

GGPLOT 2

Rindra LUTZ / William ROBACHE

19/12/2020

Introduction

GGPLOT2 est une bibliothèque R qui vous permet de visualiser les données développées par Hadley Wickham. GGPLOT 2 est une extension *tidyverse* qui permet la génération de graphiques en utilisant une syntaxe cohérente et puissante. L'une des caractéristiques de GGPLOT 2 est qu'il suppose que les données liées aux graphiques sont stockées dans une table de données (data frame ou autre).

Afin de découvrir au mieux ggplot2, nous allons réaliser l'analyse d'un jeu de données de 20 états occidentaux.

Préparation

```
#charger ggplot2
library(ggplot2)

# charger l'extension lisant le format CSV
library(readr)

# emplacement souhaité pour le jeu de données
file <- "debt.csv"

# télécharger le jeu de données s'il n'existe pas
if(!file.exists(file))
  download.file("http://www.stat.cmu.edu/~cshalizi/uADA/13/hw/11/debt.csv",
               file, mode = "wb")

# charger les données dans l'objet 'debt'
debt <- read_csv(file)

## Warning: Missing column names filled in: 'X1' [1]

##
## -- Column specification -----
##
## cols(
##   X1 = col_double(),
##   Country = col_character(),
##   Year = col_double(),
##   growth = col_double(),
##   ratio = col_double()
## )
```

Nettoyage des données

L'ensemble de données contient un échantillon de 20 États membres occidentaux de la région de l'OCDE, la croissance de leur produit intérieur brut (PIB) et le ratio de leur dette publique au produit, en dette / PIB. Les données vont du milieu des années 1940 à la fin des années 2000, la première colonne de l'ensemble de données ne contient que le numéro de ligne, nous le supprimons donc immédiatement:

```
# inspection du jeu de données
str(debt)

## tibble [1,171 x 5] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ X1      : num [1:1171] 147 148 149 150 151 152 153 154 155 156 ...
## $ Country: chr [1:1171] "Australia" "Australia" "Australia" "Australia"
## ...
## $ Year    : num [1:1171] 1946 1947 1948 1949 1950 ...
## $ growth  : num [1:1171] -3.56 2.46 6.44 6.61 6.92 ...
## $ ratio   : num [1:1171] 190 177 149 126 110 ...
## - attr(*, "spec")=
## .. cols(
## ..   X1 = col_double(),
## ..   Country = col_character(),
## ..   Year = col_double(),
## ..   growth = col_double(),
## ..   ratio = col_double()
## .. )

# suppression de la première colonne
debt <- debt[, -1]
```

Il faut également souligner d'emblée que certaines mesures font défaut: pour certains pays, il n'existe pas de mesures fiables du PIB et / ou de la dette publique. En conséquence, le nombre d'observations dans chaque pays / région est différent, allant des observations annuelles dans 40 pays / régions de Grèce aux observations annuelles dans 64 pays / régions de plusieurs pays (comme l'Australie ou les États-Unis):

```
table(debt$Country)
```

##						
##	Australia	Austria	Belgium	Canada	Denmark	Finland
##	64	59	63	64	56	64
##	France	Germany	Greece	Ireland	Italy	Japan
##	54	59	40	63	59	54
##	Netherlands	New Zealand	Norway	Portugal	Spain	Sweden
##	53	64	64	58	42	64
##	UK	US				
##	63	64				

Recodage d'une variable

Dernière opération préliminaire avant l'analyse: nous allons calculer la décade de chaque observation en divisant l'année de mesure par 10 et en multipliant la partie entière du résultat par 10. Cette opération très simple donne la valeur mesurée de l'année de 1940 à 1949, de 1950 à 1959, et ainsi de suite.

```
debt$Decade <- factor(10 * debt$Year %/% 10)
```

Voici, pour terminer, les premières lignes du jeu de données sur lequel on travaille :

