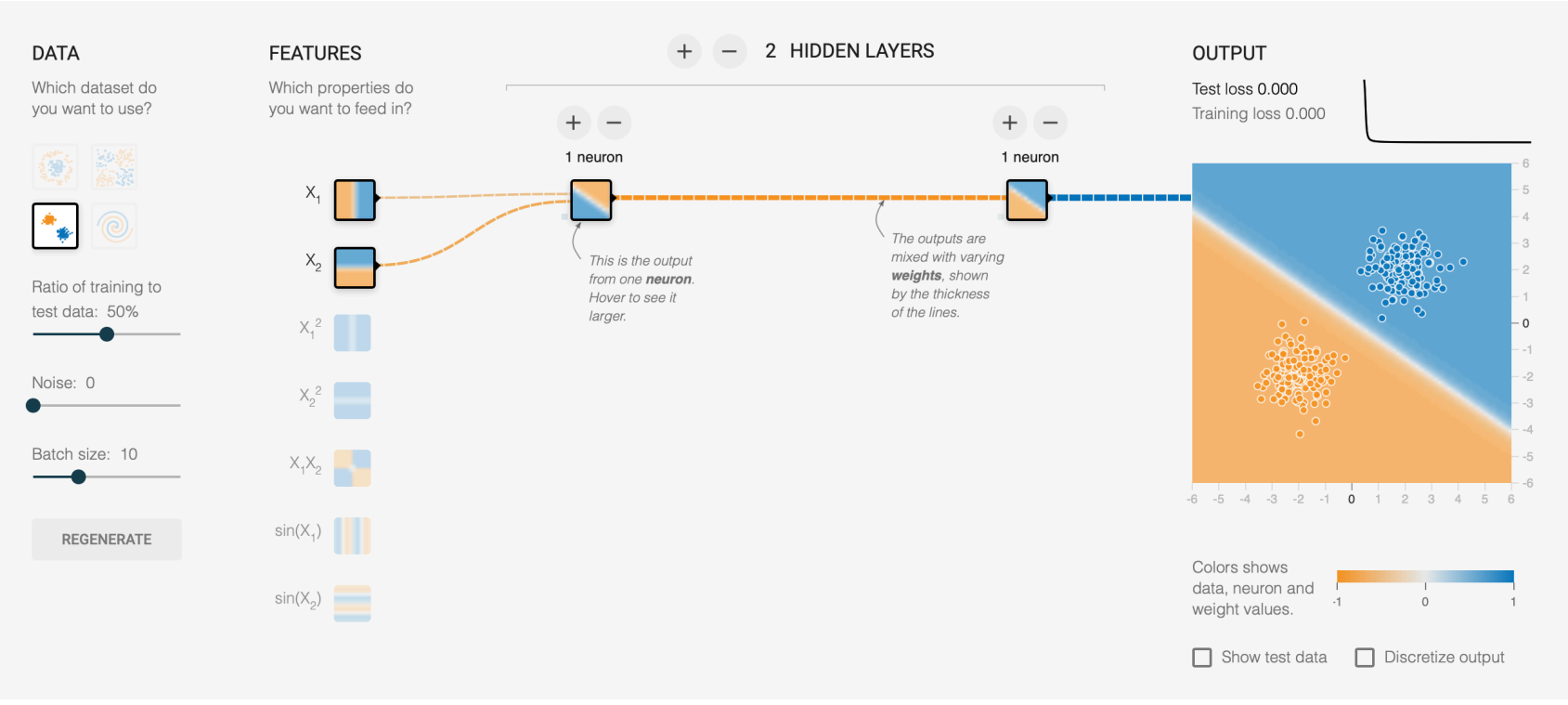
Régression II



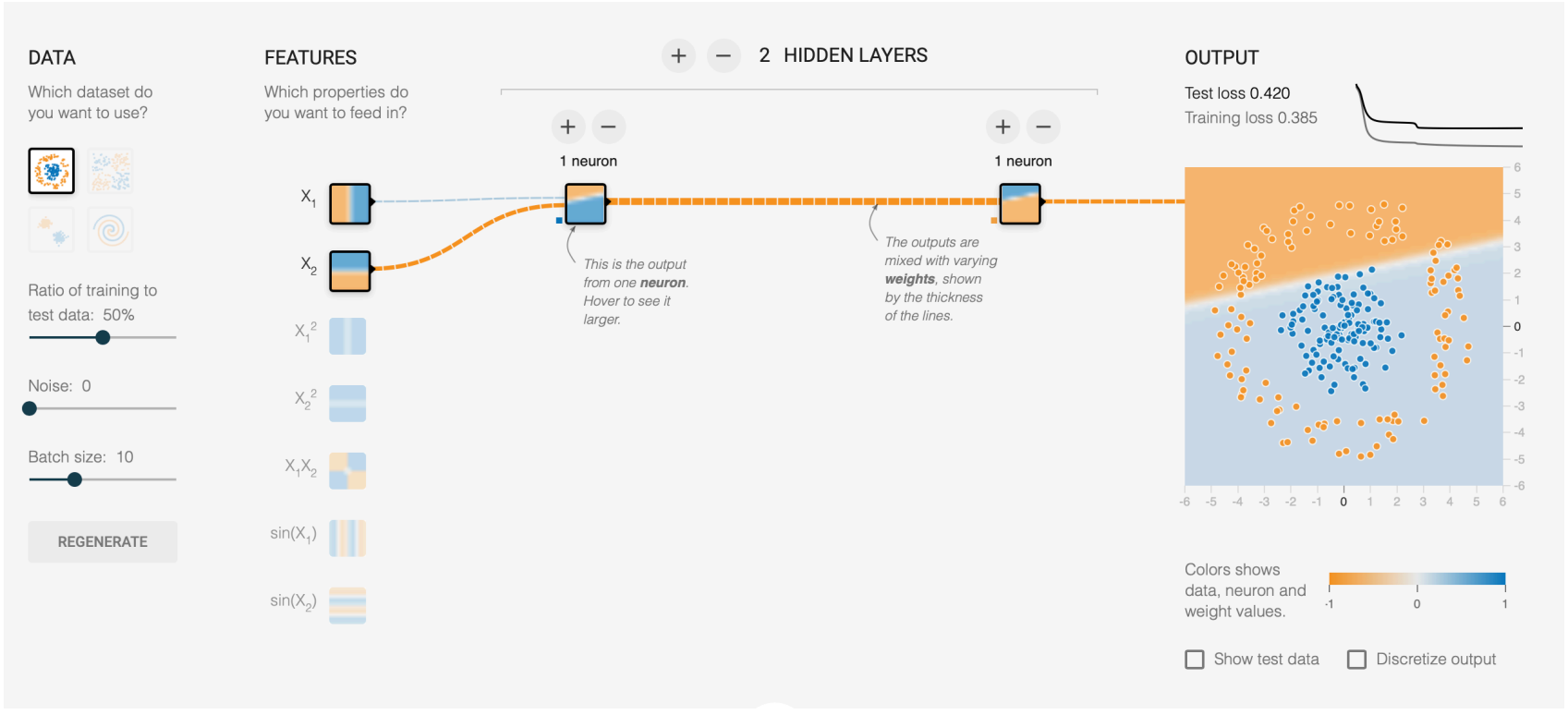
Classification



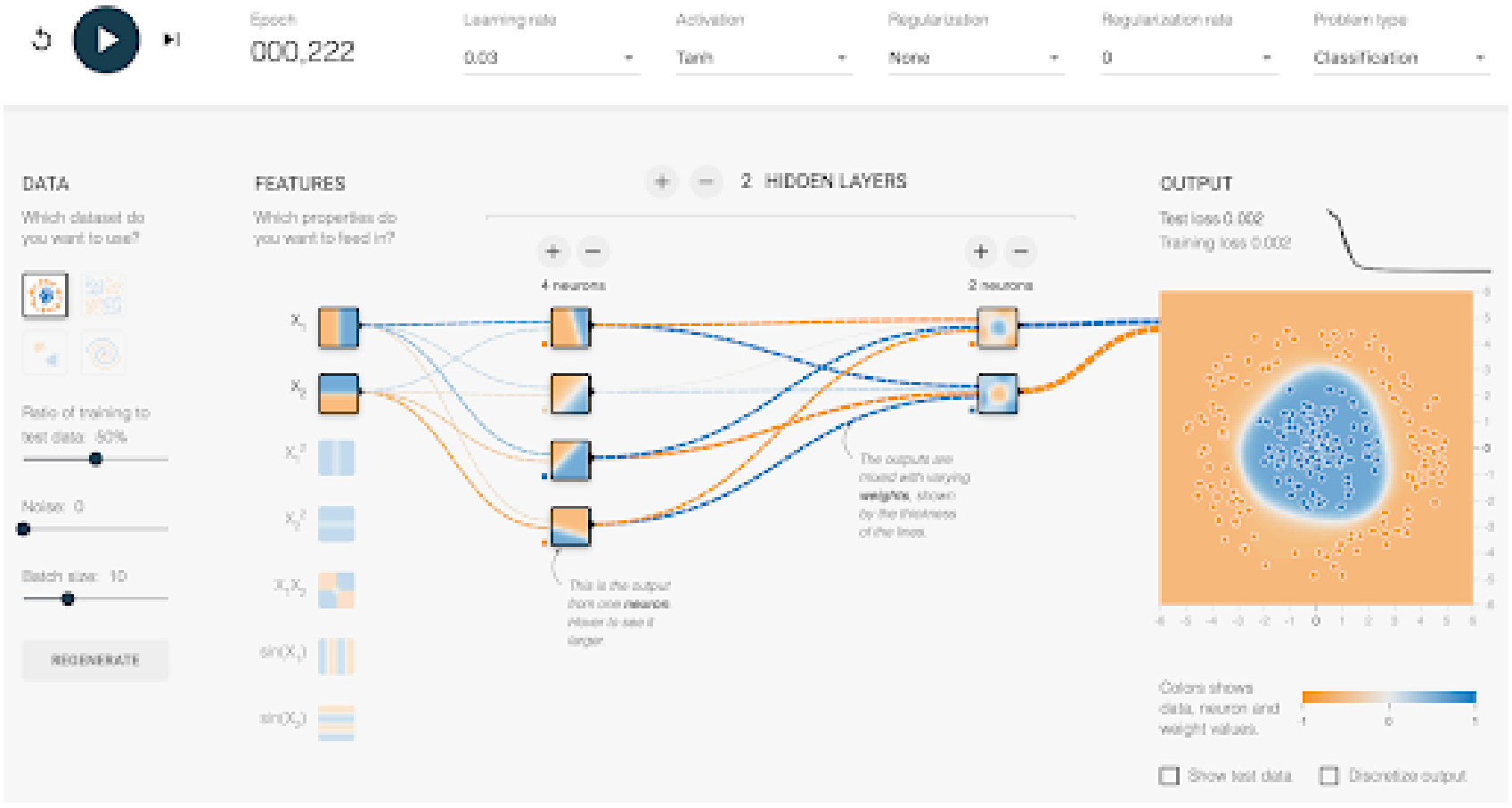
Régression logistique: une exemple simple



