Visualisation de données avec R — retour TP1

Rémi Pépin (sur base de Arthur Katossky)

Janvier 2019

Contents

La visualisation de données, une représentation *visuelle*

La visualisation repose sur la vision car c'est le sens le plus adapté comme support de mémoire externe:

- parallélisation inconsciente
- perception simultanée
- zone cérébrales développé
- possibilité technique

Une représentation *efficace*

La visualisation se distingue du design, de la publicité ou de l'art par son intérêt pour l'efficacité de la représentation à communiquer une information ; nous ne poserons donc pas la question de savoir si un graphique est "joli" mais de s'il "fonctionne".

Malheureusement définir l'efficacité c'est compliqué car elle dépend du but à atteindre. Il n'y a donc pas une solution magique. Cela demande de la réflexion et de la remise en question.

Dans ce cours, nous nous intéresserons surtout à la tâche de comparer des grandeurs dans des graphiques finis, c'est-à-dire publiables. D'un côté parce que la comparaison est une tâche fondamentale (elle conditionne, par exemple, la découverte de régularités). De l'autre parce que c'est l'une de plus étudiées.

Une représentation subjective

Rien n'est objectif dans ce monde, ne vous pensez pas au dessus de tout le monde. Utilisez votre subjectivité pour produire une représentation qui "claque" au lieu de la rendre lisse.

Les données

Nous travaillerons avec des données Eurostat, principalement des données de population au niveau NUTS 2 (le découpage officiel statistique européeen, qui possède 3 niveaux). Le tableau de données `NUTS2_year` possède une ligne par région NUTS 2 et par année d'observation:

```
## # A tibble: 9,019 x 15
##    id_anc année superficie comments population_femm~ population_homm~
##    <chr>  <int>      <dbl> <chr>               <int>            <int>
##  1 AT11    2015       3669 <NA>               147246           141110
##  2 AT12    2015      18917 <NA>               832975           803803
##  3 AT13    2015        395 <NA>               929704           867633
##  4 AT21    2015       9360 <NA>               286371           271270
##  5 AT22    2015      16251 <NA>               621265           600305
##  6 AT31    2015      11717 <NA>               727840           709411
##  7 AT32    2015       7050 <NA>               276378           262197
##  8 AT33    2015      12514 <NA>               370936           357890
##  9 AT34    2015       2534 <NA>               191814           186778
```

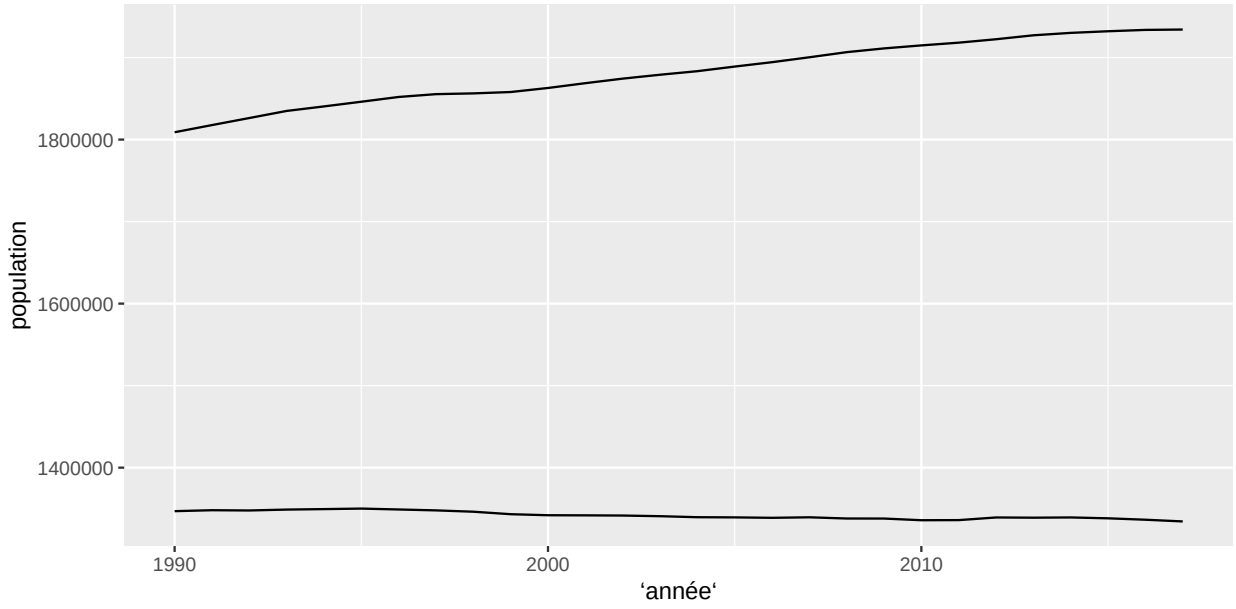
```
## 10 BE10    2015        161 <NA>               605416           578685
## # ... with 9,009 more rows, and 9 more variables: population <int>,
## #   population_0_19 <int>, population_20_59 <int>,
## #   population_60_plus <int>, id <chr>, nom_anc <chr>, nom <chr>,
## #   chgt <fct>, anc <list>
```

1. Faire un graphique avec ggplot2

1.1 Graphique basique

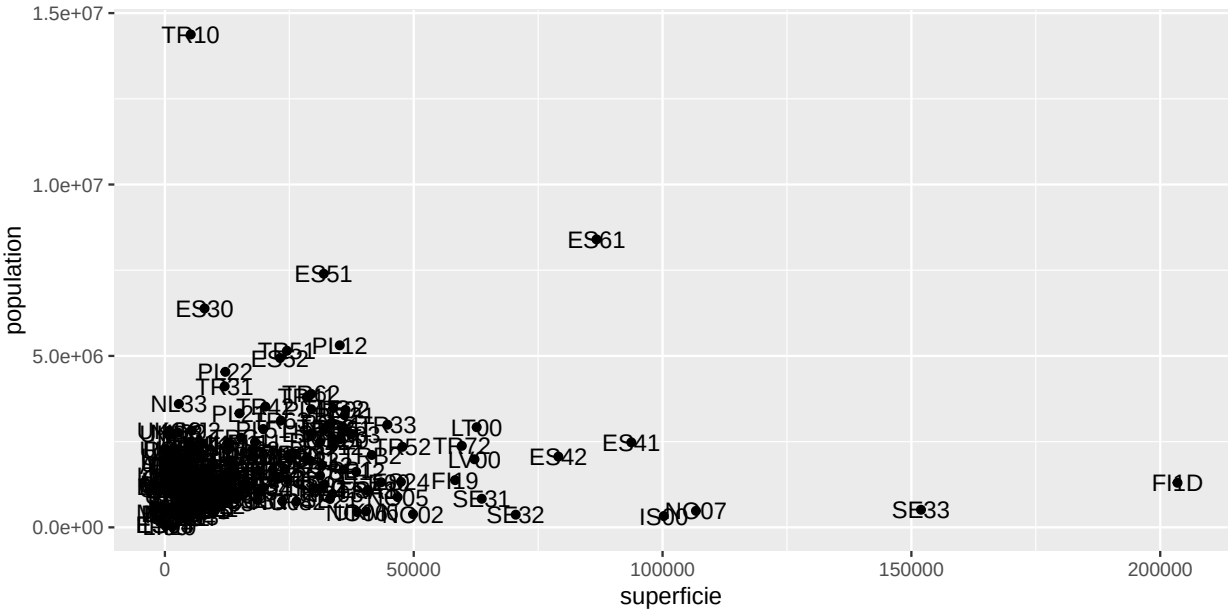
Commençons par un exemple: l'évolution de la population en Champagne-Ardenne et en Picardie entre 1990 et 2015.

```
NUTS2_year %>%
  filter(nom %in% c('Champagne-Ardenne', 'Picardie')) %>%
  ggplot(aes(x=année, y=population)) +
  geom_line(aes(group=nom))
```



Exercice 1.1.2 Il est possible de superposer plusieurs couches graphiques. Pouvez-vous rajouter des étiquettes à chaque observation? Notez le nom de la fonction que vous avez utilisé.

