

Introduction aux Bases de Données

Master IM, Paris 5

@comeetie

Recueil de données en ligne, techniques du web

Master IM, Paris 5

@comeetie

Data Science Hacking Basics

avec Linux, R, et Chrome

Master IM, Paris 5

@comeetie

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

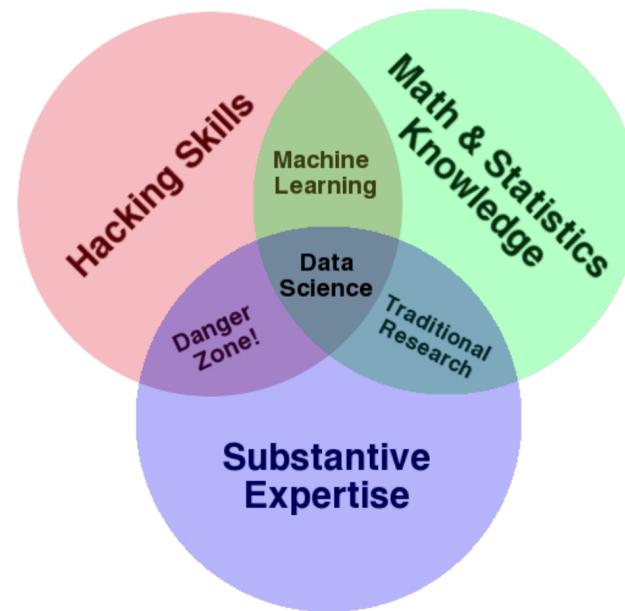
Data Science ?

“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

Un domaine en construction

A screenshot of a Twitter post from Arthur Charpentier (@freakonometrics). The post includes a profile picture of Arthur, his name and handle, a "Suivre" (Follow) button, and the tweet text. It also shows the timestamp, retweet count, and favorite count.

Arthur Charpentier
@freakonometrics

Suivre

"Data science: how is it different to statistics ?"
bulletin.imstat.org/2014/09/data-s... / by @hadleywickham

22:41 - 5 Sept 2014

16 RETWEETS 18 FAVORIS

Un domaine en construction



Ralph Winters
@RDub2

Suivre

@freakonometrics @hadleywickham Statisticians not cleaning
reshaping data? What have I been doing the last 10 years?

23:39 - 5 Sept 2014

3 FAVORIS



Hadley Wickham
@hadleywickham

Suivre

@RDub2 they are, but they're not teaching/research it (by and
large)

23:59 - 5 Sept 2014

2 FAVORIS

Un domaine en construction



Ralph Winters
@RDub2

Suivre

@hadleywickham I think the collection analysis, insight generation, cleaning, communication is multidisciplinary not just #ds

01:41 - 6 Sept 2014



Dr. Diego Kuonen
@DiegoKuonen

Suivre

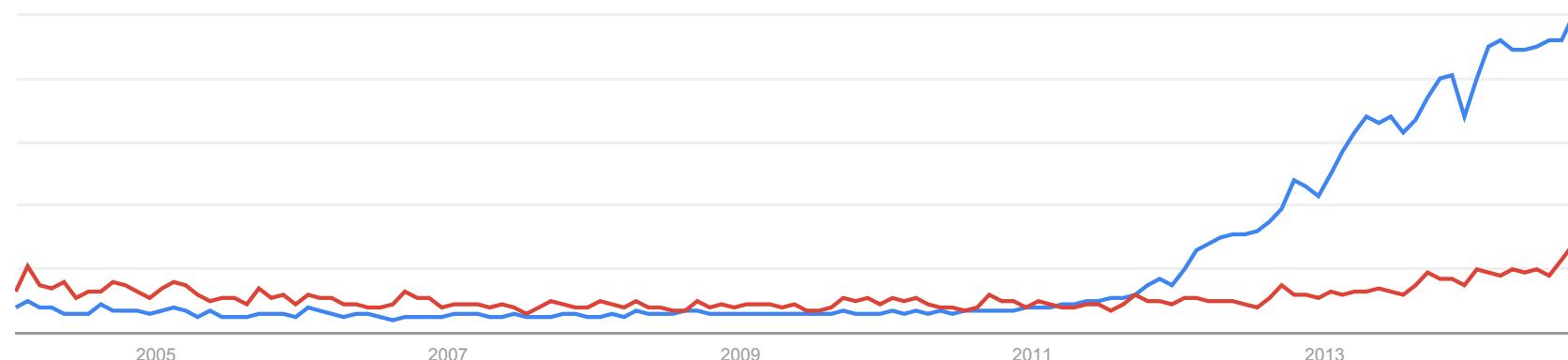
.@freakonometrics @hadleywickham, please find my view on #BigData, #DataScience + #Statistics at goo.gl/nMbecn
@Rdub2

