

Representado las series temporales con ggplot2

Análisis estadístico de series económicas

Xavier Barber

Departamento de Estadística, Matemáticas e Informática
Centro de Investigación Operativa
Universitas Miguel Hernández de Elche

17/Feb/2019

Plot Time Series Data Using GGPlot

Valencia Bayesian Research group

Datos

```
library(ggplot2)
theme_set(theme_minimal())
# Demo dataset
(head(economics))
```

date	pce	pop	psavert	uempmed	unemploy
1967-07-01	507.4	198712	12.5	4.5	2944
1967-08-01	510.5	198911	12.5	4.7	2945
1967-09-01	516.3	199113	11.7	4.6	2958
1967-10-01	512.9	199311	12.5	4.9	3143
1967-11-01	518.1	199498	12.5	4.7	3066
1967-12-01	525.8	199657	12.1	4.8	3018

Gráfico básico

```
# Gráfico básico de líneas
```

```
ggplot(data = economics, aes(x = date, y = pop)) +  
  geom_line(color = "#00AFBB", size = 2)
```

```
# Sólo un subconjunto
```

```
ss <- subset(economics, date > as.Date("2006-1-1"))  
ggplot(data = ss, aes(x = date, y = pop)) +  
  geom_line(color = "#FC4E07", size = 2)
```

Gráfico básico

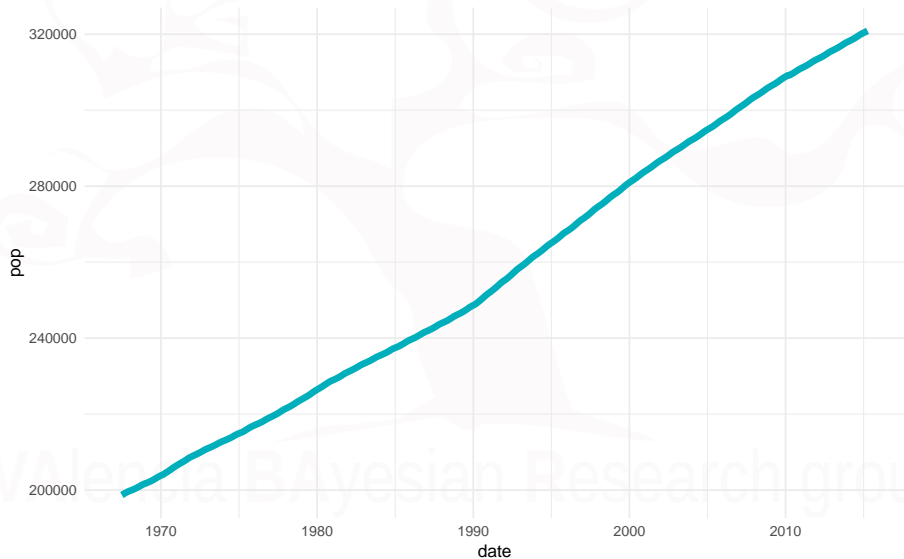
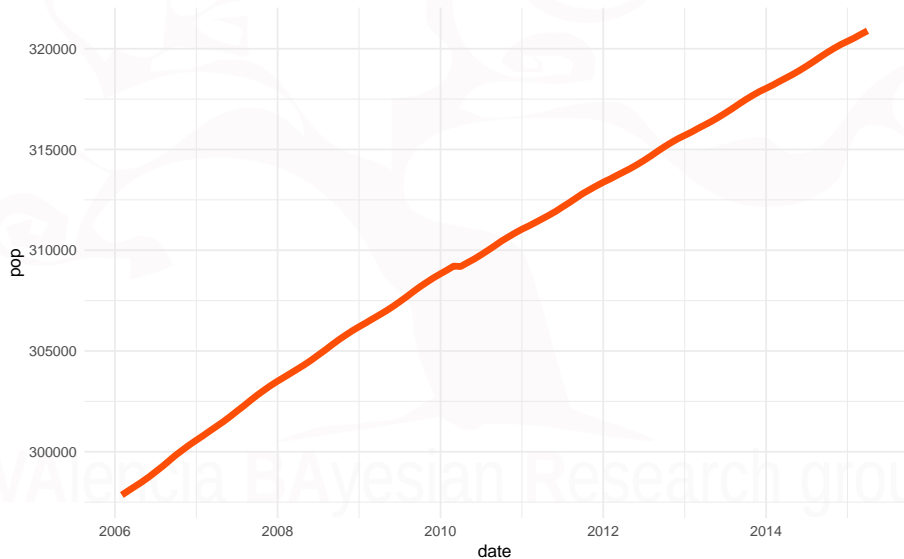