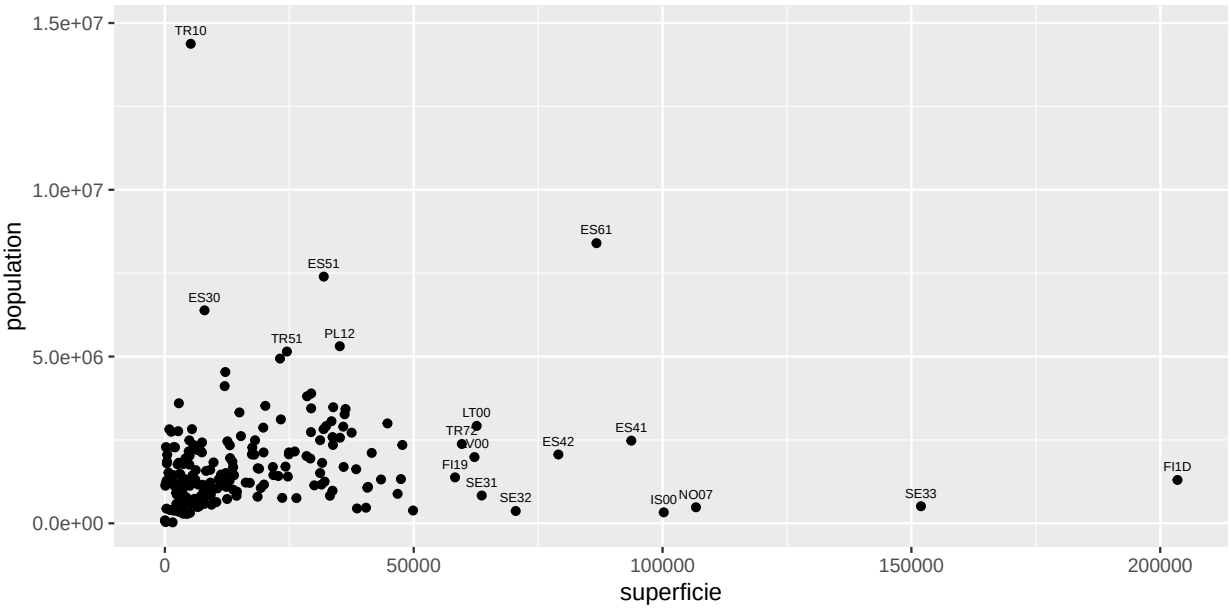
```
NUTS2_year %>%
  filter(année == 2015, !is.na(population), !is.na(superficie)) %>%
  ggplot(aes(x=superficie, y=population)) +
  geom_point() +
  geom_text(aes(label=id_anc))
```



Problème de ce graph : label sur les points, bouilli illisible vers l'origine

Solution :

```
NUTS2_year %>%
  filter(année == 2015, !is.na(population), !is.na(superficie)) %>%
  ggplot(aes(x=superficie, y=population)) +
  geom_point() +
  geom_text(
    # Je ne conserve que les données dans les zones les moins denses.
    data = . %>% filter(population>5000000 | superficie>50000),
    aes(label=id_anc),
    # Pour changer la taille du texte, pour des raisons arbitraires
    # et indépendantes des données, je place le paramètre graphique
    # **en dehors** de la fonction aes().
    size=2,
    # nudge_y permet de décaler le texte verticalement ;
    # la valeur est donnée en unités réelles (ici des habitants)
    nudge_y=400000
  )
```

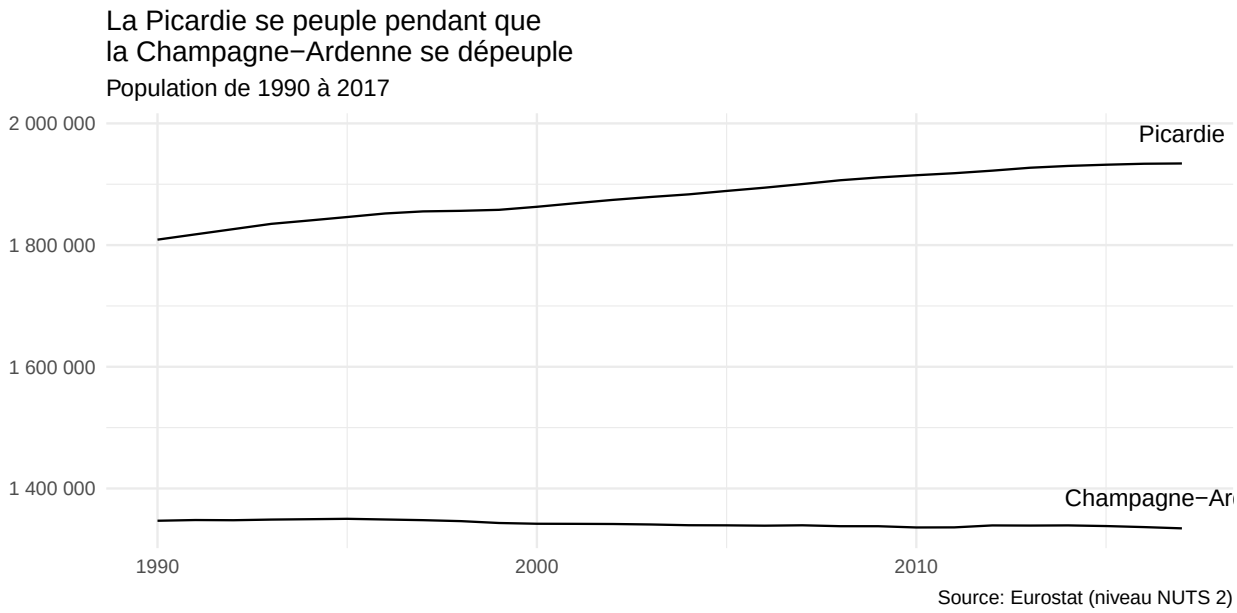


1.2 Options de présentation

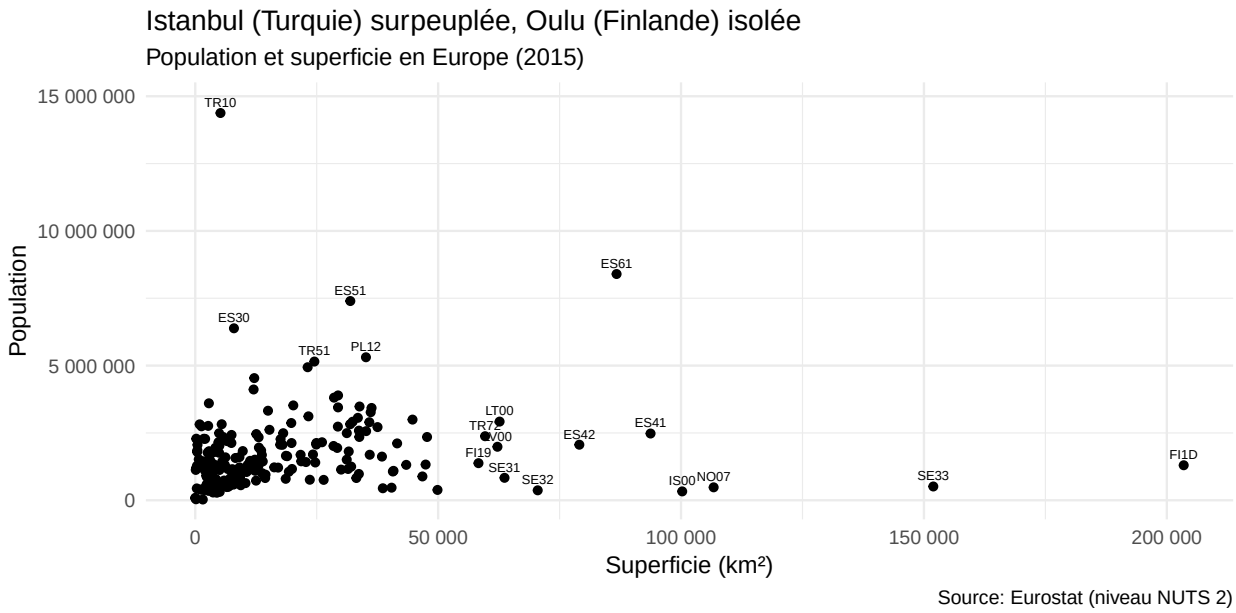
Le graphique Picardie—Champagne-Ardenne obtenu précédemment n'est pas satisfaisant:

- OÙ EST LE TITRE ???!!!
- C'EST QUOI LES LIGNES ???
- ECRITURE SCIENTIFIQUE ???
- ELLES VIENNENTS D'OÙ LES DONNEES ????
- Fond gris ?
- Les labels des axes pas forcément utile

```
NUTS2_year %>%
  filter(nom %in% c('Champagne-Ardenne', 'Picardie')) %>%
  ggplot(aes(x=année, y=population, group=nom)) +
  geom_line() +
  geom_text(
    data = . %>% filter (année == 2017)
    , aes(label=nom)
    , nudge_y=50000)+
  # changer le style de graphique
  theme_minimal() +
  # supprimer les titres des deux axes
  # utiliser un format plus lisible sur l'axe des ordonnées
  scale_x_continuous(name = NULL) +
  scale_y_continuous(name = NULL, labels = scales:::number)  +
  # ajouter titre et sous-titre
  labs(
    title    = "La Picardie se peuple pendant que\nla Champagne-Ardenne se dépeuple",
    subtitle = "Population de 1990 à 2017",
    caption = "Source: Eurostat (niveau NUTS 2)"
  )
```



