Techniques de visualisation de données

Matthieu Cisel - D.U. Data Analyst

Mai 2022

1 Objectifs

Être capable de produire des figures d'une qualité irréprochable constitue l'une des qualités attendues d'un Data Analyst. Dans ce cours, nous allons nous concentrer sur trois axes principaux : l'amélioration d'un graphique sur le plan ésthétique, la production de graphiques interactifs, et la production de cartes, avec notamment l'outil Kepler développé par Uber.

2 Dimension esthétique de la production d'un graphe

Pour les utilisateurs de R, nous nous focaliserons sur le package ggplot2. Pour les utilisateurs de Python, nous nous concentrerons sur matplotlib. Le suivi des cours de Datacamp correspondants est facultatif, mais nous vous recommandons les suivants : Introduction to Data Visualization with ggplot2 pour R, et Introduction to Data Visualization with Matplotlib pour Python.

Vous utiliserez le jeu de données nommé PhD.v3, qui, par contraste avec les jeux de données que vous avez utilisés précédemment, comporte la discipline de rattachement de la thèse (obtenue après un travail de classification par réseau de neurones). Dans votre notebook, le titre de l'exercice doit apparaître, utilisez du markdown.

2.1 Exercice 1

Nous vous demandons de représenter l'évolution quantitative des différentes disciplines sur la période 1985-2018, via deux types de graphiques. Le premier est le "stacked area plot", le second est le stacked bar chart. Tous ces graphiques seront stockés uniquement dans le notebook Jupyter, et seuls quelques-uns seront mobilisés dans un court rapport de quelques pages dans lequel nous allons nous entraîner à décrire des figures.

Vous trouverez le code pour réaliser ce travail à cette adresse pour Python, et à cette adresse pour R.

Dans un second temps, nous vous proposons de réaliser un graphique analogue, mais en utilisant un "stacked barplot". Voici un exemple avec Python.

A vous de sélectionner l'un des deux graphes. Nous allons maintenant vérifier votre capacité à jouer sur des petites variations des caractéristiques du graphiques. Nous allons vous donner ici peu d'indices quant à la manière de réaliser ces variations. Tout l'objet de l'exercice consiste à utiliser les bons mots-clés pour trouver en ligne les sites susceptibles de vous aider.

2.2 Exercice 2

Pour commencer, produisez un graphe comprenant une grille en fond (en anglais, background grid). Pour matplotlib, vous pouvez trouver la procédure à cette adresse. Avec ggplot2 (R), la grille apparaît par défaut.

Vous devez maintenant produire un nouveau graphique, légèrement transparent, de sorte que la grille apparaisse derrière le graphe, comme ci-dessous.Réalisez plusieurs niveaux de transparence. Nous vous donnons un mot-clé seulement : alpha.

