Résumé

Projet réalisé dans le cadre du cours INFO0801  
Modélisation connaissances   
& humanités numériques

Table des matières

[I. INTRODUCTION 2](#_Toc32917024)

[II. DÉVELOPEMENT TECHNIQUE 3](#_Toc32917025)

[1) OBJECTIFS DU PROJET 3](#_Toc32917026)

[2) STRATÉGIE 3](#_Toc32917027)

[3) STRUCTURATION DU CODE 4](#_Toc32917028)

[III. PRÉSENTATION DES DONNÉES 5](#_Toc32917029)

[IV. OUTILS UTILISÉS 7](#_Toc32917030)

[1) R STUDIO 7](#_Toc32917031)

[2) OPENWEATHER 7](#_Toc32917032)

[3) Les sites utilisÉs pour le web scrapping 8](#_Toc32917033)

[4) GitHub 8](#_Toc32917034)

[V. BILAN 9](#_Toc32917035)

# INTRODUCTION

Dans de nombreux domaines comme l’industrie, le secteur de la finance ou la vente au détail, l’extraction de données est le processus d’exploration de vastes ensembles de données visant à trouver des renseignements pertinents qui pourraient être utilisés dans un but précis. C’est aussi un processus de récupération de plusieurs types de données à partir de différentes sources, dont certaines peuvent être mal structurées. L'extraction des données permet de consolider, traiter et affiner les données, puis de les stocker dans un emplacement bien défini.

L'extraction des données est la première étape du processus ETL (Extract, Transform, Load) qui désigne une séquence d'opérations portant sur les données, permettant de consolider et regrouper différents types de données provenant de différentes sources sous un format commun. Ce processus comprend trois étapes :

L’Extraction ou Extract : Identifie les données pertinentes, puis les prépare pour traitement ou transformation ;

La Transformation ou Transform : Les données sont triées, structurées et nettoyées, les entrées en double sont supprimées de manière à conserver une seule instance, les valeurs manquantes sont supprimées ou enrichies et des vérifications sont effectuées de manière à obtenir des données cohérentes, exploitables et fiables.

Le Chargement ou Load : Une fois transformées, les données sont de haute qualité, et elles sont conservée dans un emplacement analyse.

Ce processus est de plus en plus utilisé dans la plupart des secteurs d'activité avec des objectifs variables. Une fois que les données ont été collectées et stockées, l’étape suivante consiste à leur donner du sens.

Le web représente une des plus importantes sources d’informations où il est possible d’acquérir des données dans les contenues des sites par l’utilisation du Web Scraping, permettant d’exploiter la structuration par balises du web pour en extraire du contenu, et par l’utilisation des APIs qui augmente la capacité et la vitesse d’accès à l’information.

Dans le cadre de notre première année de master en informatique parcours Intelligence Artificielle à la Faculté des Science de Reims, il nous est proposé de mettre en pratique nos connaissances et nos compétences professionnelles au travers d’un projet ayant pour objectif principal la construction d’un jeu de données utilisable au moyen des différentes méthodes et techniques énoncées ci-dessus.

# DÉVELOPEMENT TECHNIQUE

## OBJECTIFS DU PROJET

Ce projet consiste à utiliser des techniques d’extraction de données dont le web scraping et l’utilisation des APIs pour construire un jeu de données contenant un minimum de 100 individus décrits par 10 caractéristiques au minimum.

Dans le cadre de ce projet nous avons utilisé 198 pays du monde qui avaient les informations les plus significatives dont il y aura 17 variables à observer lesquelles seront des différents indicateurs comme par exemple l’espérance de vie, la mortalité, la natalité, les tourisme entre autres qui ont été extraites avec la technique du web scraping et des différents conditions météorologiques qu’on a récupéré en utilisant l’API par le service en ligne Openweathermap.

Ce jeu de données sera utilisé dans le cadre du prochain module INFO0808 Visualisation des données où nous allons l’utiliser comme ressource pour visualiser et analyser les résultats.

## STRATÉGIE

Dans la réalisation du projet nous avons choisis de travailler dans un seul fichier d’extension .R qui contiendra tout le code nécessaire pour l’extraction et la construction du jeu de données.