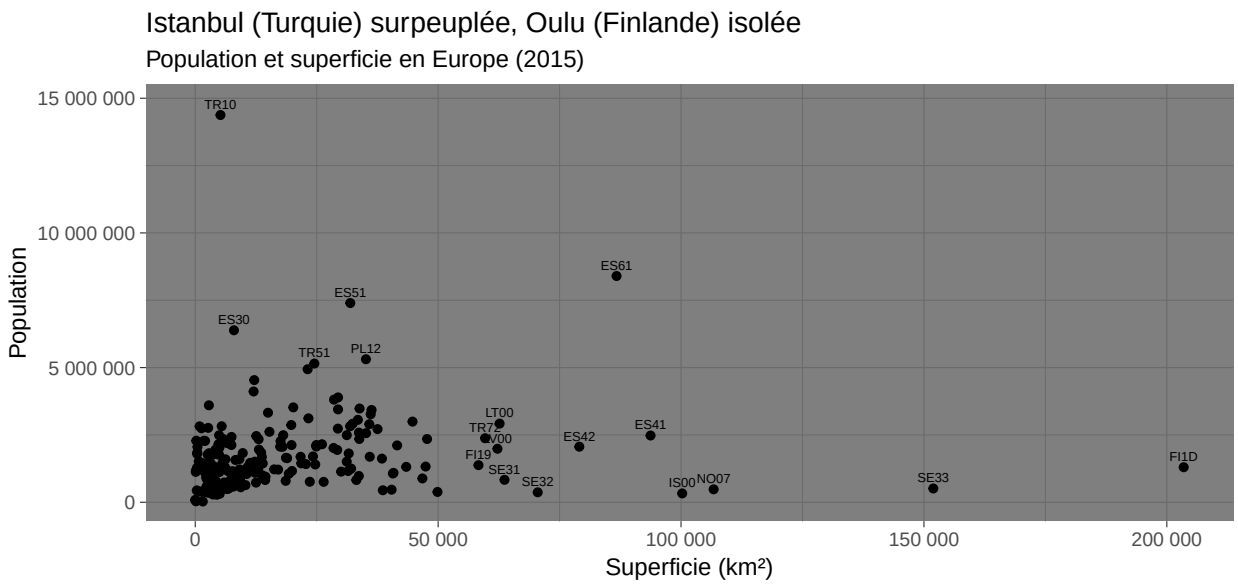

```
NUTS2_year %>%
  # pour cette partie du code, voir exercice bonus A
  filter(année == 2015, !is.na(population), !is.na(superficie)) %>%
  ggplot(aes(x=superficie, y=population)) +
  geom_point() +
  geom_text(
    data = . %>% filter(population>5000000 | superficie>50000),
    aes(label=id_anc),
    size=2,
    nudge_y=400000
  ) +
  # utiliser un format plus naturel (les titres des axes peuvent être spécifiés ici...)
  scale_x_continuous(name='Superficie (km²)', labels = scales:::number) +
  scale_y_continuous(labels = scales:::number) +
  # changer le style de graphique
  theme_minimal() +
  # ajouter titre, sous-titre...
  labs(
    y        = 'Population', # les titres des axes peuvent être spécifiés ici...
    title    = "Istanbul (Turquie) surpeuplée, Oulu (Finlande) isolée",
    subtitle = "Population et superficie en Europe (2015)",
    caption  = "Source: Eurostat (niveau NUTS 2)"
  )
```



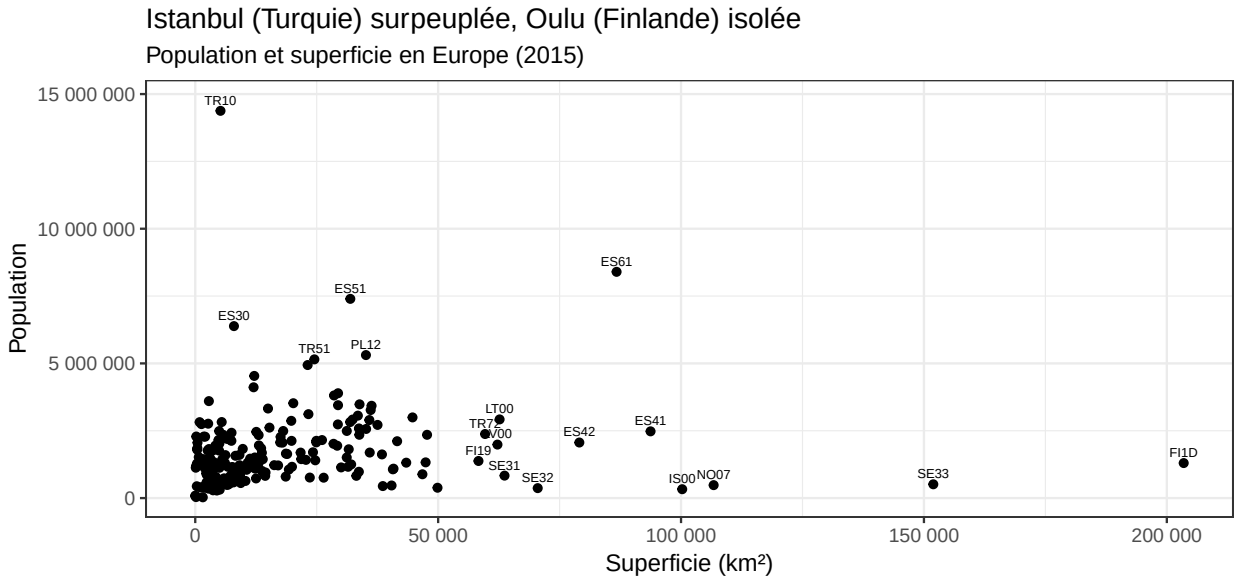
> N'oubliez pas, un grahique doit véhiculer un message. Si vous n'arrivez pas à en trouver un alors le graphique est sûrement inutile !

Quelques exemples de thème

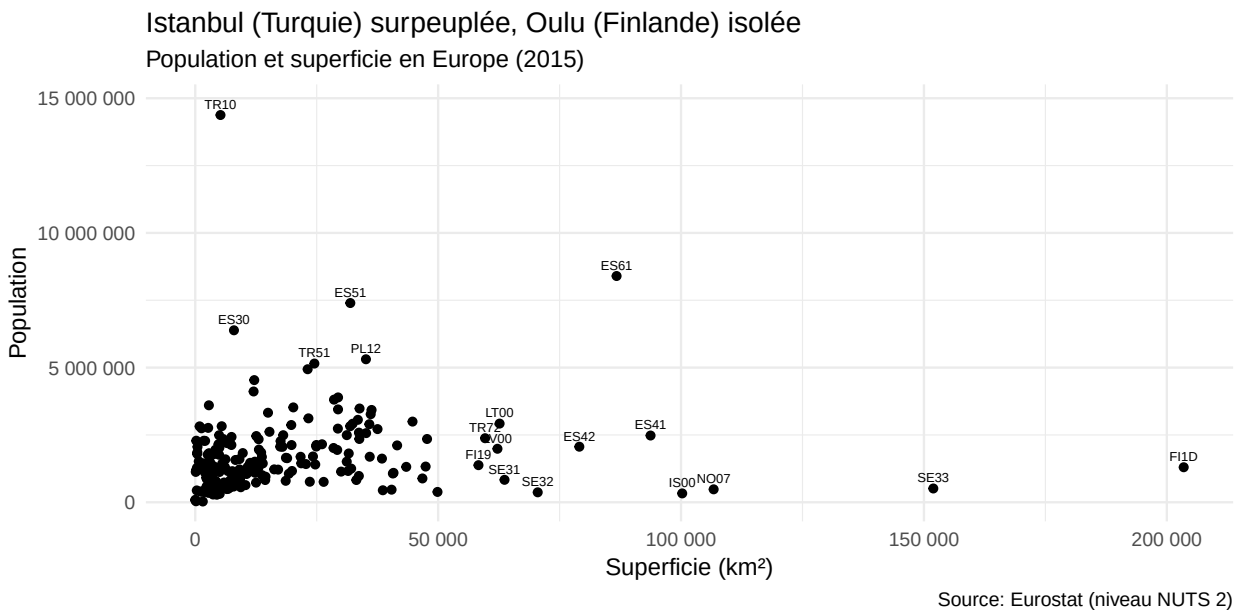
```
plot <- NUTS2_year %>%
  # pour cette partie du code, voir exercice bonus A
  filter(année == 2015, !is.na(population), !is.na(superficie)) %>%
  ggplot(aes(x=superficie, y=population)) +
  geom_point() +
  geom_text(
    data = . %>% filter(population>5000000 | superficie>50000),
    aes(label=id_anc),
    size=2,
    nudge_y=400000
  ) +
  # utiliser un format plus naturel (les titres des axes peuvent être spécifiés ici...)
  scale_x_continuous(name='Superficie (km²)', labels = scales:::number) +
  scale_y_continuous(labels = scales:::number) +
  # changer le style de graphique
  # ajouter titre, sous-titre...
  labs(
    y        = 'Population', # les titres des axes peuvent être spécifiés ici...
    title    = "Istanbul (Turquie) surpeuplée, Oulu (Finlande) isolée",
    subtitle = "Population et superficie en Europe (2015)",
    caption  = "Source: Eurostat (niveau NUTS 2)"
  )

plot + theme_dark()
```