Gráfico básico (subconjunto)

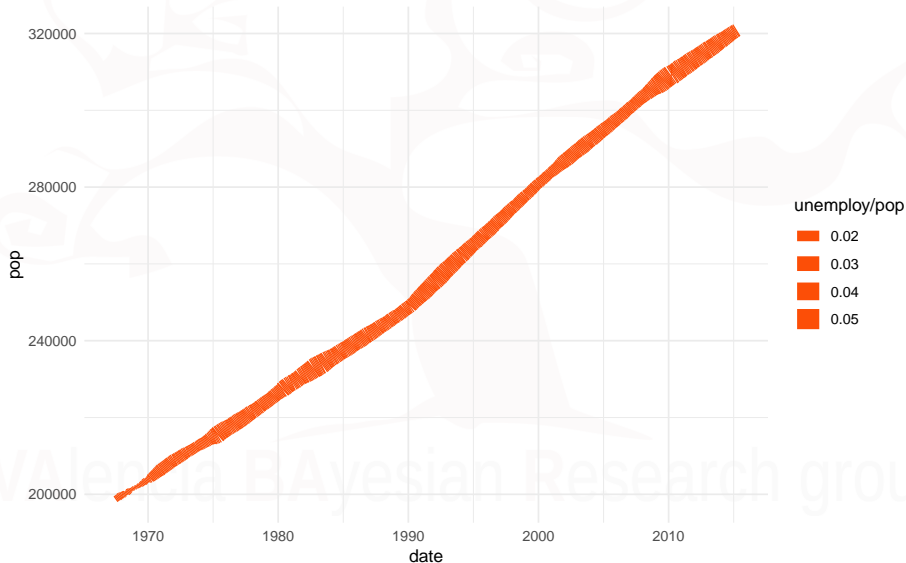


Grosor de línea según valor

```
ggplot(data = economics,  
       aes(x = date, y = pop)) +  
  geom_line(aes(size = unemploy/pop),  
           color = "#FC4E07")
```

Valencia Bayesian Research group

Grosor de línea según valor



Dibujando 2 series

```
library(tidyr)
library(dplyr)
df <- economics %>%
  select(date, psavert, uempmed) %>%
  gather(key = "variable", value = "value", -date)
head(df, 3)
```

date	variable	value
1967-07-01	psavert	12.5
1967-08-01	psavert	12.5
1967-09-01	psavert	11.7

psavert: personal savings rate

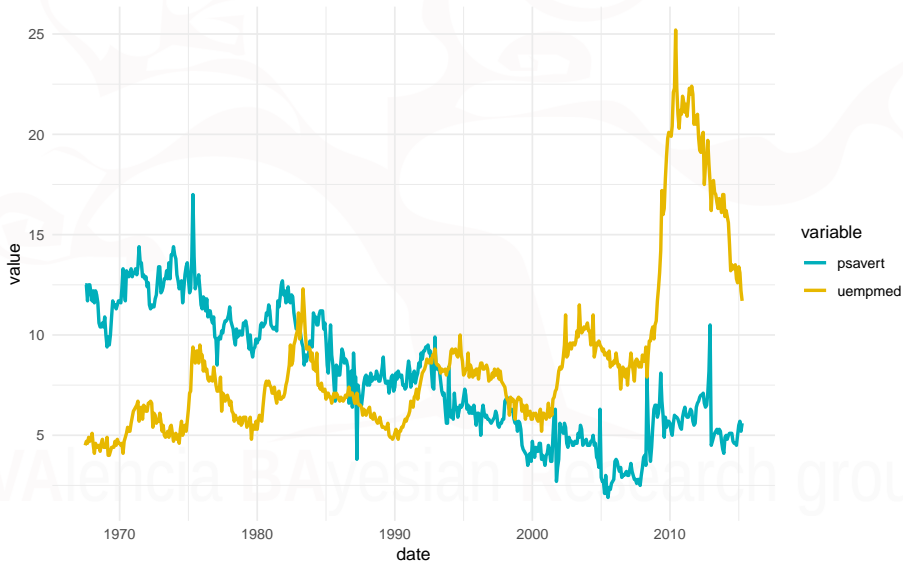
unemploy: number of unemployed in thousands

