



DASHBOARDS, PIZZAS, CONTAMINACIÓN Y R... ¿UN MIX CON POCA RELACIÓN?

María Medina Pérez
@mariamedp



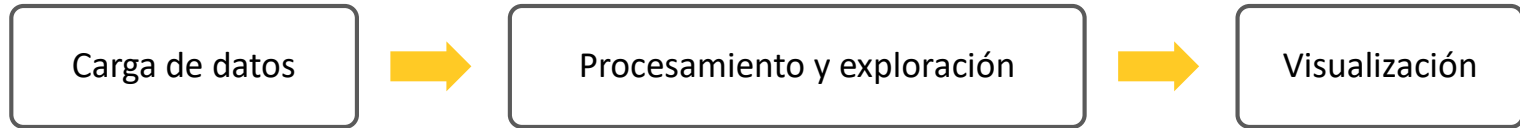
¿QUÉ OS VOY A CONTAR?



@mariamedp

R-Ladies Madrid Meetup: R + Negocios - 31/01/2017

ANÁLISIS DESCRIPTIVO



¿Cómo presentamos los resultados?

- Slides
- Reporting
- **FLEXDASHBOARD**



ANÁLISIS DESCRIPTIVO



```
1 ---
2 title: "Flexdashboard de ejemplo en R Ladies"
3 author: "María Medina"
4 date: "31/01/2017"
5 output:
6   flexdashboard::flex_dashboard:
7     orientation: rows
8 ---
9
10 {r, echo=FALSE}
11 library(DT); library(ggplot2); library(plotly)
12
13
14 El conjunto de datos
15
16 El dataset 'diamonds' contiene una serie de atributos de distintos diamantes. Está en el paquete 'ggplot2'.
17
18 **Datos:**
19
20 {r, echo=FALSE}
21 DT::datatable(diamonds)
22
23
24 Visualizaciones
25
26
27 Row
28
29 {r, fig.width=20, fig.height=5}
30 p <- ggplot(diamonds, aes(x=price, fill=cut)) + geom_histogram(binwidth=200) +
31   labs(x="Precio", y="Frecuencia", fill="Calidad") + ggtitle("Histograma de precios")
32 ggplotly(p)
33
34 Column
35
36 {r, fig.width=10, fig.height=5}
37 p <- ggplot(diamonds, aes(x=carat, y=price, color=cut)) + geom_point(size=0.5) +
38   labs(x="Peso", y="Precio", color="Calidad") + ggtitle("Precio según peso - valores")
39 ggplotly(p)
40
41 {r, fig.width=10, fig.height=5}
42 p <- ggplot(diamonds, aes(x=carat, y=price, color=cut)) + geom_smooth() +
43   labs(x="Peso", y="Precio", color="Calidad") + ggtitle("Precio según peso - tendencia")
44 ggplotly(p)
45
46
47
48
49
50
51
```

<http://rmarkdown.rstudio.com/flexdashboard/>



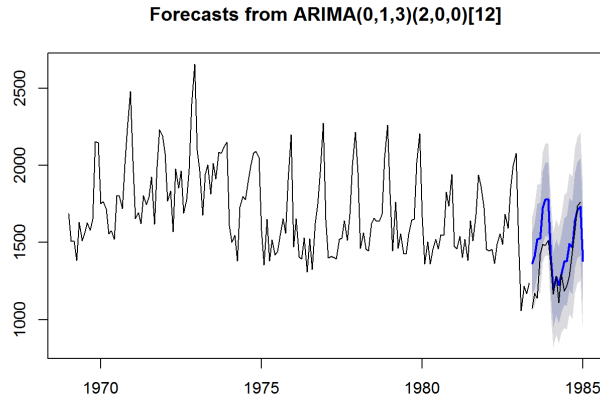
@mariamedp

R-Ladies Madrid Meetup: R + Negocios - 31/01/2017

PREDICCIÓN DEMANDA ALIMENTARIA

Objetivo: estimar los kg de producto que se van a vender cada día en España.

Aproximación 1: usar únicamente el histórico de ventas.



```
library(forecast)
```

<http://robjhyndman.com/software/forecast/>

```
lambda <- BoxCox.lambda(tseries_train, lower=0)
fit <- auto.arima(tseries_train, lambda=lambda,
                 stepwise=FALSE, approximation=FALSE)

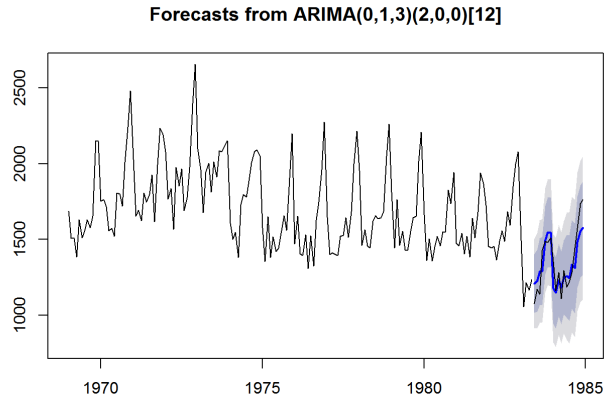
fc <- forecast(fit, h=length(tseries)-ind_ini_test+1)

plot(fc)
lines(tseries_test)
```

PREDICCIÓN DEMANDA ALIMENTARIA

Objetivo: estimar los kg de producto que se van a vender cada día en España.