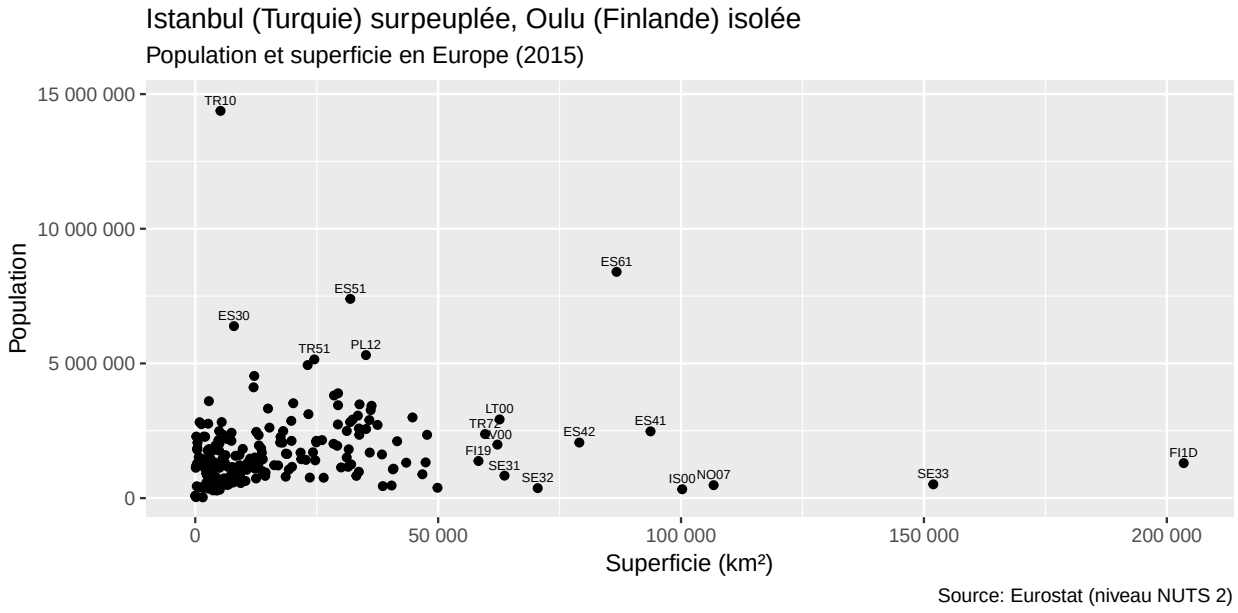

```
plot + theme_bw()
```



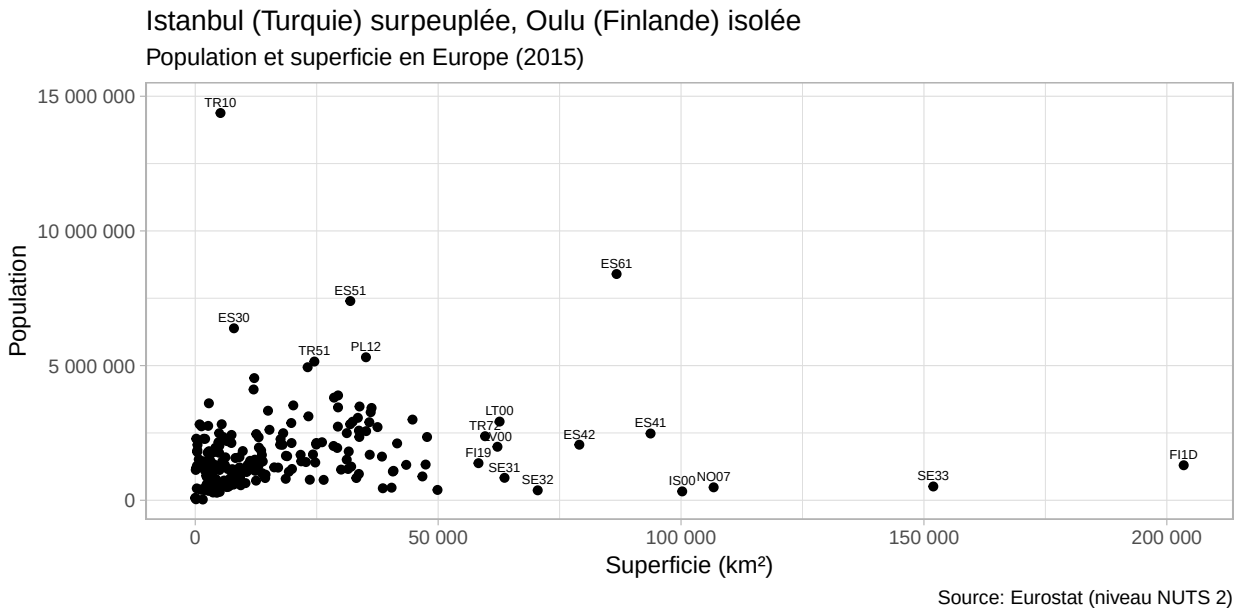

```
plot + theme_minimal()
```



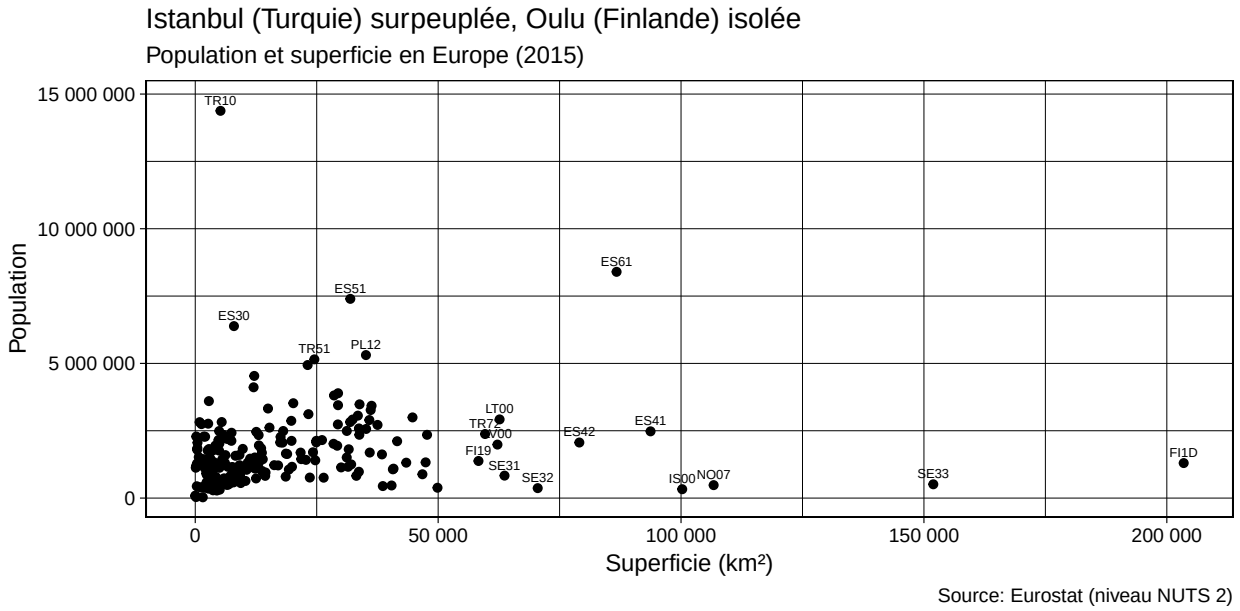

```
plot + theme_gray()
```



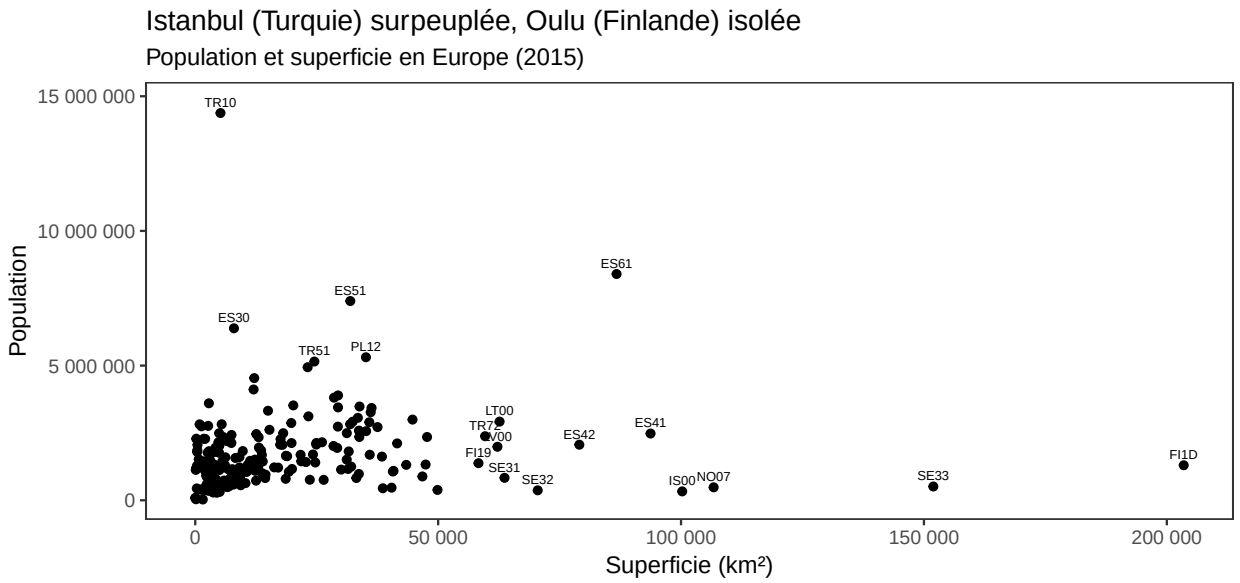

```
plot + theme_light()
```



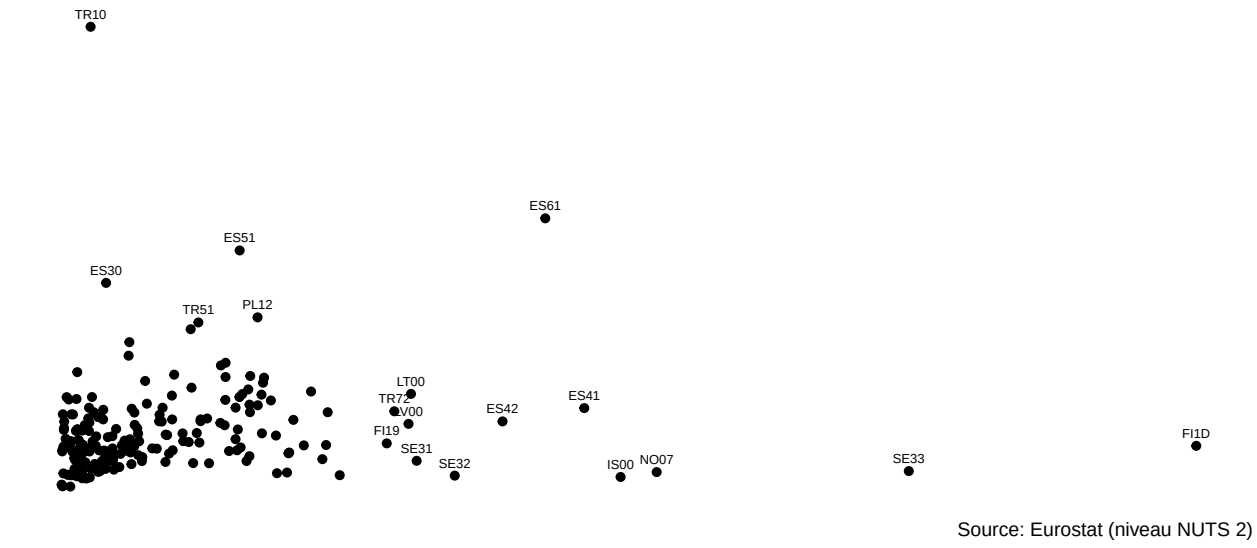

```
plot + theme_linedraw()
```




```
plot + theme_test()
```




```
plot + theme_void()
```



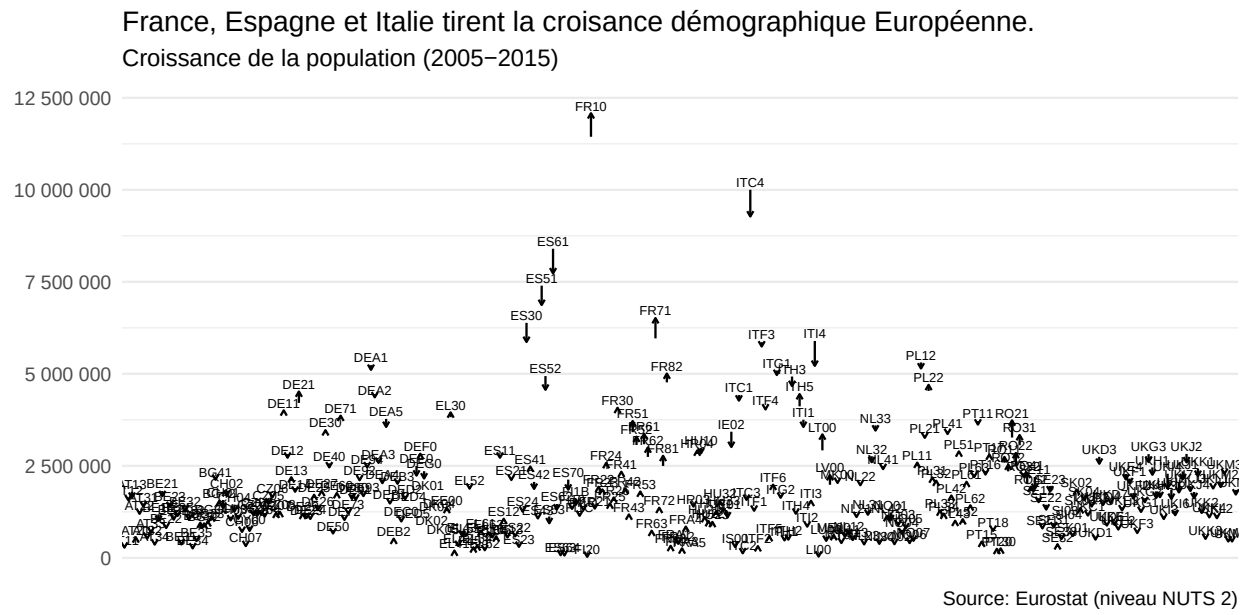
2. Représenter des données continues

2.1 Longueurs

Commentaire directement sur le code

```
NUTS2_year %>%
  # pré-traitement
  filter(!str_detect(id_anc, '^TR')) %>% # suppression des données turques (expression régulière)
  filter(année %in% c(2005,2015), !is.na(population)) %>%
  # graphique (cœur)
  ggplot(aes(x=id_anc, y=population)) +
```

```r
  geom_line(arrow = arrow(length = unit(0.1, "cm"))) +
  # ajouter des étiquettes
  geom_text(
    # Si je ne change pas les données en entrée, chaque étiquette est imprimée
    # pour chaque observations, soit deux fois pour chaque région (2005 et 2015).
    # Je décide de ne retenir pour chaque region NUTS 2 que la plus grande
    # population atteinte, de façon à que ce que l'étiquette soit toujours
    # au-dessus de la flèche.
    data = . %>%
      group_by(id_anc) %>%