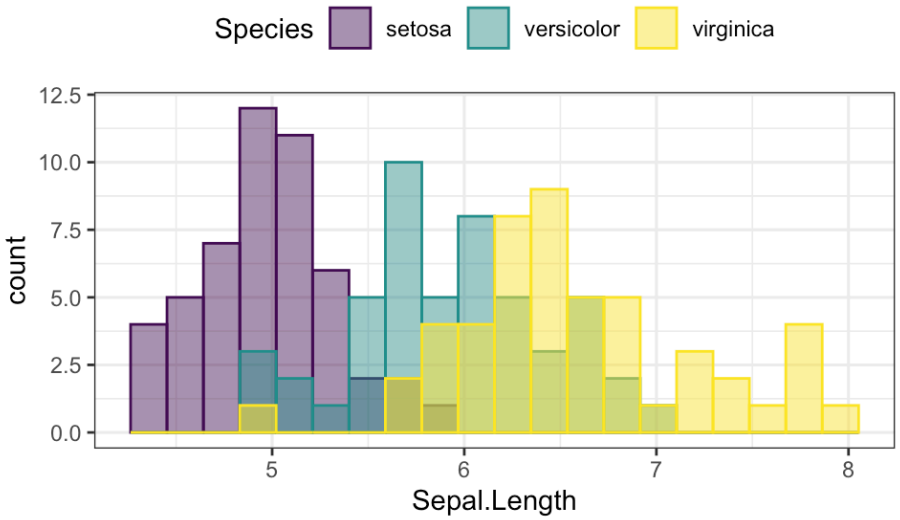

Figure 1: Jouer sur la transparence d'un graphique

2.3 Exercice 3

Dans l'exercice 3, nous repartons sur des graphes non transparents, mais avec toujours une grille. Nous allons jouer sur la distance entre l'axe et les labels correspondants. Dans la figure ci-dessous, nous illustrons comment pour l'axe des X, nous pouvons éloigner les labels de l'axe (flèches rouges). Trouvez sur Internet la méthode pour le faire, et produisez un graphe avec les labels de l'axe X clairement décollés de l'axe.

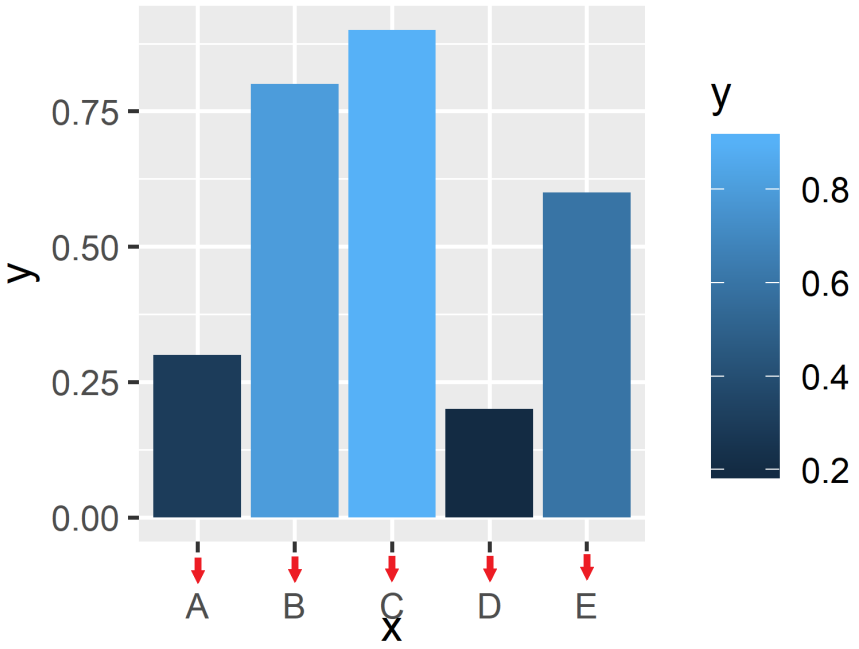

Figure 2: Jouer sur la distance des labels à l'axe

Dans un second temps, produisez un graphique où ces labels sont légèrement inclinés de 45°. Ci-dessous, nous avons choisi un angle de 90°, conservez quant à vous l'angle de 45°. Mot-clé : tilt.

