

EMSBD 6 - Apprentissage non-supervisé

Paris et sa sphère d'influence

Bruno KUBECZKA

31 mars 2024

Ce document a pour objet l'étude de l'évolution de la population à Paris et dans ses 3 couronnes de départements de 1780 à 2022.

Sommaire

1	Introduction	3
2	Préparation des données	4
2.1	Chargement	4
2.2	Extraction de l'historique de population de tous les départements	5
2.3	Extraction de l'historique de population de la sphère parisienne	6
3	1ère approche des données	8
3.1	Population française	8
3.2	Population parisienne	9
3.3	Population des couronnes	10
4	Stratégie de l'étude	14
5	Profilage des départements	15
5.1	Mise à l'échelle min-max	15
5.2	Visualisation des profils temporels	16
5.3	Regroupement des profils	18
5.3.1	Clustering hiérarchique	18
5.3.2	Visualisation des groupes	20
5.3.3	Profils moyens	22
5.4	Projection	25
5.4.1	ACP	25
5.4.2	isomap	27
6	Identification des moments clés dans la population parisienne	29
6.1	Préparation des données	29
6.2	Détection des points de rupture (PELT pénalité CROPS)	30
6.3	Projection des moments clés sur les profils temporels moyens	36
7	Conclusion	39
8	Références	39

1 Introduction

L’objet de cette étude est de mettre en évidence d’éventuels **phénomènes de population entre Paris et sa sphère d’influence**.

Par **sphère d’influence**, on entend les **3 couronnes de départements** au sens administratif tel que défini par l’INSEE à savoir

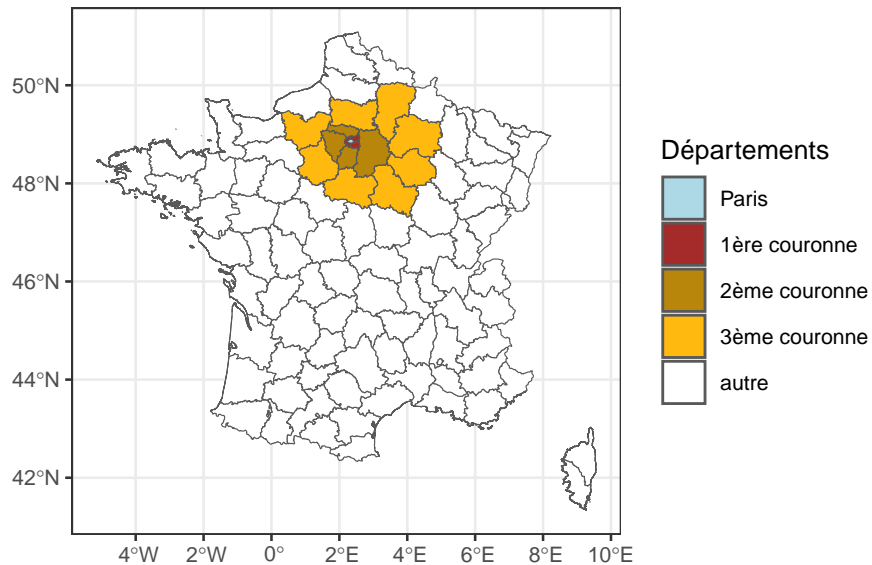
- La **1ère couronne** (petite couronne)
 - 92 : Hauts-de Seine
 - 93 : Seine-Saint-Denis
 - 94 : Val-de-Marne
- La **2ème couronne** (grande couronne)
 - 77 : Seine-et-Marne
 - 78 : Yvelines
 - 91 : Essonne
 - 95 : Val-d’Oise
- La **3ème couronne**
 - 02 : Aisne
 - 10 : Aube
 - 27 : Eure
 - 28 : Eure-et-Loir
 - 45 : Loiret
 - 51 : Marne
 - 60 : Oise
 - 89 : Yonne

Dans les chapitres suivants, on se propose

- de vérifier si la notion de couronnes telle que définie par l’INSEE est cohérente et homogène au sens de la population
- d’identifier les mouvements de population historique au sein la sphère d’influence et de les rapprocher le cas échéant d’événements historiques sous-jacents

Sphère d'influence parisienne

Paris et ses 3 couronnes



2 Préparation des données

Les données travaillées sont les **recensements de population des départements métropolitains de 1780 à 2022**.

Elles sont issues du site [une histoire du conflit politique](https://unehistoireduconflitpolitique.fr/).

💡 Source des données

source : <https://unehistoireduconflitpolitique.fr/telecharger.html>

section : Taille d'agglomération et de commune > Base de données en format CSV

fichier : *popdepartements.csv* (130 Mo)

2.1 Chargement

```
file = file.path(DATA_DIR, POP_DEPARTEMENTS_CSV)
df_pop_departements_raw = read.csv(file)
```

```
# dimensions du jeu de données
dim(df_pop_departements_raw)
```

[1] 96 1219

Les données se présentent sous la forme d'une table composée

- des **96 départements métropolitains** en ligne, indexés par leur numéro et leur nom (champs `dep` et `nomdep`)
- de **1219 colonnes** dont des colonnes au nom sous la forme `popAAAA` où `AAAA` est une année comprise entre 1780 et 2022 (soit 243 colonnes pertinentes). Ces colonnes sont les **recensements de population du département correspondant**.

2.2 Extraction de l'historique de population de tous les départements

Du jeu de données brut, on extrait dans `df_pop_tous_departements` les **populations de 1780 à 2022** pour **TOUS les départements**, en maintenant les 2 colonnes descriptives :

- `dep` contient le **code du département** au format factoriel
- `nomdep` contient le **nom du département** au format factoriel

```
# dataframe : population de tous les départements
# extraction des colonnes
# . dep : numéro de département
# . nomdep : nom des départements
# . toutes colonnes popxxx : population de 1780 à 2022

df_pop_tous_departements = df_pop_departements_raw %>%
  dplyr::select(c("dep", "nomdep"), matches("^pop[0-9]{4}$") )

# Indexation des lignes par le code département
row.names(df_pop_tous_departements) = df_pop_tous_departements$dep

# colonnes : retrait du préfix pop devant les années
colnames(df_pop_tous_departements) <- gsub('^pop', '',
                                           colnames(df_pop_tous_departements))

# catégorisation des colonnes dep et nomdep
df_pop_tous_departements$dep = as.factor(df_pop_tous_departements$dep)
df_pop_tous_departements$nomdep = as.factor(df_pop_tous_departements$nomdep)

dim(df_pop_tous_departements)
```

[1] 96 245

```
# Décompte des Données manquantes dans le dataframe
df_pop_tous_departements %>%
  is.na() %>%
  sum()
```

```
[1] 0
```

`df_pop_tous_departements` contient donc le recensement de population des **96 départements métropolitains** sur **243 ans** (de 1780 à 2022).

Les lignes sont indexées par le **code de département**.

Les colonnes sont indexées par

- `dep` code de département
- `nomdep` nom de département
- `AAAA` année de l'observation

il ne contient **pas de données manquantes**.

2.3 Extraction de l'historique de population de la sphère parisienne

On définit **4 listes de départements** représentant les 4 couronnes :

- `sphere_couronne_paris`, département parisien
- `sphere_couronne_1`, département de la petite couronne
- `sphere_couronne_2`, département de la grande couronne
- `sphere_couronne_3`, département de la 3ème couronne

```
sphere_couronne_paris = c("75")
sphere_couronne_1 = c("92", "93", "94")
sphere_couronne_2 = c("77", "78", "91", "95")
sphere_couronne_3 = c("02", "10", "27", "28", "45", "51", "60", "89")
```

On sélectionne dans le jeu de données initial les départements qui nous intéressent à savoir les départements de Paris et des 3 couronnes.

```
# population des couronnes
df_pop_couronnes = df_pop_tous_departements %>%
  filter(dep %in% c(sphere_couronne_paris, sphere_couronne_1,
                    sphere_couronne_2, sphere_couronne_3))
```

```
dim(df_pop_couronnes)
```

```
[1] 16 245
```

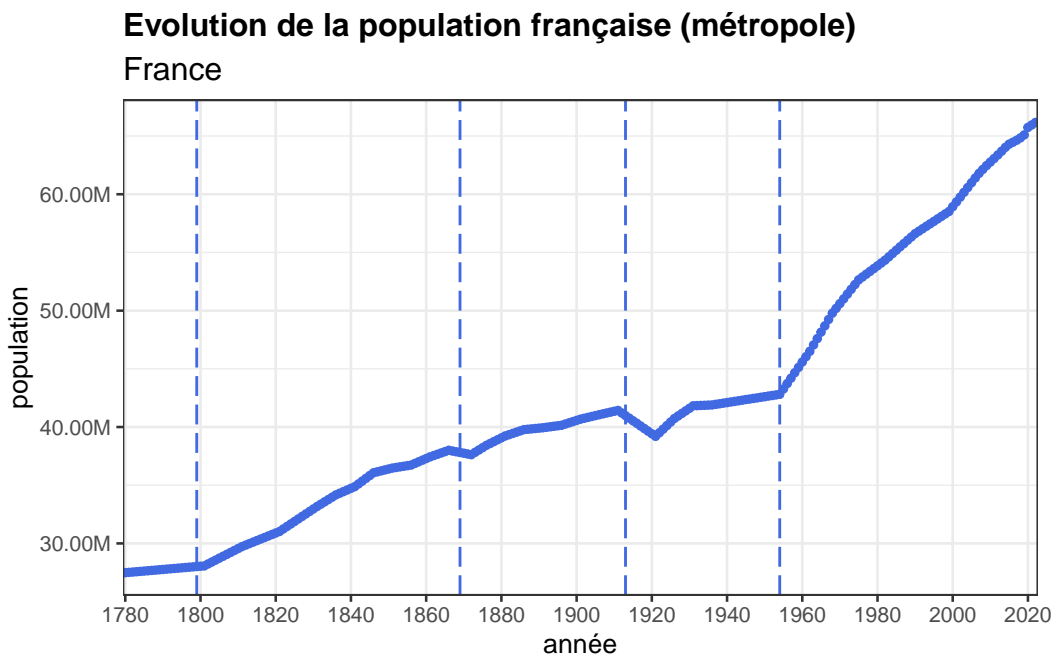
On a collecté dans `df_pop_couronnes` **243** ans de recensement pour les **16** départements **Paris** et ses **3** couronnes.