      summarise(population=max(population)),
    aes(label=id_anc),
    size=2, # diminuer la taille de la police sans référence aux données
    nudge_y=200000 # écarter le label des flèches
  ) +
  # graphique (mise en page)
  scale_x_discrete(name=NULL, breaks=NULL) + # break les identifiants NUTS 2 n'étant pas des grandeurs
#  scale_x_discrete(name=NULL) +
  scale_y_continuous(name=NULL, labels = scales:::number) +
  theme_minimal() +
  labs(
    title    = "France, Espagne et Italie tirent la croisance démographique Européenne.",
    subtitle = "Croissance de la population (2005-2015)",
    caption = "Source: Eurostat (niveau NUTS 2)"
  )
```



Source: Eurostat (niveau NUTS 2)

Exercice 2.1.6 La croissance démographique est-elle plus grande en île de France (`FR10`) ou à Rome (`ITI4`)?

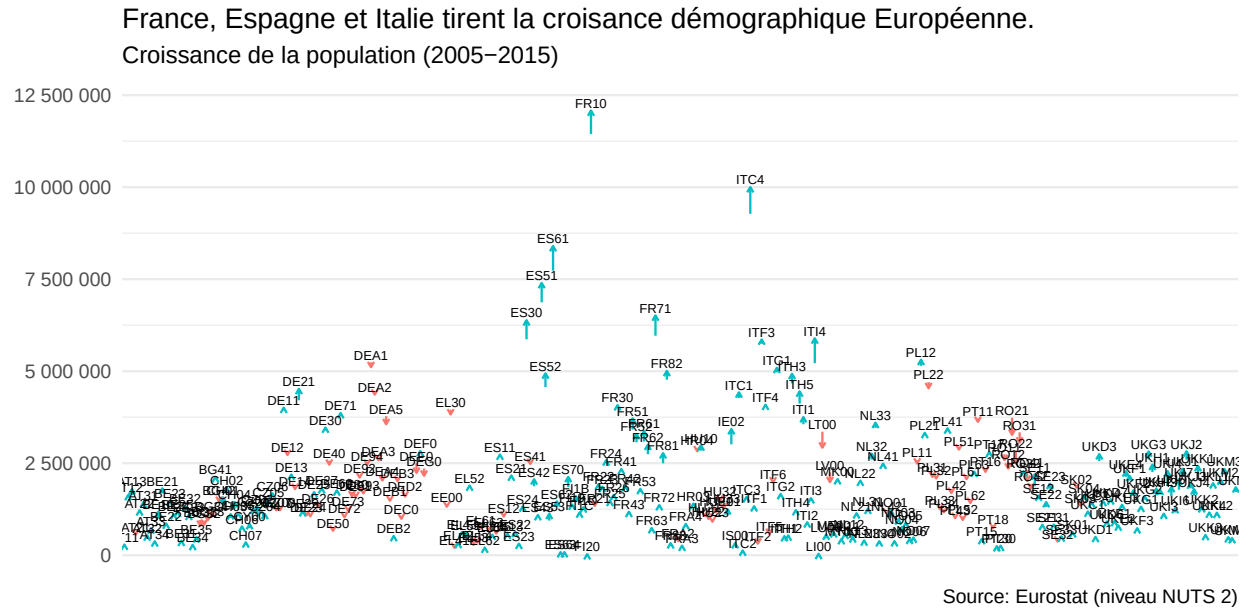
C'est chaud à lire … Plusieurs raisons. Déjà il faut trouver les deux zones en question (classemement des flèched dans l'ordre alphabétique). Ensuite le diagramme nous propose 2 infos, la population et la croissance. Et cela cause un tassement sur les "croissances". Personnellement j'arrive globalement à dicerner 3 grands niveaux de croissance (6 si on compte positif/négatif), mais pas à comparer deux valeurs de niveau équivalent. Cette représentation est donc mauvaise !