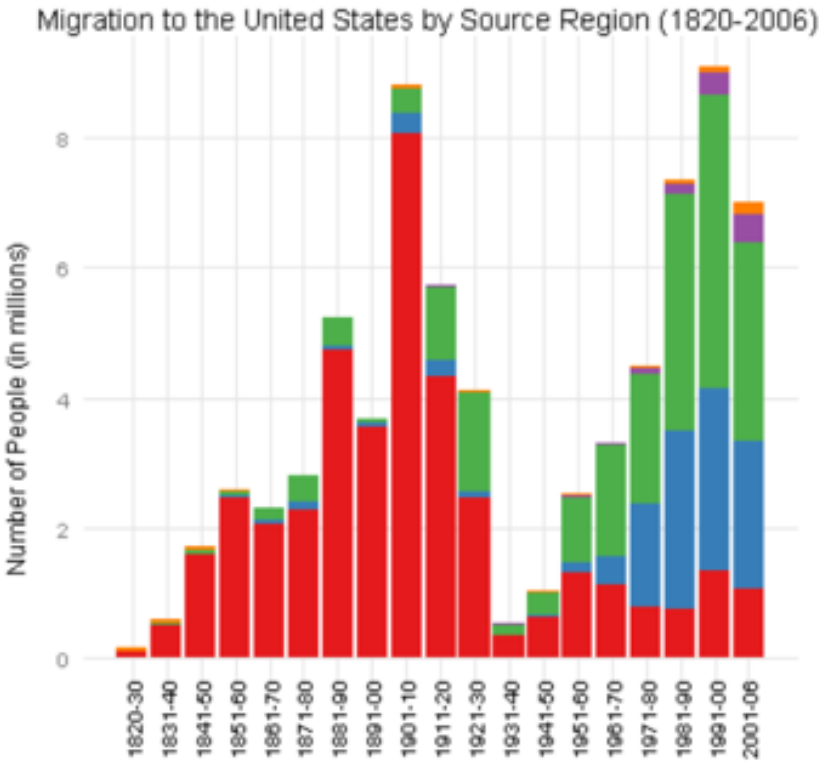

Figure 3: Jouer sur l'inclinaison des labels

2.4 Exercice 4

Dans l'exercice 4, nous vous demandons de reproduire le même graphique, mais en changeant la police pour utiliser du Times New Roman ou du Garamond. Mot-clé pour R : extrafonts. Produisez ensuite différents graphiques pour montrer que vous êtes capables de jouer sur la taille de la police pour les labels des axes et pour le titre.

Produisez un second graphique sur lequel vous jouez sur la taille des marges, pour "écraser" un peu le graphique vers le centre, comme dans la figure ci-dessous.

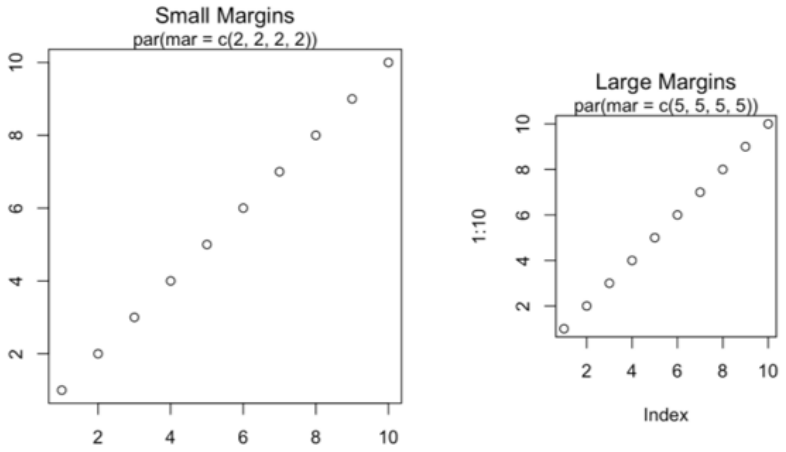

Figure 4: Jouer sur la taille des marges

2.5 Exercice 5

Dans l'exercice 5, nous vous demandons de produire un graphe où vous changez l'échelle des Y par une échelle logarithmique. Cela ne présente que peu d'intérêt, c'est simplement pour être certain que vous êtes capable de le faire.

2.6 Exercice 6

Dans l'exercice 6, nous vous demandons de jouer sur la position de la légende (reprenant les différentes disciplines). Produisez des graphiques avec deux positions différentes de la légende.

