

TP 6 : Visualisation des gares

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

Les chunks que vous devez modifier ont un nom qui commence par ANS. 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 le fichier Rmd.

Chargeons les librairies. En dehors de la librairie `ggplot2`, les autres font partie du package `tidyverse`. On peut donc se contenter de charger ce package.

```
library(ggplot2)
suppressMessages(library(tidyverse))
# Si tidyverse est mal installé chez vous, chargez :
# library(tibble)
# library(tidyr)
# library(readr)
# library(dplyr)
```

L'objectif des parties 1 à 5 est de reproduire la figure qui se trouve dans le fichier PDF `gares.pdf`. Ce graphique contient les quatre couches ci-dessous :

1. des polygones pour différencier les départements,
2. des points pour représenter les gares,
3. une couche de textes, en noir et gras, pour les gares d'intérêt national,
4. une couche de textes, en gris 20%, pour les gares d'intérêt régional.

L'objectif de la partie 6 est de reproduire la figure qui se trouve dans le fichier PDF `CSP.pdf`.

1 Collecte des données

1.1 Les gares SNCF

On commence par récupérer les données utiles, c'est-à-dire les données géographiques, et les données sur les gares. Le fichier `liste_gares.rda` contient la table `liste_gares` telle qu'obtenu à la fin du TP4, enregistrée au format R. Et la table `fr` contient des points sur les frontières des départements de France. On peut charger ces données ainsi :

```
load("liste_gares.rda")
fr <- map_data("france")
```

Ecrire du code R qui permette de créer (en utilisant les tuyaux, ou 'pipe' en anglais) la table `df_gares`, issue de `liste_gares`, après avoir fait les actions ci-dessous :

- filtrer la partie qui concerne les gares de la région "Nouvelle-Aquitaine", en utilisant `filter` sur la colonne `Region_SNCF` ;
- sélectionner uniquement les colonnes "Intitule_gare", "latitude", "longitude", "Segment_DRG", à l'aide de `select`.

```
## A compléter...
```

La colonne `Segment_DRG` permet de différencier les gares suivant 3 catégories :

- catégorie “a” : gares de voyageurs d’intérêt national. Ces gares sont celles dont la fréquentation par des voyageurs des services nationaux et internationaux de voyageurs est au moins égale à 250 000 voyageurs par an ou dont ces mêmes voyageurs représentent 100% des voyageurs.
- catégorie “b” : gares de voyageurs d’intérêt régional. Le périmètre de gestion correspond, dans chaque région, à l’ensemble des gares n’appartenant pas à la catégorie a mais dont la fréquentation totale est au moins égale à 100 000 voyageurs par an.
- catégorie “c” : gares de voyageurs d’intérêt local. Leur périmètre de gestion correspond, dans chaque région, à l’ensemble des gares de cette catégorie. La redevance est fixée, par région, pour l’ensemble des gares de cette catégorie.

On veut supprimer certaines lignes.

Que veut dire NA en R ?

Réponse :

1. Calculer le nombre de lignes dans `df_gares`.
2. Mettre à jour `df_gares`, en appliquant la fonction `na.omit` à ce tibble, pour supprimer toutes les lignes qui contiennent des NA.
3. Calculer le nombre de lignes dans la version mise à jour de `df_gares`.

A compléter...

Combien de lignes ont disparu dans l’opération du chunk que vous venez de remplir ? Est-ce bien ce que l’on souhaitait ?

Réponse :

1.2 Les départements de PACA

Récupérer ensuite les données géographiques concernant la région Nouvelle-Aquitaine. Pour cela, filtrer la table `fr` sur la colonne `region` en conservant les départements suivants:

“Charente”, “Charente-Maritime”, “Correze”, “Creuse”, “Dordogne”, “Gironde”, “Landes”, “Lot-et-Garonne”, “Pyrenees-Atlantiques”, “Deux-Sevres”, “Vienne”, “Haute-Vienne”.

On enregistrera le résultat de ce filtrage dans une table nommée `df_dpt`.

A compléter...

2 Le fond de carte

Les sections 2 à 5 ajoutent des couches au graphique `p` défini dans le chunk ci-dessous, pour lequel il n’y a par défaut, ni jeu de données, ni correspondance esthétique.

```
p <- ggplot()           # ligne 1
p <- p + coord_quickmap() # ligne 2
p <- p + theme_void()    # ligne 3
```

À quoi sert la ligne 2 dans le chunk `p_initialisation` ci-dessus ? Et la ligne 3 ?

Réponse :

On commence par ajouter le fond de carte en traçant les contours des départements.

En s’inspirant des lignes 2 et 3, créer `p1` à partir de `p` en ajoutant le fond de carte avec une couche de lignes polygonales où :

- le jeu données est `df_dpt`, passé par l’argument `data` de `geom_polygon`,

- les correspondances esthétiques, passées par l'argument `mapping` sont : `long` sur l'axe des abscisses, `lat` sur l'axe des ordonnées, `region` en couleur de fond (utiliser l'argument `fill`), et `group = group` pour regrouper ces polygones correctement,
- le paramètre `show_legend` est réglé à `TRUE` pour afficher une légende.

Puis afficher `p` ainsi mis à jour.

```
## A compléter...
```

Que se passe-t-il si l'on met le paramètre `show.legend` à `FALSE` ?