Colorez la flèche en fonction de si la croissance et positive ou négative.

```
NUTS2_year %>%
  filter(!str_detect(id_anc, '^TR')) %>%
  filter(année %in% c(2005, 2015)) %>%
  arrange(année) %>%
  group_by(id_anc) %>% mutate(
    max      = max(population),
    increase = first(population)<last(population)
  ) %>% ungroup %>%
  ggplot(aes(x=id_anc, y=population)) +
  geom_line(
    aes(color=increase),
    arrow = arrow(length = unit(0.1, "cm"))
  ) +
  geom_text(
    data = . %>% group_by(id_anc) %>% slice(1),
    aes(y=max, label=id_anc),
    size=2,         # diminuer la taille de la police
    nudge_y=200000 # écarter l'étiquette de l'extrémité de la flèche
  ) +
  scale_x_discrete(name=NULL, breaks=NULL) +
  scale_y_continuous(name=NULL, labels = scales:::number) +
  guides(color=FALSE) + # suppression de la légende
  theme_minimal() +
  labs(
    title    = "France, Espagne et Italie tirent la croisance démographique Européenne.",
    subtitle = "Croissance de la population (2005-2015)",
    caption = "Source: Eurostat (niveau NUTS 2)"
  )
```

```
## Warning: Removed 11 rows containing missing values (geom_path).
```

```
## Warning: Removed 11 rows containing missing values (geom_text).
```



Source: Eurostat (niveau NUTS 2)

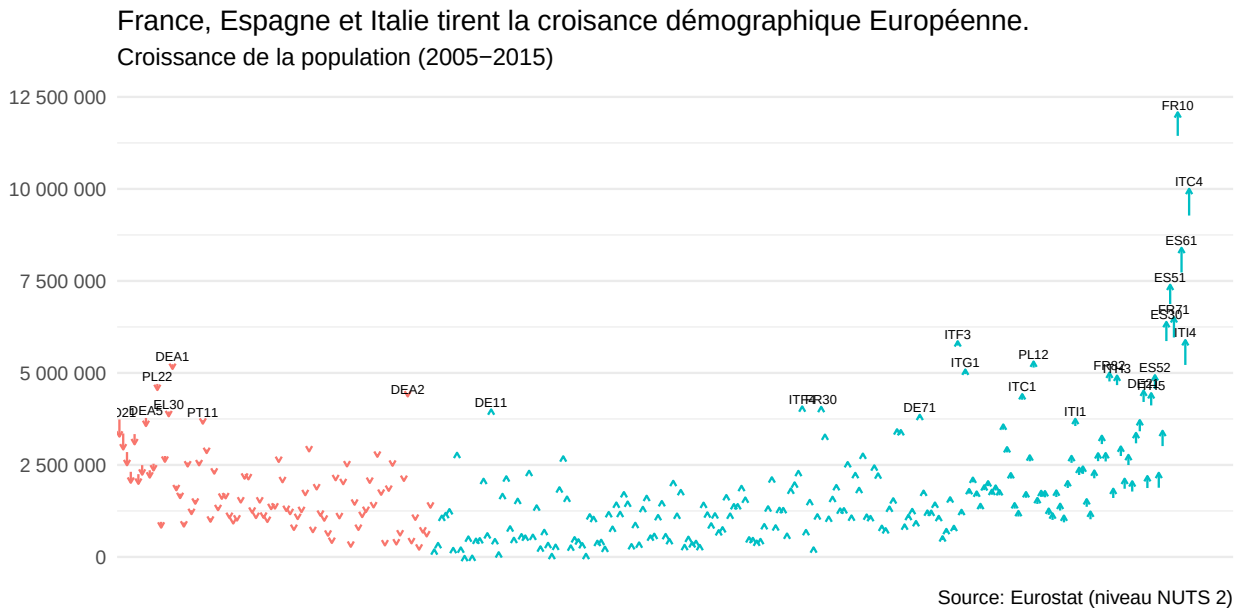
Correction: On va classer par origine pour lire la réponse. Mais bon, si le moyen le plus efficace pour

comparer deux croissances (variable que l'on représente sur l'axe des ordonnées) est sa position sur l'axe des abscisses c'est qu'il y a un problème !!!!

Pour une meilleure lecture je retire également des labels

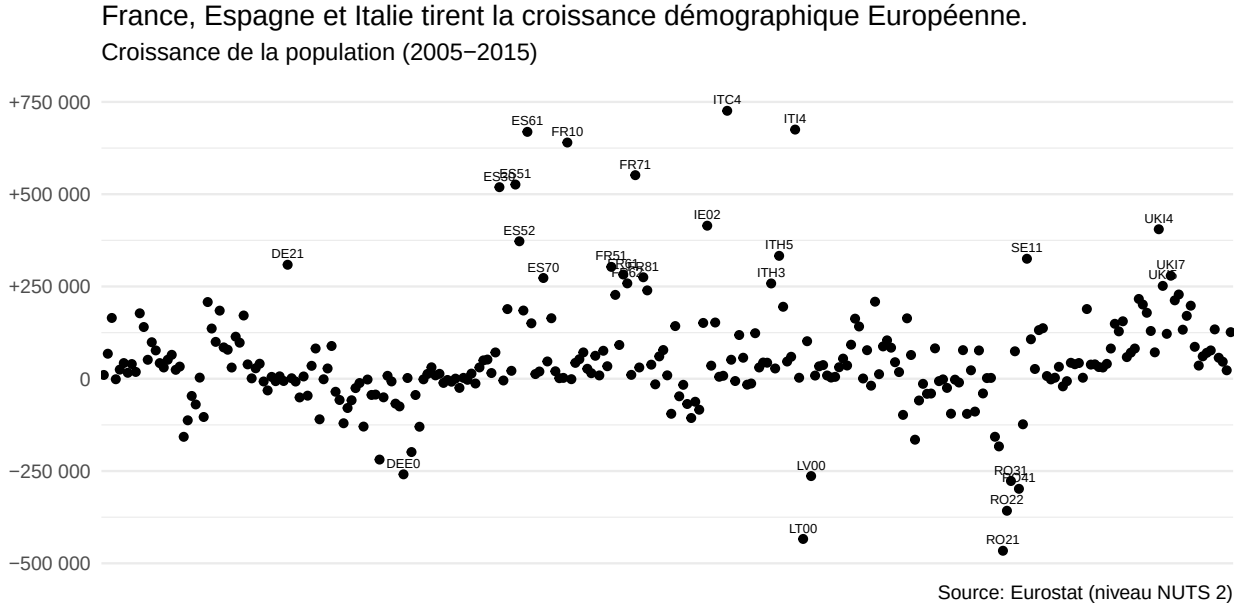
```
NUTS2_year %>%
  filter(!str_detect(id_anc, '^TR')) %>%
  filter(année %in% c(2005, 2015)) %>%
  arrange(année) %>%
  group_by(id_anc) %>% mutate(
    max      = max(population),
    increase = first(population)<last(population)
  ) %>% ungroup %>%
  # passer par reorder() pour choisir l'ordre d'apparition des modalités
  mutate(
    # x_min    = reorder(id_anc, population, min),
    # x_max    = reorder(id_anc, population, function(x) max(x)),
    id_anc  = reorder(id_anc, population, function(x) last(x)-first(x))
  ) %>%
  ggplot(aes(x=id_anc, y=population)) +
  geom_line(aes(color=increase), arrow = arrow(length = unit(0.1, "cm"))) +
  geom_text(
    data = . %>% group_by(id_anc) %>% slice(1) %>% filter (population > 3500000),
    aes(y=max, label=id_anc),
    size=2,
    nudge_y=200000
  ) +
  scale_x_discrete(name=NULL, breaks=NULL) +
  scale_y_continuous(name=NULL, labels = scales:::number) +
  guides(color=FALSE) +
  theme_minimal() +
  labs(
    title    = "France, Espagne et Italie tirent la croisance démographique Européenne.",
    subtitle = "Croissance de la population (2005-2015)",
    caption = "Source: Eurostat (niveau NUTS 2)"
  )
```

```
## Warning: Removed 11 rows containing missing values (geom_path).
```