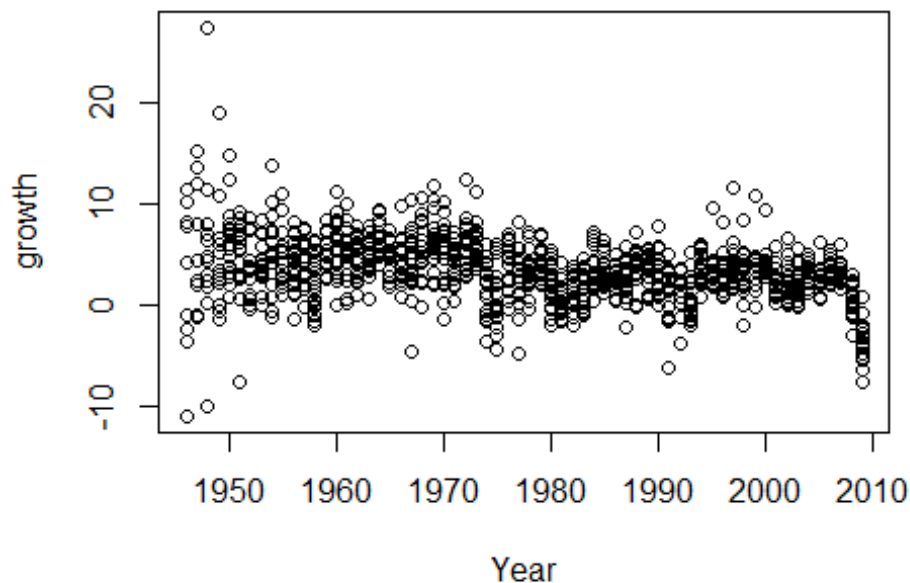
```
head(debt)
```

```
## # A tibble: 6 x 5
##   Country    Year growth ratio Decade
##   <chr>      <dbl> <dbl> <dbl> <fct>
## 1 Australia  1946  -3.56 190.  1940
## 2 Australia  1947   2.46 177.  1940
## 3 Australia  1948   6.44 149.  1940
## 4 Australia  1949   6.61 126.  1940
## 5 Australia  1950   6.92 110.  1950
## 6 Australia  1951   4.27  87.1 1950
```

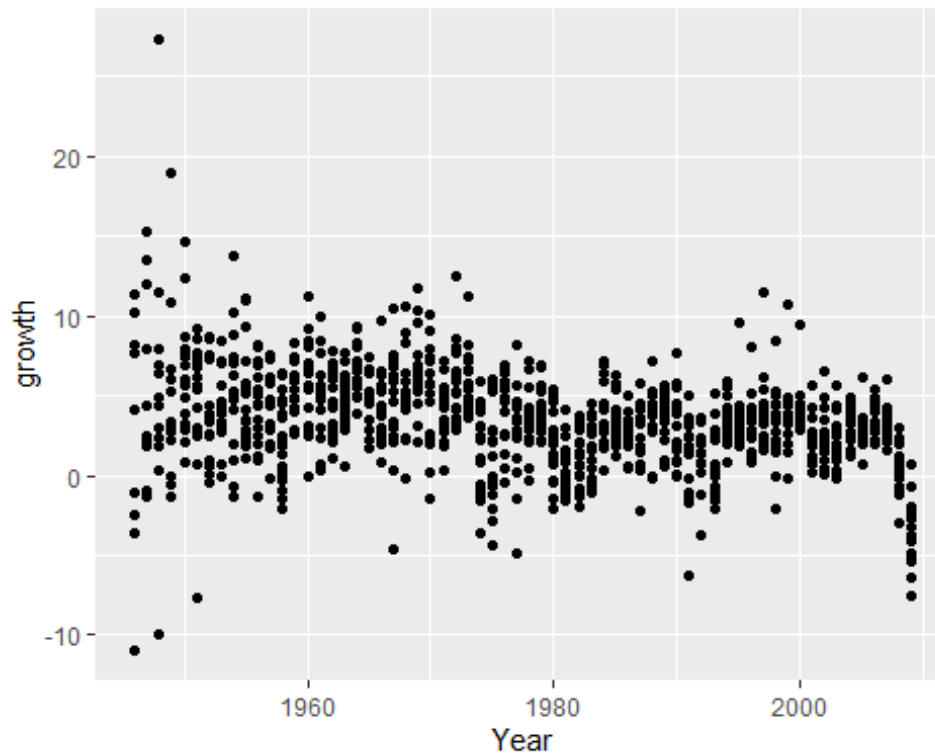
Visualisation des données

#La commande en R pour visualiser les données

```
with(debt, plot(Year, growth))
```



```
#La commande avec ggplot2 pour visualiser les données  
with(debt, qplot(Year, growth))
```