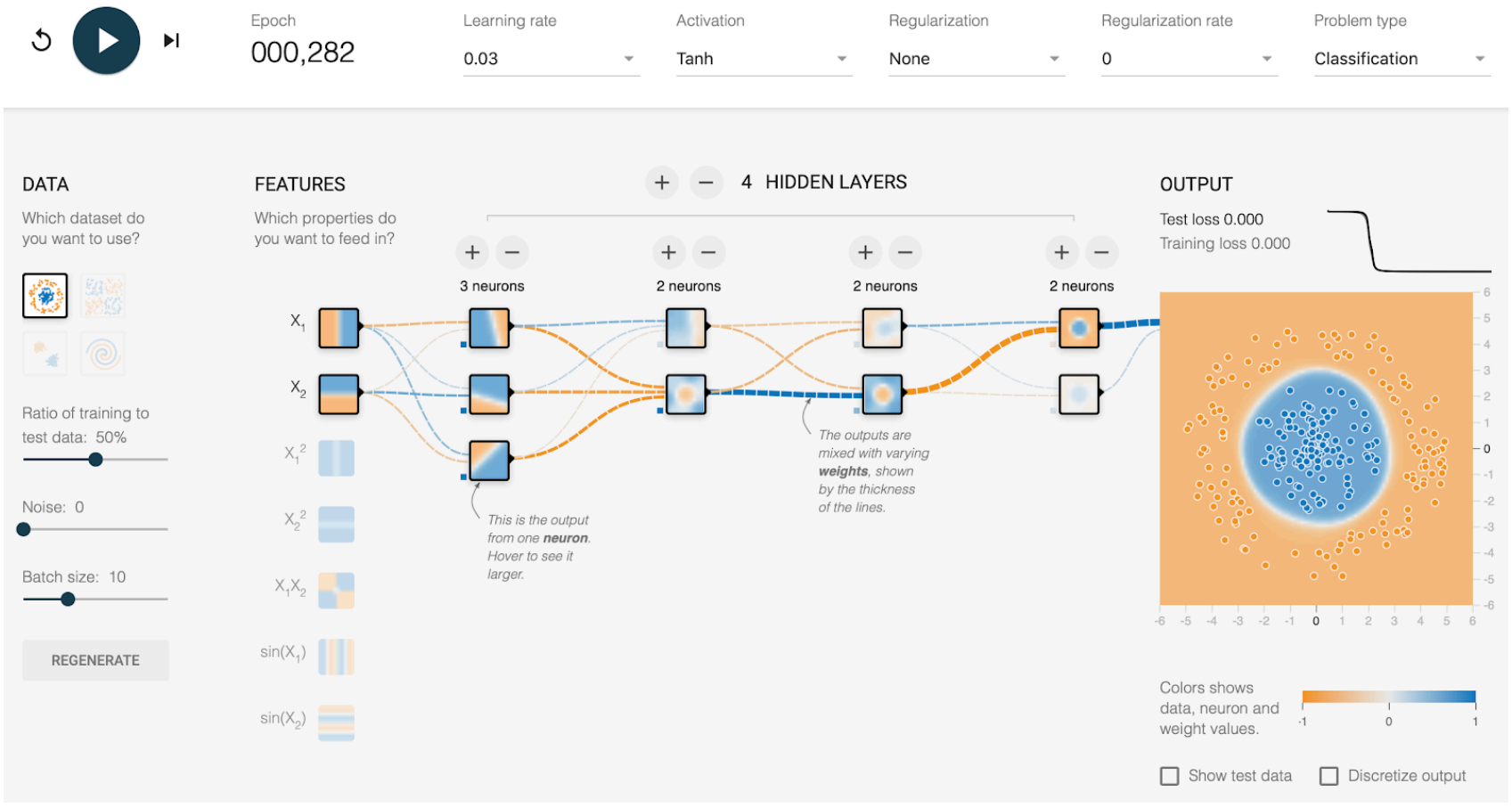
Régression logistique: une exemple plus compliqué I