3 1ère approche des données

3.1 Population française

Population française métropolitaine

```
# Population = Somme par colonne de la population de tous les départements
df_pop_france = t(data.frame(colSums(df_pop_tous_departements[,3:ncol(df_pop_tous_departements)])))
row.names(df_pop_france) = "france"
```



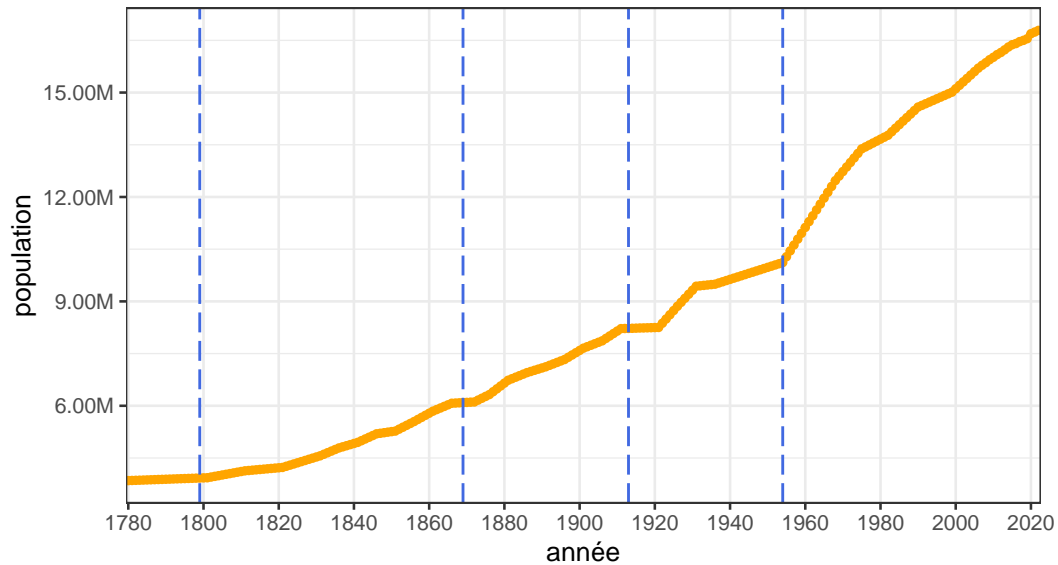
On note “à l’œil” que la population française métropolitaine connaît plusieurs phases

- jusqu’en 1800, la population stagne
- A partir de 1800 jusqu’en 1945, la population présente une évolution régulière et relativement lente, qui connaît 2 “accidents” correspondant à la guerre franco-prussienne de 1870 et à la 1ère guerre mondiale.
- après 1955, une augmentation plus forte de la démographie post-seconde guerre mondiale (Baby Boom)

```
df_pop_couronnes_marginale = t(data.frame(colSums(df_pop_couronnes[,3:ncol(df_pop_couronnes)])))
row.names(df_pop_france) = "france"
```


Evolution de la population

Sphère d'influence parisienne (Paris + couronnes)



Population de la sphère (Paris + couronnes)

En reportant les paliers de population française sur l'évolution de la population de la sphère, on constate

- que la courbe suit celle de la démographie française,
- qu'elle connaît les mêmes inflexions, dans les mêmes proportions.

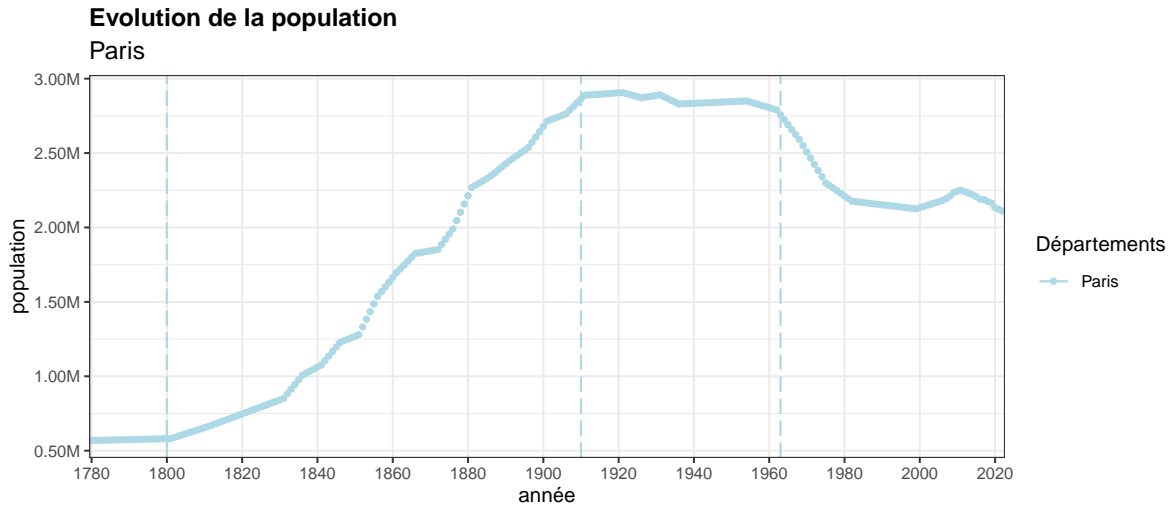
On supposera ici que si des mouvements de populations atypiques sont constatés dans les différentes couronnes de la sphère, ils sont dus à des mouvements intrinsèques.

Regardons plus précisément l'évolution de population des couronnes de la sphère.

3.2 Population parisienne

```
table = df_pop_couronnes %>%  
  filter(dep %in% sphere_couronne_paris)
```

Paris connaît une évolution de population qui se distingue nettement de celle de la population française :



- Jusqu’au début de la 1^{ère} guerre mondiale, la population parisienne croît dans des proportions plus fortes que la moyenne française. Cette phase coïncide avec la phase d’exode rural que connaît la France à cette période.
- Alors que la population française poursuit sa croissance après la 1^{ère} guerre mondiale, la population parisienne plafonne jusqu’à l’après Seconde Guerre Mondiale, puis commence à décroître au moment où la France connaît le “Baby Boom”
- La population parisienne décroît jusqu’aux années 2000 pour connaître un rebond au début du 21^{ème} siècle.

La littérature tend à justifier la stagnation et la décroissance par un équilibrage des populations entre Paris et sa banlieue (saturation urbaine, plans immobiliers dans les années 70, augmentation du prix du logement). Le rebond serait dû à une augmentation de la natalité dans les classes les plus défavorisées.

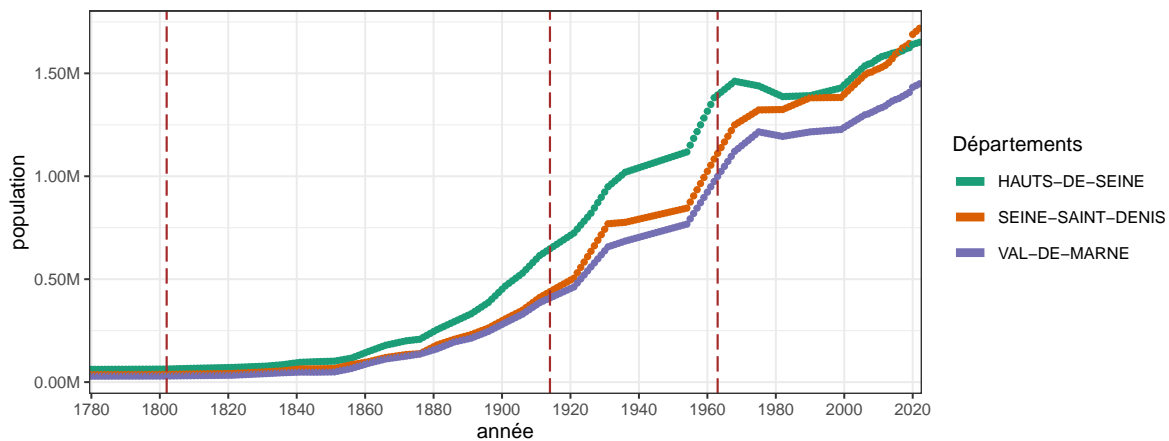
3.3 Population des couronnes

```
table = df_pop_couronnes %>%
  filter(dep %in% sphere_couronne_1)
```

```
table = df_pop_couronnes %>%
  filter(dep %in% sphere_couronne_2)
```