Réponse :

Supprimer `p1`, puis exécuter de nouveau le chunk précédent en fixant le paramètre `show.legend` à `FALSE` pour mettre à jour `p`.

```
## A compléter...
```

3 Ajout des gares

On ajoute ensuite la couche de points qui symbolisent les gares.

Créer le nouvel objet `p1` contenant le graphique `p` auquel on ajoute une couche de points construite à partir des données de `df_gares`, que l'on passera à l'aide de l'argument `data`, en utilisant comme correspondance esthétique :

- `longitude` sur l'axe de abscisses,
- `latitude` sur l'axe des ordonnées,
- `Segment_DRG` sur taille (`size`), forme (`shape`) et couleur (`color`).

Et afficher `p1`.

```
## A compléter...
```

On remarque dans la légende que les trois modalités de la variable `Segment_DRG` apparaissent dans l'ordre `a`, `b`, puis `c`. Le choix des couleurs et des tailles n'est pas le bon. En outre, on veut supprimer le guide à droite concernant la variable `Segment_DRG`.

Cette fois-ci, mettez à jour `p`, en faisant un copier-coller de la couche que vous venez de construire, mais en retirant la légende.

```
## A compléter...
```

On veut maintenant régler manuellement l'échelle des tailles et des couleurs pour que :

- les gares de type `a` (nationales) soient en taille 4, de couleur "black",
- les gares de type `b` (régionales) soient en taille 2, de couleur "grey20",
- les gares de type `c` (locales) soient en taille 1, de couleur "grey50".

Pour les couleurs, il faut utiliser la fonction `scale_color_manual`, et donner les valeurs que l'on souhaite dans l'ordre d'apparition des modalités. Ce qui donne ici :

```
p <- p +
  scale_color_manual(values = c("black", "grey20", "grey50"))
p
```

En s’inspirant de ce qui est écrit dans le chunk `choix_couleurs`, utiliser la fonction `scale_size_manual` pour choisir les tailles des points comme souhaité ci-dessus.

```
## A compléter...
```

4 Ajout des noms des grandes gares

Mettre à jour `p` en ajoutant une couche de texte avec `geom_text`, où

- les données (`data`) sont issues de la table `df_gares`, filtrée pour ne garder que les lignes correspondant aux gares de type “a”,
- les correspondances esthétiques sont : `longitude` sur l’axe des abscisses, `latitude` sur l’axe des ordonnées, et `Intitule_gare` comme étiquette (`label`)
- le texte est écrit en gras (“bold”) de taille 2,7 (paramètres `fontface` et `size`),
- les paramètres optionnels `hjust`, `vjust` et `nudge_y` sont fixés à “left”, “bottom” et -0.08 respectivement.

Puis afficher `p`.

```
## A compléter...
```

À quoi servent les paramètres `hjust`, `vjust` et `nudge_y` ?

Réponse :

5 Ajout des noms des gares régionales

En vous inspirant de ce que vous avez fait dans le chunk `ANSlabel_a` ci-dessus, ajouter les noms des gares régionales en mettant à jour `p`, puis en affichant `p`. On souhaite que la couleur du texte soit “gray20”, sans être en gras, de taille 1.8, et avec un `nudge_y` égal à 0.02. Puis afficher `p`.

```
## A compléter...
```

Enfin, sauvegarder le graphique obtenu dans un fichier PDF appelé, `NOM_gares.pdf`, après avoir **remplacé NOM par votre nom de famille**, en utilisant le chunk ci-dessous.

```
ggsave(filename = "NOM_gares.pdf",
        plot = p,
        width = 9,
        height = 9)
```

À quoi servent les paramètres `width` et `height` de la commande ci-dessus ?

Réponse :

6 Comparaison des CSP des usagers des gares de Nice et de Marseille Saint-Charles

Le fichier `comparaison.rda` contient une table concernant la répartition par catégorie socio-professionnelle des usagers des gares de Marseille Saint-Charles et Nice.

```
load("comparaison.rda")
prof_deux_gares

## # A tibble: 2 x 13
##   Gare      Agricult. `Artisan, TPE, ~ Autre `Cadre, Prof. libér, ~ Enseignant
##   <chr>      <dbl>      <dbl> <dbl>      <dbl>      <dbl>
## 1 Marseille ~      0.3          3      1          20         4
## 2 Nice          0.1          5      2          13         4
## # ... with 7 more variables: Etudiant post-bac <dbl>, Militaire <dbl>,
## #   Ouvrier, employé <dbl>, Retraité <dbl>, Sans emploi <dbl>,
## #   Scolaire pré-bac <dbl>, Tech., Agent <dbl>
```

6.1 Ré-organisation

Utiliser la fonction `gather` sur le tableau `prof_deux_gares` pour créer un nouveau tableau `CSP_deux_gares` dont les trois colonnes sont : “Gare”, “CSP” et “Pourcentage”. Puis trier le tableau par “Gare” (pour sélectionner toutes les colonnes sauf `Gare`, en utilisant `-Gare`).

```
## A compléter...
```

6.2 Graphique

1. On cherche à reproduire le graphique du fichier `csp.pdf`. Indication: on pourra utiliser le paramètre `labels` de la fonction `scale_fill_discrete` pour raccourcir le nom des gares, ainsi que la fonction `coord_flip`.
2. Enregistrer le résultat dans un fichier PDF, de taille 10 x 6, `NOM_CSP.pdf` après avoir **remplacé NOM par votre nom de famille**.

```
#1.
## A compléter...
```

```
#2.  
## A compléter...
```