Régression logistique: une exemple plus compliqué II



Régression logistique: une exemple plus compliqué III



Apprentissage non-supervisé

Réduction de la dimension / visualisation I

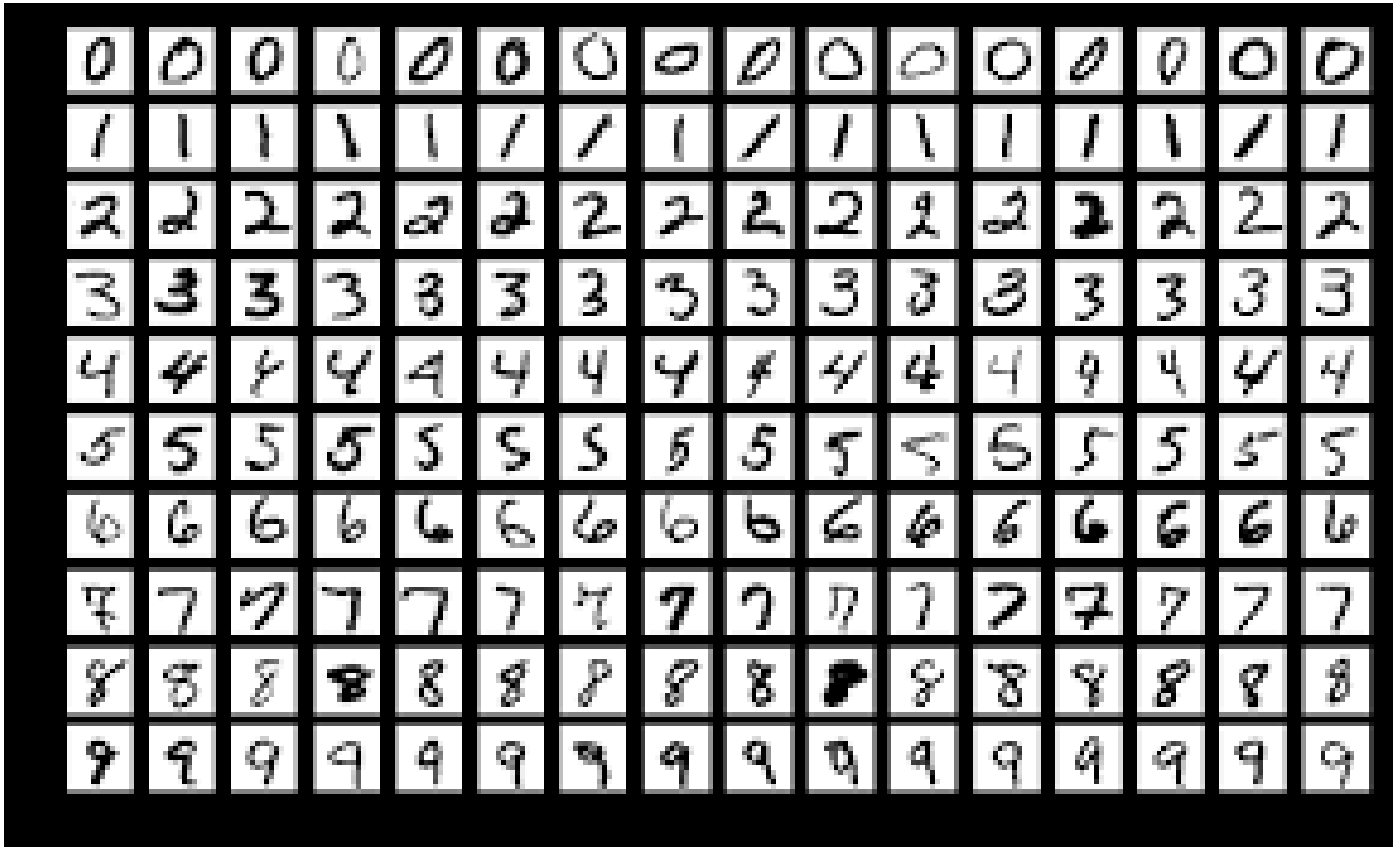


Figure 1: MNIST data

Réduction de la dimension / visualisation II

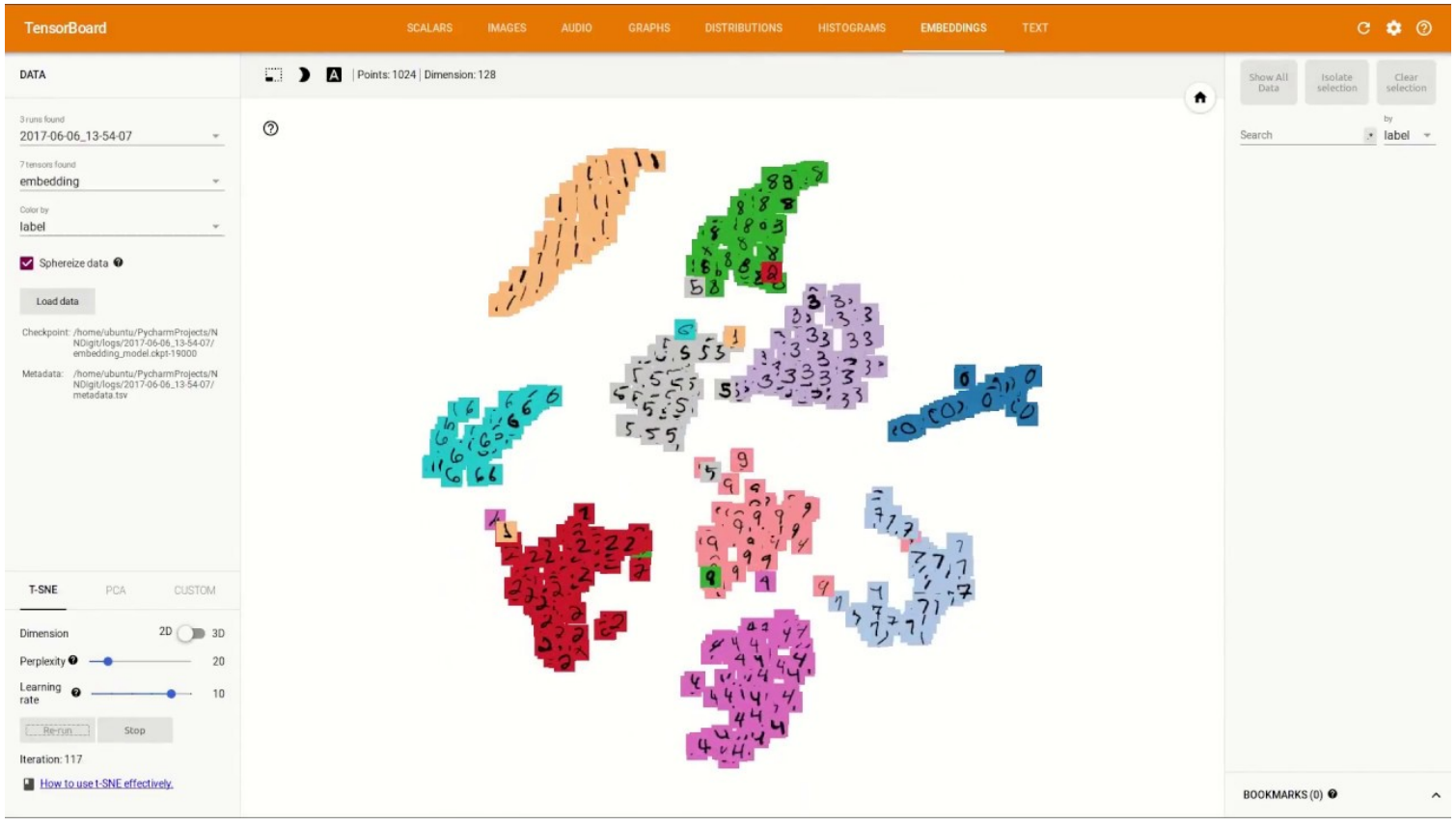


Figure 2: T-SNE

Clustering

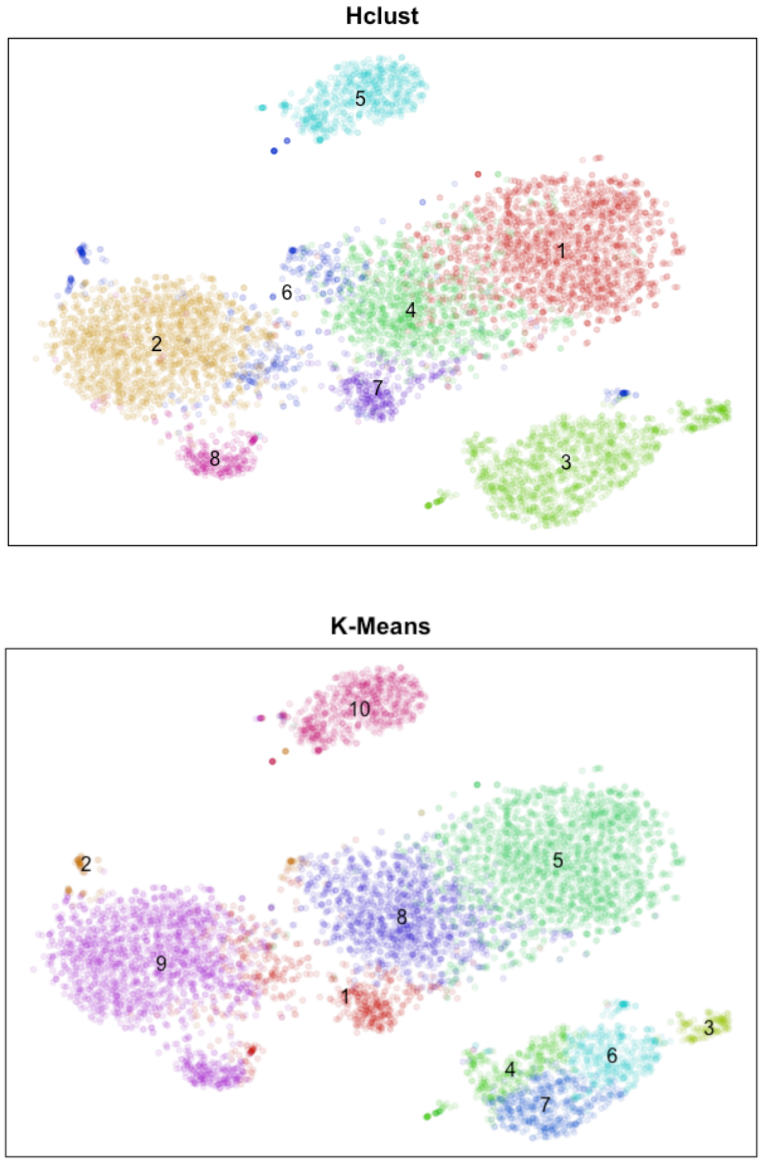


Figure 3: Hierarchical clustering et K-means

Retour sur les cas d'usage

Churn/attrition

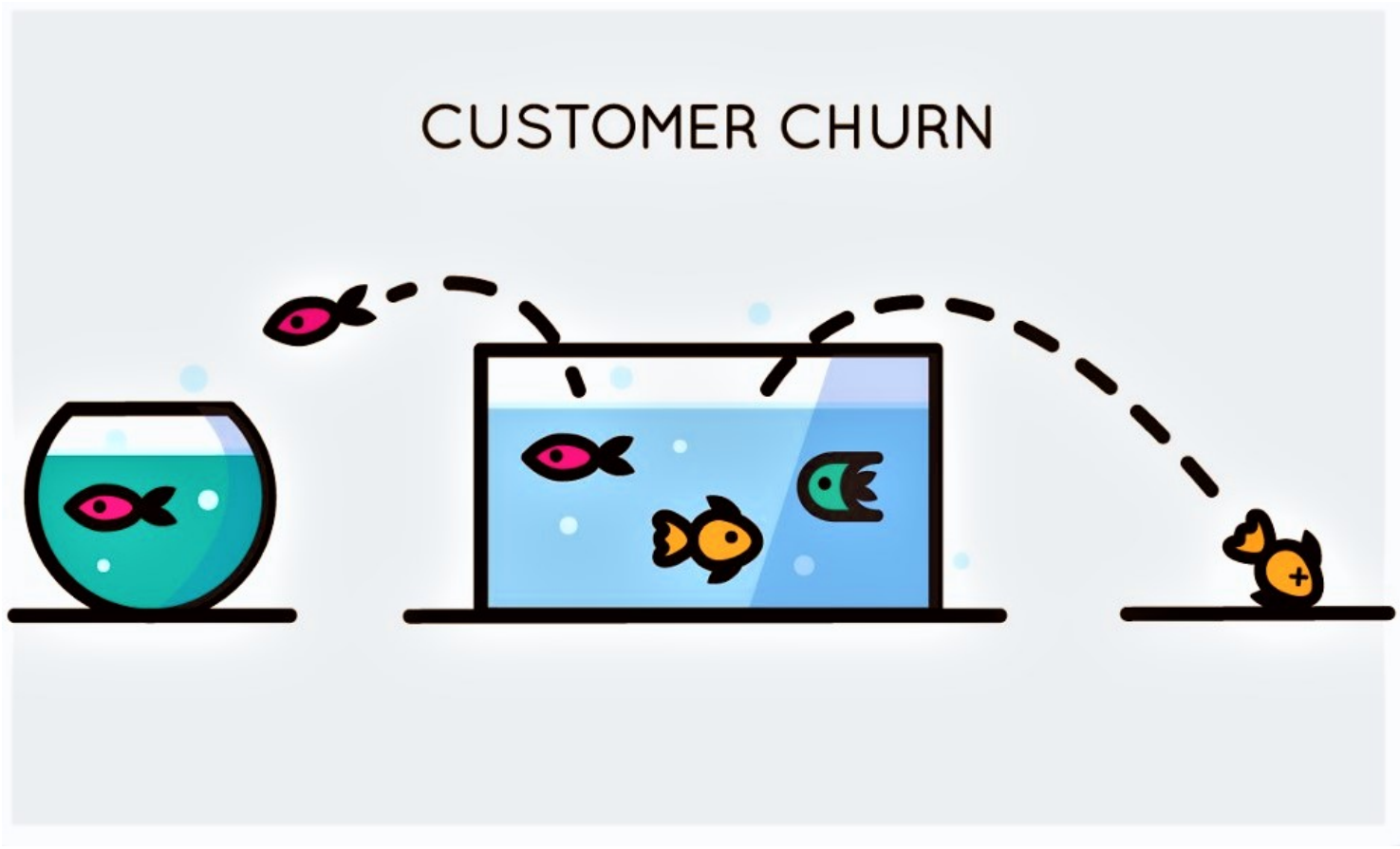


Figure 4: Classification : le client reste ou part

Segmentation des clients

Figure 5: Clustering

Sentiment analysis

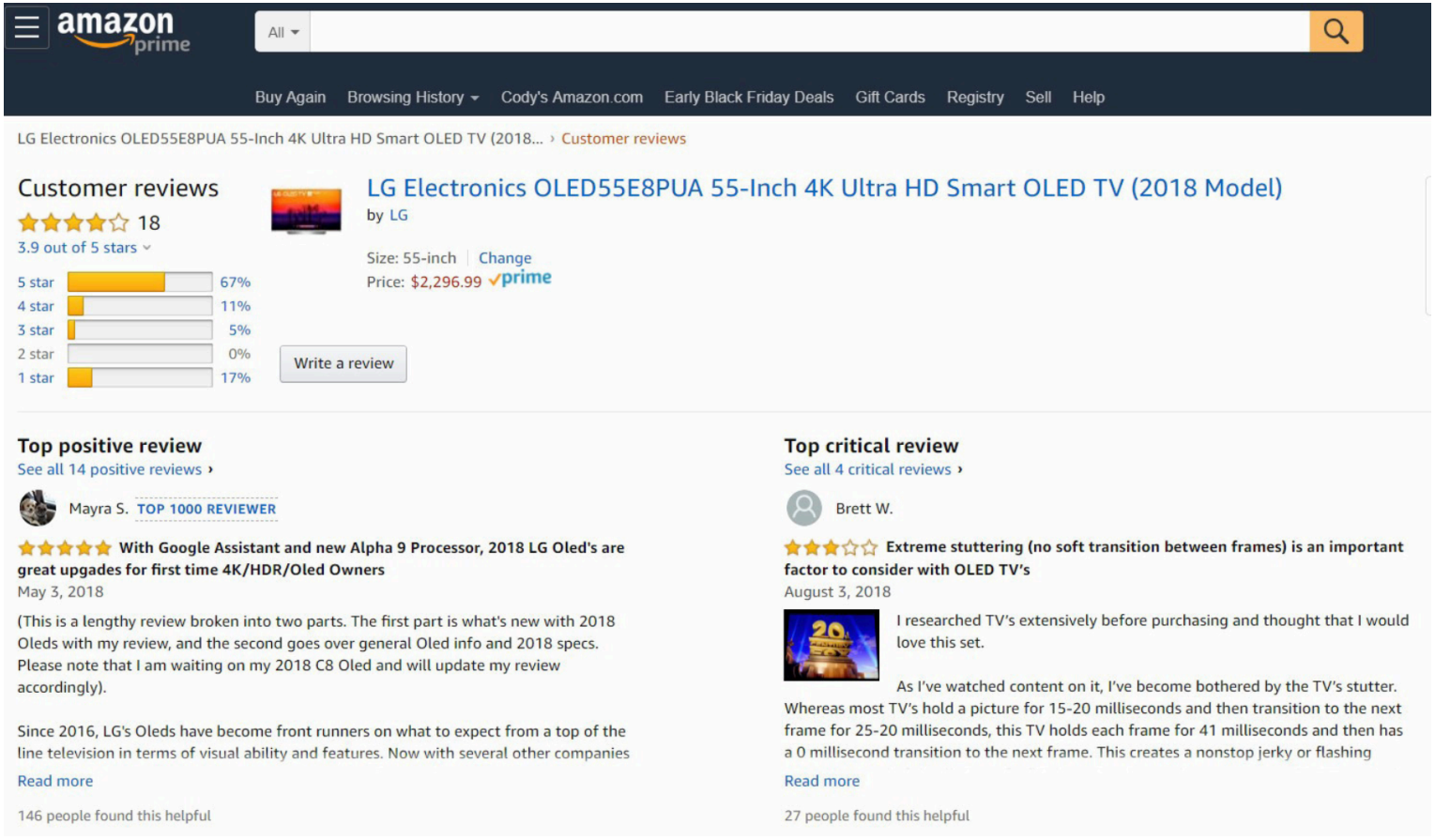
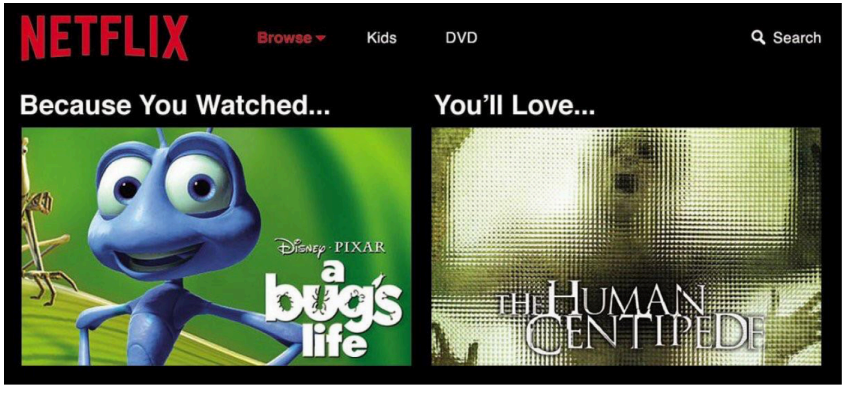


Figure 6: Classification : le commentaire est positif ou non // Régression : note

Recommandation



Because You Watched...You'll Love... — What Problem Does Movie Recommendation Help Solve?

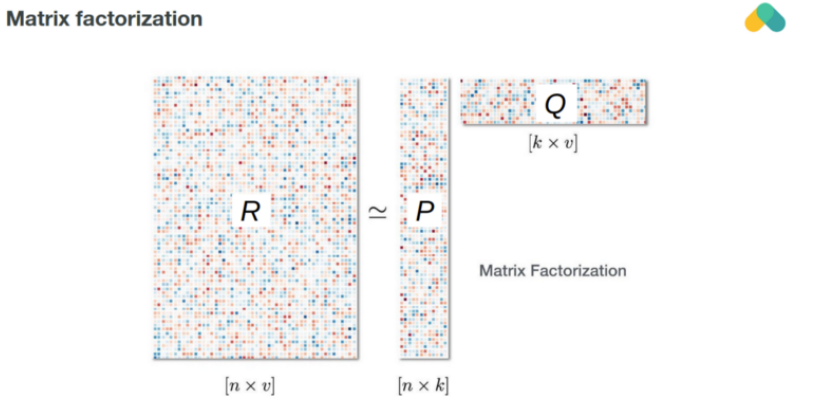


Figure 7: Réduction de la dimension

Déclaration de Montréal pour une IA responsable

Dix principes :

- le bien-être,
- le respect de l'autonomie,
- la protection de l'intimité et de la vie privée,
- la solidarité, la participation démocratique,
- l'équité,
- l'inclusion de la diversité,
- la prudence,
- la responsabilité et
- le développement soutenable.

