

# Préparation de données

Daniel Aloise <[daniel.aloise@polymtl.ca](mailto:daniel.aloise@polymtl.ca)>

# *Data munging*

- Les scientifiques de données passent la plupart de leur temps à nettoyer et à structurer leurs données

# *Data munging*

- Les scientifiques de données passent la plupart de leur temps à nettoyer et à structurer leurs données
- L'autre partie ils se plaignent de l'absence de données disponibles
- **Data munging** est l'art d'acquérir des données et de les préparer pour l'analyse

# Langages

Librairies utiles Python: scikit-learn, SciPy, Pytorch, matplotlib, seaborn

Python contient des librairies et des fonctionnalités pour faciliter le *data munging* (p. ex.p. expressions régulières)

R en général préféré par les statisticiens

Matlab/Julia rapide pour manipuler des matrices

Java/C informatique distribuée, Big Data

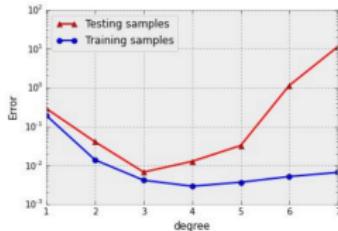
Excel outil de base

# Notebooks TP1,2,3

## Jupyter

- Permettent de combiner code, données, résultats, texte
- Rendent les projets :
  - reproductibles
  - *tweakable*
  - documentés

```
In [48]: degrees = range(1, 8)
errors = np.array([regressor3(d) for d in degrees])
plt.plot(degrees, errors[:, 0], marker='^', c='r', label='Testing samples')
plt.plot(degrees, errors[:, 1], marker='o', c='b', label='Training sample')
plt.yscale('log')
plt.xlabel("degree"); plt.ylabel("Error")
= plt.legend(loc='best')
```



By sweeping the degree we discover two regions of model performance:

- **Underfitting** (degree < 3): Characterized by the fact that the testing error will get lower if we increase the model capacity.
- **Overfitting** (degree > 3): Characterized by the fact the testing will get higher if we increase the model capacity. Note, that the training error is getting lower or just staying the same!

# Notebooks

- Notebooks **maintiennent séquence des procédures utilisées** sur les données du début à la fin du projet
- Attendez-vous à refaire votre analyse à partir de zéro, alors construisez votre code pour le rendre possible

# Format de données

**CSV** pour des tableaux bien structurés

**XML** données structurées, mais pas sous la forme d'un tableau

**JSON** *Javascript Object Notation* pour l'échange des messages entre objets (POO)

**BDD SQL** pour des tableaux multiples et liés

# Où sont les données ?

- La première étape de n'importe quel projet en fouille de données est de **trouver** où sont les données dont on aura besoin.
- Les grandes bases de données contiennent souvent de *métadonnées* :
  - ex. Wikipedia edits
- La réutilisation des métadonnées pour d'autres objectifs est un exercice d'imagination.

# Possibles sources de données

- Données privées [Google, Fb...](#)
- Données gouvernementales
- Données académiques
- Web
- IoT

# Données privées

- Facebook, Google, Twitter, Amazon, LinkedIn, etc. ont de données super intéressantes.
- La plupart des organisations ont (ou devraient avoir) des bases de données internes d'intérêt pour leur business.
- Obtenir un accès extérieur est généralement impossible.
- Au moins, nous aimerais croire cela 
- Les entreprises publient parfois des API qui permettent de récupérer “quelques” données.

# Données privées

- Facebook, Google, Twitter, Amazon, LinkedIn, etc. ont de données super intéressantes.
- La plupart des organisations ont (ou devraient avoir) des bases de données internes d'intérêt pour leur business.
- Obtenir un accès extérieur est généralement impossible.
- Au moins, nous aimerais croire cela 
- Les entreprises publient parfois des API qui permettent de récupérer “quelques” données.

# Données gouvernementales

- Les villes et les pays s'engagent de plus en plus à ouvrir leurs données.
- Le gouvernement du Canada a environ 10,000 jeux de données ouverts .