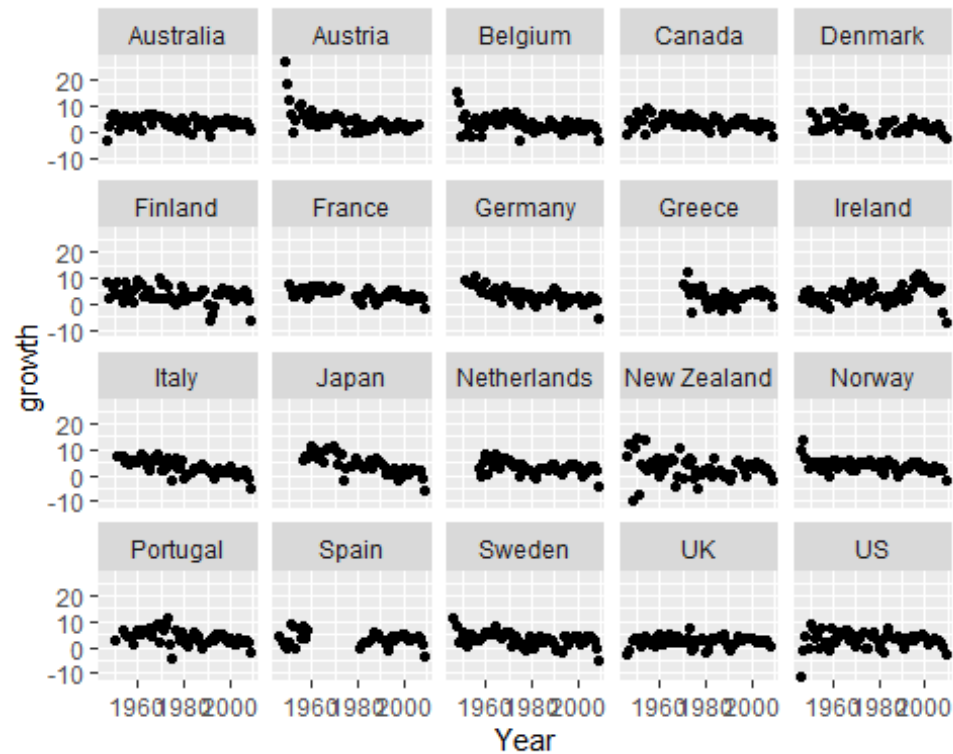


Visualisation par “petits multiples”

Maintenant, essayons de mieux comprendre l'évolution de ces taux de croissance du PIB au fil des ans en fonction des pays.

L'élément `facet_wrap(~ Country)` signifie construire le graphique pour chaque valeur différente de la variable “Country”.

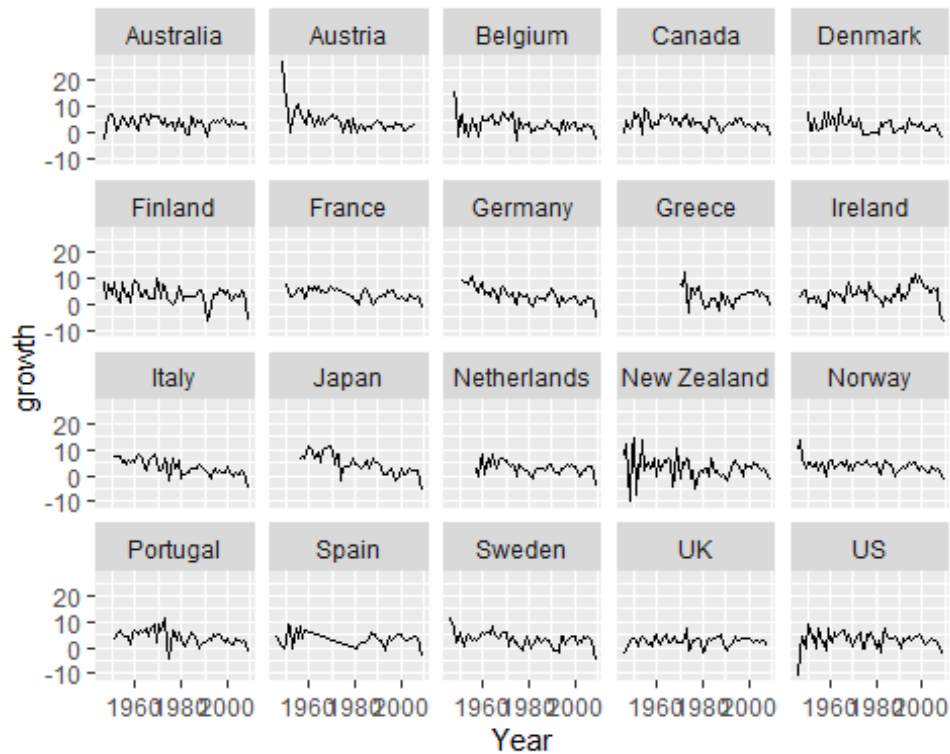
```
qplot(Year, growth, data = debt) +  
  facet_wrap(~ Country)
```