https://www.declarationmontreal-iaresponsable.com/la-declaration

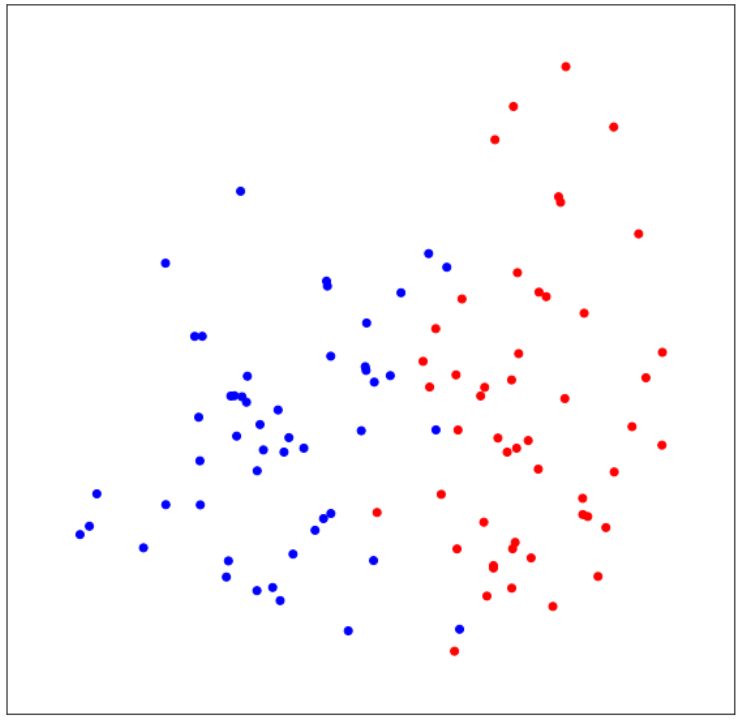
Le problème de classification

Exemples

Spam detection

- Données : emails
- Input : email
- Output : Spam or No Spam

Classification binaire : toy datasets



- But : retrouver la classe
- Input : 2 predicteurs
- Output : classe

Classification multi-classes

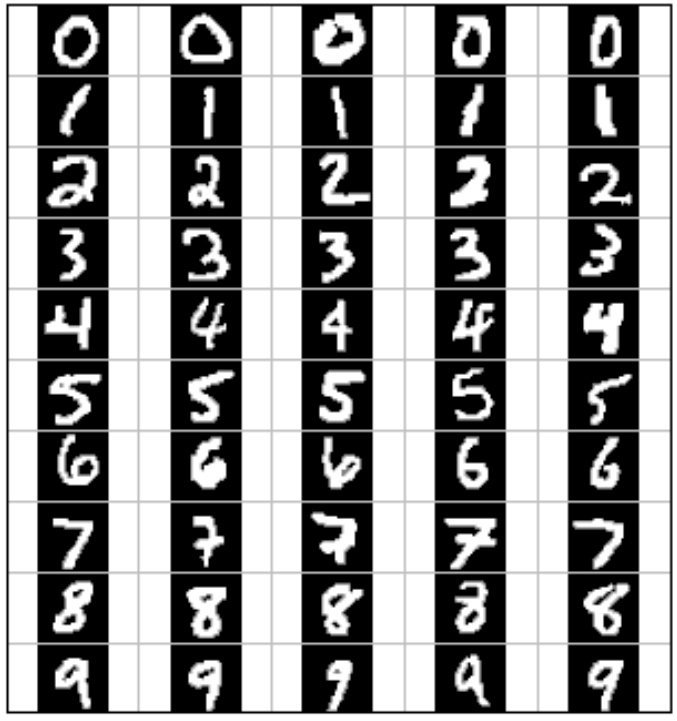
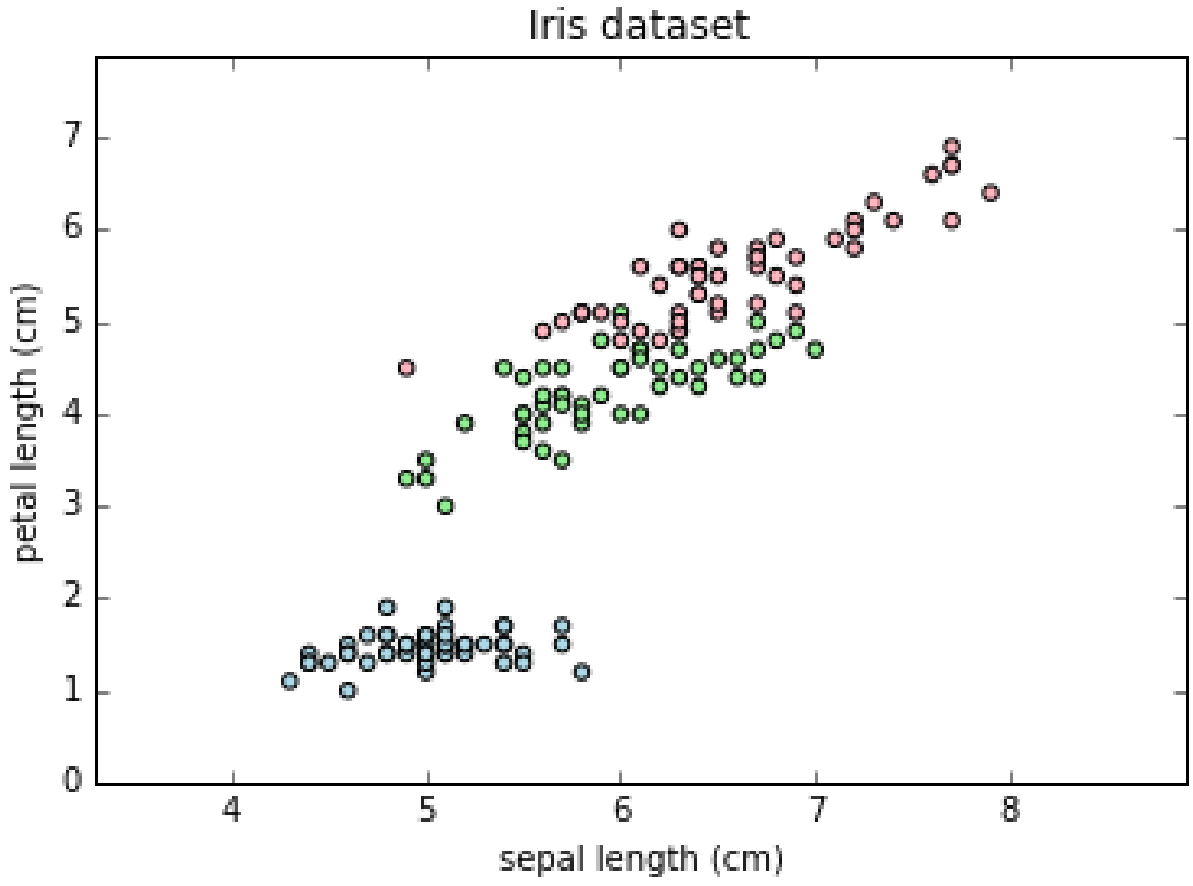


Figure 8: Jeu de données MNIST

- Lire un code postal sur une enveloppe.
- But : assigner un chiffre à une image.
- Input : image.
- Output : chiffre correspondant.

Classification multi-classes : Iris dataset



- But : retrouver la classe
- Input : 2 predicteurs
- Output : classe

Classification

Le problème de classification binaire

On a des données d'apprentissage (learning data) pour des individus $i = 1, \ldots, n$. Pour chaque individu i :

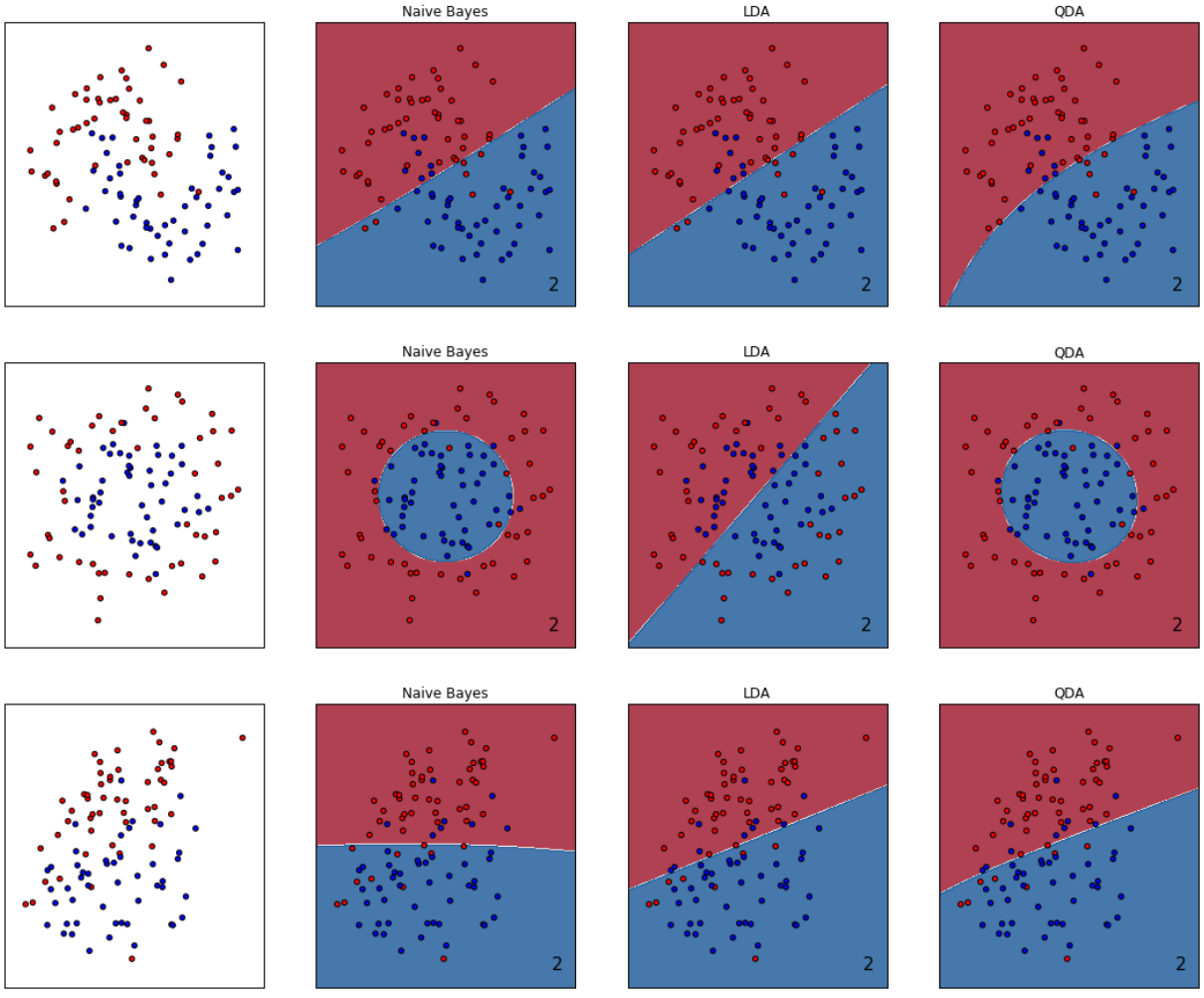
- on a un vecteur de covariables (features) $X_i \in \mathcal{X} \subset \mathbb{R}^d$
- la valeur de son label $Y_i \in \{-1, 1\}$.
- on suppose que les couples (X_i, Y_i) sont des copies i.i.d. de (X, Y) de loi inconnue.

But

- On veut pour un nouveau vecteur X_+ de features prédire la valeur du label Y_+ par $\hat{Y}_+ \in \{-1, 1\}$
- Pour cela, on utilise les données d'apprentissage $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ pour construire un **classifieur** $\hat{c}$ de telle sorte que

$$\hat{Y}_+ = \hat{c}(X_+).$$

Classification binaire : toy datasets



Le problème de classification multi-classes

On a des données d'apprentissage (learning data) pour des individus $i = 1, \ldots, n$. Pour chaque individu i :

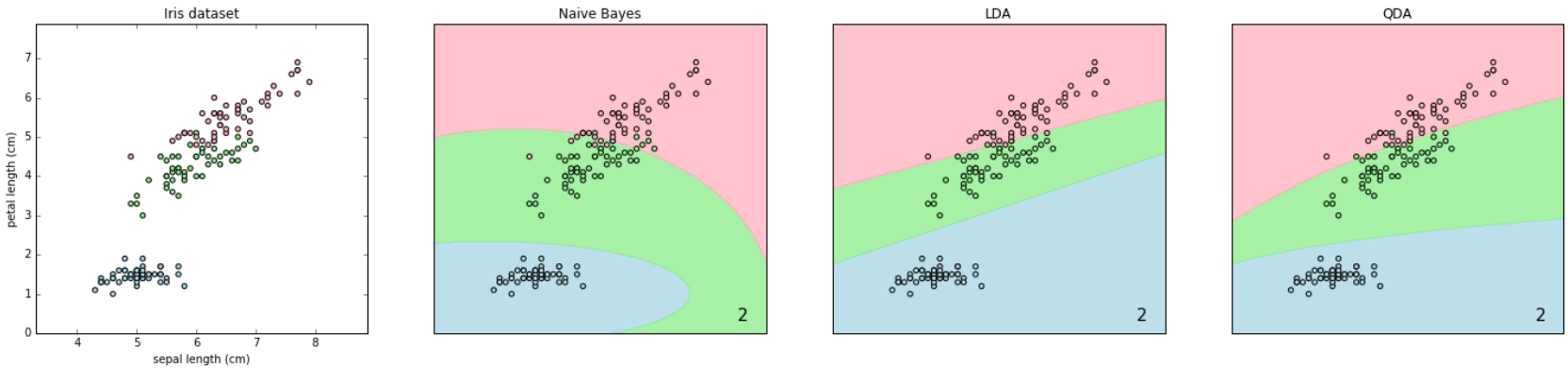
- on a un vecteur de covariables (features) $X_i \in \mathbb{R}^d$
- la valeur de son label $Y_i \in \mathcal{C} = \{1, \ldots, K\}$.
- on suppose que les couples (X_i, Y_i) sont des copies i.i.d. de (X, Y) de loi inconnue.