Aproximación 2: incorporar variables externas (calendarios, promociones).



```
library(forecast)
```

<http://robjhyndman.com/software/forecast/>

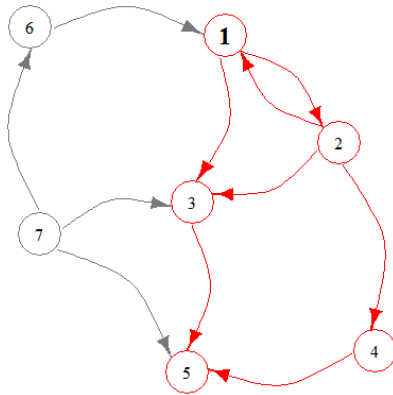
```
lambda <- BoxCox.lambda(tseries_train, lower=0)
fit_varextra <- auto.arima(tseries_train, lambda=lambda, xreg=varextra_train,
                           stepwise=FALSE, approximation=FALSE)

fc_varextra <- forecast(fit_varextra, xreg=varextra_test, h=length(tseries)-ind_ini_test+1)

plot(fc_varextra)
lines(tseries_test)
```

ACCESIBILIDAD INFORMACIÓN EN LA WEB

Objetivo: medir cómo de buena es la estructura de una página web.



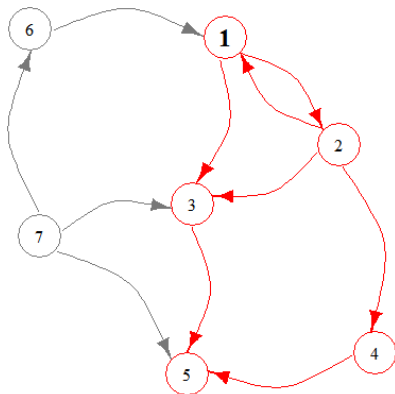
```
library(igraph) http://igraph.org/r/
```

```
g <- graph.data.frame(crawler_data[c("url", "link")])  
E(g)$weight <- crawler_data$count
```

```
subg <- graph.neighborhood(g, nodes=main_web, order=MAX_DEPTH, mode="out")[[1]]
```

ACCESIBILIDAD INFORMACIÓN EN LA WEB

Objetivo: medir cómo de buena es la estructura de una página web.



```
n_reachable_pages <- vcount(subg) - 1

reachable_pages <- setdiff(V(subg)$name, main_web)

sp_forw <- shortest.paths(subg, mode="out", weights=NA, v=main_web, to=reachable_pages)
avg_distance_from_index <- mean(sp_forw)


sp_backw <- shortest.paths(subg, mode="out", weights=NA, v=reachable_pages, to=main_web)
n_unreachable_index <- sum(is.infinite(sp_backw))
avg_distance_to_index <- mean(sp_backw[!is.infinite(sp_backw)])
```


CONTAMINACIÓN EN MADRID

Objetivo: informar de los niveles de contaminación por NO₂ en Madrid.

 @datoxnitro_bot

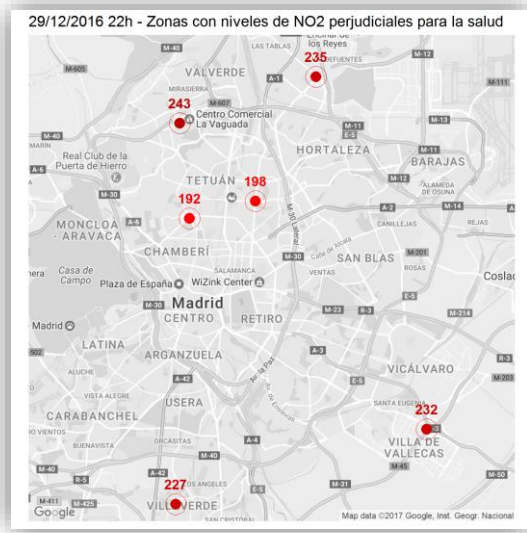


 @datoxnitro_bot

ÚLT. HORA 🚨 Niveles para Escenario 1 por alta #contaminaciónmadrid.
El Ayto. de @MADRID podría activar restricciones para el sábado 31dic

CONTAMINACIÓN EN MADRID

Objetivo: informar de los niveles de contaminación por NO₂ en Madrid.



`library(ggmap)`

<https://github.com/dkadle/ggmap>

```
center_coords <- c(-3.670205, 40.425208)
map <- get_map(location=center_coords, zoom=12, maptype="roadmap", color="bw", source="google")
```

```
# Niveles en la escala de color
cortes <- seq(180, 300, 20)
red_palette <- colorRampPalette(colors=c("red", "dark red"))(length(cortes) - 1)
airp_umbral$no2_bin <- cut(airp_umbral$no2, cortes)

# Visualización mapa
ggmap(map) +
  geom_point(data=airp_umbral, aes(x=lon, y=lat, col=no2_bin), size=5, shape=19) +
  geom_point(data=airp_umbral, aes(x=lon, y=lat, col=no2_bin), size=10, shape=1) +
  geom_text(data=airp_umbral, aes(x=lon, y=lat, label=no2, col=no2_bin),
            hjust=0.5, vjust=-1.5, size=5, fontface="bold") +
  scale_color_manual(values=setNames(red_palette, levels(airp_umbral$no2_bin))) +
  guides(col=FALSE) + labs(x=NULL, y=NULL) +
  ggtitle("29/12/2016 22h - Zonas con niveles de NO2 perjudiciales para la salud") +
  theme(title=element_text(size=13), axis.ticks=element_blank(), axis.text=element_blank())
```

¿UN MIX CON POCA RELACIÓN?

¡NO! ES UN -MIX



¡GRACIAS!



<https://github.com/mariamedp/rladies-negocio>



@mariamedp