La première étape a consisté à écrire les différentes fonctions nécessaires. Nous avons pour commencer écris trois fonctions, la première permettant la collecte des différentes informations sur le site que nous avons choisis, la deuxième permettant de supprimer les caractères accentues et espace superflus et la troisième servant à remplacer un ensemble de nom enregistré dans un vecteur par un autre ensemble enregistré lui aussi dans un vecteur.

La deuxième étape se divise en plusieurs partie, d’abord nous avons créé une première data frame contenant un ensemble de 242 (Deux-cent-quarante-deux) observations et 2 (deux) variables qui sont les pays et leur continent respectif. La variable utilisée porte le nom de « information.pays ». Nous avons construit juste après la data frame contenant les noms des pays comme observations et 8 (huit) variables.

Nous avons par la suite utilisé ces deux data frame pour réaliser une jointure et obtenir un jeu de données contenant 253 (Deux-cent-cinquante-trois) et 10 (dix) variables qui sont : les pays, les continents, les espérances de vie par pays, la mortalité infantile, les indice de performance environnementale, la mortalité par pays et le nombre de touristes.

Ensuite, après avoir créé la première data frame, l’objectif était de lui ajouter toutes les capitales nécessaires à notre jeu de données. Nous avons donc créé une data frame contenant l’ensemble des pays et leur capitale collecté sur un autre site. Pour assurer une certaine concordance avec certains individus lors de la jointure, nous avons utilisé la fonction permettant de remplacer un vecteur par un autre.

Une fois notre nouvelle data frame correctement remplie, nous avons effectué une jointure. Cette jointure a créé une data frame contenant 256 (Deux-cent-cinquante-six) observations et 11 (onze) variables. Il est important de signaler que nous avons, apres parcours des données, supprimé certaines observations ne contenant pas d’information significatives.

Enfin, nous avons créé une fonction permettant de faire des requêtes à l’API OPENWEATHER et de récupérer les informations mise à notre disposition par l’API en format JSON. Avec cette fonction nous avons créé une data frame contenant 194 observations et 8 (huit) variables. Avec cette nouvelle data frame nous allons faire une dernière jointure avec la data frame précédente, renommer les colonnes et la transformer en Tibble afin d’obtenir notre jeu de données final.

## STRUCTURATION DU CODE

# PRÉSENTATION DES DONNÉES

Après avoir collecté et nettoyé différent type d’information, nous avons obtenu un jeu de données contenant 194 individus qui sont les différents pays et 17 variables. Parmi les différentes variables contenues dans notre jeu de données, 7 proviennent des informations récupérées avec l’API par utilisation de la capitale du pays et 10 proviennent des informations récupérées par web scraping. Les différentes informations contenues dans les variables récupérées par web scraping ont été récoltées entre 2016 et 2017.

Les 17 différentes variables sont les suivantes :

|  |  |
| --- | --- |
| NOM DE LA VARIABLE | DESCRIPTION |
| Capitals | Variable de type chaine de caractère, qui représente la capitale du pays. C’est le contenu de cette variable qui sera utilisé comme paramètre pour l’exécution de la fonction mise à disposition par l’API. |
| Continent | Variable de type facteur à cinq niveaux, représentant le contient auquel le pays appartient. |
| Esperance\_vie | Variable de type numérique, représentant l’espérance de vie dans le pays. |
| Mortalite\_inf | Variable de type numérique, représentant le taux de mortalité infantile dans le pays. |
| Indice\_perf\_env | Variable de type numérique, représentant l’indice de performance environnemental du pays. |
| Mortalite | Variable de type numérique, représentant le taux de mortalité global du pays. |
| tourisme | Variable de type numérique, représentant le nombre de touriste enregistré au cours de cette période. |
| pib-par-habitant | Variable de type numérique, représentant le PIB par habitant. |
| Natalite | Variable de type numérique, représentant le taux de natalité dans le pays. |
| Superficie | Variable de type numérique, représentant la superficie du pays. |
| Longitude | Variable de type numérique, représentant la longitude de la capitale du pays. |
| Temp\_actu | Variable de type numérique, représentant la température actuelle de la capitale du pays. |
| Temp\_min | Variable de type numérique, représentant la température minimale du jour de la capitale du pays. |
| Temp\_max | Variable de type numérique, représentant la température maximale du jour de la capitale du pays. |
| Humidity | Variable de type numérique, représentant l’humidité du jour de la capitale du pays. |
| Type\_temps | Variable de type facteur à quinze niveaux, représentant le type de temps. |