But

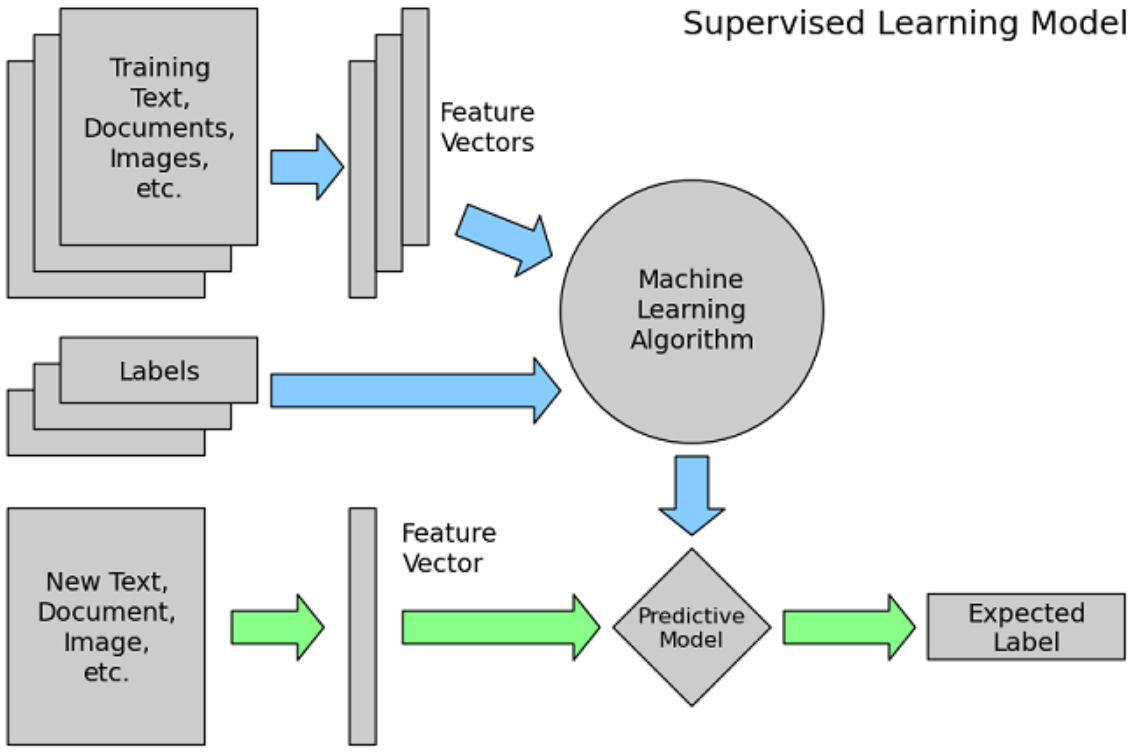
- On veut pour un nouveau vecteur X_+ de features prédire la valeur du label Y_+ par $\hat{Y}_+ \in \mathcal{C} = \{1, \ldots, K\}$
- Pour cela, on utilise les données d'apprentissage $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ pour construire un **classifieur** $\hat{c}$ de telle sorte que

$$\hat{Y}_+ = \hat{c}(X_+).$$

Classification multi-classes : Iris dataset



Apprentissage statistique supervisé



- Input : covariables, variables explicatives, features $X = (X^1, \ldots, X^d)$
- Ouput : variable à expliquer, variable dépendante, réponse, label Y

Approche probabiliste / statistique

Approche probabiliste / statistique en classification binaire

- Pour construire le classifieur $\hat{c}$, on construit des estimateurs $\hat{p}_1(x)$ et $\hat{p}_{-1}(x)$ de

$$p_1(x) = \mathbb{P}(Y = 1 | X = x) \quad \text{et} \quad p_{-1}(x) = 1 - p_1(x)$$

- en modélisant la loi de $Y|X$.
- Puis, conditionnellement à $X_+ = x$, on classifie en utilisant la règle

$$\hat{Y}_+ = \hat{c}(x) = \begin{cases} 1 & \text{si } \hat{p}_1(x) \geq s \\ -1 & \text{sinon} \end{cases}$$

pour un seuil $s \in (0, 1)$.

- Si on choisit $s = 1/2$, cela revient à classifier suivant la plus grande valeur entre $\hat{p}_1(x)$ et $\hat{p}_{-1}(x)$ (on retient cette règle dans la suite).

Classifieur bayésien (1)

Formule de Bayes

Nous savons que

$$\begin{aligned} p_y(x) = \mathbb{P}(Y = y | X = x) &= \frac{f_y(x)\mathbb{P}(Y = y)}{f(x)} \\ &= \frac{f_y(x)\mathbb{P}(Y = y)}{\sum_{y'=-1,1} f_{y'}(x)\mathbb{P}(Y = y')} \\ &\propto f_y(x)\mathbb{P}(Y = y), \end{aligned}$$

où f est la densité jointe de X et f_y est la densité de X condionnellement à $Y = y$.

Donc si on estime les loi de $X|Y$ et de Y, on a un estimateur de celle de $Y|X$.

Classifieur bayésien (2)

Classifieur bayésien

On construit un classifieur grâce à la formule de Bayes

- en modélisant la loi $X|Y$ puis en l'estimant
- et en estimant la loi marginale de Y.

On aura donc

$$\hat{p}_y(x) = \widehat{\mathbb{P}}(Y = y|X = x) \propto \hat{f}_y(x)\widehat{\mathbb{P}}(Y = y)$$

puis

$$\begin{aligned}\hat{Y}_+ = \hat{c}(x) &= \begin{cases} 1 & \text{si } \hat{p}_1(x) \geq \hat{p}_{-1}(x) \\ -1 & \text{sinon} \end{cases} \\ &= \begin{cases} 1 & \text{si } \hat{f}_1(x)\widehat{\mathbb{P}}(Y = 1) \geq \hat{f}_{-1}(x)\widehat{\mathbb{P}}(Y = -1) \\ -1 & \text{sinon} \end{cases} \\ &= \text{argmax}_{y=-1,1}\hat{f}_y(x)\widehat{\mathbb{P}}(Y = y).\end{aligned}$$

Maximum a posteriori en classification binaire

Si, conditionnellement à $X_+ = x$, on classifie en utilisant la règle

$$\hat{Y}_+ = \hat{c}(x) = \begin{cases} 1 & \text{si } \hat{p}_1(x) \geq \hat{p}_{-1}(x) \\ -1 & \text{sinon ,} \end{cases}$$

c'est équivalent à utiliser une fonction discriminante.

Fonction discriminante

$$\hat{\delta}_y(x) = \log \widehat{\mathbb{P}}(X = x | Y = y) + \log \widehat{\mathbb{P}}(Y = y)$$

pour $y = 1, -1$ et de classifier suivant sa valeur pour chaque y.

Classification multi-classes et maximum a posteriori

- On modélise la distribution de $Y|X$
- On construit des estimateurs $\hat{p}_k(x)$ pour $k \in \mathcal{C}$ tels que

$$\sum_{k \in \mathcal{C}} \hat{p}_k(x) = 1$$

- Conditionnellement à $X_+ = x$, on classifie en utilisant la règle

$$\hat{Y}_+ = \text{argmax}_{k \in \mathcal{C}} \hat{p}_k(x).$$

On peut alors définir des fonctions discriminantes pour $k \in \mathcal{C}$

$$\hat{\delta}_k(x) = \log \widehat{\mathbb{P}}(X = x | Y = k) + \log \widehat{\mathbb{P}}(Y = k).$$

et décider de classifier avec la règle

$$\hat{Y}_+ = \text{argmax}_{k \in \mathcal{C}} \hat{\delta}_k(x).$$

Remarque

- On peut choisir plusieurs modèles pour la loi de $X|Y$ qui donnent des classifieurs différents.
- Le plus simple est le "Naive Bayes"
- Nous verrons aussi l'analyse discriminante linéaire (Linear discriminant analysis - LDA) et quadratique (Quadratic discriminant Analysis -QDA)

Naive Bayes

- On veut un modèle pour la loi de $X|Y$.
- Le plus simple est de considérer que les features X^j $(j = 1, \ldots, d)$ sont indépendantes conditionnellement à Y.
- C'est équivalent à supposer que la densité conditionnelle $f_y(x)$ de $X|Y = y$ est donnée par

$$\prod_{j=1}^{d} f_y^j(x^j)$$

où f_y^j est la densité de $X^j|Y = y$.

Quelle loi pour $X^j | Y = y$?

- Si X^j est une variable continue, on choisit la loi normale

$$X^j | Y = y \sim \mathcal{N}(\mu_{j,y}, \sigma^2_{j,y}),$$

  dont les paramètres $\mu_{j,k}$ et $\sigma^2_{j,k}$ sont estimés par maximum de vraisemblance.
  Il suffit alors d'estimer les $\mu_{j,y}, \sigma^2_{j,y}$ pour tous les $j = 1, \ldots, d$ et les $y \in \mathcal{C}$.
- Si X^j est une variable discrète, on choisit la loi de Bernoulli ou la loi multinomiale.

Analyse discriminante

Analyse discriminante

Supposons que

$$X|Y = y \sim \mathcal{N}(\mu_y, \Sigma_y).$$

On rappelle que la densité de la loi $\mathcal{N}(\mu, \Sigma)$ est donnée par

$$f(x) = \frac{1}{(2\pi)^{d/2}\sqrt{\det \Sigma}} \exp\Big(-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)\Big).$$

Estimation

On estime les paramètres inconnus par maximum de vraisemblance. Donc pour $y \in \mathcal{C}$, on pose

$$I_y = \{i = 1, \ldots, n \ : \ Y_i = y\} \quad n_y = |I_y|$$

et

$$\widehat{\mathbb{P}}(Y = y) = \frac{n_y}{n}, \quad \hat{\mu}_y = \frac{1}{n_y} \sum_{i \in I_y} X_i,$$

$$\hat{\Sigma}_y = \frac{1}{n_y} \sum_{i \in I_y} (X_i - \hat{\mu}_y)(X_i - \hat{\mu}_y)^\top.$$

Ce sont simplement les proportions, moyennes et variances de chaque sous-groupe de données défini par la valeur du label.

Fonctions discriminantes

Dans ce cas, les fonctions discriminantes sont

$$\begin{aligned}\hat{\delta}_y(x) &= \log \widehat{\mathbb{P}}(X = x | Y = y) + \log \widehat{\mathbb{P}}(Y = y) \\ &= -\frac{1}{2}(x - \hat{\mu}_y)^\top \hat{\Sigma}_y^{-1}(x - \hat{\mu}_y) - \frac{d}{2}\ln(2\pi) \\ &\quad - \frac{1}{2}\log\det\hat{\Sigma}_y + \log \widehat{\mathbb{P}}(Y = y)\end{aligned}$$

Linear Discriminant Analysis (LDA)

Supposons que $\Sigma = \Sigma_y$ pour tout $y \in \mathcal{C}$ (cela revient à supposer qu'il y a la même structure de corrélation dans chaque groupe)

Linear Discriminant Analysis (LDA)

Les frontières de décision sont linéaires et les régions ont la forme (pour la classification binaire) $\langle x, w \rangle \geq c$ avec

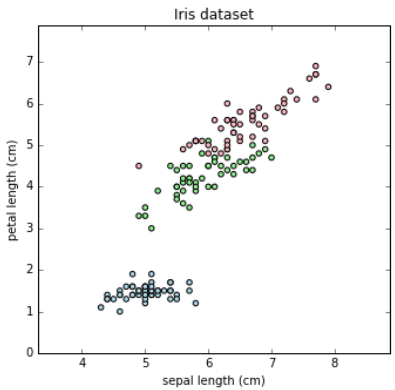
$$w = \hat{\Sigma}^{-1}(\hat{\mu}_1 - \hat{\mu}_{-1})$$

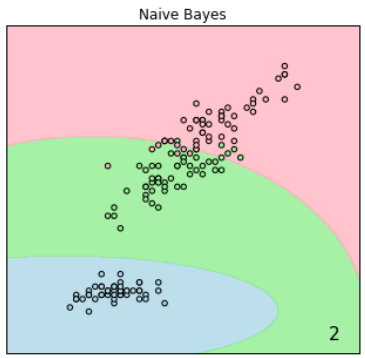
$$c = \frac{1}{2}(\langle \hat{\mu}_1, \hat{\Sigma}^{-1}\hat{\mu}_1 \rangle - \langle \hat{\mu}_{-1}, \hat{\Sigma}^{-1}\hat{\mu}_{-1} \rangle) + \log\left(\frac{\widehat{\mathbb{P}}(Y=1)}{\widehat{\mathbb{P}}(Y=-1)}\right).$$

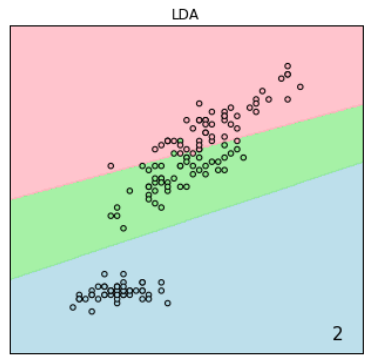
Quadratic Discriminant Analysis (QDA)

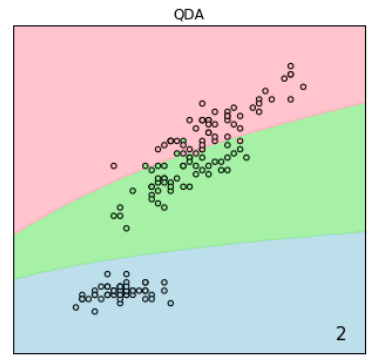
On n'assume plus que $\Sigma = \Sigma_y$ pour tout $y \in \mathcal{C}$. Dans ce cas, les frontières sont quadratiques.