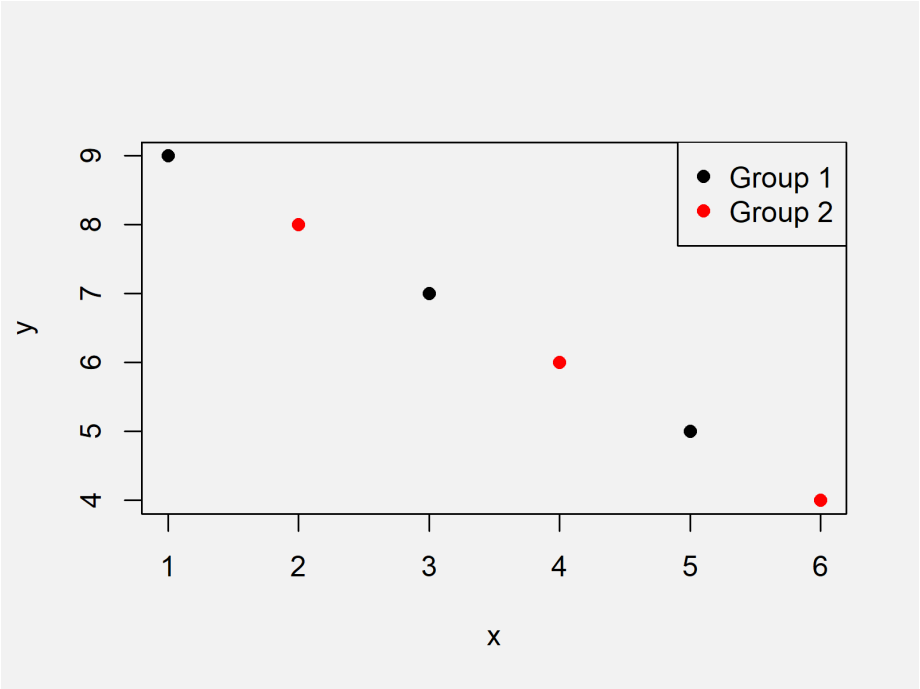

Figure 5: Une légende positionnée en haut à droite

2.7 Exercice 7

Dans l'exercice 7, changez la palette de couleur utilisée pour représenter les différentes disciplines.

2.8 Exercice 8

Dans l'exercice 8, changez l'ordre (de bas en haut) dans lequel apparaissent les disciplines. Par exemple, si la biologie est en bas, touchant l'axe des X, elle doit apparaître en haut.

3 Production de graphes animés et interactifs

3.1 Exercice 9 : production d'un GIF

Nous allons maintenant nous intéresser à l'évolution de la langue d'écriture au fil des ans (sur la période 1985-2018). Nous allons commencer par produire un graphe animé (sous la forme d'un GIF). La barre que vous devez représenter correspond au pourcentage des thèses écrites en anglais. Vous devez faire en sorte que la discipline avec le plus haut taux de thèses en anglais apparaisse en

haut. Puis, lorsque le GIF se déclenche, le temps passe et la discipline qui reste en haut évolue. C'est ce que l'on nomme une "bar chart race" en anglais.

Vous trouverez sur ce site la manière de procéder avec R, et sur ce site la manière de procéder avec Python.