# Données gouvernementales

- Les villes et les pays s'engagent de plus en plus à ouvrir leurs données.
- Le gouvernement du Canada a environ 10,000 jeux de données ouverts .
- La question clé quand on rend une base de données publique est la **préservation de l'anonymat**.

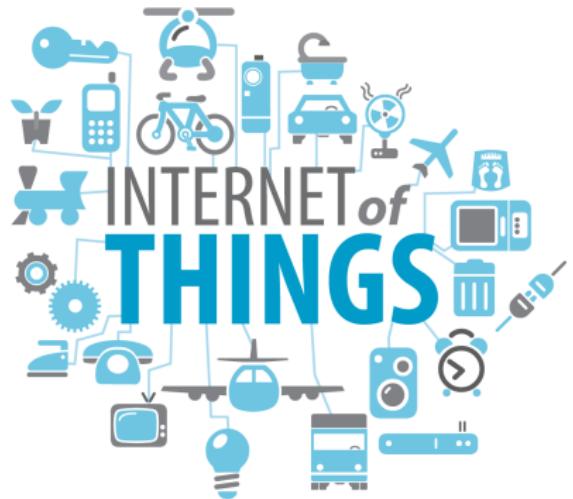
# Données académiques

- Rendre les données disponibles est maintenant une exigence pour les publications.
- Attendez-vous à pouvoir trouver des données économiques, médicales, démographiques et météorologiques si vous cherchez bien.
- Repérez les documents pertinents et demandez aux propriétaires.
- Cependant, c'est probable que ces données ont été déjà très explorées.

# Web search et web scrapping

- Le **scraping** est l'art de dépouiller des données d'une page Web.
- Python fournit des bibliothèques pour aider à analyser / gratter le web, mais il faut d'abord vérifier si :
  - il y a des APIs rendues disponibles par le site ?
  - quelqu'un a déjà codé un *scraper* ?
- Les conditions de service limitent ce que vous pouvez légalement faire.
- **Faites attention !**

# Internet of Things



- Les sources de données sont aujourd'hui très variées.
- Le stockage n'est pas cher !

# Type de données

**Données indépendantes** : soit au niveau des individus, soit au niveau des caractéristiques.

**Données dépendantes** : présentent des relations (ex. réseau sociaux, séries temporelles).  
du temps

Ex: chaque élève on enregistre son poids --> caractéristiques (ex: taille et poids) dépendantes et (ex: taille et lunettes) indépendantes

# Type de données

Supposons que pour chaque élève (*observation*, enregistrement ou lignes), j'enregistre sa taille et son poids (*features*, attribut ou colonnes). Ces données sont-elles dépendantes ?

oui

# Type de données

Supposons que pour chaque élève (*observation*, enregistrement ou lignes), j'enregistre sa taille et son poids (*features*, attribut ou colonnes). Ces données sont-elles dépendantes ?

- Oui, les attributs sont dépendants.

# Type de données

Supposons que pour chaque élève (*observation*, enregistrement ou lignes), j'enregistre sa taille et son poids (*features*, attribut ou colonnes). Ces données sont-elles dépendantes ?

- Oui, les attributs sont dépendants.
- Non, deux individus sont indépendants.

# Type de données

Supposons que pour chaque élève (*observation*, enregistrement ou lignes), j'enregistre sa taille et son poids (*features*, attribut ou colonnes). Ces données sont-elles dépendantes ?

- Oui, les attributs sont dépendants.
- Non, deux individus sont indépendants.

# Données indépendantes

## Données multidimensionnelles

Un jeu de données  $D$  est un ensemble de  $n$  enregistrements  $X_1, \dots, X_n$  où chaque enregistrement  $X_i$  contient un ensemble  $d$  d'attributs nommé  $(X_i^1, \dots, X_i^d)$

chien	personne	plante
- race	- grandeur	- nom latin
- couleur	- poids	- médicaments utilisés

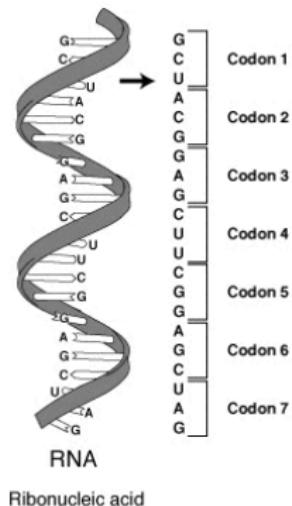
La nature des attributs classifie les données comme **numériques, catégoriques, binaires, textuelles, etc.**

# Données dépendants

- La connaissance des dépendances préexistantes modifie considérablement la fouille de données.
- En général, les données dépendantes sont plus complexes en raison des relations préexistantes entre les éléments.