Exemple sur le dataset `Iris`









Classifieur constants sur une partition

Rappel sur la classification

On a pour $i = 1, \ldots, n$

- $X_i \in \mathbb{R}^d$
- $Y_i \in \mathcal{C} = \{-1, 1\}$ ou $\{1, \ldots, K\}$.

But

- On veut pour un nouveau vecteur X_+ de features prédire la valeur du label Y_+ par $\hat{Y}_+ \in \mathcal{C} = \{1, \ldots, K\}$
- Pour cela, on utilise les données d'apprentissage $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ pour construire un **classifieur** $\hat{c}$ de telle sorte que

$$\hat{Y}_+ = \hat{c}(X_+).$$

Classifieur constants sur une partition

On va considérer

- une partition $\mathcal{A} = \{A_1, \ldots, A_M\}$ de $\mathcal{X}$ (qui peut dépendre des données)
- l'ensemble $\mathcal{F}_{\mathcal{A}}$ des fonctions constantes sur $\mathcal{A}$
- la perte 0/1 $\ell(y, y') = \mathbb{1}_{yy' \leq 0}$

on cherche alors un classifieur $\hat{c}$ qui vérifie

$$\hat{c}_{\mathcal{A}} = \operatorname*{argmin}_{c \in \mathcal{F}_{\mathcal{A}}} \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, c(X_i)) = \operatorname*{argmin}_{c \in \mathcal{F}_{\mathcal{A}}} \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{Y_i c(X_i) \leq 0}.$$

Vote majoritaire

En classification binaire, on sait alors que $\hat{c}_{\mathcal{A}}$ vérifie pour $x \in A_m$

$$\hat{c}_{\mathcal{A}}(x) = \begin{cases} 1 & \text{si } \#\{i : X_i \in A_m, Y_i = 1\} > \#\{i : X_i \in A_m, Y_i = -1\} \\ -1 & \text{sinon} \end{cases}$$

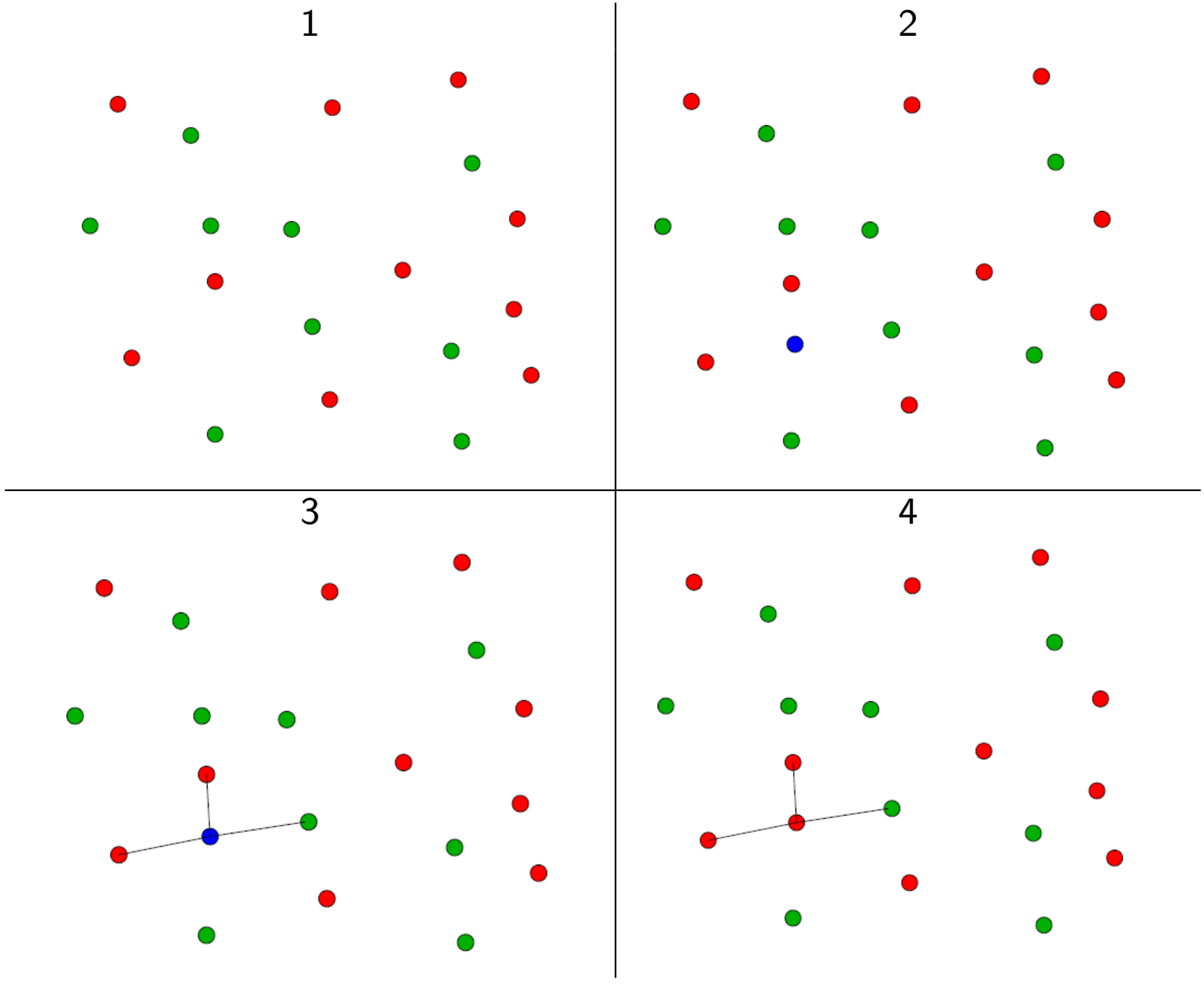
$$= \begin{cases} 1 & \text{si } \bar{Y}_{A_m} > 0 \\ -1 & \text{sinon} \end{cases}$$

En classification multi-classes, on posera pour $x \in A_m$

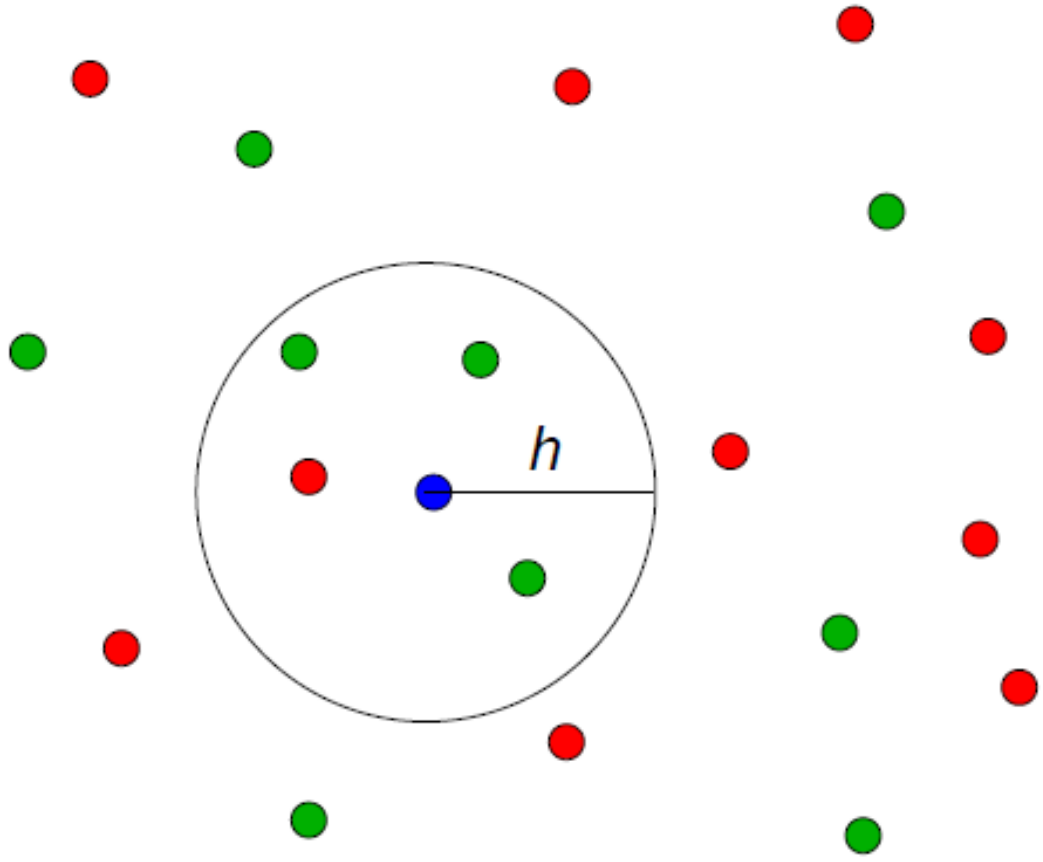
$$\hat{c}_{\mathcal{A}}(x) = \underset{k \in \{1,\dots,K\}}{\text{argmax}} \ \#\{i : X_i \in A_m, Y_i = k\}$$

Il reste à choisir la partition $\mathcal{A} = \{A_1, \dots, A_M\}$ de $\mathcal{X}$!

Exemple: k plus proches voisins (avec $k = 3$)



Exemple: k plus proches voisins (avec $k = 4$)



k plus proches voisins

On considère l'ensemble $\mathcal{I}_x$ composé des k indices de $\{1, \ldots, n\}$ pour lesquels les distances $\|x - X_i\|$ sont minimales.

On pose

$$\hat{c}(x) = \underset{k \in \{1,\ldots,K\}}{\text{argmax}} \ \#\{i \in \mathcal{I}_x, Y_i = k\}.$$

- En pratique, il faut choisir la distance
- et k !!

Partition

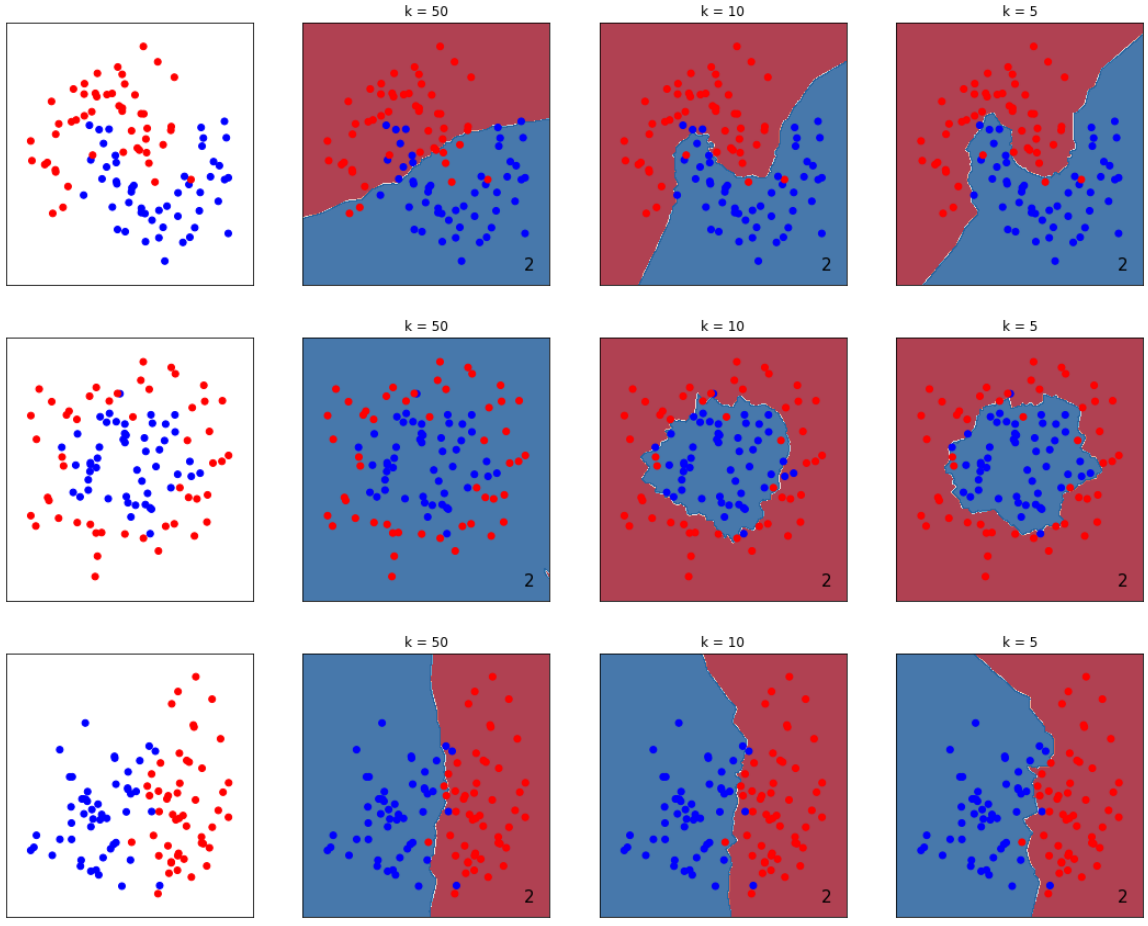
On remarque que $\mathcal{I}_x$ appartient à l'ensemble $\{\phi^1, \ldots, \phi^M\}$ des combinaisons de k éléments parmi n avec