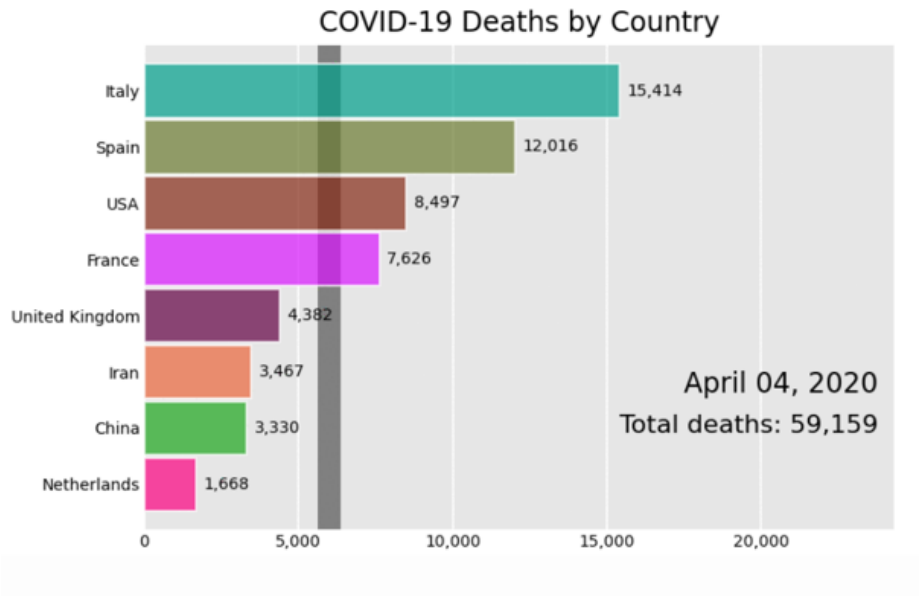

Figure 6: Extrait instantané d'une "bar chart race"

3.2 Exercice 10 : Graphique interactif contenant un slider

Dans cet exercice, nous vous laissons toute latitude pour choisir ce que vous souhaitez représenter. Nous vous donnons une seule contrainte : incorporez comme dans la figure ci-dessous un "slider" (barre horizontale sur laquelle vous pouvez faire varier le temps, entre autres). Vous devrez utiliser plotly, que ce soit sur R ou sur Python. Il n'y a pas obligation à produire un bubble plot, qui ne sert ici que d'illustration pour le slider. Vous devrez ensuite produire un second graphique, mais utilisant cette fois un "selector".

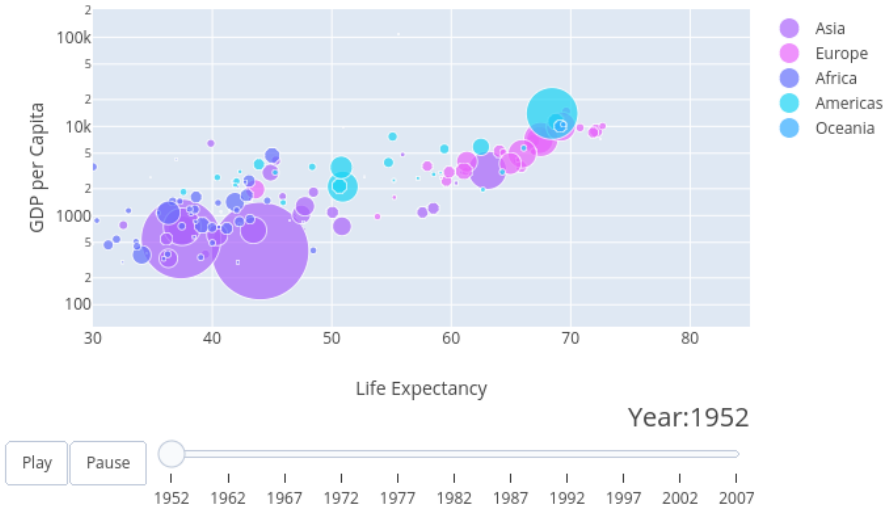

Figure 7: Graphique interactif comportant un slider

En plus de la production de ce graphique, nous vous demanderons de suivre le cours sur Datacamp consacré à plotly (Interactive Data Visualization with plotly in R ou Introduction to Data Visualization with plotly in Python), et d'en fournir le certificat à la fin.

Vous réaliserez des exports des widgets produits, que vous soumettrez ensuite en addition du rapport et du notebook Jupyter.

4 Visualisation de données spatialisées

Nous vous avons fourni un jeu de données comportant l'ensemble des vols d'avions civil sur une période d'un mois, à l'échelle planétaire. Pour la plupart des vols, l'origine et la destination sont fournis, ainsi qu'une dizaine de points le long de la trajectoire. La problématique consiste ici à sélectionner les seuls vols en direction de la Russie, et en partance de la Russie, quelques jours avant le début de la guerre, et quelques jours après le début de la guerre.