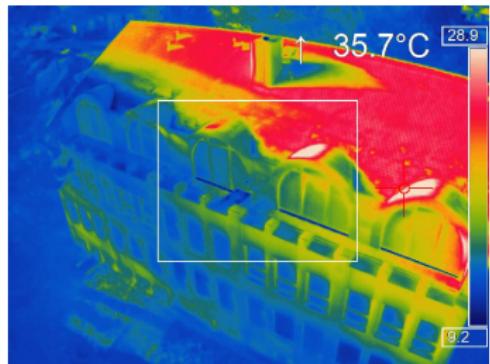
# Séries temporelles

- Les séries temporelles contiennent des valeurs généralement générées par des mesures continues dans le temps.
- Ces données ont généralement des dépendances **implicites** intégrées aux valeurs observées au fil du temps.
- Certaines séries peuvent montrer des patrons périodiques d'attribut mesuré au fil du temps.
- Les **séquences discrètes** sont la version catégorique des séries temporelles.



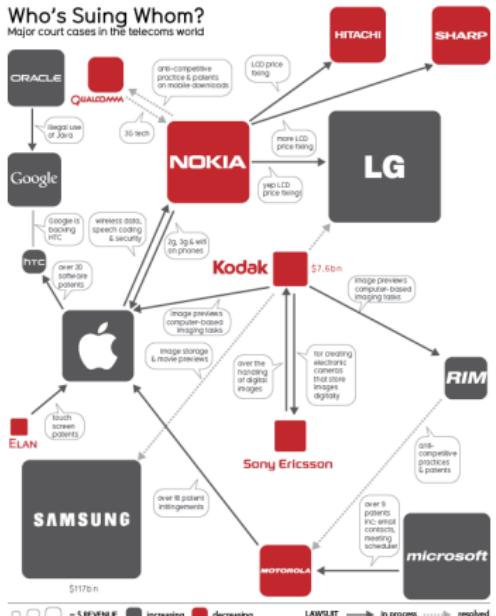
# Données spatiales

- Des attributs non spatiaux (ex. température, pression, intensité de couleur de pixel d'image) peuvent être mesurés dans l'espace.
- Une forme particulière de données spatiales sont les données spatio-temporelles, qui contiennent à la fois des attributs spatiaux et temporels.