$$M = \binom{n}{k}.$$

On peut donc poser

$$A_m = \{x \in \mathcal{X}, \mathcal{I}_x = \phi^m\}.$$

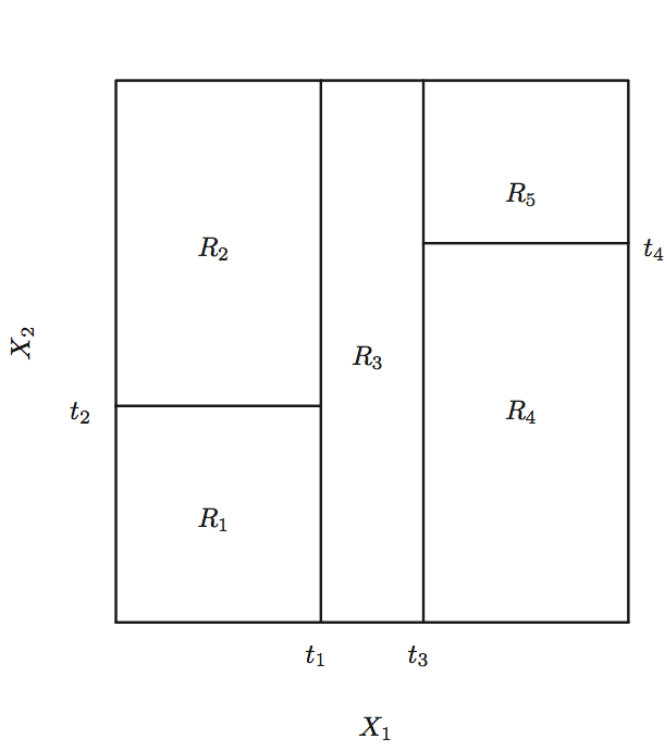
k-NN

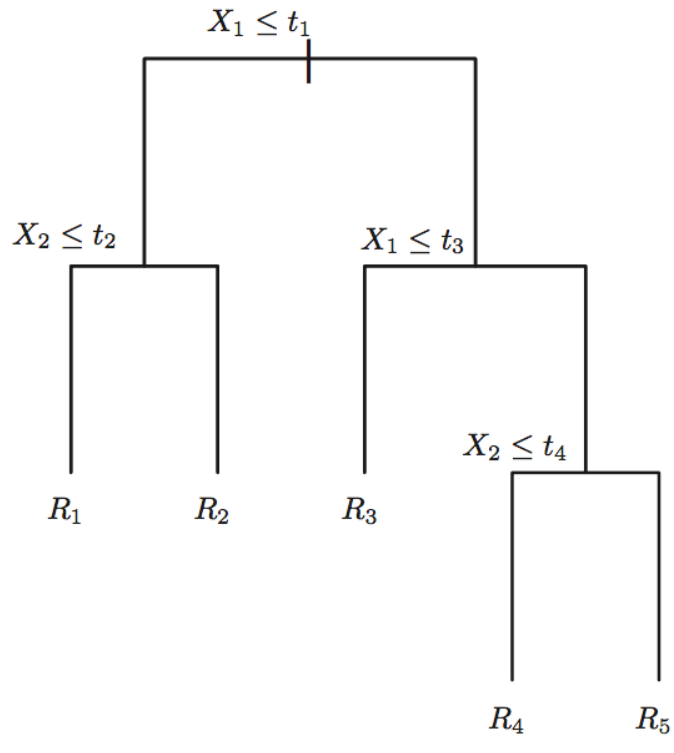


Arbres de décision

Approche “top-bottom”

- On commence par une région qui contient toutes les données
- On coupe récursivement les régions par rapport à une variable et une valeur de cette variable





Heuristique:

On veut choisir la valeur du “split” de telle sorte que les deux nouvelles régions sont les plus **homogènes** possible...

L'**homogénéité** peut se définir par différents critères

- la variance empirique
- l'indice de Gini
- l'entropie.

Arbre de classification à partir de l'indice de Gini

On coupe une région R en deux parties R_- and R_+. Pour chaque variable $j = 1, \ldots, p$ et chaque valeur de "split" t, on définit

$$R_-(j,t) = \{x \in R : x^j < t\} \quad \text{et} \quad R_+(j,t) = \{x \in R : x^j \geq t\}.$$

on cherche j et t qui minimisent

$$\text{Gini}(R_-) + \text{Gini}(R_+)$$

where

$$\text{Gini}(R) = \frac{1}{|\{i, X_i \in R\}|} \sum_{k \in \mathcal{C}} \hat{p}_{R,k}(1 - \hat{p}_{R,k})$$

où $\hat{p}_{R,k}$ est la proportion d'observations avec le label k dans l'ensemble des $\{i, X_i \in R\}$.

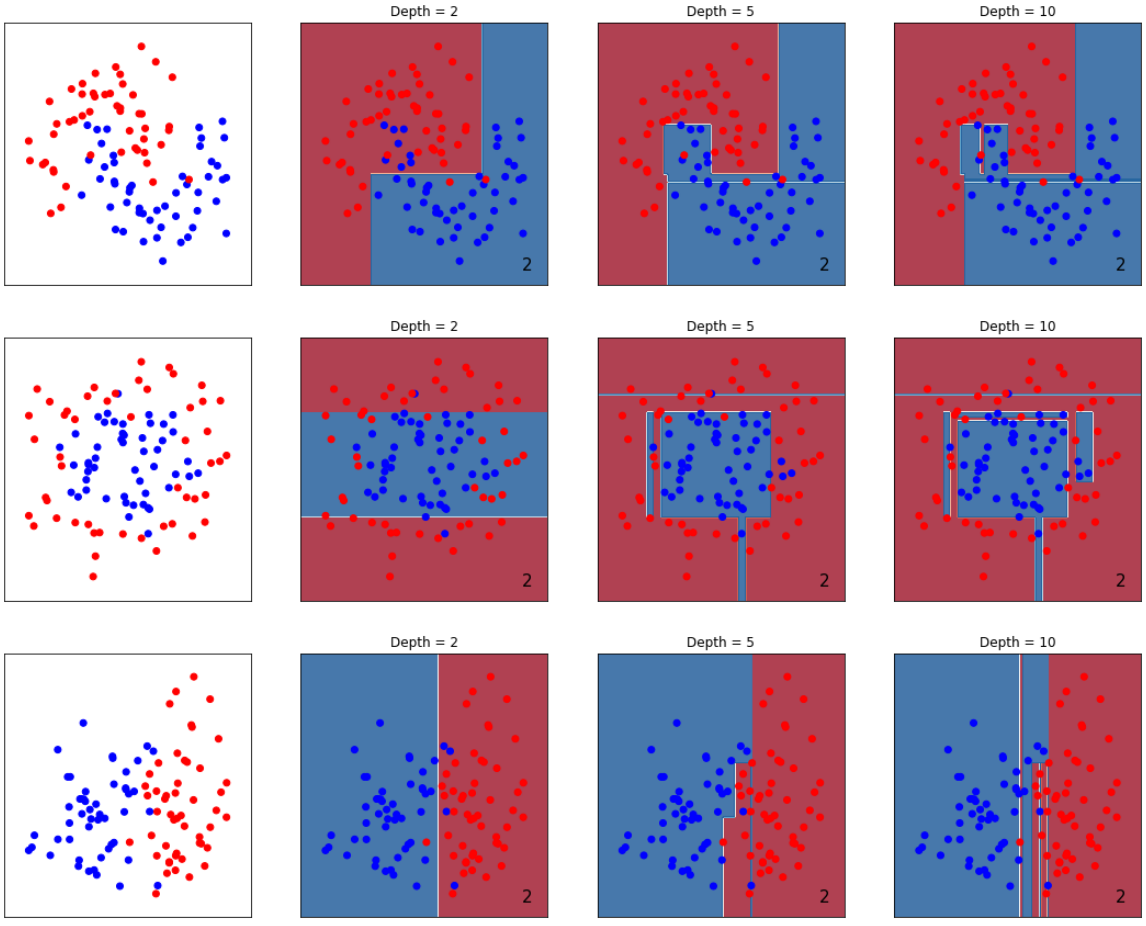
Algorithmes CART, C.4.5

- l'algorithme CART utilise l'indice de Gini
- l'algorithme C.4.5 (pas implementé dans `sklearn`) utilise l'entropie

$$E(R) = -\sum_{k \in \mathcal{C}} \hat{p}_{R,k} \log(\hat{p}_{R,k})$$

- il y a d'autres critères possibles (χ^2, etc)

CART


Depth = 2
Depth = 5
Depth = 10
Depth = 2
Depth = 5
Depth = 10
Depth = 2
Depth = 5
Depth = 10

Règles d'arrêt et algorithmes dérivés

Règles d'arrêt

On arrête l'algorithme quand

- l'arbre a atteint une taille maximale (fixée à l'avance)
- le nombre de feuilles atteint une valeur maximale (fixée à l'avance)
- quand les effectifs des noeuds terminaux atteignent une valeur minimale (fixée à l'avance)

En pratique

En pratique, ce sont des algorithmes instables et qui sur-apprennent, on les utilisent dans des algorithmes plus complexes qui "mélangent" des arbres

- les forêts alétoires (random forests)
- le boosting.

Minimisation de l'erreur, méthodes basées sur l'optimisation

Régression logistique

- C'est le plus utilisé des algorithmes de classification.
- On modélise la loi de $Y|X$

Modèle logistique

Pour $y \in \{-1, 1\}$, on considère le modèle

$$\mathbb{P}(Y = 1|X = x) = \sigma(x^\top \beta + \beta_0)$$

où $\beta \in \mathbb{R}^d$ est un vecteur de régression et $\beta_0 \in \mathbb{R}$ est l'intercept, σ est la fonction sigmoïde

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Fonction de perte logistique

On calcule $\hat{\beta}$ et $\hat{\beta}_0$ comme suit

$$(\hat{\beta}, \hat{\beta}_0) \in \underset{\beta \in \mathbb{R}^d, \beta_0 \in \mathbb{R}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \log(1 + e^{-Y_i(\langle X_i, \beta \rangle + \beta_0)})$$

C'est un problème de minimisation convexe et régulier, il y a de nombreux algorithmes (descente de gradient, Newton, etc)

On peut introduire la fonction de **perte logistique**

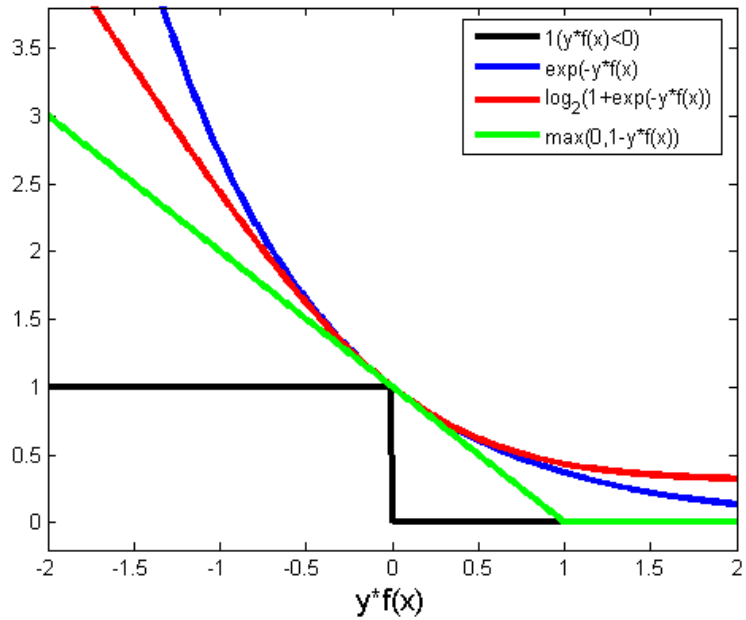
$$\ell(y, y') = \log(1 + e^{-yy'})$$

alors

$$(\hat{\beta}, \hat{\beta}_0) \in \underset{\beta \in \mathbb{R}^d, \beta_0 \in \mathbb{R}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, \langle X_i, \beta \rangle + \beta_0)$$

Autres fonctions de perte classiques en classification binaire

- Hinge loss (SVM), $\ell(y, y') = (1 - yy')_+$
- Quadratic hinge loss (SVM), $\ell(y, y') = \frac{1}{2}(1 - yy')_+^2$
- Huber loss $\ell(y, y') = -4yy' \mathbb{1}_{yy' < -1} + (1 - yy')_+^2 \mathbb{1}_{yy' \geq -1}$



- Toutes ces pertes peuvent être vues comme des approximations convexe de la perte 0/1 $\ell(y, y') = \mathbb{1}_{yy' \leq 0}$

Erreur empirique / erreur de généralisation

On se donne une fonction classifiante déterministe $c : \mathbb{R}^d \in \mathcal{C}$, on définit $\mathcal{L}$ la loi inconnue des couples (X_i, Y_i) et

Erreur empirique ou erreur visible

$$R_n(c) = \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, c(X_i)).$$

Erreur de généralisation

$$R(c) = \mathbb{E}_{\mathcal{L}} \left(\ell(Y_+, c(X_+)) \right)$$

où (X_+, Y_+) est un couple indépendant de $\mathcal{D}_n$

- En classification, on prend souvent $\ell(y, y^{'}) = \mathbb{1}_{y \neq y'}$, dans ce cas $1 - R_n(c)$ est appelé “accuracy” de c.