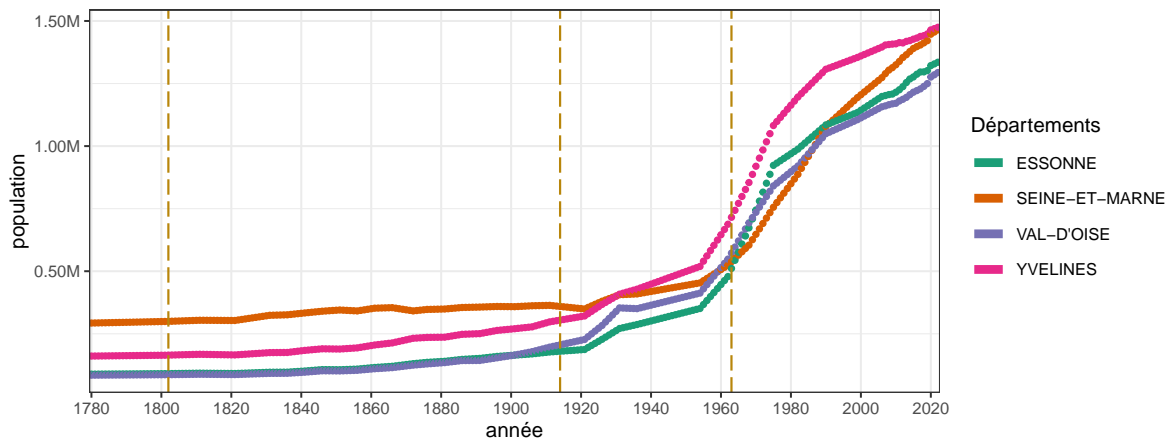
Evolution de la population

1ère couronne

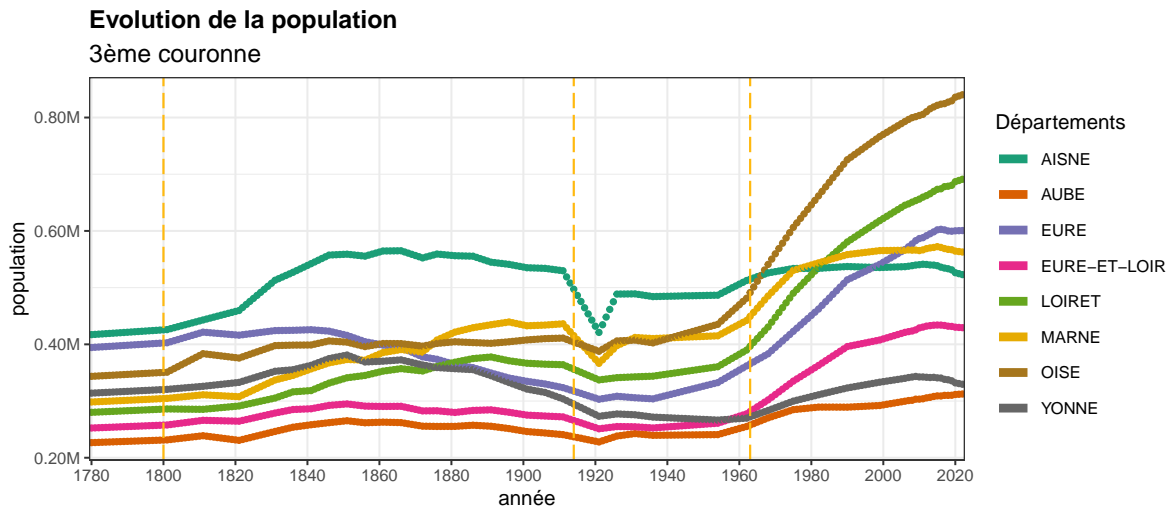


Evolution de la population

2ème couronne



```
# population dans la couronne 3
table = df_pop_couronnes %>%
  filter(dep %in% sphere_couronne_3)
```



Le comportement des départements de la **1ère couronne** est plutôt homogène :

- Leur population commence à croître fortement à partir de 1850 (contre 1800 à Paris)
- Une “cassure” dans le rythme de croissance (sans pour autant la rompre) est constatée dans le courant des années 60, au moment où la population parisienne commence à décroître.
- La croissance se poursuit jusqu’en 2022

Le comportement des départements de la **2ème couronne** est homogène :

- Leur population plutôt basse stagne jusqu’au début de la 1ère guerre mondiale (alors que pendant ce temps, la population française augmente et que la population parisienne explose).
- La population commence à croître fortement après-guerre jusqu’à nos jours (alors que la population parisienne stagne puis décroît)

En ce qui concerne la **3ème couronne**, les comportements des départements sont diverses ; il est complexe d’en sortir des généralités si ce n’est que ces départements ne connaissent pour la plupart ni la croissance française, ni la croissance parisienne jusqu’en 1945.

On commence ici à voir se dessiner la notion de sphère d’influence en termes de population.

Alors que la population de la sphère suit globalement la croissance de la population française, on constate des particularités à l’intérieur de la sphère :

- Paris et sa 1ère couronne ont jusqu'à la 1ère guerre mondiale une croissance plus forte que celle de la France, vraisemblablement alimentée par les populations stagnantes des couronnes 2 et 3,
- Une stagnation et une décroissance de la population parisienne au moment du Baby Boom vraisemblablement au profit des 3 couronnes (principalement 2 et 3)

4 Stratégie de l'étude

Pour approfondir l'étude des mouvements de population au sein de la sphère d'influence, on propose une étude du **profil temporel de chaque département**.

L'idée est de classifier les départements en fonction de la façon dont leur population évolue dans le temps.

Dans un second temps, on identifiera les moments clés de l'évolution de la population parisienne et on les rapprochera des profils moyens pour mettre en avant des mouvements de population entre Paris et ses couronnes.

Pour ce faire,

- on appliquera par département (i.e. par ligne) une **normalisation min-max** de telle sorte que pour chaque département,
 - le minimum de sa population dans le temps soit 0
 - le maximum de sa population dans le temps soit 100

La mise à la même échelle de tous les départements les rend indépendants de la taille du département et permet de concentrer l'étude sur la façon dont la population évolue.

- On procédera alors à une **classification** pour rapprocher les profils similaires et on en déduira des profils moyens.
- Enfin, on effectuera sur la population parisienne une **détection de ruptures** pour identifier les moments clés.

Ces moments clés seront rapprochés de la courbe des profils moyens pour mettre en avant d'éventuelles coïncidences dans leurs caractéristiques.

5 Profilage des départements

Dans ce chapitre, on cherche

- à mettre en avant les caractéristiques de l'évolution de la population d'un département indépendamment de sa taille
- à rapprocher des départements ayant connu une évolution similaire

5 étapes

- **Normaliser** la population des départements sur une échelle de 0 à 100 par une technique de **mise à l'échelle Min-Max**
- **Visualiser** couronne par couronne les profils des départements
- **Classifier** les profils par une technique de **clustering hiérarchique**
- Calculer un **profil moyen** par classe
- Valider "géométriquement" la classification par méthodes ACP et Isomap

5.1 Mise à l'échelle min-max

```
# normalisation min-max
# -----
minmax_scaling = function(row) {

  row_min <- min(row)
  row_max <- max(row)

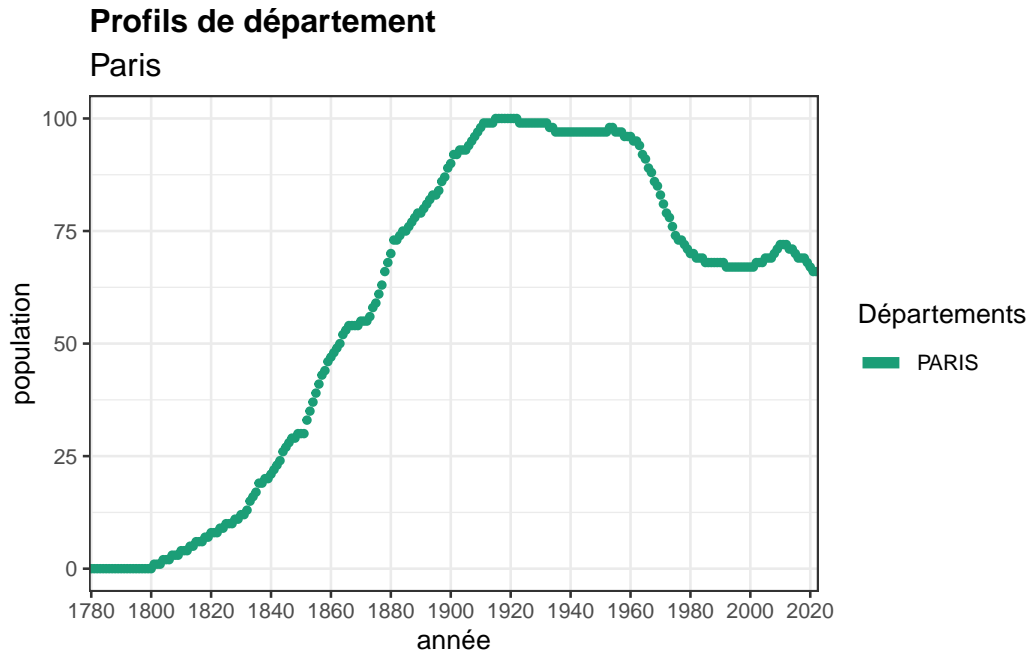
  normalized_row <- (row-row_min)/(row_max-row_min)

  return(normalized_row)
}

# Application aux départements de la sphère
# -----
df_profils_dep = apply( df_pop_couronnes[,3:ncol(df_pop_couronnes)],
                        MARGIN=1, # par ligne
                        FUN=function(row_departement)
                          100 * round( minmax_scaling(row_departement), 2)
                        )

# ajouter les colonnes dep et nomdep
df_profils_dep = cbind(df_pop_couronnes[,1:2], t(df_profils_dep))
```

5.2 Visualisation des profils temporels



Indépendamment de leur taille respective, les départements de la **1ère et 2ème couronne** ont respectivement une évolution de leur population similaire à travers le temps.

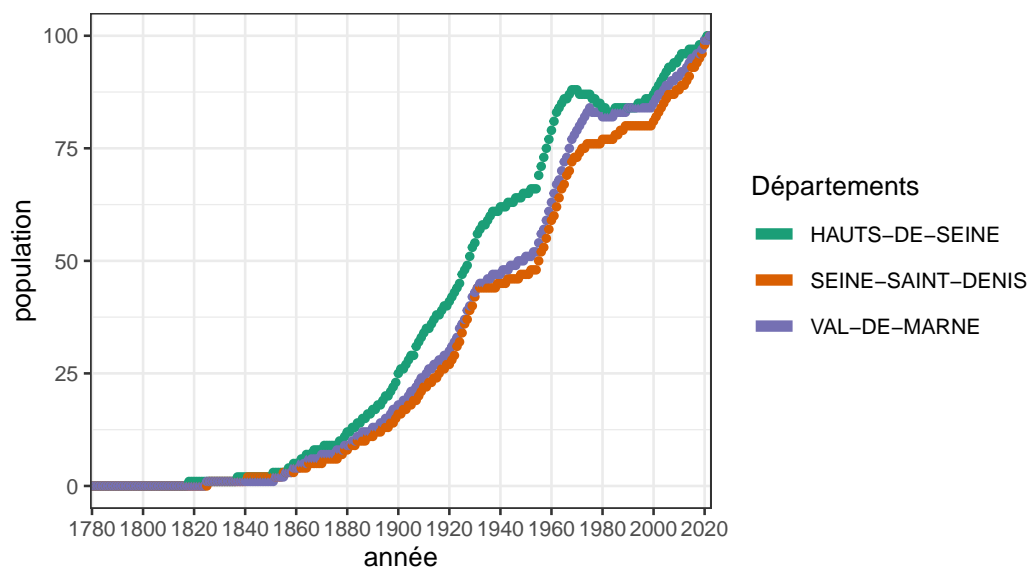
L'évolution de la population des départements de la **3ème couronne** est plus **chaotique** : il est complexe à l'oeil de confirmer / infirmer la proximité des départements. Le constat est sans doute dû au fait que l'on a à faire à des départements à faible population en valeur absolue de telle sorte que tout mouvement de population a un impact fort sur le département.

