

# TP 4 : Projet Gares SNCF

Mettre son nom ici

Les lignes qui commencent par > dans ce fichier markdown Rmd sont des lignes qui correspondent à des questions qui vous sont posées.

Les chunks ont un nom pour les repérer en cas de besoin. Ceux que vous devez modifier comportent la mention “A compléter”. Les questions qui nécessitent une réponse rédigée sont en italique, donc entourées d’étoiles dans le fichier Rmd. Vous devez y répondre après le mot “Réponse :” dans ce fichier markdown Rmd.

## 1 Utiliser des données de la SNCF

La SNCF met en ligne un certain nombre de données sur le site web SNCF Open Data

On se propose d’étudier les gares de la SNCF, sous divers aspects concernant essentiellement la fréquentation de ces gares.

Executer le code ci-dessous pour charger les packages utiles.

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.1 --
## v ggplot2 3.3.5      v purrr 0.3.4
## v tibble 3.1.4       v dplyr 1.0.6
## v tidyr 1.1.3        v stringr 1.4.0
## v readr 1.4.0        v forcats 0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()

# ou pour ceux qui n'arrivent pas à installer tidyverse en entier :
library(tibble)
library(magrittr)

##
## Attaching package: 'magrittr'

## The following object is masked from 'package:purrr':
##
##     set_names

## The following object is masked from 'package:tidyr':
##
##     extract

library(readr)
library(tidyr)
library(dplyr)
```

## 2 Importer les jeux de données

### 2.1 Téléchargement des données

Sur le site web SNCF Open Data : on ne le fera pas dans ce TP!

Télécharger depuis Ametice et enregistrer dans votre répertoire de travail les fichiers `liste_gares.csv`, `freq_gares.csv`, `age_gares.csv`, `prof_gares.csv`, et `genre_gares.csv`.

### 2.2 Chargement des jeux de données

À l'aide des fonctions de `tidyverse`, charger les 5 jeux de données dans `R` et leur donner les noms associés.

```
## A compléter (décommenter et compléter les parties ...)  
#liste_gares <- ...  
#freq_gares <- ...  
#age_gares <- ...  
#prof_gares <- ...  
#genre_gares <- ...
```

## 3 Mise en forme des jeux de données

### 3.1 Référentiel des gares

On se propose de ne garder que certaines colonnes de `liste_gares` en modifiant cette table. > Sélectionner les colonnes suivantes à l'aide de la fonction `select`: “Code gare”, “Intitulé plateforme”, “Gare DRG”, “Gare étrangère”, “Code UIC”, “Région SNCF”, “Code postal”, “Segment DRG”, “WGS 84”.

```
## A compléter
```

On se propose maintenant de renommer (grâce à la fonction `colnames`) les colonnes de `liste_gares` pour ne pas avoir de problèmes d'accents et d'espace dans ces noms :

```
## A compléter
```

À chaque gare est affectée un code unique appelé UIC, qui se trouve maintenant dans la colonne 5 de cette table. Dans de nombreuses autres tables de données SNCF, les deux chiffres “87” qui commencent chaque code UIC ici ne sont pas présent.

1. Quel est le type des éléments de la colonne “Code\_UIC”
2. Convertir la colonne “Code\_UIC” en double
3. À l'aide d'une opération vectorielle, enlever 87000000 de chaque nombre dans la colonne UIC de cette table.
4. Afficher les gares de la “REGION PROVENCE ALPES COTE D'AZUR” à l'aide de la fonction `filter`.

```
# 1.  
## A compléter  
  
# 2.  
## A compléter  
  
# 3.  
## A compléter  
  
# 4.  
## A compléter
```

5. Comprendre ce que fait la commande suivante: `liste_gares <- liste_gares %>% distinct(Code_UIC, .keep_all = TRUE)`

Réponse :

Et l'évaluer avec R.

```
# 5.  
## A compléter
```

6. Quelle commande permet d'afficher la ligne de la gare dont le code UIC est 271007 ?