06:23 - 6 Sept 2014

Une mode anglo-saxone qui s'exporte

Évolution de l'intérêt pour cette recherche. Recherche sur le Web. Dans tous les pays, De 2004 à ce jour.

■ big data ■ data science

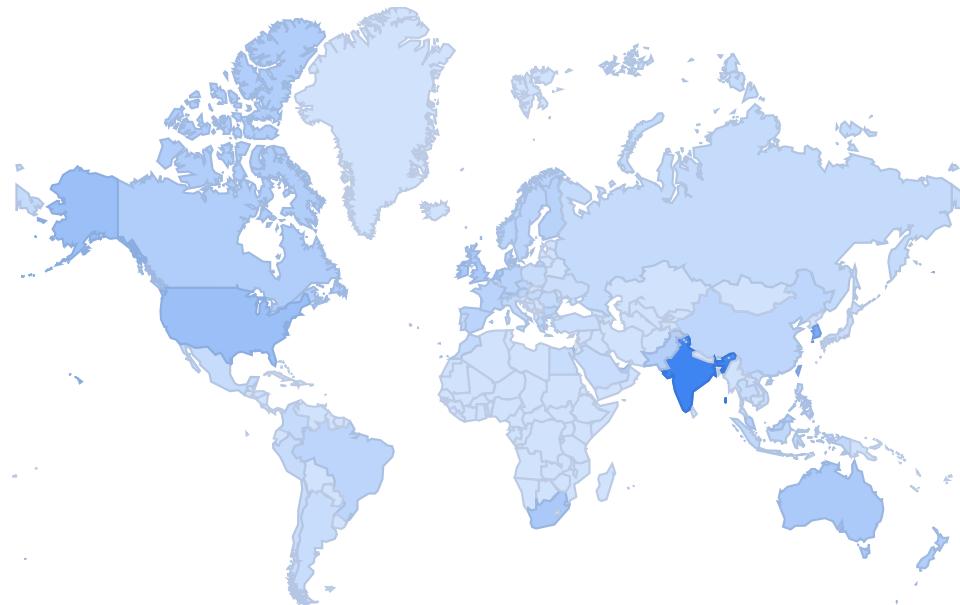


Google™

Afficher le rapport complet dans [Tendances des recherches](#)

Une mode anglo-saxone qui s'exporte

Rechercher le volume pour big data. Recherche sur le Web. Dans tous les pays, De 2004 à ce jour.

