# Data Science, Séance 1 : introduction

Etienne Côme

21 novembre 2019

# Data Science ?

*The next sexy job*

*The ability to take data to be able to understand it, to process it, to extract value from it, to visualize it, to communicate it, that's going to be a hugely important skill.*

– **Hal Varian, Google**

# Data Science ?

*Data science, as it's practiced, is a blend of Red-Bull-fueled* ***hacking*** *and espresso-inspired* ***statistics****.*

*Data science is the civil engineering of data. Its acolytes possess a practical knowledge of tools & materials, coupled with a theoretical understanding of what's possible*

**– Mike Driscoll, CEO of metamarkets**

# Drew Conway's Data Science Venn Diagram

# Data Science ?

*A data scientist is someone who can obtain, scrub, explore, model and interpret data, blending hacking, statistics and machine learning. Data scientists not only are adept at working with data, but appreciate data itself as a first-class product.*

**– Hilary Mason, chief scientist at bit.ly**

# Data Science ?

*Parler de la donnée, c'est aussi évoquer le datascientist, ce mouton à 5 pattes de la data disposant de compétences statistiques, informatiques, comprenant parfaitement les enjeux métier de l'entreprise... Est-il aussi un fantasme du discours ambiant sur le big data ?*

*S'il peut exister des profils qui s'approchent de cette description, la réalité démontre le plus souvent que la datascience, comme la science en générale, ne se produit pas seule mais en groupe. La bonne question à se poser est donc plutôt celle de la synchronisation des différentes compétences au sein de l'organisation. Nous plaidons davantage pour un datascientism que pour des datascientists.*

*Une autre réalité méconnue sur le datascientist est qu'il s'agit avant tout d'un métier d'artisan. Chaque problème et chaque jeu de données demande toujours une démarche spécifique qui n'est pas industrialisable, ce que beaucoup de gens ne comprennent pas encore.*

# Une mode avec des origines anciennes

Johann Kepler

# Une mode avec des origines anciennes

Charles Joseph Minard

# Une mode avec des origines anciennes

Charles Joseph Minard

# Une mode avec des origines anciennes

William Sealy Gosset (Student)

# Des compétences clés

1. Préparer les données (DB)

Récupérer, mélanger, enrichir, filtrer, nettoyer, vérifier, formater, transformer des données...

2. Mettre en œuvre une méthode un modèle (ML/Stats)

Arbre de décision, régression, clustering, Modèle graphique, SVM...

3. Interpréter les résultats (Vis)

Graphiques, Data visualisation, Cartes...

# Des compétences clés

1. Préparer les données (DB) – 80% du boulot

Récupérer, mélanger, enrichir, filtrer, nettoyer, vérifier, formater, transformer des données...

2. Mettre en œuvre une méthode un model (ML/Stats)

Arbre de décision, régression, clustering, Modèle graphique, SVM...

3. Interpréter les résultats (Vis) – 80% du boulot

Graphiques, Data visualisation, Cartes...

# Des compétences clés

## 1. Data Munging

Récupérer, mélanger, enrichir, filtrer, nettoyer, vérifier, formater, transformer des données

## 2. Statistiques

Analyse de données traditionnelle

## 3. Visualisation

Graphiques, Data visualisation, Cartes...

# Plan du cours

## Data-munging

les fichiers textes csv, json, xml, . . . et la ligne de commande
base de donnée et algèbre relationnel
trouver des données, et les manipuler en R
manipuler des données en R avec dplyr
api, web et scraping, . . .
données spatiale

## Visualisation

introduction à la visualisation, bonnes pratiques & erreurs communes
ggplot et la grammaire graphique
introduction à la cartographie avec le package cartography

Quelques exemples de projets

http://www.comeetie.fr/map_lbc.php

Quelques exemples de projets

http://www.comeetie.fr/galerie/francepixels/

Quelques exemples de projets

http://www.comeetie.fr/galerie/francepixels/

Quelques exemples de projets

http://vlsstats.ifsttar.fr/

Quelques exemples de projets

http://vlsstats.ifsttar.fr/atNight/

Quelques exemples de projets

http://www.comeetie.fr/galerie/velib/

# Organisation du cours

Cours orienté pratique et mise en œuvre

Outils principaux : linux, R et Chrome

Exercices de mise en oeuvre pratique

+ projet

+ contrôle continu

# Projet, foot data

- Données : https://figshare.com/collections/Soccer_match_event_dataset/4415000/2

