**TD 2 : dplyr, carte leaflet et Rmarkdown**

L’objectif de ce TD est d’étudier une autre méthode de Data Mining via le package ***dplyr*** (se lit deeplayer), de reprendre l’utilisation des graphiques vue au TD précédent et d’introduire l’utilisation des cartes en R via le package ***leaflet***, en plus des rapports Rmarkdown.

Dans un premier temps, nous allons travailler avec les données des aéroports de New-York (année 2013). Ce set est directement accessible dans R grâce au package suivant qu’il faut installer : ***nycflights13***. Lors de ce TD, vous pourrez vous servir de ce lien : <https://datacarpentry.org/R-genomics/04-dplyr.html>.

data = flights

1. **Data mining avec dplyr**
2. Afin d’utiliser le package ***dplyr***, vous devriez avoir besoin du package ***Rcpp***. En cas d’erreur lors de l’installation, suivre ce lien : <http://web.pdx.edu/~gerbing/R/ErrorRcppInstall.pdf> et le fichier zip disponible sur Teams.
3. Déterminer la classe de la variable *flights*. Elle est disponible suite à l’installation du package ***nycflights13***.
4. Utiliser les fonctions de base vues la dernière fois pour mieux comprendre le contenu du jeu de données : ***dim()***, ***head()***, ***summary()***. Visiter également la documentation du package pour en savoir plus : <https://cran.r-project.org/web/packages/nycflights13/nycflights13.pdf>.
5. En utilisant les fonctions ***select()*** et ***unique()***, déterminer les aéroports de New York recensés dans ces données.
6. En utilisant la fonction ***View()***, observer que les vols ne sont pas dans l’ordre chronologique.
7. Utiliser la fonction ***arrange()*** pour classer les vols dans l’ordre chronologique. Stocker ce résultat dans une nouvelle variable que nous utiliserons à présent.
8. A l’aide de la même fonction, classer les vols en fonction de l’ordre croissant des retards au départ de l’avion. De même dans l’ordre décroissant.
9. Avec la fonction ***filter()***, conserver les vols avec plus de 5 minutes de retard au décollage. Conserver ensuite les vols qui rattrapent ce retard à l’arrivée (qui arrivent avec moins de 5 minutes de retard).
10. Avec la fonction ***mutate()***, ajouter une colonne qui définit le gain de temps entre le retard à l’arrivée et le retard au départ. Conserver cette information en stockant le résultat dans une nouvelle colonne.
11. La fonction ***summarise()*** permet de connaitre la moyenne de ce gain de temps sur l’année 2013.
12. **Dplyr avancé**
13. Avec les fonctions ***group\_by()*** et ***summarise()***, comptez le nombre de vols qu’il y a par mois. Vous aurez besoin de la fonction ***n()*** également.
14. Avec les mêmes fonctions, comptez le nombre de vols qu’il y a par jour. En traçant la série temporelle de cette nouvelle donnée, observer les variations irrégulières comme celle de juillet. Zoomer sur les données de ce mois-ci à l’aide d’un nouveau graphique et tenter d’expliquer les variations.
15. (Faire cette question si vous êtes en avance) Avec les mêmes fonctions, déterminer le retard maximum au départ et à l’arrivée ainsi que le retard accumulé qu’il y a eu chaque jour de l’année.
16. **Cartes et rapport : introduction à Rmarkown**
17. Ouvrir le fichier *Carte leaflet\_Template.Rmd*.
18. Lancer l’édition du rapport avec le bouton ***Knit*** dans la barre de menu de Rstudio. Il peut y avoir une erreur avec le package appelé ***stringi***,. SI oui, téléchargez le package au format .zip ici : <https://cran.r-project.org/web/packages/stringi/index.html> puis installez le manuellement via Rstudio.
19. Observer l’architecture du rapport et faire le lien avec l’architecture du code. De plus, ouvrir le fichier HTML présent dans votre dossier courant. Vous venez de générer un rapport Rmardkown (vide, pour l’instant).
20. Installer le package ***leaflet***. En vous inspirant du tutoriel accessible via le lien suivant, remplir l’espace de code dédié à la carte (<https://rstudio.github.io/leaflet/>). Vous pourrez choisir le *Tile* (fond de carte) de votre choix via ce lien : <https://leaflet-extras.github.io/leaflet-providers/preview/>. Une fois choisi et intégré, relancer l’édition du rapport.
21. Télécharger à partir de Teams le fichier TD 2 - Annexe.R.
22. Décommenter la ligne « *source »* de la partie carte du code et adapter le chemin pour accéder au fichier téléchargé. Analyser rapidement les deux variables intégrées après avoir exécuté la ligne de code. Il s’agit de données météorologiques Météo-France pour 3 stations sur l’année 2018.
23. Les metadata vont nous permettre de placer des marqueurs pour les 3 stations sur la carte et d’éditer les pop ups. Pour cela, vous pouvez utiliser la fonction ***unique()***.
24. En vous inspirant du TD précédent, tracer dans l’onglet *températures*, les courbes des données de températures pour les 3 stations.
25. De même pour les précipitations.
26. (Faire cette question si vous êtes en avance) Tester le package ***highcharter*** pour faire des graphiques plus esthétiques (<http://jkunst.com/highcharter/highcharts.html>). Tracer les 3 séries temporelles de température sur un même graphe. S’inspirer du bout de code à la fin de ce document.
27. Pour finir, créer une data frame avec la fonction ***data.frame()*** pour présenter la moyenne et l’écart-type des températures pour chacune des stations. Faire de même pour la somme annuelle et le maximum de précipitations.
28. Installer le package ***DT***. Ce package met en forme de tableau les data frame. Intégrer grâce à la fonction ***DT()***, les deux data frame fraichement créées.