Il est aussi complexe de déterminer des rapprochements entre départements de couronnes différentes.

Pour regrouper les profils, on va appliquer une méthode de **classification hiérarchique**.

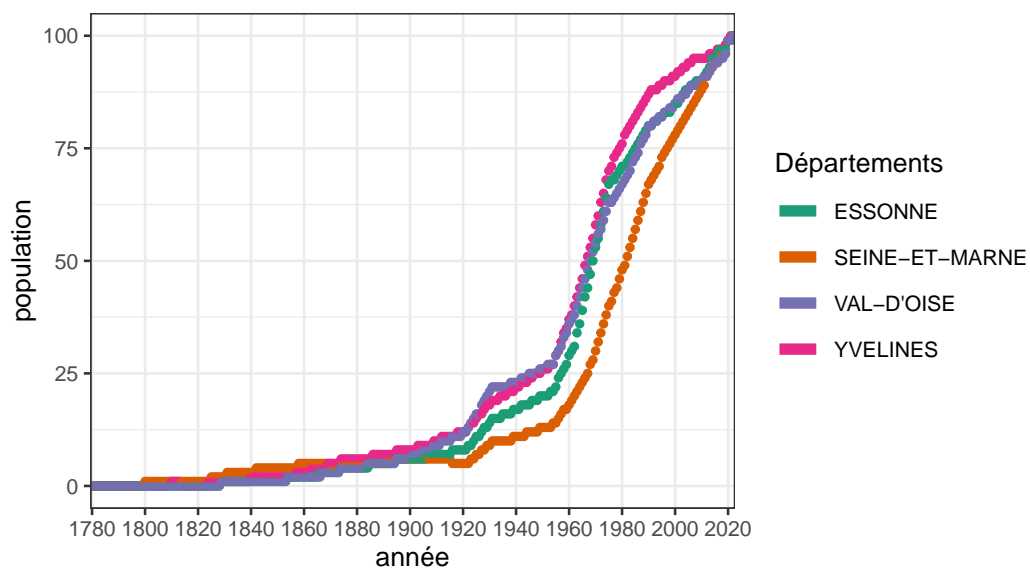
Profils de département

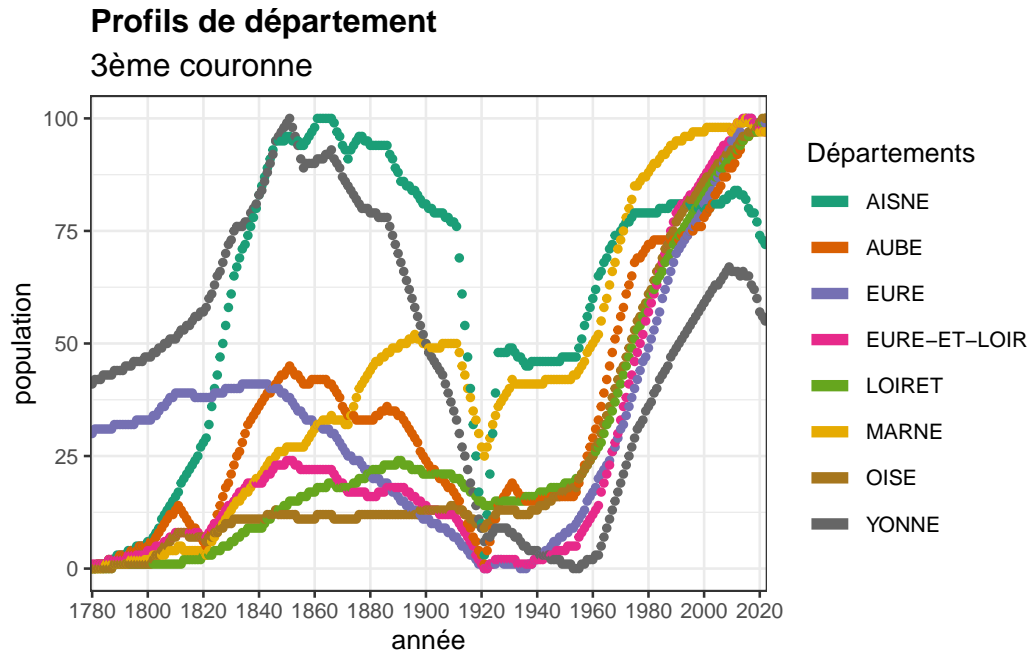
1ère couronne



Profils de département

2ème couronne





5.3 Regroupement des profils

A noter

Dans le cadre de la classification et aussi de la projection à venir, on utilisera la **distance euclidienne entre les séries temporelles** pour en simplifier l'analyse.

Une approche avec des distances “optimal matching” du type Dynamic Time Warping, basées plutôt sur la forme des courbes, permettrait certainement d'optimiser les résultats.

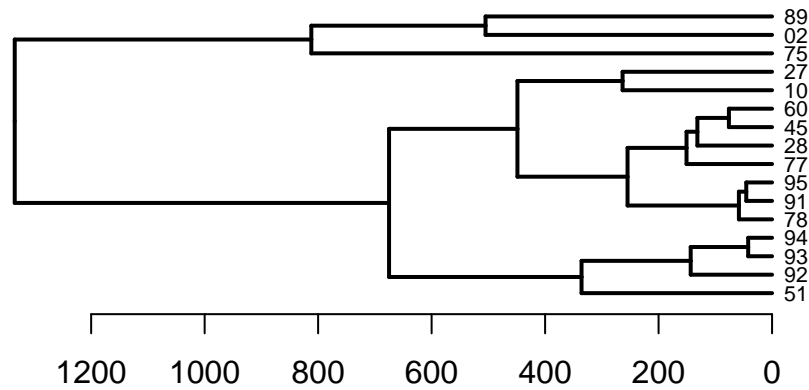
5.3.1 Clustering hiérarchique

```
library(dendextend)

hc = hclust(dist(df_profils_dep[,3:ncol(df_profils_dep)],
               method="euclidean"), method="ward.D2")

dend = as.dendrogram(hc)
```

Profils de départements de la sphère parisienne



dendrogramme

! Important

On décide de scinder les profils en **7 classes**.

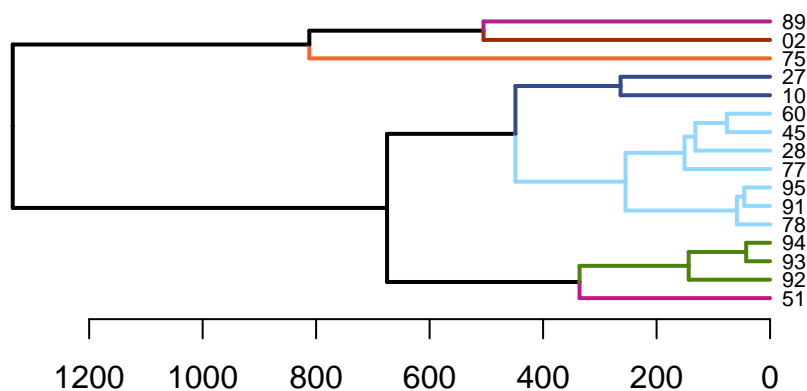
```
# Nombre de groupes
nb_classes=7
```

```
# classification hiérarchique : assignation des groupes
# -----
clusters = cutree(dend, k=nb_classes)
```

```
# Visualisation du dendrogramme avec identification des groupes
# -----
dend.plot <- as.dendrogram(dend) %>%
  set("branches_lwd", 2) %>% # Branches line width
  set("branches_k_color", palette_groupes,
      k = nb_classes) %>% # Color branches by groups
  set("labels_cex", 0.7) # Change label size

plot(dend.plot,
     horiz=TRUE,
     main="Profils des départements de la sphère parisienne",
     sub="dendrogramme")
```

Profils des départements de la sphère parisienne



dendrogramme

```
clusters = as.factor(clusters)
clusters
```

```
02 10 27 28 45 51 60 75 77 78 89 91 92 93 94 95
 1  2  2  3  3  4  3  5  3  3  6  3  7  7  7  3
Levels: 1 2 3 4 5 6 7
```

```
# Assignation d'un nom aux groupes
# (nom = concaténation des numéros de départements du groupe)
# -----
clusters_name = c()
for (g in levels(clusters)) {

  # concaténation des numéros des départements du groupe
  name = paste(names(clusters)[clusters==g], collapse = "-")
  clusters_name[g] = name
}
```

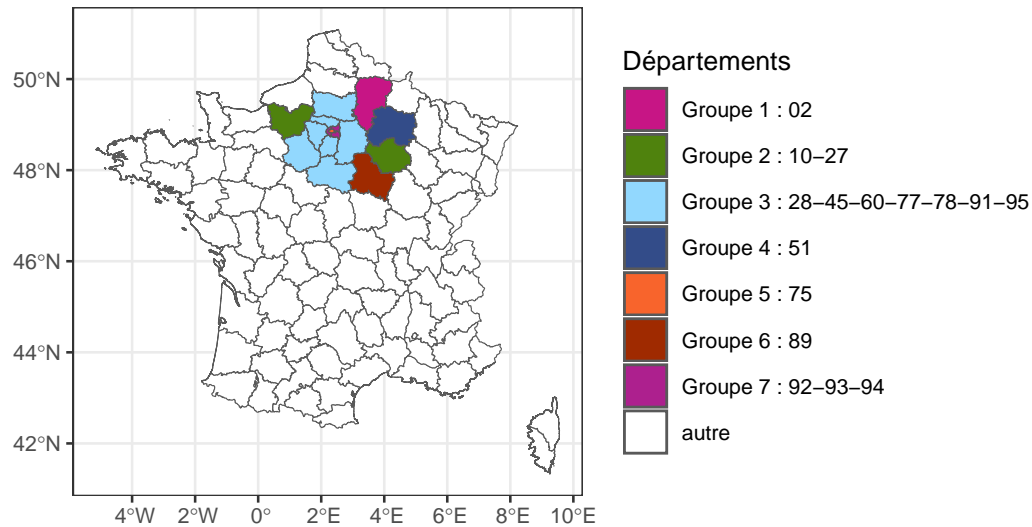
5.3.2 Visualisation des groupes

La classification en 7 classes rebat les cartes des couronnes.