# Quelques pointeurs

Dépot web du cours : https://github.com/comeetie/dsp5.git

Mon adresse mail : etienne.come@ifsttar.fr

Mon compte twitter : @comeetie

Outils en ligne : Google,StackOverflow, gitHub, github.io

Cours intéressants : Stat221 (Harvard), CS294-10 (Berkley)

# Reprise en douceur avec des fondamentaux

fichiers textes et ligne de commande

# Fichiers textes

Formats très simples et pérenne pour stocker des données et les échanger Exemples :

CSV : Comma Separated Value

XML : Extensible Markup Language

JSON : JavaScript Object Notation

# Les fichiers type csv

Fichier texte simple pour stocker des données tabulaire. Les différentes variables sont séparées grâce à une

‘,’ ou autre ‘;’, ‘|’, ‘#’ et ’’

Possible de mettre une ligne pour le header

Exemple

# Compétence : Savoir importer un fichier malgré des problèmes d'encodage et/ou de formatage

# La ligne de commande

obtenir de l'aide

man

rediriger les sorties

$<$, $>$, $>>$, ...

enchainer des commandes

| et script bash

afficher un fichier

head, tail, cat et more

# La ligne de commande

analyser le fichier

grep : global regular expression Filtrer toute les lignes contenant ‘tot’

Filtrer toute les lignes contenant un chiffre de 0 à 4 suivi d’un nombre quelconque de caractères et d’un chiffre de 5 à 9

Filtrer toute les lignes commençant par un tirer

options -i, -n et -c,. . .

# La ligne de commande

éditer le fichier

nano et gedit

modifier, analyser le fichier

sed : stream editor (lecture ligne/ligne) remplacer toutes les occurrences de “ficheir” par “fichier” :

supprimer toutes les lignes vides :

supprimer les lignes 7 à 9 :

# La ligne de commande

modifier, analyser le fichier

perl : practical extraction and reporting language substitution multiples et mise en mémoire des pattern matchés:

# La ligne de commande

problème d'encodage

file : informations sur un fichier

iconv : changement d'encodage iso-8859 $\rightarrow$ utf8

# Import en R de fichiers type csv

problème de formatage et import dans R

- au séparateur de champs
- à l'en-tête
- au conversion de chaîne de caractère en facteur
- au séparateurs décimaux : , ou .
- au séparateurs de chaîne de caractère " ou ' ?

# Exercice (20 mn) :

Importer proprement dans R le fichier ./data/exo1.csv qui contient des problèmes d'encodage et de formatage avec read.table

Vérifier que les variables numériques sont bien numériques, que les chaines de caractères sont bien des chaines de caractères ....

Même chose avec read_csv de la library readr

# Compétence : Savoir lire et remettre en forme un fichier JSON en R

# Package Rjson

Lecture ecriture de JSON

lire un fichier JSON :

exporter un objet R en JSON :

manipuler :

# Exercice (20mn):

Créer la data.frame suivante :

qui contient les id des stations velib et la moyenne du nombre de bornes disponibles sur la période enregistrée. Pour cela vous utiliserez le fichier ./data/exo2.json qui à la forme suivante : tableau de stations ayant chacune une id (id) et trois tableaux associés; nombre de vélos (available_bikes), nombre de bornes (available_bike_stands), date de la mesures (download_date)

# Les fichiers type XML

Fichier texte contenant des balises imbriquées

Chaque balise peut être décrite par différents attributs

Les balises et attributs devraient être décrit par une dtd et/ou des namespaces

Parser un fichiers

2 méthodes :

SAX: api pour la lecture en ligne d'un document récupération d'évènements correspondant à la lecture d'une balise particulière

DOM: méthode de construction de l'arbre DOM du document

# Exemple

# Package XML

permet de parser un fichier et de construire un arbre DOM

fournis des fonctions pour parcourir et extraire les données de l'arbre créer

Exemple d'utilisation

# Exercice (20mn)

Construire à partir du fichier ./data/exo3.xml une data.frame contenant les variables suivantes :

'id','lat','long','nbBikes','nbEmptyDocks'

# Correction

```
library(XML)
data     = xmlTreeParse("./data/exo3.xml") # parser le fic
stations = xmlChildren(xmlRoot(data)) # liste des stations
vars = c('id','lat','long','nbBikes','nbEmptyDocks')
resMatrix=sapply(stations,function(x){
  # extraction des variables
  clist = lapply(xmlChildren(x),xmlValue)
  # sélection des variables
  sel   = names(clist) %in% vars
  # et conversion des variables
  as.numeric(unlist(clist[sel]))
  })
# mise sous forme de data.frame
res=data.frame(t(resMatrix),row.names = NULL)
names(res)=c('id','lat','long','nbBikes','nbEmptyDocks')
```