# Graphes et réseaux

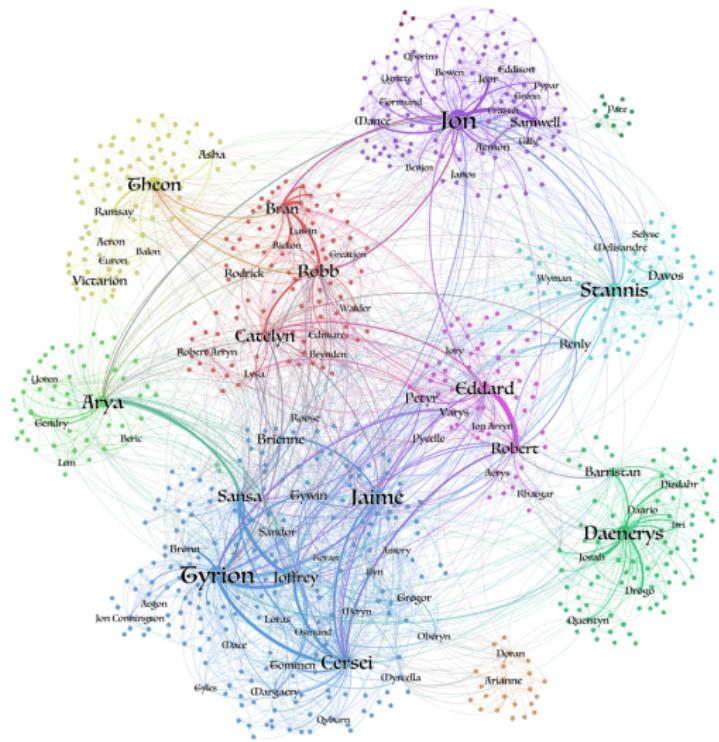
Relations explicites entre les données



David McCandless & James Key // v1.2 // Oct 10  
InformationIsBeautiful.net

Idea.Guardian.Tech, Ny Times /data.billyesquire.com  
source: Bloomberg, BBC, DigitalTrends.com  
only highlights or summaries of lawsuits

# Graphes et réseaux



# Portabilité

- La **portabilité** du type de données est un élément crucial du processus de fouille de données, car les données sont souvent hétérogènes et peuvent contenir plusieurs types d'attributs.
- C'est toujours plus compliqué et long (quoiqu'idéal) de développer des algorithmes qui puissent marcher pour des jeux de données hétérogènes.
- Les outils *off the shelf* sont souvent pour un type spécifique de données.

# Numérique → Catégorique

- **Discrétisation** : divise l'intervalle de valeurs d'un attribut numérique en  $\phi$  intervalles.
- Alors, l'attribut est supposé d'avoir  $1, \dots, \phi$  valeur catégorique.
- Ex. attribut numérique **âge**  
Création de 4 intervalles : [0, 20]; [20, 40]; [40, 60]; [60, 100]  
La valeur **symbolique** d'un enregistrement dans l'intervalle [20, 40] est 2, tandis qu'un autre dans l'intervalle [60, 100] est 4

# Numérique → Catégorique

- Les variations dans un intervalle sont perdues après la discréétisation.
- En plus, les données peuvent être non uniformément distribuées à travers les intervalles.
- Il faut faire attention aux tailles des intervalles en fonction de la distribution des données par rapport aux attributs discréétisés.

# Catégorique → Numérique

- **Binarisation** : prends les  $\phi$  valeurs possibles d'un attribut catégorique et crés  $\phi$  nouveaux attributs binaires (numériques).
- Ainsi, exactement l'un des  $\phi$  attributs prend la valeur de 1, et le reste prend la valeur 0.

# Catégorique → Numérique

- **Binarisation** : prends les  $\phi$  valeurs possibles d'un attribut catégorique et crés  $\phi$  nouveaux attributs binaires (numériques).
- Ainsi, exactement l'un des  $\phi$  attributs prend la valeur de 1, et le reste prend la valeur 0.
- Pourquoi pas un *mapping* direct entre l'attribut catégorique et ses  $\phi$  valeurs à un seul attribut numérique entier ?

# Texte → numérique

- *Bag-of-words* : les attributs correspondent aux mots, et les valeurs correspondent aux fréquences de ces attributs.  
exemple des 1 et 0 quand blond, brun roux
- Représentation très creuse ( $\approx 10^5$  mots différents dans la langue anglaise).  
par contre si on a 10 000 mots, pas possible de tous les représenter avec des 1 et des 0
- En plus, on perd les relations entre les mots.
- *Word embeddings* 
  - 10 valeurs aléatoires pour chaque mot
  - on donne en entrée les 10 valeurs d'un mot
  - on veut en sortie les mots à proximité

Roi - Homme + Femme = Reine

# Texte → numérique

- *Bag-of-words* : les attributs correspondent aux mots, et les valeurs correspondent aux fréquences de ces attributs.
- Représentation très creuse ( $\approx 10^5$  mots différents dans la langue anglaise).
- En plus, on perd les relations entre les mots.
- *Word embeddings* 
- Un domaine en vogue aujourd'hui est la transformation de n'importe quel type de données en numérique avec de l'apprentissage profond.