- Paris et la 1ère couronne sont maintenus dans leur spécificité et leur homogénéité en 2 classes distinctes

Profils des départements dans la sphère parisienne

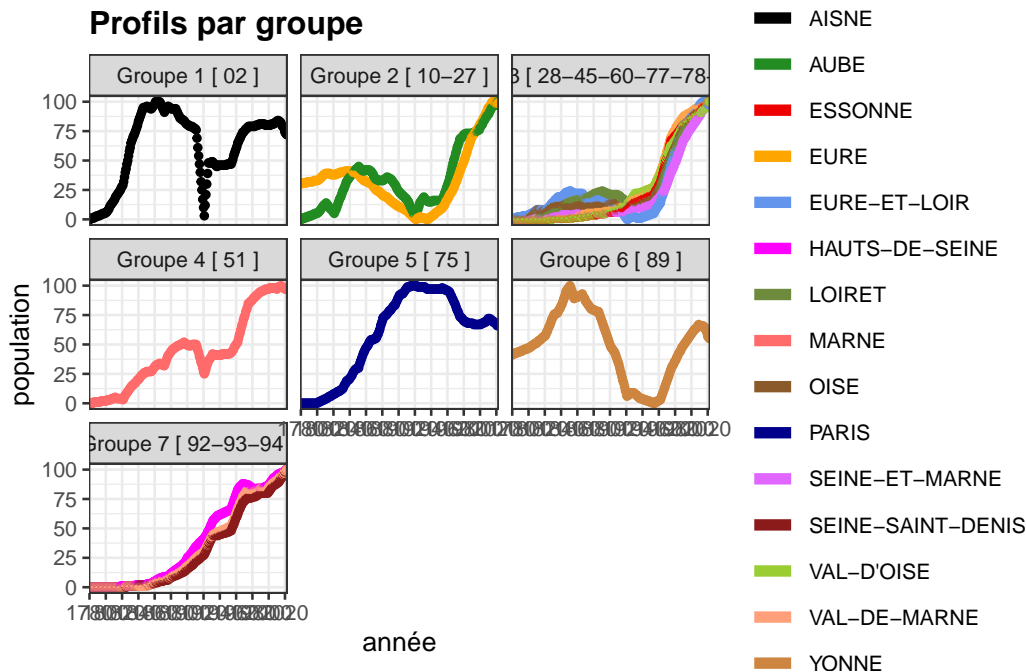
Regroupement par classification hiérarchique



- Tout en maintenant en cohésion les départements de la 2ème couronne (77/78/91/95), elle a aussi associé un certain nombre de départements de la 3ème couronne (28/45/60).
- Elle isole le comportement du reste des départements de l'Est de la 3ème couronne (02, 51, 89)
- Bien qu'éloignés géographiquement, les profils des départements 10 et 27 sont suffisamment proches pour être rapprochés dans un même groupe.

Visualisation des profils par classe

```
# Ajout du groupe aux départements (comme groupe)
df_profils_dep_w_groups = df_profils_dep
df_profils_dep_w_groups = cbind(groupe=clusters[as.character(df_profils_dep_w_groups$dep)],
                                df_profils_dep_w_groups)
df_profils_dep_w_groups$groupe = as.factor(df_profils_dep_w_groups$groupe)
```



La visualisation permet de confirmer “à l’œil” l’homogénéité des profils d’une même classe et d’identifier les caractéristiques qui isolent certains départements (la forte décroissance de la population de l’Yonne (89) ou les “fortes cassures” au moment de la 1^{ère} guerre mondiale (02-51)).

On comprend aussi la “proximité” sur le dendrogramme du département 51 (Marne - groupe 4) et de la classe 7 (1^{ère} couronne) : l’évolution de leur population est similaire à un “accident important” près lors de la 1^{ère} guerre mondiale.

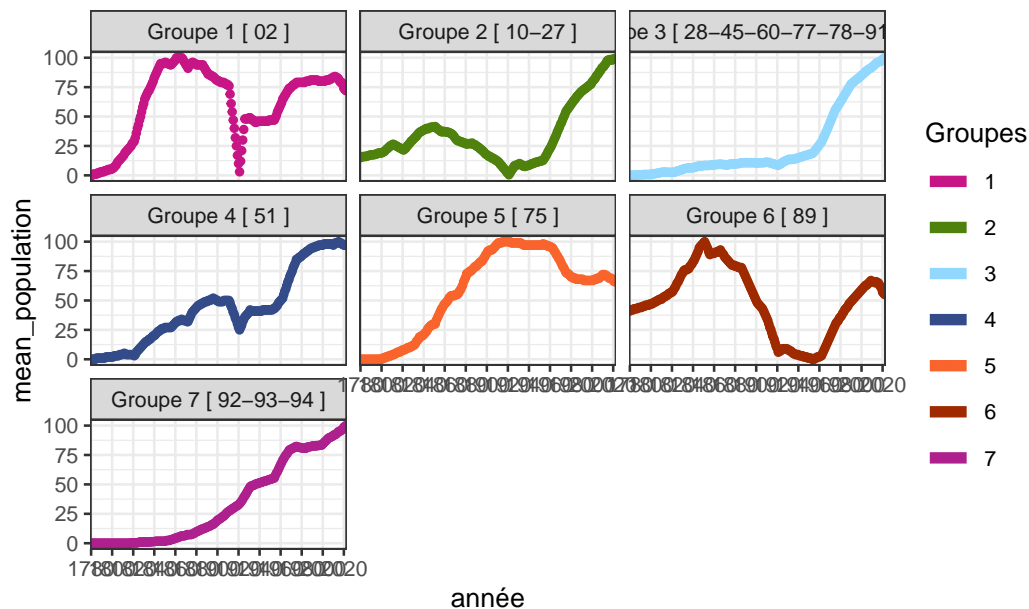
5.3.3 Profils moyens

On crée un nouveau Data Frame de profils moyens par classe à partir du Data Frame de tous les profils.

```
df_profils_dep_moyen = df_profils_dep_w_groups %>%
  group_by(groupe) %>%
  summarise(across(starts_with("1"), mean)) %>%
  column_to_rownames(var = "groupe") %>%
  as.data.frame()
```

```
# Création d'un Dataframe avec
# . groupes en colonne
# . années en ligne
```

Profils moyens par groupe



```
table = df_profils_dep_w_groups %>%
  group_by(groupe) %>%
  summarise(across(starts_with("1"), mean)) %>%
  column_to_rownames(var = "groupe") %>%
  as.data.frame() %>%
  t()
```

```
# Corrélation du groupe 5 avec les autres groupes
cor(table, method="pearson")[,5]
```

	1	2	3	4	5	6	7
0.3830966	-0.1426154	0.3647162	0.6800904	1.0000000	-0.5516320	0.6647601	

Un calcul de corrélation entre le département parisien (Groupe 5) comme référence et le profil moyen des autres classes montre

- des corrélations positives fortes entre Paris et les groupes 4 (51-Marne) et Groupe 7 (petite couronne)

On peut supposer que ces départements ont connu des mouvements de population similaires en tant que pôle d'attraction des populations.

- des corrélations négatives
 - relativement fortes entre Paris et le département de l'Yonne (89) (Groupe 6)
 - relativement faible entre Paris et les départements du Groupe 2 (27-Eure et 10-Aube)

On peut supposer des transferts de population entre ces départements et Paris.

5.4 Projection

Les valeurs des individus étant normalisées “en ligne” (valeurs de 0 à 100), on se propose d’utiliser des techniques de projection en dimension restreinte pour apprécier la cohérence “géométrique” des résultats de la classification.

En retenant la **distance euclidienne** pour mesurer les distances entre profils, il s’agit d’identifier des plans sur lesquels les individus sont les mieux représentés.

On appliquera 2 techniques :

- une **analyse en composantes principales**
- **Isomap**, que l’on sait conserver les distances entre individus après projection

5.4.1 ACP

ACP sur les profils de départements de la sphère d’influence

```
profils_dep_pca = PCA(df_profils_dep %>% dplyr::select(-dep, -nomdep),
                      scale.unit = FALSE,
                      graph = FALSE,
                      ncp=ncol(df_profils_dep)-2)
```

Visualisation de la part de variance portée par les axes

```
profils_dep_pca$eig[1:5, 2]
```

comp 1	comp 2	comp 3	comp 4	comp 5
56.085389	36.727800	3.852381	2.183053	0.674125