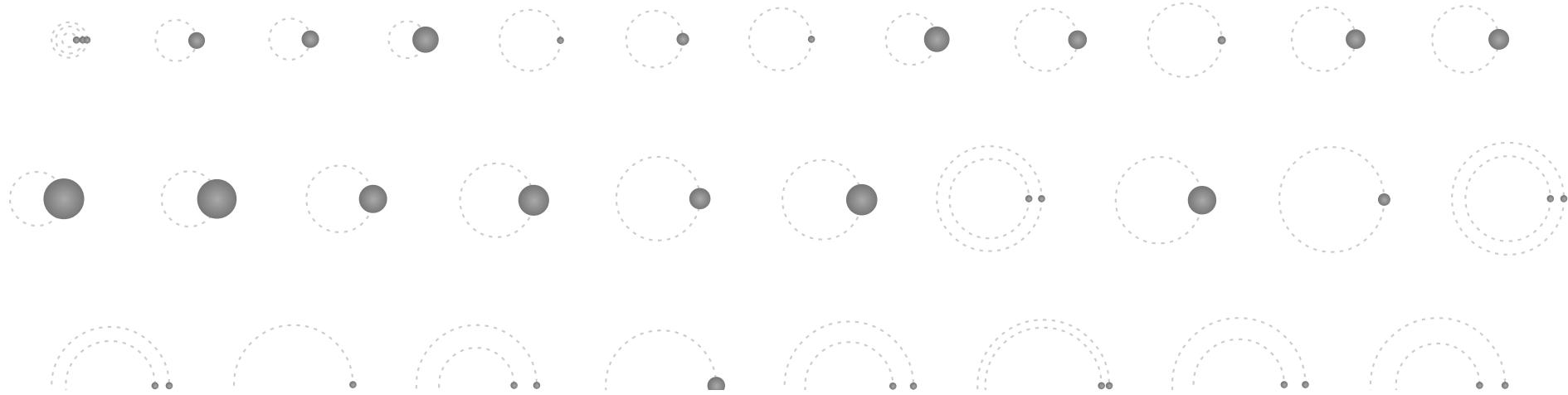


avec des origines anciennes



Johann Kepler

avec des origines anciennes



Johann Kepler, mbostock blo.cks // NYT

avec des origines anciennes



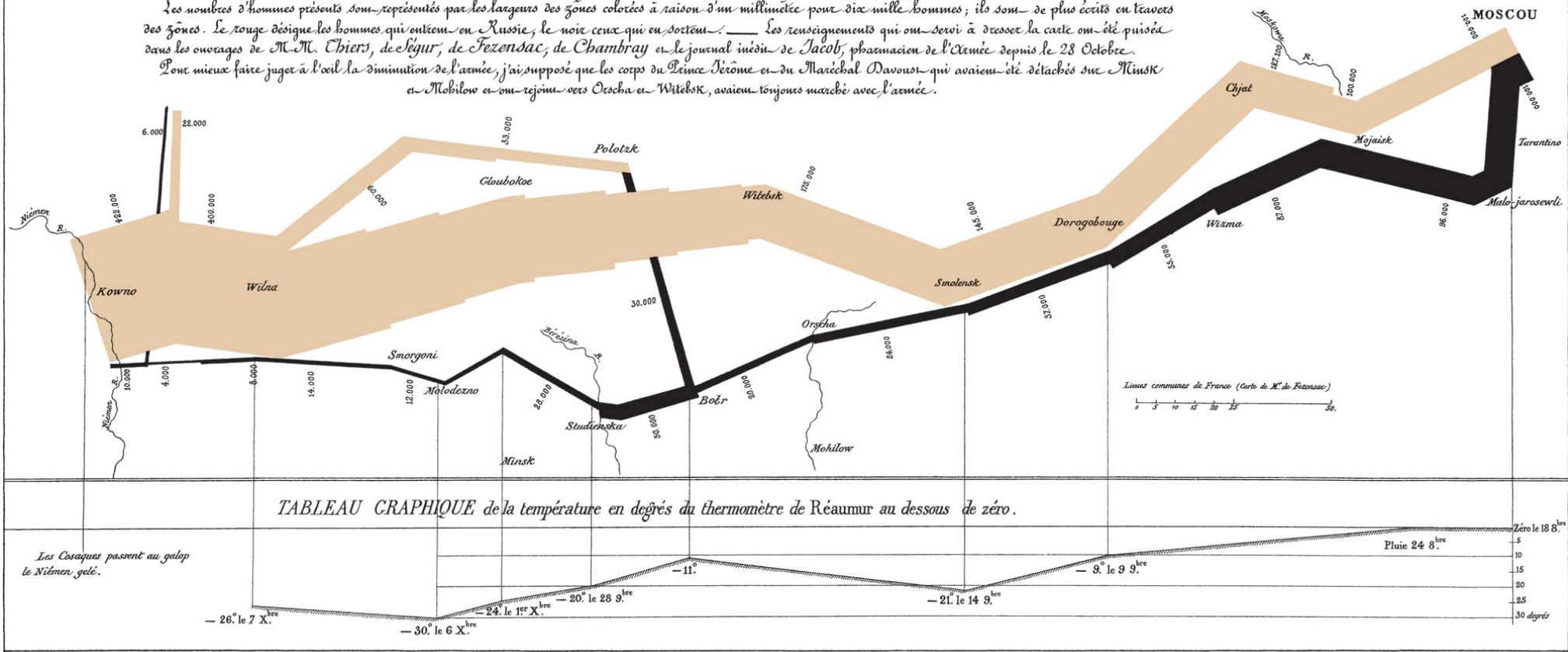
Charles-Joseph Minard

Carte Figurative des pertes successives en hommes de l'Armée Française dans la campagne de Russie 1812-1813.