Dibujando 2 series

```
ggplot(df, aes(x = date, y = value)) +  
  geom_line(aes(color = variable), size = 1) +  
  scale_color_manual(values=c("#00AFBB", "#E7B800")) +  
  theme_minimal()
```

Valencia Bayesian Research group

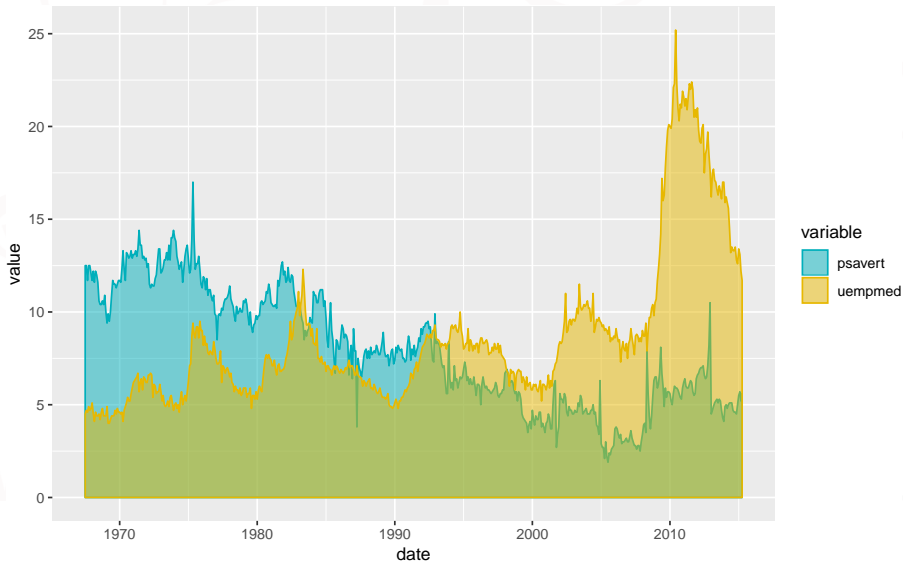
Dibujando 2 series



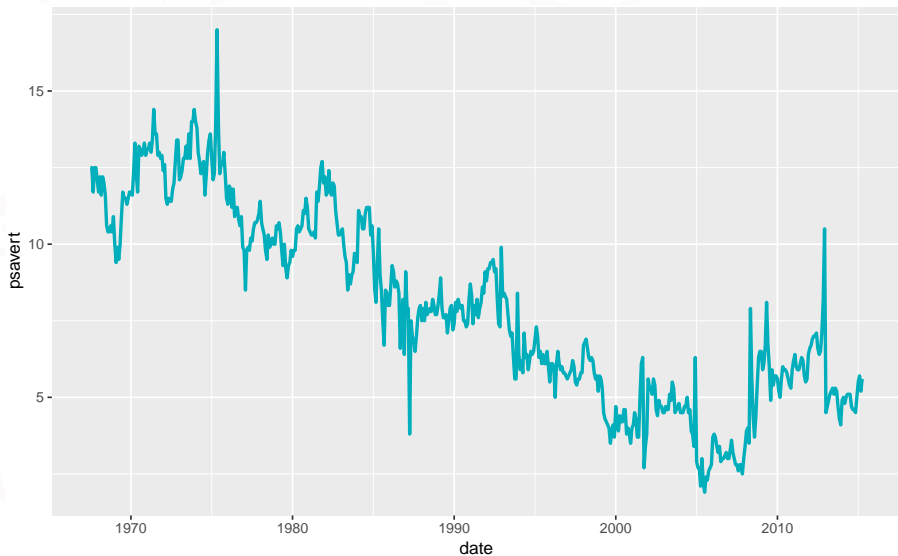
Dibujando 2 series (áreas)

```
ggplot(df, aes(x = date, y = value)) +  
  geom_area(aes(color = variable, fill = variable),  
            alpha = 0.5, position=position_dodge(0.8))+  
  scale_color_manual(values=c("#00AFBB", "#E7B800"))+  
  scale_fill_manual(values=c("#00AFBB", "#E7B800"))
```

Dibujando 2 series (áreas)