# OUTILS UTILISÉS

## R STUDIO

R est un langage de programmation et un logiciel libre destiné aux statistiques et à la science des données. Il permet de traiter, organiser et analyser des volumes importants de données de manière rapide et flexible, afin de pouvoir y appliquer des tests statistiques et se représenter ces données graphiquement à l’aide d’une grande variété de libraires disponibles telles que : Tidyverse (visualisation et manipulation des tableaux de données, importation et exportation de données, manipulation de variables, extraction de données du web …), Xml2, Rvest ou encore Jsonlite.



## OPENWEATHER

OPENWEATHER est un site qui propose un service permettant de fournir des informations sur la météo et les prévisions actuelles dans une ville. Ce site dispose d’une API permettant d’obtenir des données météorologiques en temps réel telles que : la température actuelle, la température maximale, la température minimale, l’humidité, la direction du vent… Dans le but d’utiliser l’API nous avons demandé l’obtention d’une clé gratuite nous permettant d’effectuer 60 requêtes par minutes et à l’option « Current weather API » parmi toutes les options disponible sur le site. L’utilisation de l’API se fait comme suit :

* L’url : <http://api.openweathermap.org/data/2.5/weather>?
* Les paramètres utilisés dans notre projet à ajouter à l’url:
* q : désigne dans notre cas le nom de la capitale (q=Paris)
* units : désigne l’unité de la valeur (units=metric)
* appid : désigne la clé api utilisée

(appid=9ada210033e2363be58a9fac5b682c4f)

* lang : Désigne la langue (lang=fr)

L’APIs propose bon nombre de paramètres et de méthodes permettant d’accéder à l’information. Nous avons choisi la méthode qui n’inclue que les paramètres cité ci-dessus.



## Les sites utilisÉs pour le web scraping

Les sites utilisés pour le web scraping contiennent plusieurs informations sur bon nombre de pays. Ce sont :

* <https://www.populationdata.net/palmares>
* <https://jeretiens.net/tous-les-pays-du-monde-et-leur-capitale/>

## GitHub

GitHub est un service web d'hébergement et de gestion de développement de logiciels, utilisant le logiciel de gestion de versions Git. Il un contrôle d'accès et des fonctionnalités destinées à la collaboration comme le suivi des bugs, les demandes de fonctionnalités, la gestion de tâches et une sorte de documentation pour chaque projet.

IMAGE GIT



# BILAN

Ce projet a été réalisé dans le cadre du cours de modélisation de connaissances & humanités numériques. Celui-ci exige des connaissances variées afin de mettre en œuvre les techniques d’extraction de contenu de sites web, au moyen d’un script ou un programme dans le but de le transformer pour permettre son utilisation dans un autre contexte. Ces différentes connaissances ont été utilisées pour la construction d’un jeu de données, qui sera utilisé pour faire de la visualisation de données.

Nous avons, au cours de ce projet, rencontré bon nombre de difficultés liées en grande partie à l’utilisation de l’API. En effet, certaines erreurs telles que des erreurs de requêtes, de permissions ou de réponses nous ont posé problème et ont nécessité un travail de recherche sur les différents types d’erreurs http, appelés encore les statuts http et codes d'erreur concernant le JSON. Nous avons donc découvert que l’erreur 400 fait référence à une mauvaise requête, l’erreur 403 à une restriction au niveau des permissions et l’erreur 404 à une réponse vide, c’est-à-dire que l’élément recherché n’existe pas.

En définitive, il a été l’occasion pour nous, étudiants, d’avoir une première expérience pratique et ainsi se familiariser avec le domaine du web scraping, l’utilisation des APIs ou encore le choix des ressources adéquates pour la réalisation d’un projet.