Visualisation en séries temporelles

L'argument `geom = "line"` permet d'utiliser des lignes plutôt que des points, cependant cela peut poser problème pour les graphiques où il manque des données. (cf: Espagne)

```
qplot(data = debt, y = growth, x = Year, geom = "line") +  
  facet_wrap(~ Country)
```



Composition graphique avec ggplot2

Commençons par créer un treillis de base au graphique :

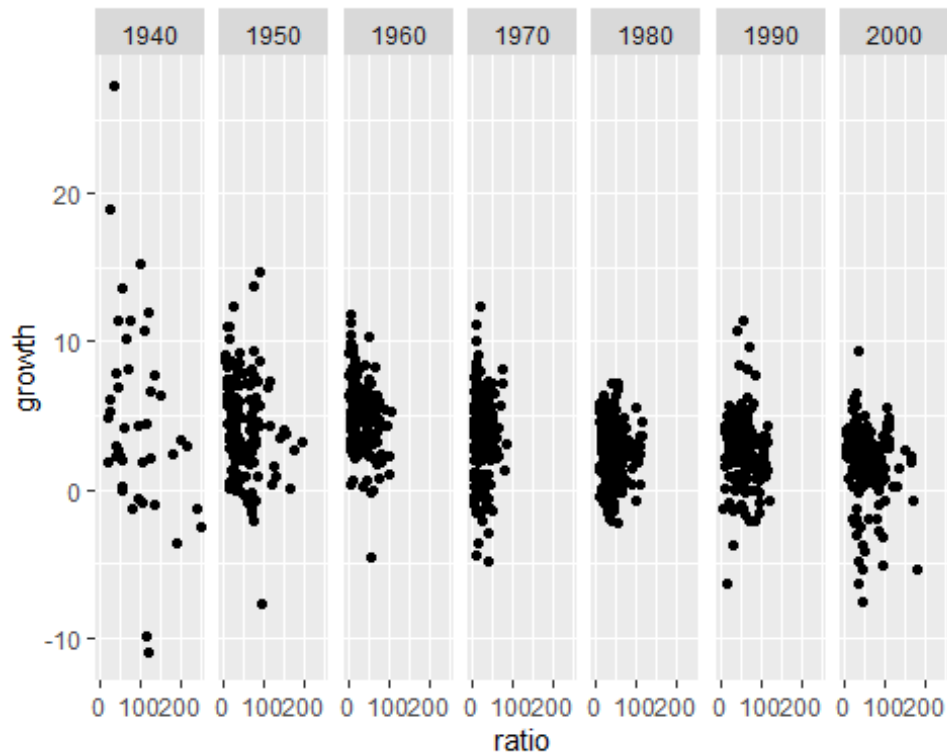
```
p <- ggplot(data = debt, aes(y = growth, x = ratio))
```

Cette commande permet de stocker dans p une base de travail.

Growth = le taux de croissance du PIB

ratio = Dette publique / PIB

```
p + geom_point() +  
  facet_grid(. ~ Decade) +  
  scale_x_continuous(breaks = seq(0, 200, by = 100))
```



Ce graphique propose une décomposition temporelle de la relation.

Nous pouvons donc maintenant sauvegarder dans l'objet `p` afin de pouvoir le modifier.