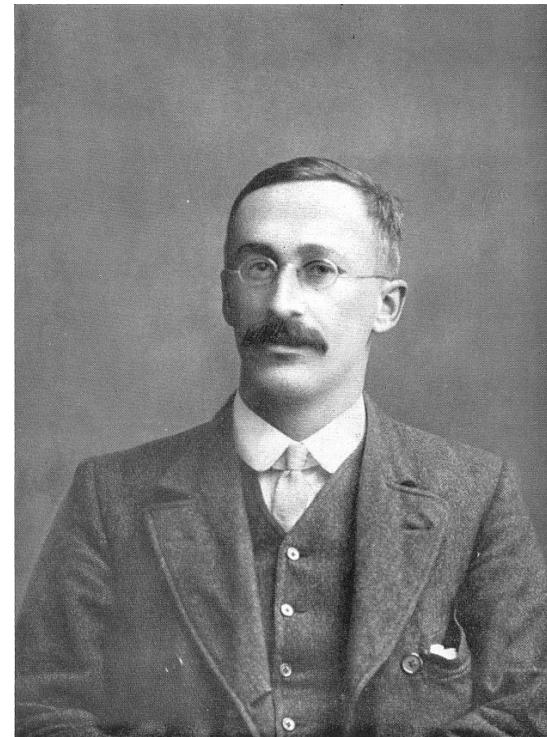
Dressée par M. Minard, Inspecteur Général des Ponts et Chaussées en retraite à Paris, le 20 Novembre 1869.

Les nombres d'hommes présents sont représentés par les largures des zones colorées à raison d'un millimètre pour dix mille hommes; ils sont de plus écrits en travers des zones. Le rouge désigne les hommes qui entrent en Russie, le noir ceux qui en sortent. Les renseignements qui ont servi à dresser la carte ont été puisés dans les ouvrages de M. M. Chiers, de Cléger, de Fezensac, de Chambray et le journal intime de Jacob, pharmacien de l'Armée depuis le 28 Octobre.

Pour mieux faire juger à l'œil la diminution de l'armée, j'ai supposé que les corps du Prince Jérôme et du Maréchal Davout, qui avaient été détachés sur Minsk et Mohilow et se rejoignirent vers Orscha en Witebsk, avaient toujours marché avec l'armée.

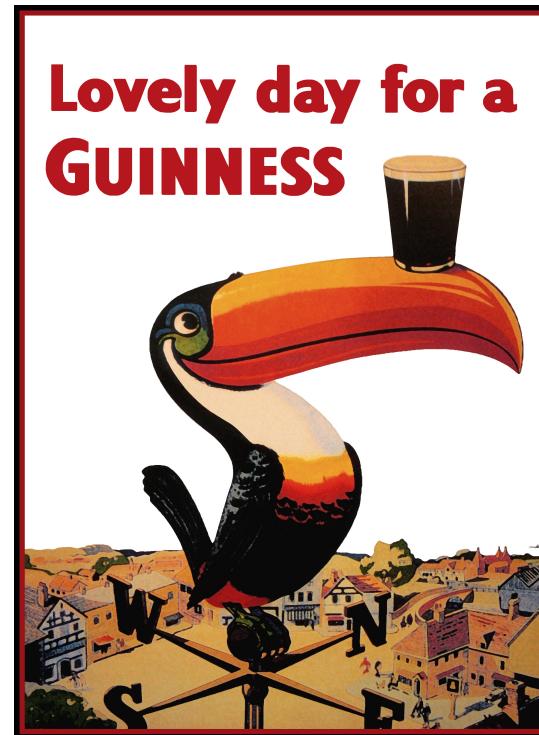


avec des origines anciennes



William Sealy Gosset (Student)

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

Data Munging

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

Statistiques

Analyse de données traditionnelle

Visualisation

Graphiques, Data visualisation, Cartes...