2.2 Position

Exercice 2.2.1 : j'ai également fait un peu de vide dans les labels

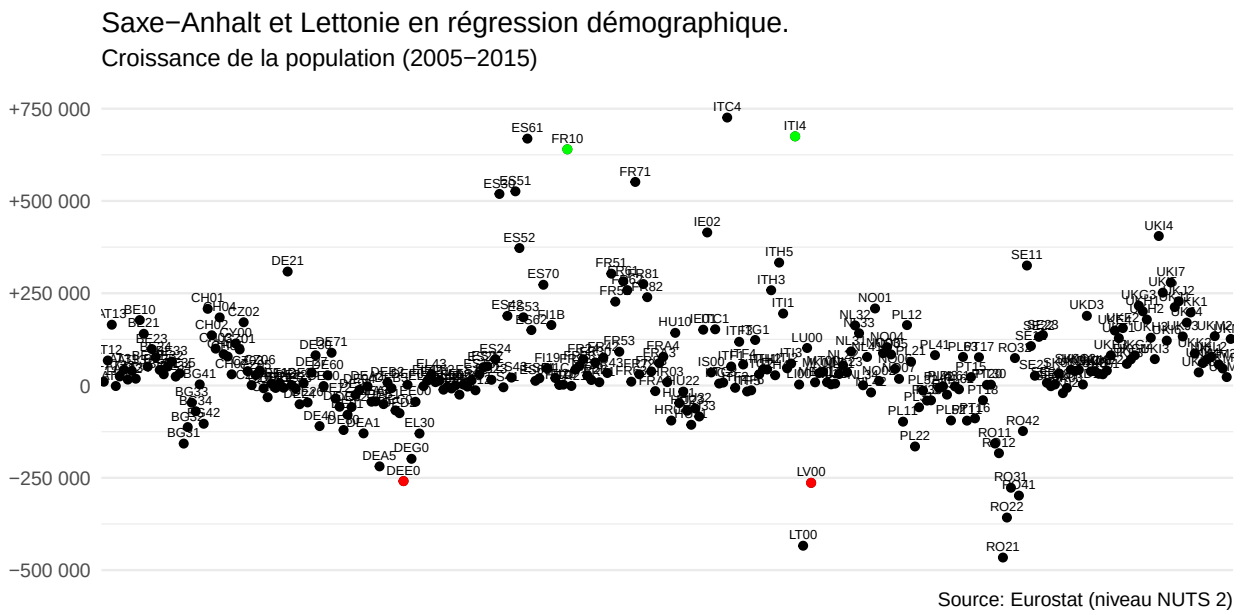


Exercice 2.2.2 La perte démographique est-elle plus grande au Saxe-Anhalt (`DEE0`) ou en Lettonie (`LV00`)?

¯_( )_/¯, mais comparer Paris et Rome c'est possible maintenant ø/

```
NUTS2_year %>%
  filter(année %in% c(2005, 2015)) %>%
  # calculer de la croissance démographique
  group_by(id_anc) %>% summarize(
    croissance = population[année=='2015']-population[année=='2005']
  ) %>%
  # supprimer les valeurs manquantes
  filter(!is.na(croissance)) %>%
```

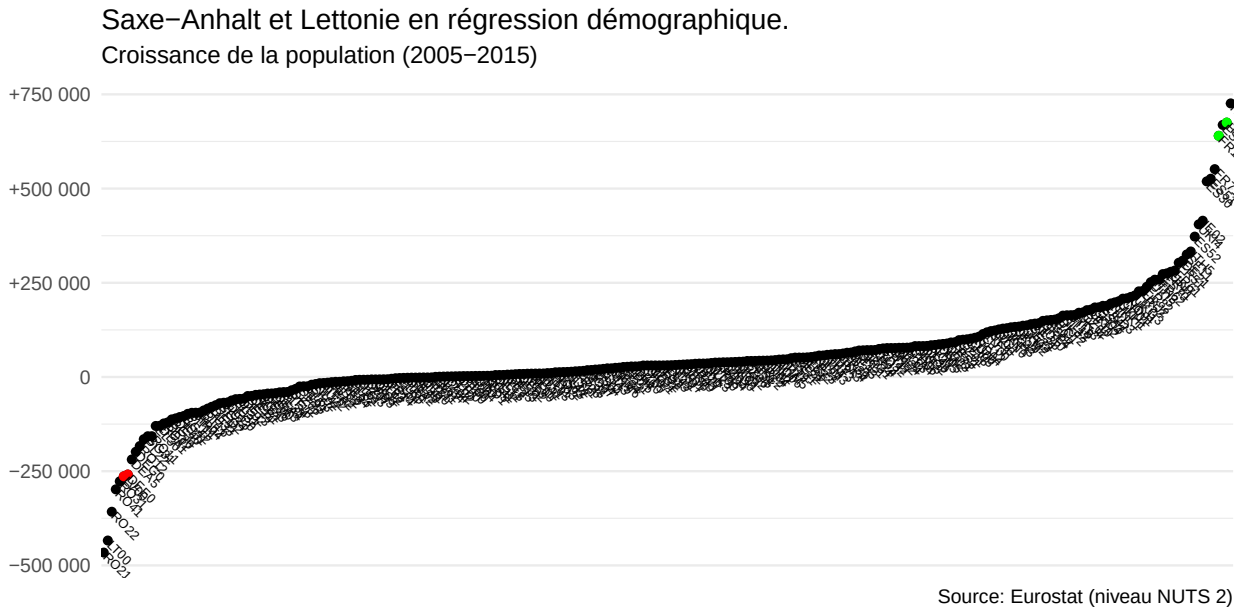
```
  # produire le graphique
  ggplot(aes(x=id_anc, y=croissance)) +
  geom_point() +
  geom_text(
    aes(label = id_anc),
    size=2, nudge_y=30000
  ) +
  geom_point(data = . %>% filter(id_anc %in% c("DEE0", "LV00")), color='red') +
  geom_point(data = . %>% filter(id_anc %in% c("FR10", "ITI4")), color='green') +
  # simplifier la mise en page
  scale_x_discrete(  name=NULL, breaks=NULL) +
  scale_y_continuous(name=NULL, labels = number_plus) +
  theme_minimal() +
  labs(
    title    = "Saxe-Anhalt et Lettonie en régression démographique.",
    subtitle = "Croissance de la population (2005-2015)",
    caption  = "Source: Eurostat (niveau NUTS 2)"
  )
```



Réorganisons tout cela !

```
NUTS2_year %>%
 filter(année %in% c(2005, 2015)) %>%
 group_by(id_anc) %>% summarize(
   croissance = population[année=='2015']-population[année=='2005']
 ) %>%
  #Réorganisation des données
 mutate(id_anc = reorder(id_anc, croissance)) %>%
 filter(!is.na(croissance)) %>%
 ggplot(aes(x=id_anc, y=croissance)) +
 geom_point() +
 geom_point(data = . %>% filter(id_anc %in% c("DEE0", "LV00")), color='red') +
 geom_point(data = . %>% filter(id_anc %in% c("FR10", "ITI4")), color='green') +
# étiquettes en biais. C'est plus pour le "fun" car c'est toujours illisibles ...
```

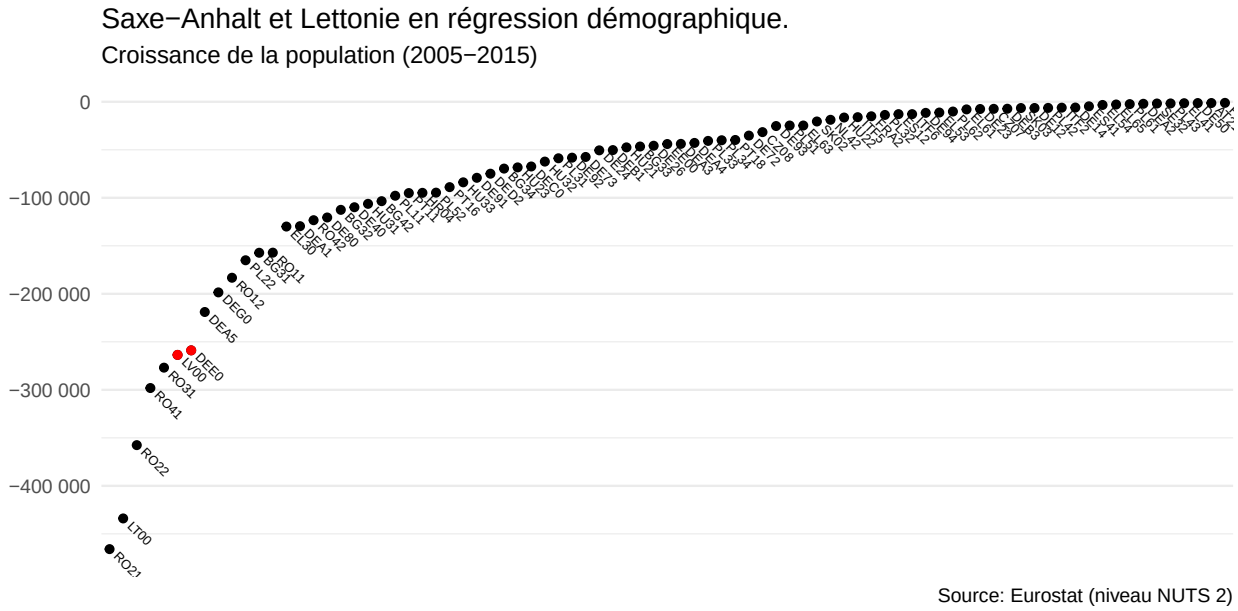
```
  geom_text(
    aes(label = id_anc),
    size=2, nudge_y=-5000, nudge_x=0.5, angle=-45, hjust=0
  ) +
  scale_x_discrete(  name=NULL, breaks=NULL) +
  scale_y_continuous(name=NULL, labels = number_plus) +
  theme_minimal() +
  labs(
    title    = "Saxe-Anhalt et Lettonie en régression démographique.",
    subtitle = "Croissance de la population (2005-2015)",
    caption  = "Source: Eurostat (niveau NUTS 2)"
  )
```