Definiendo el eje x



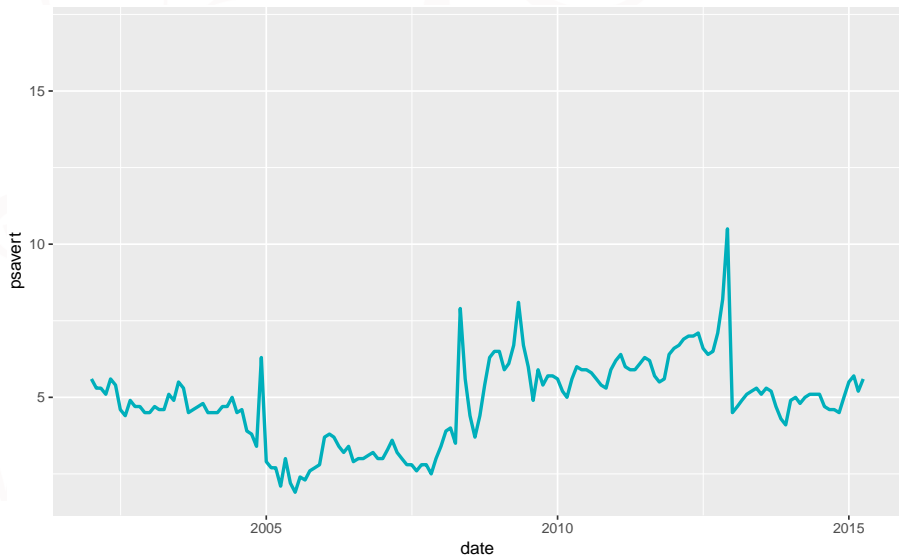
Cambiando el eje x

```
p <- ggplot(data = economics, aes(x = date,
                                  y = psavert)) +
  geom_line(color = "#00AFBB", size = 1)

p

# Set axis limits c(min, max)
min <- as.Date("2002-1-1")
max <- NA
p + scale_x_date(limits = c(min, max))
```

Cambiando el eje y



Etiquetas del eje x

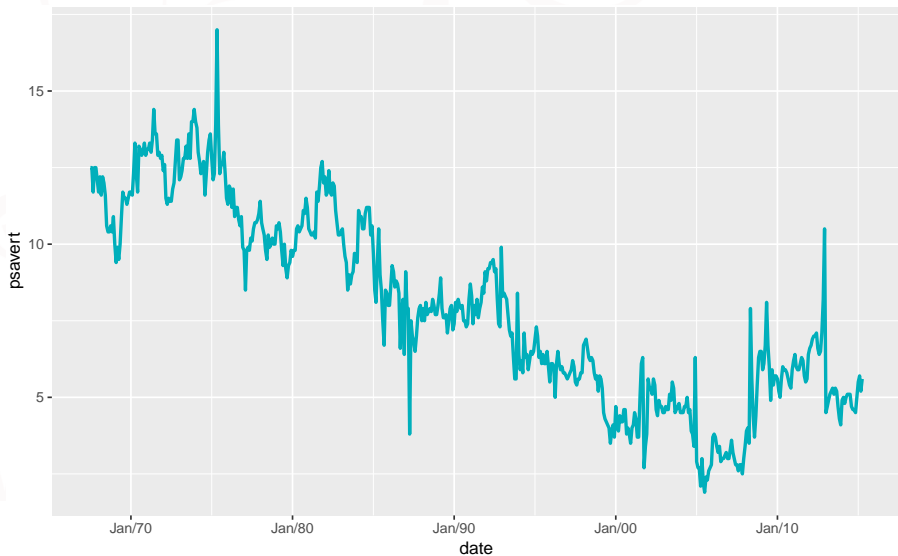
`scale_x_date()`:

- Utilizar `%a` y `%A` para el día de la semana (letra).
- Para utilizar el mes `%b` y `%B` (abreviado o completo).
- Si queremos el día en número utilizaremos `%d`.
- Para visualizar el año `%y` y `%Y` (abreviado o completo).