# Série temporelle → numérique

- **Transformée en ondelettes discrètes *wavelets*** : convertis les données de séries temporelles en données multidimensionnelles, sous la forme d'un ensemble de coefficients représentant les différences moyennes entre différentes parties de la série.
- Si souhaité, un sous-ensemble de plus grands coefficients peut être utilisé pour réduire la taille des données numériques.
- Les données résultantes sont moins dépendantes que la série temporelle originale.

# N'importe quel type → Graphe

- **Graphe de voisinage :**

- Chaque enregistrement du jeu de données est un noeud du graphe.
- Une arête existe entre deux éléments  $i$  et  $j$  si leur **distance**  $d(i,j)$  est inférieure à un seuil.
- Le poids  $w(i,j)$  de l'arête  $(i,j)$  est égal à une fonction kernel appliquée sur  $d(i,j)$ .
- Ex. *heat kernel* :

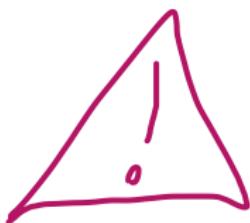
$$w(i,j) = e^{-d(i,j)^2/t^2}$$

où  $t$  est un hyperparamètre.

# Nettoyage de données

- De nombreuses questions se posent pour assurer l'analyse sensée des données provenant d'un projet réel :
  - Différence entre erreurs et artefacts
  - Compatibilité de données
  - Présence de valeurs manquantes
  - Détection de données aberrantes *outliers*

# Erreurs vs artefacts



- Les **erreurs** sont des informations perdues lors de leurs acquisitions. ex: on capte des températures et le capteur ne marche pas bien, donc données fausses, donc données perdues
- Les **artefacts** sont des problèmes découlant du traitement de données. peuvent être traités et viennent du coût de données réversibles  
sont réversibles

# Exemple

- Dans une étude bibliographique sur PubMed, on a identifié l'année de la première publication pour les 100 000 auteurs les plus prolifiques.

# Exemple

- Dans une étude bibliographique sur PubMed, on a identifié l'année de la première publication pour les 100 000 auteurs les plus prolifiques.
- Question : À quoi devrait ressembler la répartition des nouveaux auteurs les plus prolifiques selon l'année de leurs premières publications ?

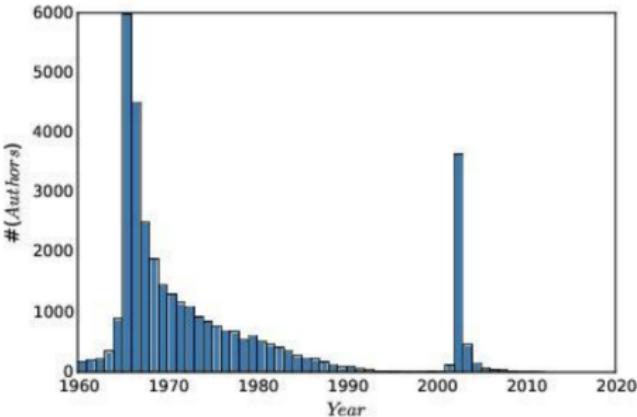
On suppose c'est des personnes + vieilles car font des publications, donc plus vieux, la courbe élevée à gauche et qui descent de plus en plus

# Exemple

- Dans une étude bibliographique sur PubMed, on a identifié l'année de la première publication pour les 100 000 auteurs les plus prolifiques.
- Question : À quoi devrait ressembler la répartition des nouveaux auteurs les plus prolifiques selon l'année de leurs premières publications ?
- Il est important d'avoir une idée préconçue de tout résultat pour aider à détecter les anomalies.