Plan du cours

Data-munging (6 séances)

Visualisation (6 séances)

Data-munging (6 séances)

1. les fichiers textes csv, json, xml, ... et la ligne de commande
2. base de donnée et algèbre relationnel -- **choix des binômes projet**
3. enrichir le jeu de données, trouver des données, et les manipuler en R
4. manipuler des données en R et dplyr -- **cc 1h**
5. api, web et scraping, ...
6. données spatiale -- **rendu des sujets de projets**

Visualisation (6 séances)

1. introduction à la visualisation, bonnes pratiques & erreurs communes
2. ggplot level 1 et la grammaire graphique
3. ggplot level 2
4. le web html, css, js, svg et d3
5. d3 level 1 -- cc 1h
6. d3 level 2 -- rendu final des projets

Quelques exemples de projets

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

Quelques exemples de projets

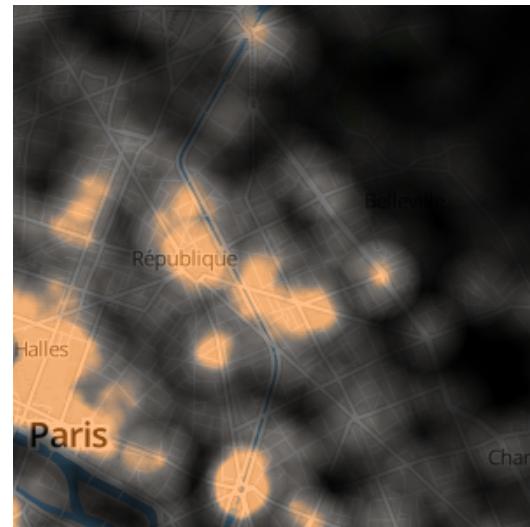
<http://www.comeetie.fr/galerie/francepixels/>

Quelques exemples de projets



<http://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

Séance de 3 heures autour d'un cours-TP

Outils principaux : linux, R et Chrome

Présentation de notions et d'outils

Exercices de mise en œuvre pratique

+ projet

+ contrôle continu

Les projets

Le projet doit vous permettre de mobiliser les différentes compétences et connaissances acquises pendant le cours et de mettre en avant votre créativité.

Pour ce faire vous vous diviserez en groupes de 2 ou 3, identifierez **1 ou 2 jeux de données** que vous voulez étudier. Vous **les mettrez en forme et les visualiserez**. Lorsque vos données seront suffisamment propres vous pourrez utiliser une méthode de stats/ML : test, régression, classification, clustering et présenterez les résultats obtenus.

Tout au long du projet vous tiendrez à jour un journal de bord sur **HackPad** de vos problèmes et réussites éventuelles.

La soumission consistera en ce journal de bord, un dépôt **gitHub** contenant vos sources et vos données ainsi qu'une présentation de 10mn

Les projets

Calendrier

- choix des binômes projet **23/09/2014**
- rendu des sujets de projets **21/10/2014**
- rendu final des projets **9/12/2014**

Le contrôle continu

Interro de type cours, exos sur feuille d'une durée de 1h

Calendrier

- Data-munging **14/10/2014**
- Visualisation **2/12/2014**

Convention des slides

- Les slides d'exercices sont en vert
- Les slides de corrections sont en rouge
- Les slides compétences sont en bleu
- Les slides dédiés aux outils sont en violet
- Les liens sont en orange

Quelques pointeurs

- Site web du cours : <http://www.comeetie.fr/datasciencenhacking/>
- Mon adresse mail : etienne.come@ifsttar.fr
- Mon compte twitter : [@comeetie](https://twitter.com/comeetie)
- Outils en ligne : [Google](#), [StackOverflow](#), [HackPad](#), [gitHub](#), [bl.ocks](#), [github.io](#)
- Cours intéressants : [Stat221 \(Harvard\)](#), [CS294-10 \(Berkley\)](#), [CS194-16 \(Berkley\)](#)

Questions ?