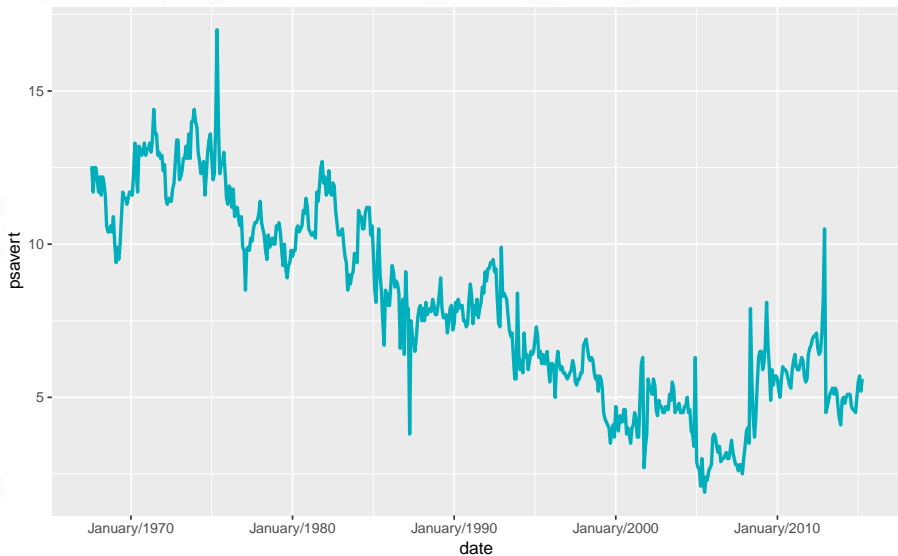
```
# Format : month/year
```

```
p + scale_x_date(date_labels = "%b/%y")
```

Etiquetas del eje x



Etiquetas del eje x



Extensiones para objetos ts

- El paquete `ggfortify` es una extensión del `ggplot`, el cual hace que representar objetos de series temporales sea más fácil (Horikoshi and Tang 2017).

`zoo::zooreg()`, `xts::xts()`,
`timeSeries::timSeries()`,
`tseries::irts()`,
`forecast::forecast()`, `vars::vars()`,

```
install.packages(
  c("ggfortify", "changepoint",
    "strucchange", "ggpmisc"))
```

Extensiones para objetos ts

- Otro paquete interesante es el `ggpmisc` (Aphalo 2017), el cual nos porvee de dos métodos muy útiles para ojetos tipo series temporales:

stat_peaks(): máximos locales en el eje x , y

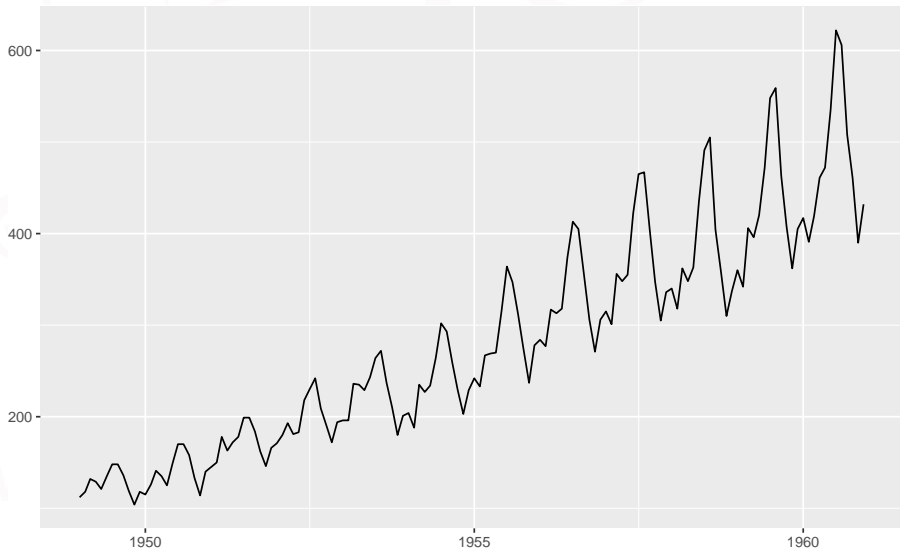
stat_valleys(): mínimos locales del eje del tiempo.