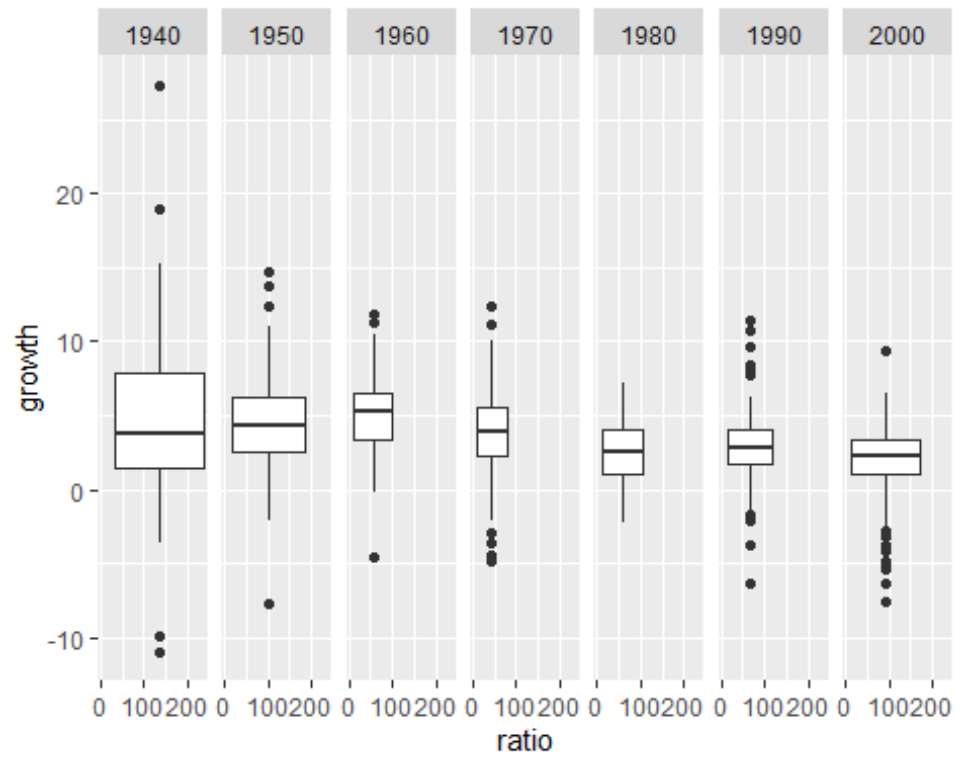
```
p <- p + geom_point() +
  facet_grid(. ~ Decade) +
  scale_x_continuous(breaks = seq(0, 200, by = 100))
```

Autres exemples avec geom

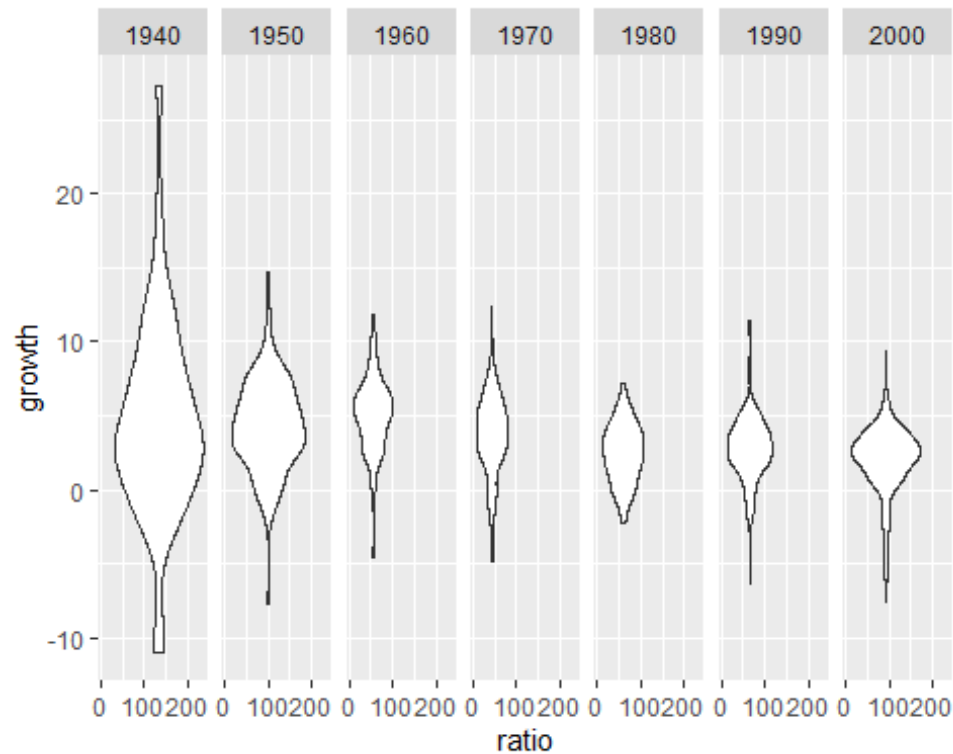
Avec des Boxplots

```
bp <- ggplot(data = debt, aes(y = growth, x = ratio))
bp + geom_boxplot() +
  facet_grid(. ~ Decade) +
  scale_x_continuous(breaks = seq(0, 200, by = 100))
```

```
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
```