```
# 6.  
## A compléter
```

Quel est le nom de cette gare ?

Réponse :

7. La colonne "WGS\_84" contient les coordonnées GPS de la gare, séparée par une virgule. À l'aide de la fonction `separate`, créer deux colonnes nommées "latitude" et "longitude" qui contiennent respectivement le premier nombre et le second nombre de cette colonne. On veut que ces deux colonnes soient au format numérique.

```
# 7.  
## A compléter
```

Pour vérifier que tout s'est bien passé, vous pouvez lancer les instructions ci-dessous. Si tout est bon, vous devez voir apparaître une carte de la France.

```
## Décommentez le code suivant avant de l'exécuter:  
#plot(liste_gares$longitude, liste_gares$latitude,  
#      col = as.factor(liste_gares$Segment_DRG), pch = 16)
```

## 3.2 Répartition par classe d'âge

On travaille maintenant à partir des données contenues dans l'objet `age_gares`.

1. En vous inspirant de ce qui est fait au dessus, dans le chunk nommé "renommer\_liste", renommer les colonnes pour que leurs noms soient "Code\_UIC", "Gare", "Classe\_age", "Pourcentage", "Annee".
2. Quel le type des éléments des colonnes Pourcentage et Année?
3. Convertir, si besoin, les colonnes Pourcentage et Année au format numeric (ou double).

```
# 1.  
## A compléter  
  
# 2.  
## A compléter  
  
# 3.  
## A compléter
```

Ici, une ligne de la table `age_gares` renseigne sur la proportion d'une classe d'âge donnée, pour une année donnée, et une gare donnée. C'est un format long.

4. Calculer la proportion annuelle moyenne des différentes classes d'âge en enchaînant (par des pipes) les commandes `group_by`, et `summarize` (en regroupant donc par UIC, Classe\_age et

- Gare). Mettre le résultat dans une colonne nommée “pourcent”. (Penser à enlever les NA). Puis appliquer la fonction `ungroup`, avant d’enregistrer le tout dans l’objet `age_gares`.
- À l’aide de la commande `spread`, étendre la colonne `pourcent` sur plusieurs colonnes en fonction de la valeur de `Classe_age` et enregistrer le résultat dans la table `age_gares`.
  - Combien y a-t-il de lignes dans la nouvelle table `age_gares` ?
  - Afficher la ligne correspondante à la gare de Paris Bercy. Cette gare n’a pas de code UIC dans cette table. Corriger ce point pour que l’UIC de cette gare soit égal à 686667.

```
# 4.
## A compléter

# 5.
## A compléter

# 6.
## A compléter

# 7.
## A compléter
```

### 3.3 Répartition par catégories socio-professionnelles (CSP)

On peut faire la même chose sur les CSP que sur les classes d’âge. Pour cela, on travaille maintenant avec les données de l’objet `prof_gares`.

Reprendre les questions 1 à 7 de la partie sur `age_gares` pour faire les mêmes transformation sur `prof_gares`, en remplaçant `Classe_age` par `CSP`. Pour la question 1, les noms des colonnes seront : “Code\_UIC”, “Gare”, “CSP”, “Pourcentage”, “Annee”.

```
# 1.
## A compléter

# 2.
## A compléter

# 3.
## A compléter

# 4.
## A compléter

# 5.
## A compléter

# 6.
## A compléter

# 7.
## A compléter
```

### 3.4 Répartition par genre

On peut faire la même chose sur la répartition par genre que sur les classes d’âge et sur les professions. Pour cela, on se concentre donc maintenant sur les données contenues dans l’objet `genre_gares`.

Reprendre les questions 1 à 7 de la partie sur `age_gares` pour faire les mêmes transformation sur

genre\_gares, en remplaçant Classe\_age par Sexe. Pour la question 1, les noms des colonnes seront : "Code\_UIC", "Gare", "Sexe", "Pourcentage", "Annee".

```
# 1.
## A compléter

# 2.
## A compléter

# 3.
## A compléter

# 4.
## A compléter

# 5.
## A compléter

# 6.
## A compléter

# 7.
## A compléter
```

### 3.5 Fréquentation

Nous nous intéressons maintenant à la fréquentation des gares en termes de nombres d'individus. Commençons par renommer les colonnes de la table `freq_gares`.