# Exemple

- Quels artefacts voyez-vous ?
- Quelles explications possibles pourraient leur causer ?



source : Skienaa, 2017

# Exemple

- D'abord, PubMed a commencé à collecter de papiers à partir de 1965    donc il n'y avait rien avant

# Exemple

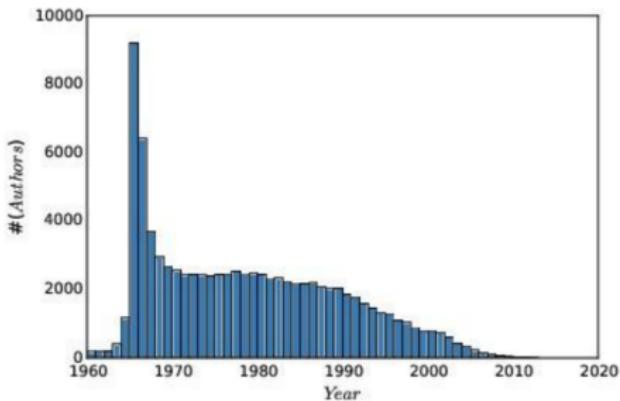
- D'abord, PubMed a commencé à collecter de papiers à partir de 1965
- En 2002, PubMed commence à utiliser les prénoms des auteurs   changer la façon dont les auteurs sont représentés

# Exemple

- D'abord, PubMed a commencé à collecter de papiers à partir de 1965
- En 2002, PubMed commence à utiliser les prénoms des auteurs
- *SSSkien* ⇒  
*StevenSSkienna* (un nouvel auteur est apparu !)  
nouvel auteur car ils l'ont stocké dans la bd différemment

# Exemple

- D'abord, PubMed a commencé à collecter de papiers à partir de 1965
- En 2002, PubMed commence à utiliser les prénoms des auteurs
- *SSSkrienna* ⇒ *StevenSSkienna* (un nouvel auteur est apparu !)
- Le nettoyage de données se débarrasse de ces artefacts



source : Skienna, 2017

courbe + élevée à gauche qu'à droite car plus les personnes sont vieilles plus elles ont le temps de publier

# Compatibilité de données

- La question de la compatibilité de données peut être critique  

- Les données ont besoin d'être compatibles pour qu'on puisse faire des comparaisons.

ex: Sur Mars, les unités pas les mêmes pour le robot, dans l'espace en Newton et sur le terrain en pounds

# Conversion d'unités

- Même la décision pour le système métrique a des incohérences potentielles : cm, m, km ?
- Attention aussi au moment d'intégrer des données de bases différentes
  - Ex. fréquences des hauteurs dans un ensemble de données combinées de mesures impériales (pieds) et métriques (mètres)
  - Si la distribution est bimodale, alors → problème de compatibilité.
- La vigilance dans l'intégration des données est donc essentielle.

# Représentation numérique

- Les attributs discrets doivent toujours être des entiers.
- Les attributs mesurés physiquement ne sont jamais quantifiés avec précision, alors ils doivent être représentés par des chiffres réels.
- Les quantités fractionnaires doivent être décimales, évitez de les représenter par  $(q, r)$  comme dans (livres, oz) ou (pieds, pouces).

# Représentation de caractères

- Un problème de nettoyage particulièrement désagréable dans les données textuelles est l'unification des représentations de code de caractères :
  - ISO 8859-1 est un code à one byte pour l'ASCII (originalement à 7 bits) qui inclut quelques caractères et ponctuations pour les langues latines
  - UTF-8 est un codage multi octets pour tous les caractères Unicode, compatible avec ASCII
- **Bonne pratique** : générez toujours de texte en UTF-8

# Unification de noms

- Utilisez des transformations simples pour unifier les noms, ex. minuscules.
- Considérez les méthodes phonétiques de *hashing* comme Soundex et Metaphone :
  - Ex. élimination des lettres doublées
- Compromis entre faux positifs et négatifs.  
deux personnes avec mm nom  
faux positifs: deux personnes pour la même personne comme Daniel  
ex: Daniel Aloise peut être compris pour Daniel Aloase

# Unification de dates dans le temps

- Aligner les événements temporels de différents ensembles de données / systèmes peut être problématique.
  - Utilisez le temps universel coordonné (UTC)
  - ou le temps UNIX

# Unification financière

- Pourquoi le prix de l'action de McDonald's et celui du pétrole sont-ils corrélés sur 30 ans ?



# Unification financière

- Pourquoi le prix de l'action de McDonald's et celui du pétrole sont-ils corrélés sur 30 ans ?