Séance 1

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 '\t'

Possible de mettre une ligne pour le header

Exemple

```
"id" "mbikestands"  
16104 30.5303867403315  
15063 5.19337016574586  
17010 16.2651933701657  
13045 8.48066298342541  
10025 9.31491712707182  
...
```

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'

```
grep tot fichier.txt
```

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

```
grep '[01234].*[56789]' fichier.txt
```

Filtrer toute les lignes commençant par un tiret

```
grep '^-' fichier.txt
```

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" :

```
sed -e 's/ficheir/fichier/g' fichier.txt > fichier.corrected.txt
```

supprimer toutes les lignes vides :

```
sed -e '/^ *$/d' fichier.txt
```

supprimer les lignes 7 à 9 :

```
sed '7,9d' fichier.txt
```

La ligne de commande

modifier, analyser le fichier

perl : practical extraction and reporting language

substitution multiples et mise en mémoire des pattern matchés:

```
perl -pe 's/last_update":([0-9]*)/last_update":0000$1}/g'
```

La ligne de commande

problème d'encodage

file : informations sur un fichier

```
file -i fichier.txt -o fichier.utf8
```

iconv : changement d'encodage

iso-8859 → utf8

```
iconv -f ISO-8859-1 -t UTF-8 fichier.txt -o fichier.utf8
```

Import en R de fichiers type csv

problème de formatage et import dans R

```
data = read.table("data.csv")
```

- ! 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 *ex01.csv* qui contient des problèmes d'encodage et de formatage.

Les fichiers JSON

Fichier texte simple pour stocker des données semi-structurées

2 éléments imbriqués à l'envie :

- objet : '{attribut1:valeur, attribut2:valeur,...}'
- tableau : '[objet1,objet2,...]'

Exemple

```
[{"_id" : "1/1/2014",
 "value" : {"bikes" : 53972,"max_bikes" : 189}},
 {"_id" : "1/10/2013",
 "value" : {"bikes" : 48375,"max_bikes" : 175}}]
```

Compétence :

Savoir lire et remettre en forme un fichier JSON en R

Package Rjson

Lecture écriture de JSON

lire un fichier **JSON** :

```
data=fromJSON(file="fichier.json")
```

exporter un objet R en **JSON** :

```
toJSON(data)
```

manipuler :

```
apply(data,function(x){})  
unlist(list)
```

Exercice (20mn):

Créer la data.frame suivante :

```
"id" "mbikestands"  
"1" 16104 30.5303867403315  
"2" 15063 5.19337016574586  
"3" 17010 16.2651933701657  
"4" 13045 8.48066298342541  
...
```

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 **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

```
<?xml version='1.0' encoding='UTF-8'?>
<stations lastUpdate="1409819465886" version="2.0">
<station>
    <id>2</id>
    <name>Dézery/Ste-Catherine</name>
    <terminalName>6002</terminalName>
...
</station>
<station>
    <id>3</id>
    <name>St-Maurice/ St-Henri</name>
    <terminalName>6003</terminalName>
...
</station>
</stations>
```

Package XML

- permet de parser un fichier et de construire un arbre DOM
- fournit des fonctions pour parcourir et extraire les données de l'arbre créé

Exemple d'utilisation

```
data    = xmlTreeParse("exo3.xml") # parser le fichier
xmltop = xmlRoot(data) # récupérer la racine
child  = xmlChildren(xmltop) # les fils
child2 = xmlValue(child) # les valeurs
val2   = xmlValue(child2)
res1   = xmlSapply(xmltop,xmlValue) # appliquer une fonction
```

Exercice (20mn)

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

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

Correction

```
data      = xmlTreeParse("exo3.xml") # parser le fichier
stations = xmlChildren(xmlRoot(data)) # liste des stations

resMatrix=sapply(stations,function(x){
  # extraction des variables
  clist = lapply(xmlChildren(x),xmlValue)
  # sélection des variables
  sel   = names(clist) %in% c('id','lat','long','nbBikes','nbEmptyDocks')
  # et conversion des variables
  as.numeric(unlist(clist[sel]))
})

# mise sous forme de data.frame
res=data.frame(t(resMatrix),row.names=1)
names(res)=c('id','lat','long','nbBikes','nbEmptyDocks')
```

Bonus Stage 1

Partager des fichiers

HackPad

Prise de note, document teste collaboratif

gitHub

Plate-forme sociale de gestion de version et de partage de code

bl.ocks

Visualisation d'un code js hébergé sur github

aithub.io

Exercice:

se créer un compte HackPad et GitHub

Bonus Stage 2

Télécharger un fichier en ligne de commande

wget

Exécuter un script à intervalles réguliers

cron, crontab