```
## Décommenter le code qui suit en enlevant le `#` en début de ligne:
#colnames(freq_gares) <- c("Gare", "Code_UIC", "Code_postal", "Segmentation",
#                           "voyageurs2020", "usagers2020",
#                           "voyageurs2019", "usagers2019",
#                           "voyageurs2018", "usagers2018",
#                           "voyageurs2017", "usagers2017",
#                           "voyageurs2016", "usagers2016",
#                           "voyageurs2015", "usagers2015")
```

1. À l'aide la fonction `mutate`, ajouter une colonne `Frequentation` qui soit la somme des six colonnes `usagers2020`, `usagers2019`, `usagers2018`, `usagers2017`, `usagers2016` et `usagers2015`, divisée par six (fréquentation annuelle moyenne depuis 6 ans).
2. Combien de NA y-a-t-il dans la colonne `Frequentation` ainsi créée ?
3. Comme pour la table `liste_gares`, à l'aide d'une opération vectorielle, enlever 87000000 à chaque élément de la colonne `Code_UIC`.

```
# 1.
## A compléter

# 2.
## A compléter

# 3.
## A compléter
```

### 3.6 Une belle jointure

On veut fusionner toutes les tables en une seule. Pour cela, on va commencer par récupérer tous les UIC qui apparaissent dans les tables concernant l'âge, le genre, la CSP des usagers :

```
## Décommenter le code qui suit en enlevant le `#` en début de ligne:
#union_UIC <- c(age_gares$Code_UIC, genre_gares$Code_UIC, prof_gares$Code_UIC)
#union_UIC <- unique(union_UIC)
#length(union_UIC)
```

Ainsi, on sait qu'on pourra fusionner l'ensemble des informations à disposition pour ces gares en particulier.

1. Créer une table nommée `enquete` qui extrait de la grande table `liste_gares` les lignes pour lesquelles l'UIC est dans le vecteur `union_UIC`. On pourra utiliser les commandes `filter` et `%in%`.
2. Faire une jointure (à gauche) de cette table avec `age_gares`, `genre_gares`, `prof_gares`, et `freq_gares`, en utilisant comme identifiant la variable `Code_UIC`. On supprimera la colonne "Gare" de toutes ces tables, ainsi que la colonne "Code\_postal" pour la table `freq_gares`, et la colonne "Non communiqué" dans `prof_gares`.

```
# 1.
## A compléter

# 2.
## A compléter
```

## 4 Premières analyses

### 4.1 Nombres d'usagers

1. En utilisant la colonne `Frequentation` que l'on a créée, quelles sont les 6 gares les plus fréquentées de la table `freq_gares` ? Et celle de la table `enquete` ? On pourra utiliser les fonctions `arrange` et `head` pour répondre à cette question.
2. Même question pour les 6 gares les moins fréquentées (avec une fréquentation non nulle).

```
# 1.
## A compléter

# 2.
## A compléter
```

### 4.2 Analyse régionale

1. Combien y a-t-il de gares en PACA (Provence Alpes Côte d'Azur) dans la table `liste_gares` (REGION PROVENCE ALPES COTE D'AZUR)? Et dans la table `enquete` ? (On pourra filtrer sur la variable "Region\_SNCF")
2. Combien y a-t-il de gares dans les Bouches-du-Rhône (code postal commençant par 13) dans la table `liste_gares` ? Et dans la table `enquete` ?
3. Quelle est la plus petite gare en termes de fréquentation dans le département 13 pour la table `enquete` ? Et pour la table `freq_gares` ?
4. A partir de la table `enquete`, quelles sont les deux gares pour lesquelles la CSP "Etudiant (après le bac)" est la plus représentée dans les gares de la région Provence-Alpes-Côte d'Azur ? Pouvez-vous expliquer ce résultat ?

```
# 1.
## A compléter
```

```
# 2.  
## A compléter  
  
# 3.  
## A compléter  
  
# 4.  
## A compléter
```