Valencia Bayesian Research group

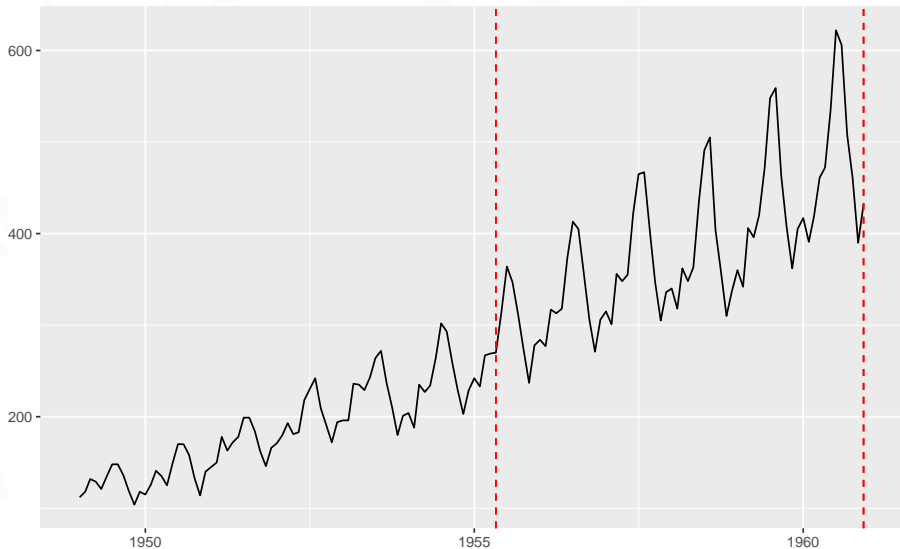
Extensiones para objetos ts

```
library(ggfortify)
library(magrittr) # for piping %>%
# Plot ts objects
autoplot(AirPassengers)
# Identify change points in mean and variance
AirPassengers %>%
  changepoint::cpt.meanvar() %>% # Identify change points
  autoplot()
# Detect jump in a data
strucchange::breakpoints(Nile ~ 1) %>%
  autoplot()
```