Pour filtrer, parmi les aéroports, ceux qui correspondent à la Russie, nous vous fournissons une liste d'aéroports russes. Vous devez ainsi effectuer un "join" des deux bases de données (celles comprenant l'ensemble des vols, et celle comprenant la liste des aéroports russes). Vous devrez ensuite ne vous focaliser que sur une seule journée de vols avant le 24 février, et une journée après le 24

février, pour que la quantité de vols soit comparable. A vous de déterminer les dates qui vous semblent les plus appropriées au regard de la problématique que nous vous proposons.

Concernant la visualisation des données, une fois celles-ci filtrées, nous vous proposons deux approches.

4.1 Exercice 11

La première approche, correspondant à l'exercice 11, consiste à utiliser les outils classiques de R ou de Python. Pour Python, le tutoriel suivant permet de produire des vols via une carte interactive. A vous de fixer les paramètres pour que la Russie apparaissent comme centrale dans votre visualisation. A vous de choisir comment représenter au mieux le contraste entre les deux journées (2 couleurs différentes sur une seule carte, ou deux cartes distinctes).

Figure 8: Une carte interactive de vols au-dessus des USA

Concernant les utilisateurs de R, nous vous proposons de suivre une approche similaire, l'interactivité en moins. Pour ce faire, nous devrez suivre le tutoriel suivant. Faites apparaître le nom de un ou plusieurs aéroports significatifs.

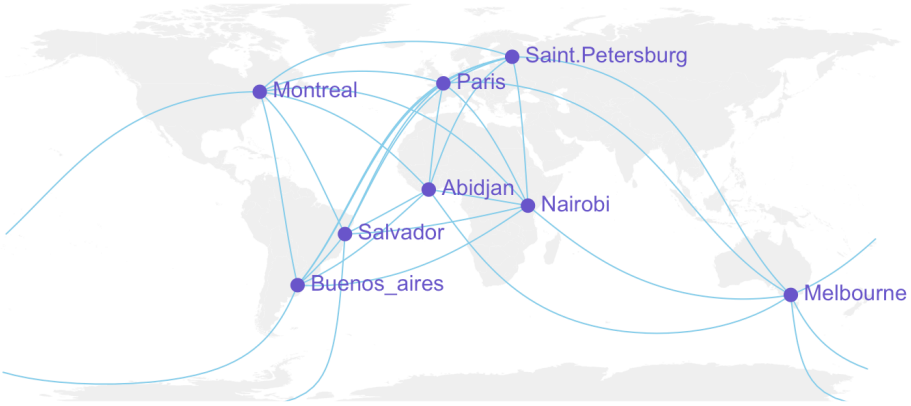

Figure 9: Visualisation de données spatialisées avec R

4.2 Exercice 12

Dans cet exercice, nous vous demandons d'utiliser le site kepler.gl développé par Uber, pour produire une visualisation des vols identique à celle que vous aurez produite dans l'exercice 11. L'objet est ici de vous sensibiliser au fait qu'il existe de nombreux sites pour la visualisation des données, au-delà de R et de Python. Vous en ferez une capture d'écran que vous inclurez dans le rapport.

Figure 10: Une carte interactive de vols au-dessus de la Russie avec Kepler

5 Rédaction d'un court rapport

En plus du notebook Jupyter, vous produirez un court rapport, dans lequel vous prendrez une figure de chaque section (une pour l'esthétique, une pour les graphiques interactifs, une pour les données spatialisées). Vous décrirez ces figures de la manière la plus précise possible (avec des valeurs chiffrées). Il faut compter environ 4 à 5 lignes par description de figure. Vous soumettrez ce rapport au format PDF. Il devra comprendre les 3 figures.