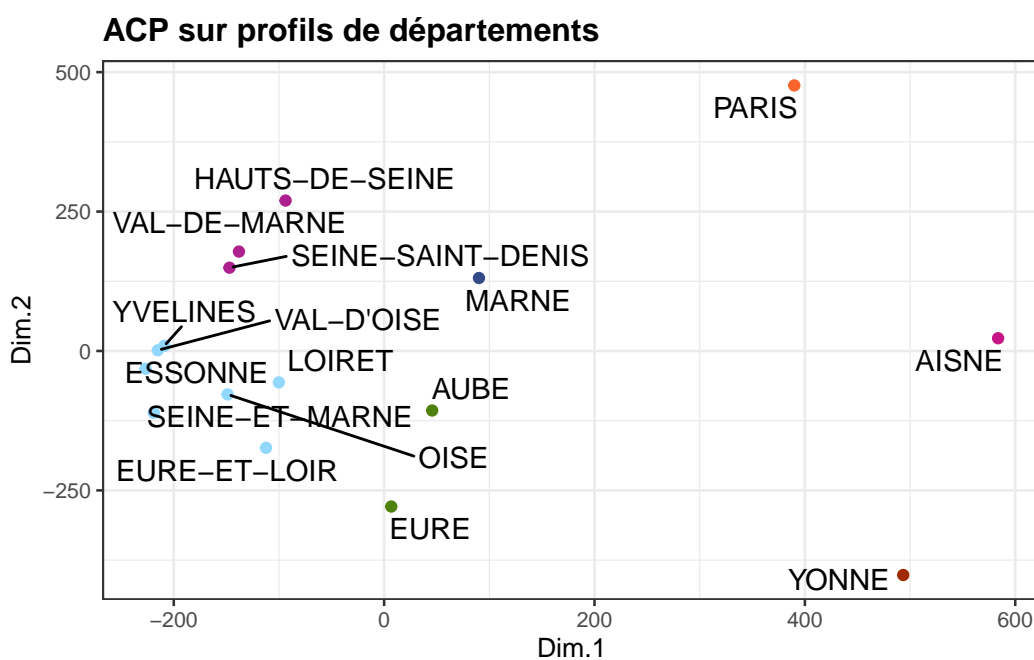
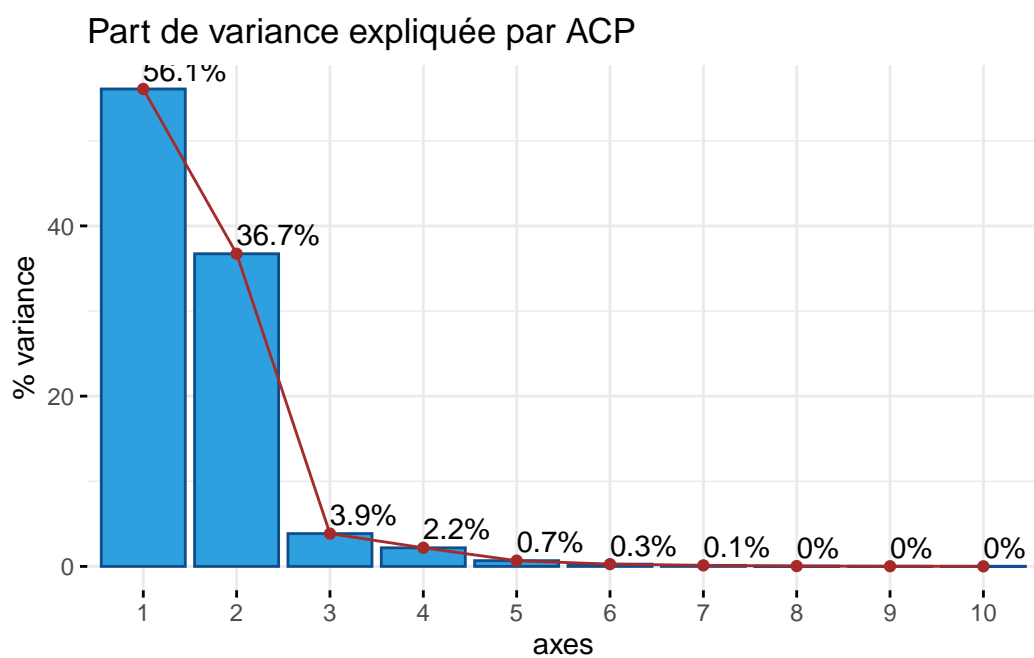
Les 2 premiers axes portent 96,7% de la variabilité du jeu de données.

```
plot(profils_dep_pca,
     title="ACP sur les profils de départements",
     col.quali="groupe",)
```

L’axe 1 oppose les départements Paris et 1ère couronne (plus la Marne, très similaire comme vu précédemment) et les départements 2ème et 3ème couronnes.

L’axe 2 oppose les classes 3 et 7 regroupant la majorité des départements et les classes isolant les départements atypiques (89-02-51).

L’ACP confirme :



- La proximité des éléments d'une même classe
- La particularité des départements 89-10-27 (Yonne-Aube-Eure) que l'on sait en corrélation négative avec les autres profils.

5.4.2 isomap

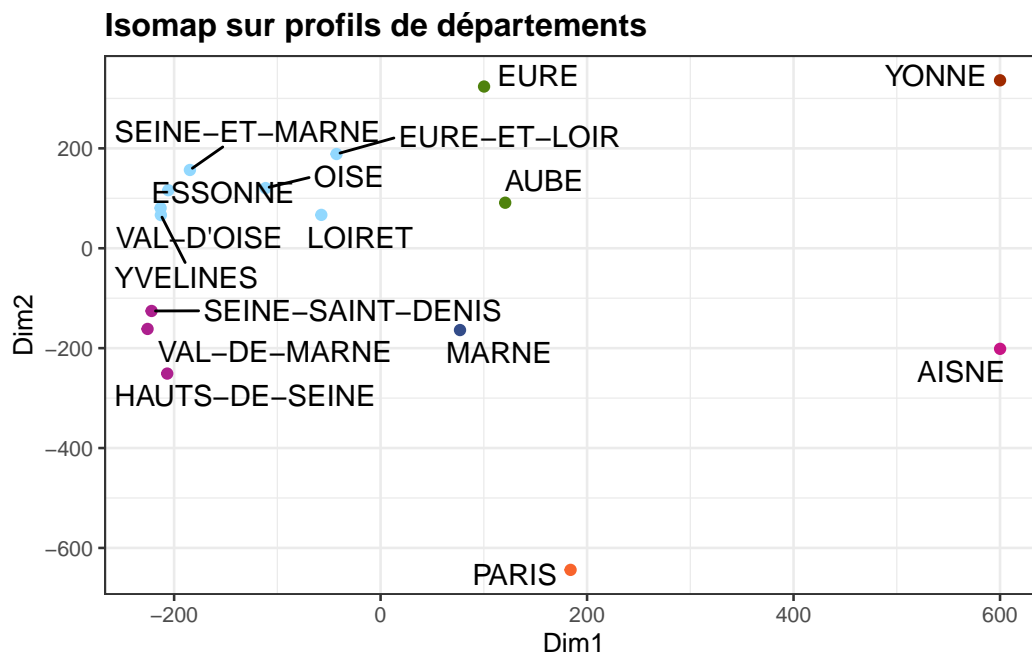
```
# Mesure des distances entre les profils de départements
profils_dep_dist = as.matrix(dist(df_profils_dep[,3:ncol(df_profils_dep)],
                                method="euclidean", diag=TRUE, upper=TRUE))

# Isomap sur les populations des couronnes
profils_dep_isomap <- isomap(profils_dep_dist, ndim=2, k=nb_classes)

# Variance expliquée avec Isomap
sum(profils_dep_isomap$eig[1:2])/sum(profils_dep_isomap$eig[1:nb_classes])
```

[1] 0.9297724

isomap capte 92% de la variabilité des données sur les 2 premiers axes.



Isomap conforte

- l'homogénéité des départements d'une même classe,
- les écarts entre classes
- le côté atypique des départements Aisne et Yonne.

6 Identification des moments clés dans la population parisienne

Dans ce chapitre, on se propose

- d'identifier des **points de rupture** dans la courbe de population parisienne par une **méthode PELT à pénalité optimisée par CROPS**.
- de les rapprocher du profil temporel moyen des groupes identifiés précédemment

Il s'agit ici de déterminer s'il est possible de mettre en corrélation les inflexions des courbes et tirer des enseignements sur de potentiels transferts de population entre départements dans la sphère.

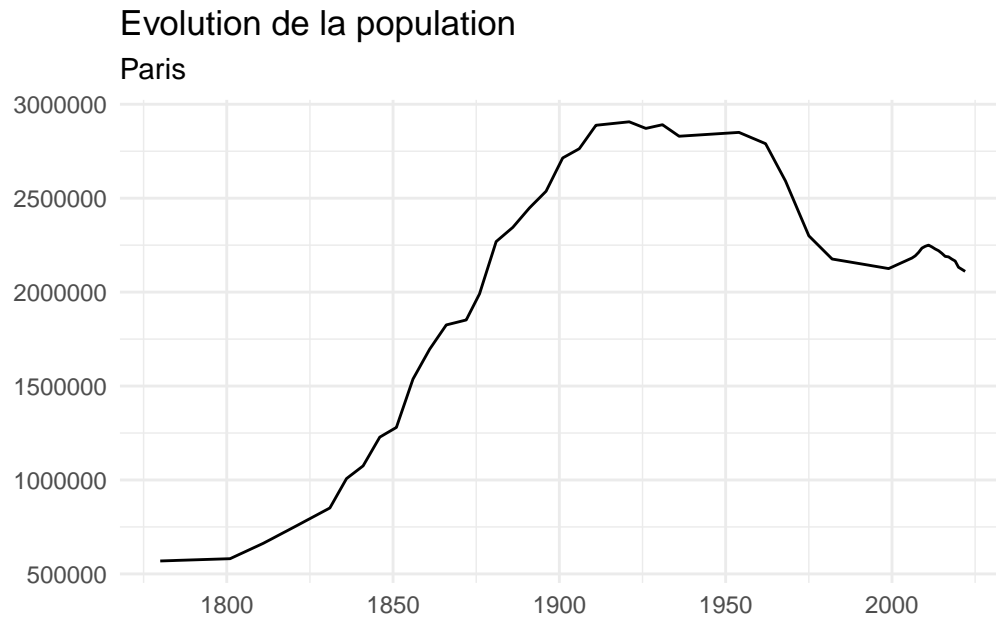
6.1 Préparation des données

```
# Transformation de la population parisienne en série temporelle
# -----

# Population de Paris en colonne
table = df_pop_couronnes %>%
  filter(dep %in% sphere_couronne_paris) %>%
  gather("année", "population", 3:ncol())

# Conversion en série temporelle
ts_paris = ts(table[, "population"], start=1780, frequency=1)
```

```
autoplot(ts_paris) +
  theme_minimal() +
  labs(title="Evolution de la population",
       subtitle="Paris")
```