Bon si le graphique était plus grand on arriverait sûrement à se prononcer, mais actuellement c'est chaud de savoir qui de Saxe-Anhalt ou de la Lettonie à plus plus grande perte de population. Pour répondre à la question je vais faire un peu de trie dans les données

```
NUTS2_year %>%
 filter(année %in% c(2005, 2015)) %>%
 group_by(id_anc) %>% summarize(
   croissance = population[année=='2015']-population[année=='2005']
 ) %>%
  #Réorganisation des données
 mutate(id_anc = reorder(id_anc, croissance)) %>%
 filter(!is.na(croissance), croissance < 0) %>%
 ggplot(aes(x=id_anc, y=croissance)) +
 geom_point() +
 geom_point(data = . %>% filter(id_anc %in% c("DEE0", "LV00")), color='red') +
 geom_point(data = . %>% filter(id_anc %in% c("FR10", "ITI4")), color='green') +
# étiquettes en biais. C'est plus pour le "fun" car c'est toujours illisibles ...
 geom_text(
   aes(label = id_anc),
   size=2, nudge_y=-5000, nudge_x=0.5, angle=-45, hjust=0
 ) +
```

```
scale_x_discrete(  name=NULL, breaks=NULL) +
scale_y_continuous(name=NULL, labels = number_plus) +
theme_minimal() +
labs(
  title    = "Saxe-Anhalt et Lettonie en régression démographique.",
  subtitle = "Croissance de la population (2005-2015)",
  caption  = "Source: Eurostat (niveau NUTS 2)"
)
```



Et voilà la réponse est que c'est la Lettonie qui a perdu le plus d'habitant

2.3 Longueurs + position

Exercice 2.3.1 Changez une ligne du code de l'exercice 2.2.1 pour maintenant représenter la croissance démographiques à l'aide d'un diagramme en barres.

- *Par soucis de lisibilité, vous pouvez également retirer les régions de croissance démographique proche de zéro.*

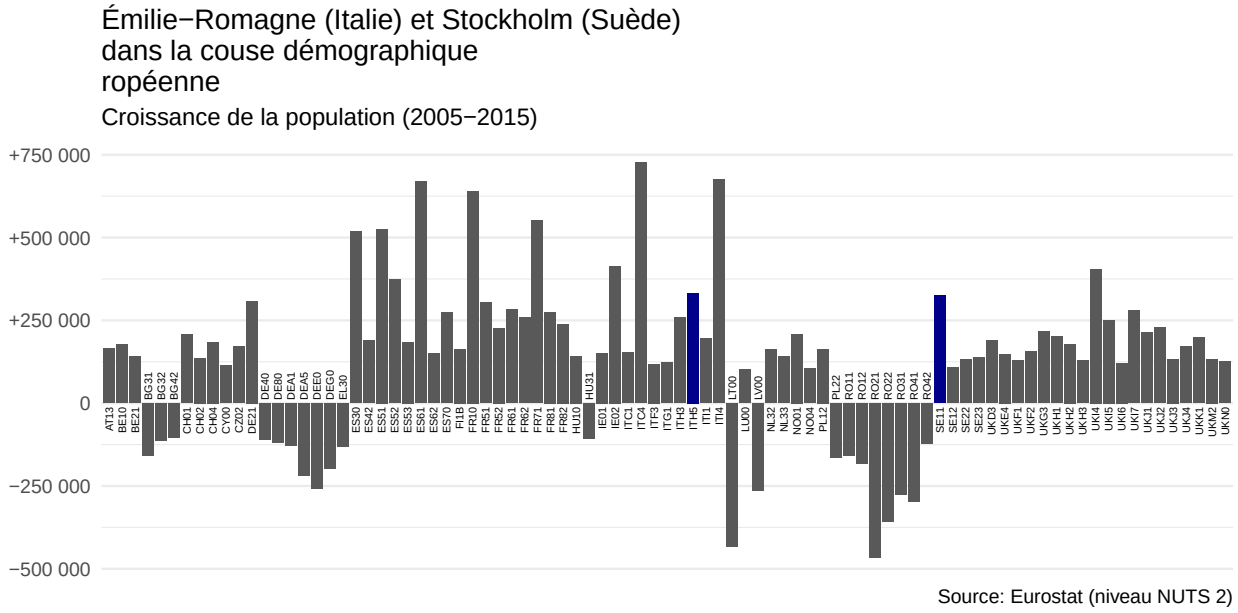
```
NUTS2_year %>%
  filter(année %in% c(2005, 2015)) %>%
  group_by(id_anc) %>% summarize(
    croissance = population[order(année)] %>% {last(.)-first(.)}
  ) %>%
  filter(!is.na(croissance), abs(croissance) > 100000) %>% # <-- FILTRE CROIS.=0
  ggplot(aes(x=id_anc, y=croissance)) +
  geom_col() + # <---------------------------------------- DIAGRAMME EN BARRES
  geom_text(
    aes(
      label = id_anc,
    # V------------------------------------------------------ ÉTIQUETTES LISIBLES
      y     = ifelse(croissance>0, -10000, 10000),
      hjust = ifelse(croissance>0, 1, 0)
    ),
    size=1.8, angle = 90) +
```

```
    # ------------------------------------------------------------------------X
  scale_x_discrete(  name=NULL, breaks=NULL) +
  scale_y_continuous(name=NULL, labels = number_plus) +
  theme_minimal() +
  labs(
    title    = "France, Espagne et Italie tirent la croissance démographique Européenne.",
    subtitle = "Croissance de la population (2005-2015)",
    caption  = "Source: Eurostat (niveau NUTS 2)"
  )
```

Il faut utiliser `geom_col`, et non `geom_bar`. Car avec `geom_bar`, `ggplot2` suppose que l'utilisateur souhaite une sorte d'histogramme. `geom_bar` va donc tenter de compter le nombre de fois où `id_anc` apparaît dans les données pour passer ce nombre en ordonnées. Il est possible d'empêcher ce comportement avec l'option `stat='identity'`.

Exercice 2.3.2 La croissance démographique est-elle plus grande en Émilie-Romagne (`ITH5`) ou à Stockholm `SE11`?

Meh ¯_( )_/¯la croissance a l'air plus forte en Émilie-Romagne, mais c'est dur à dire à cause de la distance entre les barres.



Un peu d'interactivité ici sera assez bien pour obtenir directement la différence de croissance. Et elle parait beaucoup plus naturelle ici (selon moi) que dans les digrammes précédents.

Exercice 2.3.3 Rajouter une ligne au code suivant pour réorganiser les barres par ordre croissant (ou décroissant). Est-il plus facile de répondre?

```
NUTS2_year %>%
  filter(année %in% c(2005, 2015)) %>%
  group_by(id_anc) %>% summarize(
    croissance = population[order(année)] %>% {last(.)-first(.)}
  ) %>%
  mutate(id_anc = reorder(id_anc, croissance)) %>%
  filter(!is.na(croissance), abs(croissance) > 100000) %>%
  ggplot(aes(x=id_anc, y=croissance)) +
  geom_col() +
```

```
  geom_col(data=.%>% filter(id_anc %in% c('SE11','ITH5')), fill='darkblue') +
  geom_text(
    aes(
      label = id_anc,
      y     = ifelse(croissance>0, -10000, 10000),
      hjust = ifelse(croissance>0, 1, 0)
    ),
    size=1.8, angle = 90) +
  scale_x_discrete(  name=NULL, breaks=NULL) +
  scale_y_continuous(name=NULL, labels = number_plus) +
  theme_minimal() +
  labs(
    title    = "Émilie-Romagne (Italie) et Stockholm (Suède)\ndans la course démographique Européenne",
    subtitle = "Croissance de la population (2005-2015)",
    caption  = "Source: Eurostat (niveau NUTS 2)"
  )
```

Bon pareil on triche un peu car on utilise ici encore une fois l'ordre pour se prononcer, mais des 3 représentations présentée, celle ci est clairement la meilleure !

Pour le fun je vous propose de rajouter un peu de couleur par pays. Mais ce n'est pas une bonne idée en fait ^^"

```
NUTS2_year %>%
  filter(année %in% c(2005, 2015)) %>%
  group_by(id_anc) %>% summarize(
    croissance = population[order(année)] %>% {last(.)-first(.)}
  ) %>%
  mutate(id_anc = reorder(id_anc, croissance)) %>%
  filter(!is.na(croissance), abs(croissance) > 100000) %>%
  ggplot(aes(x=id_anc, y=croissance)) +
  geom_col() +
  geom_col( aes(fill = substr(id_anc,1,2))) +
  geom_text(
    aes(
      label = id_anc,
      y     = ifelse(croissance>0, -10000, 10000),
      hjust = ifelse(croissance>0, 1, 0)
    ),
    size=1.8, angle = 90) +
  scale_x_discrete(  name=NULL, breaks=NULL) +
  scale_y_continuous(name=NULL, labels = number_plus) +
  scale_fill_discrete(name="code pays" ) +
  theme_minimal() +
  labs(
    title    = "Émilie-Romagne (Italie) et Stockholm (Suède)\ndans la course démographique Européenne",
    subtitle = "Croissance de la population (2005-2015)",
    caption  = "Source: Eurostat (niveau NUTS 2)"
  )
```