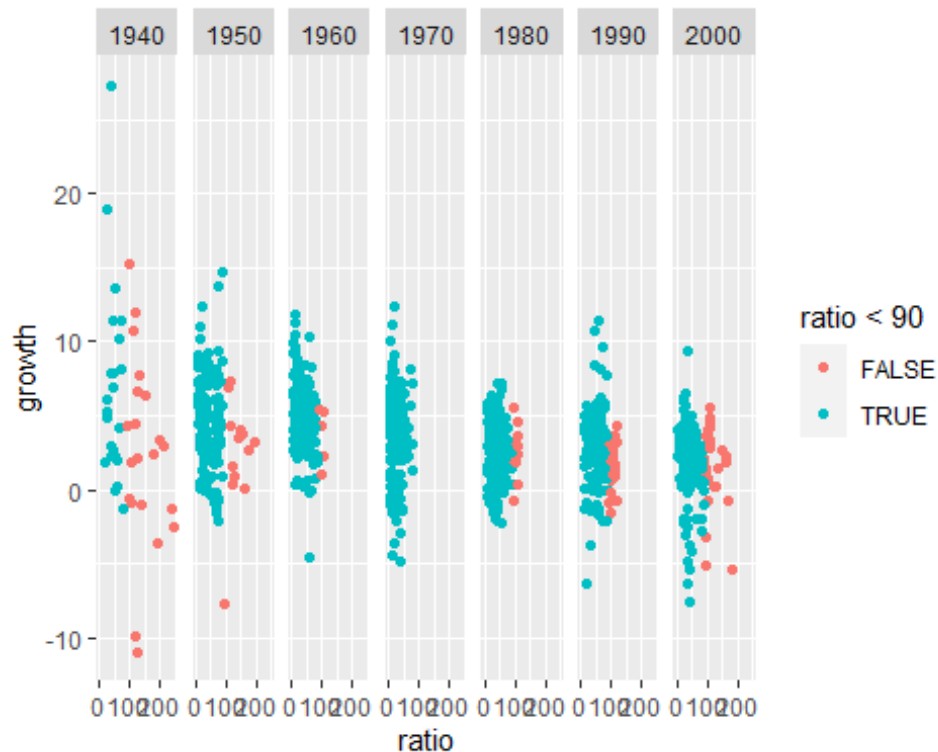
```
# En "violon"
vi <- ggplot(data = debt, aes(y = growth, x = ratio))
vi + geom_violin() +
  facet_grid(. ~ Decade) +
  scale_x_continuous(breaks = seq(0, 200, by = 100))
```

Geom est donc le moyen de mettre en forme votre graphique comme vous le souhaitez. Il existe un grand nombre de geom, décrits en détail dans la documentation officielle : <https://ggplot2.tidyverse.org/reference/>

Couleur et échelles

```
p + aes(color = ratio < 90)
```



color = ratio < 90 permet de modifier la couleur en fonction d'une inégalité => si ratio (= dette publique / PIB) inférieur à 90, alors, modifier la couleur

Exports des graphiques

ggplot2 permet également la sauvegarde des graphiques en PNG ou en PDF vectoriser pour garder la netteté du graphique. Cette commande permet d'enregistrer le dernier graphique affiché par ggplot2.

```
#Sauvegarde en PNG
ggsave("reinhart-rogooff.png", width = 11, height = 8)

#sauvegarde en PDF grâce à L'objet P
ggsave("reinhart-rogooff.pdf", plot = p,
       width = 11, height = 8)
```

Conclusion

Vous l'aurez compris, ggplot2 est une extension très complète et efficace, qui permet la réalisation de graphique complexe simplement.

Bibliographie

<http://larmarange.github.io/analyse-R/intro-ggplot2.html>

<https://juba.github.io/tidyverse/08-ggplot2.html>

<https://webdevdesigner.com/q/2-column-section-in-r-markdown-7438/>