Calcul du “log return” de la série temporelle

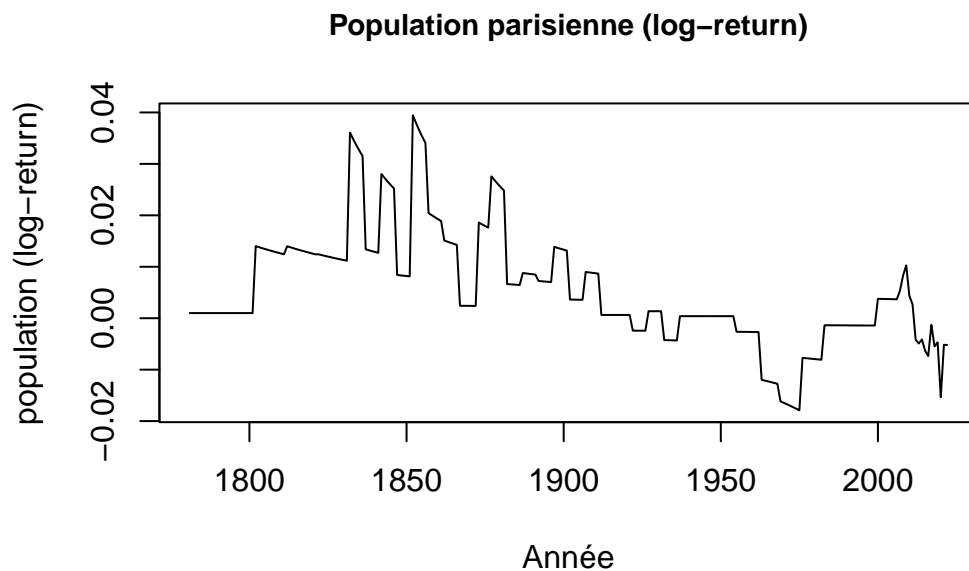
```
# Calcul du log-return  
# -----  
ts_paris_diff = diff(log(ts_paris))
```

6.2 Détection des points de rupture (PELT pénalité CROPS)

Pour identifier les moments clés dans l’évolution de la population parisienne, on va utiliser une **méthode PELT** avec une pénalité optimisée **CROPS**.

On initie l’algorithme avec les caractéristiques suivantes:

- un minimum de 10 ans entre 2 points de rupture ce qui semble raisonnable au vu des phénomènes recherchés et de l’échelle de temps.
- une large palette de pénalités candidates (1 à la taille de la série temporelle)
- une hypothèse de loi normale (communément admise pour une série en log-return)



```
# Méthode PELT CROPS (moyenne-variance)
# -----
ts_paris_crops = cpt.meanvar(ts_paris_diff,
                             test.stat = "Normal",
                             method="PELT",
                             penalty="CROPS",
                             pen.value=c(1,length(ts_paris_diff)),
                             minseglen=10)
```

```
[1] "Maximum number of runs of algorithm = 18"
[1] "Completed runs = 2"
[1] "Completed runs = 3"
[1] "Completed runs = 5"
[1] "Completed runs = 9"
[1] "Completed runs = 14"
[1] "Completed runs = 17"
```

```
# Visualisation du résultats
ts_paris_crops
```

```
Class 'cpt' : Changepoint Object
  ~~      : S4 class containing 14 slots with names
           cpts.full pen.value.full data.set cpttype method test.stat pen.type pen.value minseglen
```

Created on : Thu Mar 7 03:28:52 2024

```
summary(.) :
```

Created Using changepoint version 2.2.4

Changepoint type : Change in mean and variance

Method of analysis : PELT

Test Statistic : Normal

Type of penalty : CROPS with value, 1 242

Minimum Segment Length : 10

Maximum no. of cpts : Inf

Changepoint Locations :

Number of segmentations recorded: 14 with between 1 and 17 changepoints.

Penalty value ranges from: 1 to 219.6258

```
# Dimensions
```

```
dim(cpts.full(ts_paris_crops))
```

```
[1] 14 17
```

```
# Visualisation des pénalités "pertinentes" identifiées
```

```
# -----
```

```
round(pen.value.full(ts_paris_crops), digits=2)
```

```
[1] 1.00 1.99 3.27 13.55 18.45 27.59 34.33 37.39 70.53 107.29
```

```
[11] 119.52 120.63 124.94 219.63
```

La méthode PELT avec pénalité crops a identifié **14 pénalités pertinentes** pour un **maximum de 17 points de rupture**.

Visualisation des configurations contenant le moins de points de ruptures

```
# Pénalités les plus élevées, identifiant le moins de points de rupture
```

```
# -----
```

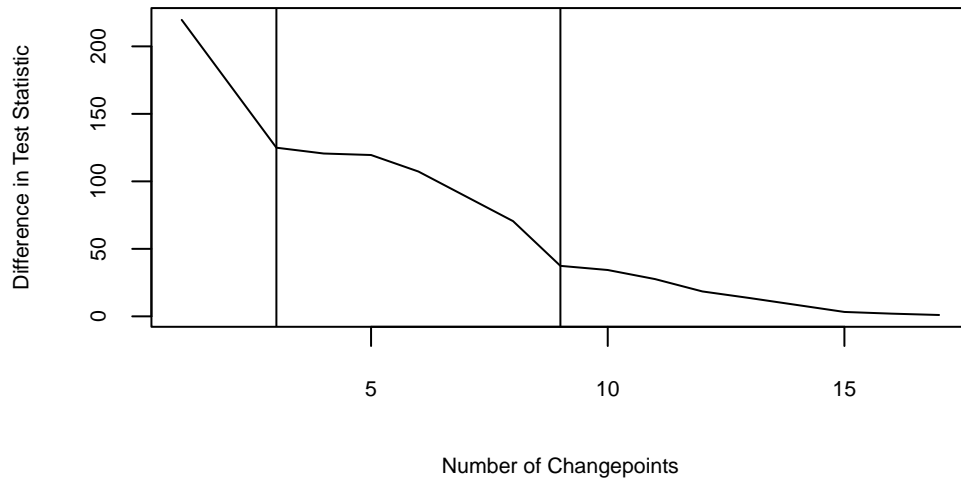
```
len = dim(cpts.full(ts_paris_crops))[1]
```

```
cpts.full(ts_paris_crops)[(len-6):len, 1:10]
```

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
[1,]	21	51	101	131	141	156	174	202	219	NA
[2,]	21	51	131	141	156	174	202	219	NA	NA
[3,]	21	51	131	141	156	174	NA	NA	NA	NA
[4,]	21	131	141	156	174	NA	NA	NA	NA	NA
[5,]	21	51	156	174	NA	NA	NA	NA	NA	NA
[6,]	21	156	174	NA	NA	NA	NA	NA	NA	NA
[7,]	21	NA	NA	NA	NA	NA	NA	NA	NA	NA

Choix de la pénalité optimale

Pénalité versus points de rupture



2 cas peuvent être envisagés :

- Une configuration à 3 points de rupture
- Une configuration à 9 points de rupture

version en 3 points de rupture

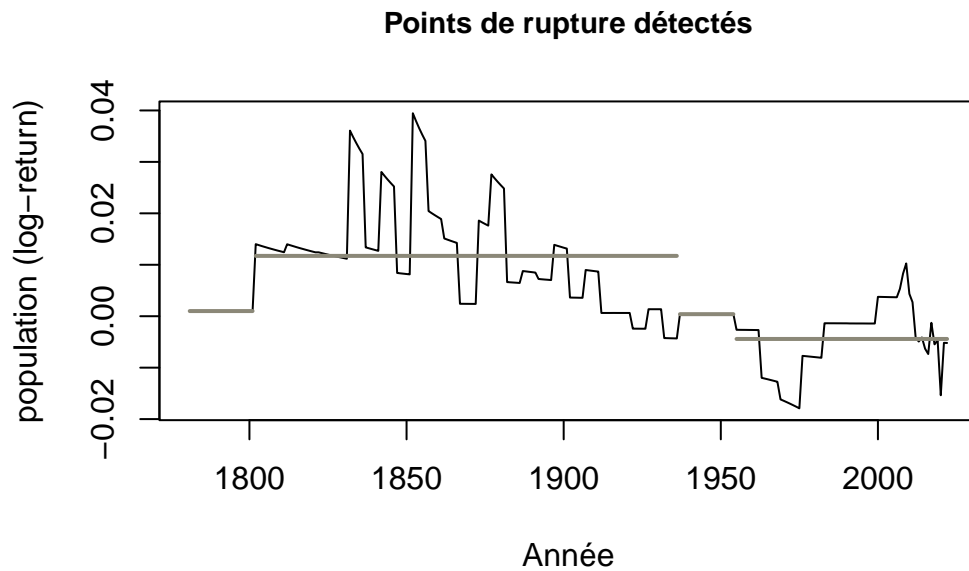
```
# Points de rupture
len = dim(cpts.full(ts_paris_crops))[1]
ts_paris_cp_index = cpts.full(ts_paris_crops)[len-1, 1:3]

# Pénalité associée
ts_paris_penalty = round(pen.value.full(ts_paris_crops)[len-1], digits=2)
ts_paris_penalty
```

```
[1] 124.94
```

```
# Années identifiées comme rupture
ts_paris_cp=time(ts_paris)[ts_paris_cp_index]
ts_paris_cp
```

```
[1] 1800 1935 1953
```