Erreur empirique / erreur de généralisation

- On suppose que les couples (X_i, Y_i) sont des copies i.i.d. de (X, Y) de loi $\mathcal{L}$ inconnue
- on note $\mathcal{D}_n = \Big\{(X_1, Y_1), \ldots, (X_n, Y_n)\Big\}$

On se donne une fonction classifiante déterministe $c : \mathbb{R}^d \in \mathcal{C}$, on définit

Erreur empirique ou erreur visible

$$R_n(c) = \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, c(X_i)).$$

Erreur de généralisation

$$R(c) = \mathbb{E}_{\mathcal{L}} \Big(\ell(Y_+, c(X_+))\Big)$$

où (X_+, Y_+) est un couple indépendant de $\mathcal{D}_n$

Remarques

- En classification, on prend souvent $\ell(y, y') = \mathbb{1}_(y \neq y')$, dans ce cas $1 - R_n(c)$ est appelé "accuracy" de c.
- On a
$$R(c) = \mathbb{E}_{\mathcal{L}}\Big(R_n(c)\Big).$$
- On voudrait retrouver
$$c^\star = \underset{c}{\text{argmin}}\, R(c)$$

Classifieur bayésien

$c^\star = \text{argmin}_c\, R(c)$ est, dans le cas de la classification et de la perte 0/1, le classifieur bayésien.

Mais on se restreint le plus souvent à un sous-ensemble $\mathcal{G}$ (par exemple les fonctions constantes sur une partition $\mathcal{A}$)

$$c_{\mathcal{G}}^{\text{oracle}} = \operatorname*{argmin}_{c \in \mathcal{G}} R(c)$$

puis, comme la loi $\mathcal{L}$ est inconnue, on remplace R par R_n

$$\hat{c}_{\mathcal{G}} = \operatorname*{argmin}_{c \in \mathcal{G}} R_n(c).$$

On a bien sûr

$$R(\hat{c}_{\mathcal{G}}) \geq R(c_{\mathcal{G}}^{\text{oracle}}) \geq R(c^\star).$$

Bornes sur les risques

Erreur visible / erreur de généralisation

Ce que l'on veut comparer

On veut comparer $R_n(\hat{c}_{\mathcal{G}})$ et $R(c^\star)$ pour mesurer "l'optimisme" quand on calcule $R_n(\hat{c}_{\mathcal{G}})$.

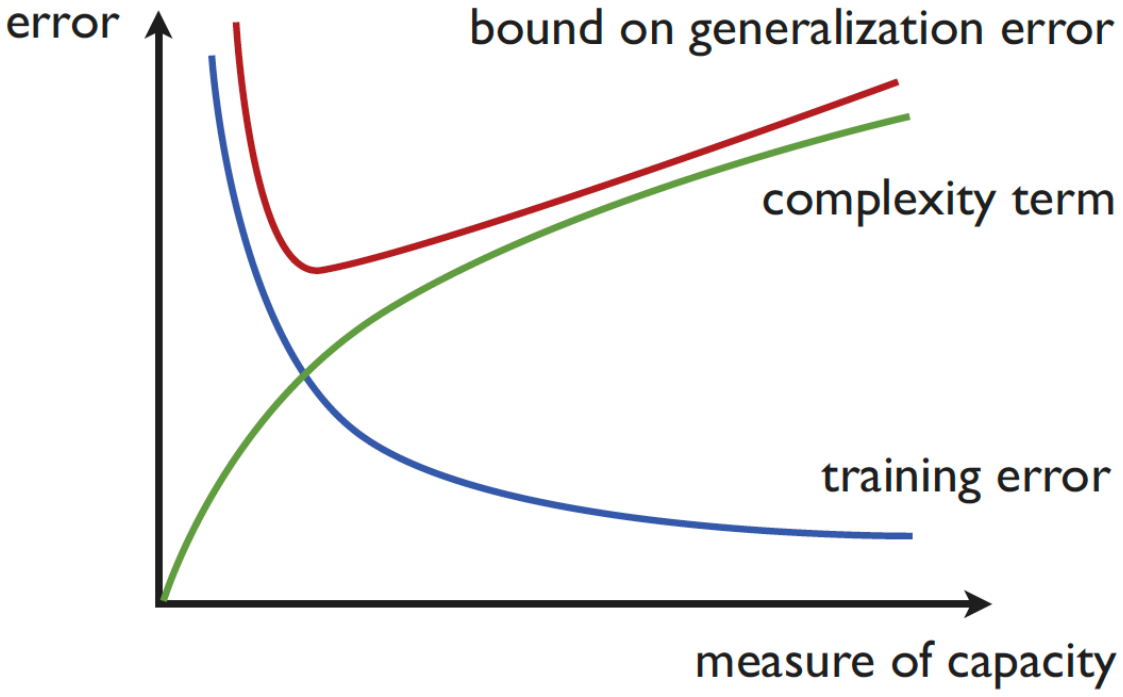


Figure 9: In **mohri2012foundations**

Première borne de risque

On montre que, avec probabilité plus grande que $1 - \varepsilon$

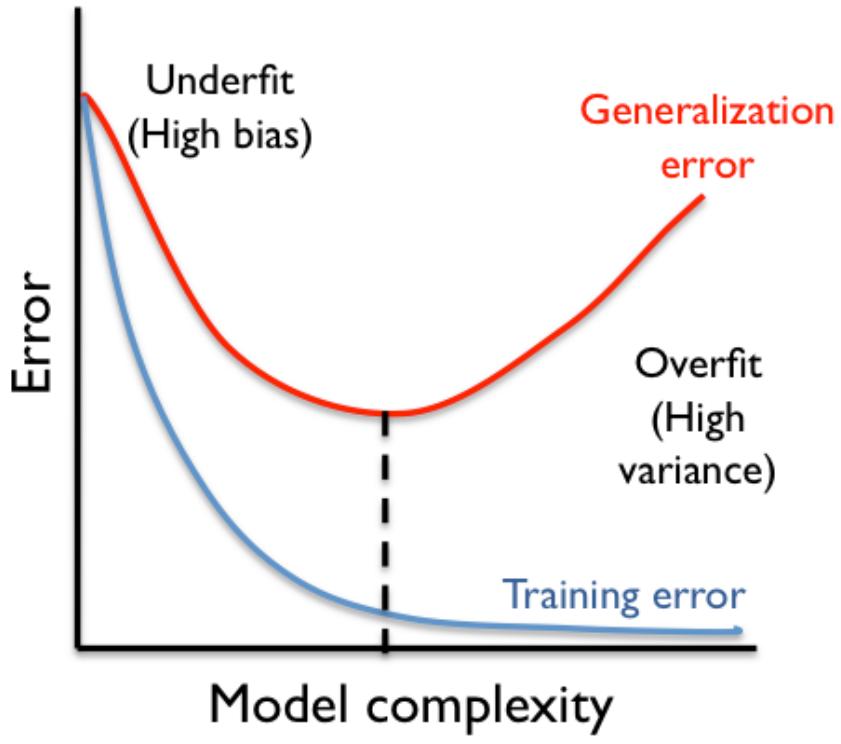
$$R(\hat{c}_{\mathcal{G}}) \leq R(c^{\star}) + \text{erreur d'approximation} + \sqrt{\frac{2\log(2|\mathcal{G}|\varepsilon^{-1})}{n}}$$

Borne “risque visible/erreur de généralisation”

On montre que, avec probabilité plus grand que $1 - \varepsilon$

$$R(\hat{c}_{\mathcal{G}}) \leq R_n(\hat{c}_{\mathcal{G}}) + \sqrt{\frac{2\log(2|\mathcal{G}|\varepsilon^{-1})}{n}}$$

Sur-apprentissage



Cross Validation

Cross Validation

- On utilise $(1-\epsilon)n$ observations pour apprendre et ϵn pour vérifier !
- On entraîne sur un jeu de données de taille $(1-\epsilon) \times n$ à la place de n!
- Assez instable si ϵn est trop petit

- Variations classiques:
  - Leave One Out,
  - K-fold cross validation ($K = 3, 5, 10$).

Text mining: comment transformer un texte en un vecteur numérique ?

Hashing

Hashing

- Idée: réduire le nombre de valeurs d'une variable nominale avec des valeurs dans un grand ensemble $\mathcal{D}$

Hashing

- Construction d'une fonction de *hashage* H: $\mathcal{D} \to \{1, \ldots, V\}$ et on utilise les valeurs hashées au lieu des valeurs originales.
- La fonction de hashage doit être la *plus injective possible...*, du moins au sens probabiliste.

Construire une telle fonction est un art !

Bag of Words

Bag of Words

- Comment transformer un texte en vecteur numérique de features ?

La stratégie "Bag of Words strategy"

- Créer un *dictionnaire* de mots,
- Calculer un *poids* pour chaque mot.

Construction d'une liste

- Faire une liste de tous les mots avec le nombre d'occurrence
- Réunir les mots qui ont la même racine (stemming)
- Hash les racines avec une fonction de hashage (MurmurHash avec 32bits par exemple)
- Calculer l'histogramme $h_w(d)$

TD-IDF

Calcul des poids

- Calculer l'histogramme $h_w(d)$
- Re-normaliser :
  - tf transformation (profil du mot):

$$\mathrm{tf}_w(d) = \frac{h_w(d)}{\sum_w h_w(d)}$$

    de telle sorte que $\mathrm{tf}_w(d)$ est la fréquence dans le document d.
  - tf-idf transformation (profil du mot re-pondéré par sa rareté):

$$\mathrm{tf-idf}_w(d) = \mathrm{idf}_w \times \mathrm{tf}_w(d)$$

    avec idf un poids dépendant du corpus

$$\mathrm{idf}_w = \log \frac{n}{\sum_{i=1}^{n} \mathbf{1}_{h_w(d_i)\neq 0}}$$

- Utiliser le vecteur $\mathrm{tf}(d)$ (or $\mathrm{tf-idf}(d)$) pour décrire un document.
- C'est le pré-processing le plus classique en textmining.

Probabilistic latent semantic analysis (PLSA)

- Modèle:

$$\mathbb{P}(\mathrm{tf}) = \sum_{k=1}^{K} \mathbb{P}(k)\,\mathbb{P}(\mathrm{tf}|k)$$

  avec k le thème caché, $\mathbb{P}(k)$ la probabilité du thème et $\mathbb{P}(\mathrm{tf}|k)$ une loi multinomiale pour le thème.

- Clustering avec un modèle de mélange

$$\mathbb{P}(k|\mathrm{tf}) = \frac{\widehat{\mathbb{P}(k)}\widehat{\mathbb{P}(\mathrm{tf}|k)}}{\sum_{k'} \widehat{\mathbb{P}(k')}\widehat{\mathbb{P}(\mathrm{tf}|k')}}$$

- Modèle de mélange
- Il existe une variante bayésienne appelée Latent Dirichlet Allocation.

Mots et Word Vectors

Word Vectors

Word Embedding

- On construit une représentation des mots dans $\mathbb{R}^d$.
- en espérant que la relation entre 2 vecteurs est liée à la relation entre les 2 mots dont ils sont issus.

Word And Context

Look ! A single word and its context

Le mot et son contexte

- Idée: caractériser un mot w par son contexte c...
- Description probabiliste:
  - Loi jointe : $f(w,c) = \mathbb{P}(w,c)$
  - Lois conditionnelles: $f(w,c) = \mathbb{P}(w|c)$ or $f(w,c) = \mathbb{P}(c|w)$.
  - Information mutuelle : $f(w,c) = \mathbb{P}(w,c) / (\mathbb{P}(w)\mathbb{P}(c))$
- Le mot w est caractérisé par le vecteur $C_w = (f(w,c))_c$ ou $C_w = (\log f(w,c))_c$.

- En pratique, on estime C sur un large corpus
- Attention : c'est un modèle de très grande dimension !

- GloVe (Global Vectors) via les moindres carrés
- Word2vec via la régression logistique
- Singular value decomposition