- Il est nécessaire de tenir compte de la valeur de l'argent au cours du temps (inflation) pour faire des comparaisons justes à long terme
- Utilisez les rendement / changement de pourcentage au lieu des changements de prix absolus pour comparer plusieurs variables dans le temps

# Normalisation

- Dans de nombreux scénarios, les différents attributs ont différentes échelles de référence et ne peuvent donc pas être comparés entre eux.
  - ex. âge et salaire
- Toute fonction agrégée calculée sur de différents enregistrements (par exemple, les distances euclidiennes) sera dominée par l'attribut de plus grande magnitude.

# Normalisation

- Considérons  $\mu_j$  la moyenne de l'attribut  $j$  et  $\sigma_j$  son écart type  
**standardisation ou z-scores**

$$z_i^j = \frac{X_i^j - \mu_j}{\sigma_j}$$

Prendre les valeurs,  
soustraire la moyenne et  
diviser l'écart type

suppose une distribution normale de moyenne 0 et  
d'écart type 1

## *min-max scaling*

$$y_i^j = \frac{X_i^j - \min^j}{\max^j - \min^j}$$

$\in [0, 1]$ , sensible aux données aberrantes

toutes les valeurs entre 0 et 1, mais le maximum n'est pas garanti  
ex: age, c'est pas garanti jusqu'à quand on va vivre

# Normalisation

- Min-max scaling implique des écarts plus petits.
- Alors, lequel de deux utiliser ?

# Normalisation

- Min-max scaling implique des écarts plus petits.
- Alors, lequel de deux utiliser ?
- Il dépend de l'application
  - clustering, PCA → z-scores
  - traitement d'images, données  $\in [0, 1]$  → min-max scaling  
on connaît le min,max

# Valeurs manquantes

- Un aspect important du nettoyage des données est de représenter correctement les **données manquantes** :
  - Quelle est l'année du décès d'une personne vivante ?
  - Qu'en est-il d'un champ laissé vide ou rempli d'une valeur étrange (ex. NULL) ?
- Mettre ces valeurs à zéro est généralement une mauvaise idée !

ex: quand un individu ne mentionne s'il a des lunettes ou pas

Personne 1 -> lunettes Oui

Personne 2 -> lunettes Non

Personne 3 -> -

Donc mettre 0 n'est pas une bonne idée

Donc on va essayer d'estimer la valeur (**imputation de valeurs manquantes**)

Daniel Aloise <[daniel.aloise@polymtl.ca](mailto:daniel.aloise@polymtl.ca)> — Préparation de données — 7 septembre 2021

44/48

# Imputation de valeurs manquantes

- Avec suffisamment de données d'entraînement, on peut supprimer toutes les données avec des valeurs manquantes.
- Pourtant, cela n'est pas toujours possible.
- Dans ce cas, il est souvent préférable d'estimer ou d'imputer des valeurs manquantes au lieu de les laisser vides.
  - ex. une bonne estimé pour votre année de décès est l'année de naissance + 80.

# Imputation de valeurs manquantes

**valeur moyenne** : conserve la moyenne.

**valeur aléatoire** : La sélection de valeurs aléatoires permet une évaluation quantitative de l'impact de l'imputation et de la qualité du modèle.

**régression** : l'utilisation de la régression linéaire pour prévoir les valeurs manquantes fonctionne bien si peu de champs sont manquants par enregistrement .

Si on peut mettre une valeur aléatoire et que ça n'a pas d'impact, c'est bon  
Si la valeur a un impact sur notre analyse, on doit faire une regression

# Détection de données aberrantes

- La plus grande vertèbre de dinosaure rapportée est 50% plus grande que toutes les autres : probablement une erreur.
- Regardez de manière critique les valeurs maximum et minimum pour toutes les variables.
- Les données normalement distribuées ne devraient pas avoir de grandes valeurs aberrantes.
- Essayer de comprendre pourquoi vous avez une valeur aberrante.
- **Évitez la solution facile de juste la supprimer**



# Détection de données aberrantes

- Visuellement, il est facile de détecter les valeurs aberrantes, mais seulement dans les espaces de faible dimension.
- Généralement considéré comme un problème d'**apprentissage non supervisé**.