La configuration en 3 points de rupture correspond

- à une pénalité de 124.94
- aux années 1800, 1935, 1953

version en 9 points de rupture

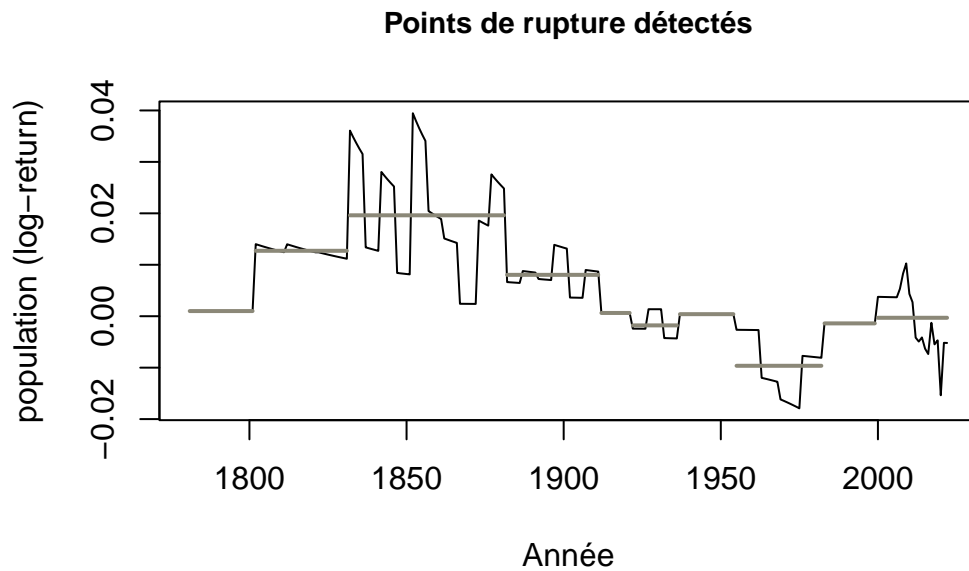
```
# Points de rupture
len = dim(cpts.full(ts_paris_crops))[1]
ts_paris_cp_index = cpts.full(ts_paris_crops)[len-6, 1:9]

# Pénalité associée
ts_paris_penalty = round(pen.value.full(ts_paris_crops)[len-6], digits=2)
ts_paris_penalty
```

```
[1] 37.39
```

```
# Années identifiées comme rupture
ts_paris_cp=time(ts_paris)[ts_paris_cp_index]
ts_paris_cp
```

```
[1] 1800 1830 1880 1910 1920 1935 1953 1981 1998
```



La configuration en 9 points de rupture correspond

- à une pénalité de 37.39
- aux années 1800 1830 1880 1910 1920 1935 1953 1981 1998

! Important

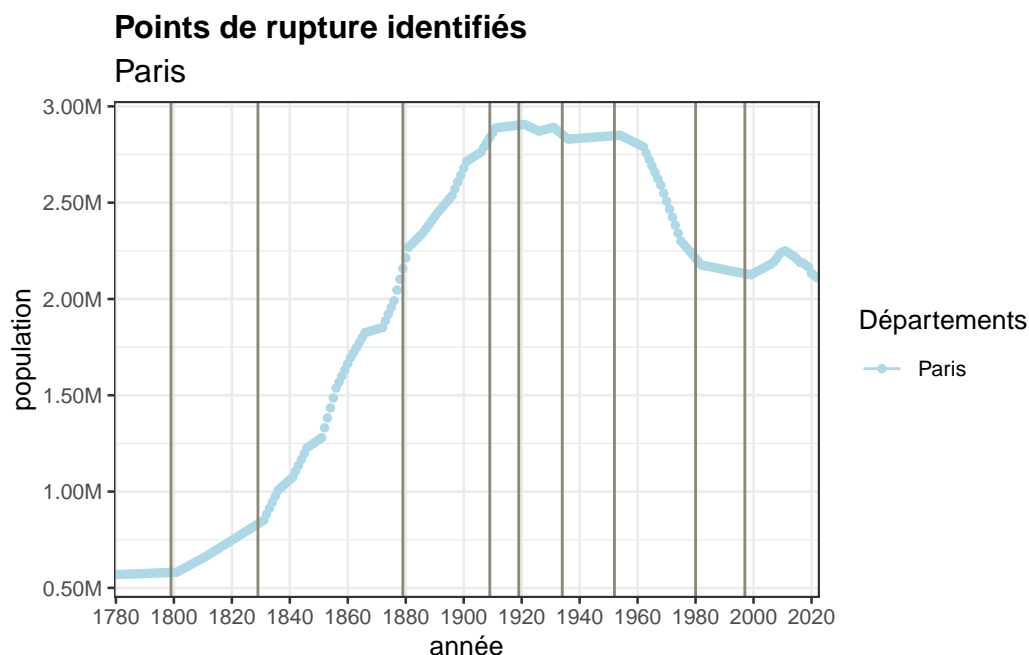
On prend le parti de conserver la version plus précise à **9 points de rupture**.

Visualisation des points de rupture identifiés

Il est intéressant de noter que les 1ères dates identifiées correspondent à des moments importants de la vie industrielle française

- 1800 : début de l'industrialisation et de l'exode rural qui en découle
- 1830 : Expansion de la machine à vapeur et développement du rail
- 1880 Fin de la période d'industrialisation, début de la grande dépression mais aussi début de l'électrification et de l'industrie automobile

Les dates intermédiaires semblent dirigées par l'entre-deux guerres.



Les dernières dates 1953-1981-1988 correspondent d'après la littérature à des moments de politique d'aménagement du territoire (développement des grands ensembles immobiliers en banlieue) ou de natalité forte.

6.3 Projection des moments clés sur les profils temporels moyens

On se propose de projeter les moments clés identifiés sur les profils temporels moyens calculés précédemment.

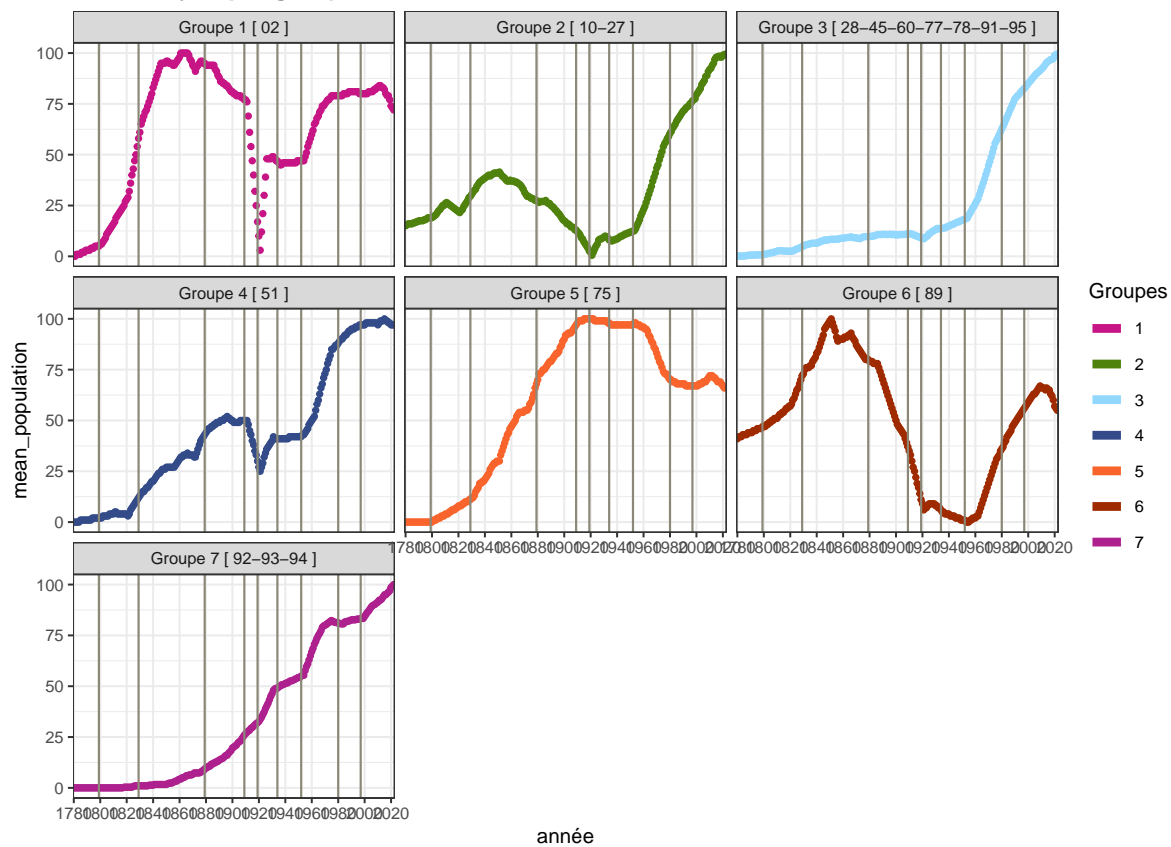
Le phénomène le plus marquant est l'effet Baby Boom (année 1953)

- On a déjà vu que le moment correspond à une décroissance forte de la population parisienne
- Il correspond aussi à une forte croissance sur l'ensemble des autres classes.

La croissance peut certes être expliquée par un accroissement de la natalité locale, mais son importance est telle qu'on peut aussi imaginer qu'un transfert de la population parisienne alimente cette croissance. C'est notamment vrai pour les départements de la classe 3 (2ème couronne + les départements de la 3ème couronne 28-45-60) et les départements de la classe 2 (10-27).

On peut aussi noter que le point de rupture 1880 est aussi significatif pour les classes dont les départements connaissent une chute de leur population au 19ème siècle : les classes 1 (02),

Profil moyen par groupe



2 (10-27) et 6 (89) connaissent à ce moment précis une forte décroissance de leur population laissant penser à un transfert de population vers Paris.

7 Conclusion

L'approche par la population montre que Paris et sa 1ère couronne tels que définis par l'INSEE forment un tout cohérent, qui a été un pôle d'attractivité jusqu'au début du Baby-boom post-seconde guerre mondiale.

La segmentation des 2ème et 3ème couronnes est plus floue ; bien que les départements de la 2ème couronne présentent un profil temporel commun, ils sont rejoints par certains départements de la 3ème couronne à savoir l'Eure-et-Loir (28), le Loiret (45) et l'Oise (60). Les autres départements de la 3ème couronne (Marne, Yonne, Aube et Eure) sont plus atypiques.

Dans tous les cas, la projection des moments-clés parisiens sur les profils temporels des autres départements laissent penser que les départements des 2ème et 3ème couronnes sont ceux qui ont contribué le plus à la forte croissance de la population à Paris et en 1ère couronne au 19ème siècle et au début du 20ème siècle (les plus contributeurs étant l'Yonne, l'Eure et l'Aube) ; ils sont aussi les bénéficiaires de la décroissance parisienne depuis le début du Baby-Boom.

8 Références

Julia Cagé et Thomas Piketty (2023) : *Une histoire du conflit politique. Élections et inégalités sociales en France, 1789-2022*, Paris, Le Seuil

Affichage de données géographiques : [GAD Maps and Data](#)

Histoire de la démographie française : [Wikipedia](#)

La démographie parisienne : [wikipedia](#)