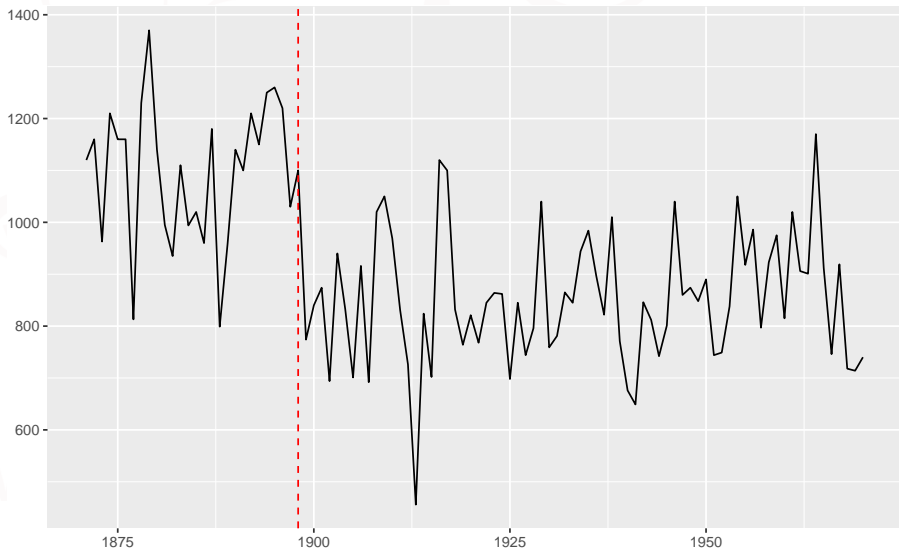
Extensiones para objetos ts



Extensiones: changepoint



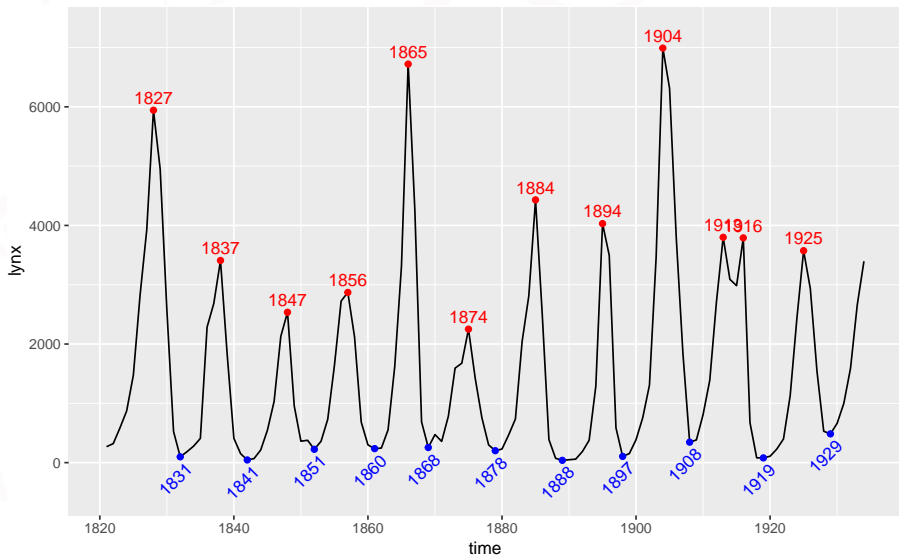
Extensiones: strucchange



Extensiones: *stat_peaks*

```
library(ggpmisc)
ggplot(lynx, as.numeric = FALSE) + geom_line() +
  stat_peaks(colour = "red") +
  stat_peaks(geom = "text", colour = "red",
            vjust = -0.5, x.label.fmt = "%Y") +
  stat_valleys(colour = "blue") +
  stat_valleys(geom = "text", colour = "blue", angle = 45,
            vjust = 1.5, hjust = 1, x.label.fmt = "%Y")+
  ylim(-500, 7300)
```

Extensiones: *stat_peaks*



Representando x meses: *facet_wrap*

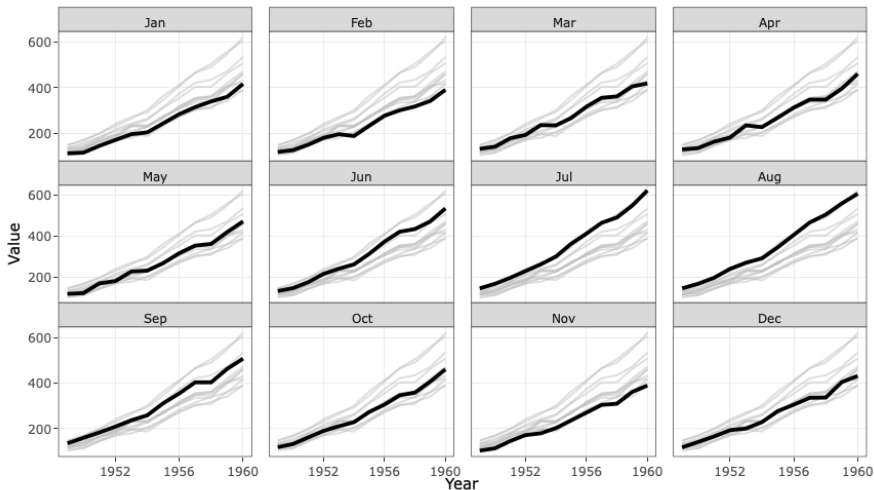
```

library(plotly)
# Air passenger data. ts converted to long matrix:
myData <- data.frame(Year=c(floor(time(AirPassengers)+.01)),
                    Month = c(cycle(AirPassengers)),
                    Value = c(AirPassengers))

# convert month numbers to names, using a built-in constant:
myData$Month <- factor(myData$Month)
levels(myData$Month) <- month.abb
# plotting reference lines across each facet:
referenceLines <- myData # \ / Rename
colnames(referenceLines)[2] <- "groupVar"
zp <- ggplot(myData, aes(x = Year, y = Value))
zp <- zp + geom_line(data = referenceLines, #Plotting the "underlayer"
                    aes(x = Year, y = Value, group = groupVar),
                    colour = "GRAY", alpha = 1/2, size = 1/2)
zp <- zp + geom_line(size = 1) # Drawing the "overlayer"
zp <- zp + facet_wrap(~ Month)
zp <- zp + theme_bw()
ggplotly()

```

Representando x meses: *facet_wrap*



Referencias

- Aphalo, Pedro J. 2017. Ggpmisc: Miscellaneous Extensions to 'Ggplot2'.
<https://CRAN.R-project.org/package=ggpmisc>.
- Horikoshi, Masaaki, and Yuan Tang. 2017. Ggfortify: Data Visualization Tools for Statistical Analysis Results.
<https://CRAN.R-project.org/package=ggfortify>.
- Plotly Team. Use plotly offline.
<https://goo.gl/DThWfV>
- Statistical tools for high-throughput data analysis.
<https://goo.gl/7rK3eX>