



Clase 8

# Factores y Anova

Miriam Lerma

Marzo 2021

# Intro

- Factores
- Analisis de varianza

# Ustedes

- Conocimientos de R (saben abrirlo, cargar paquetes y datos, saben hacer operaciones y graficos).
- Quieren saber como transformar a factor y conocer la sintaxis para hacer analisis de varianza en R.

## Notas

Ya vieron teoría.

Recuerden que los modelos dependen de sus preguntas y experimentos o muestreos.

# Créditos & materiales:

## Materiales

 Sthda por Alboukadel Kassambara


 Handbook of Regression Models in People Analytics

 Tutoriales STAT 545

 ourcodingclub

 Libro por Steve Midway

## Imágenes adicionales

 Unsplash

 Portada Unsplash por Thomas Millot



# 1. Factores

# 1.1. Titanic

Instalar y cargar el paquete

```
#install.packages('titanic')  
library(titanic)  
library(tidyverse)
```

Renombremos el dataframe que vamos a usar y agreguemoslo a nuestro environment.

```
Titanic_datos<-titanic_train
```

Vamos a usar estos datos por que no están "limpios".

La idea es que **consideren** que algunas variables se deben transformar a **factores**.

# 1.2. Columnas y valores

Renombrar columnas para que estén en español.

```
Titanic_datos <- Titanic_datos %>%  
  rename(sobrevivio=Survived,  
         clase=Pclass,  
         edad=Age,  
         sexo=Sex,  
         embarcado=Embarked,  
         precio=Fare)
```

Transformamos los espacios vacíos ("" ) a NA

```
Titanic_datos$embarcado <- ifelse(Titanic_datos$embarcado == "",  
                                 NA,  
                                 Titanic_datos$embarcado)
```

# 1.3. Factores

Que tipos de datos tenemos?

- **int**: integral, numérico sin decimales
- **chr**: character
- **dbl**: double, es un tipo numérico de doble precisión
- **fact**: factor? No hay ninguno con esta clase.

```
glimpse(Titanic_datos)
```

```
## Rows: 891
## Columns: 12
## $ PassengerId <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,
## $ sobrevivio <int> 0, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0
## $ clase <int> 3, 1, 3, 1, 3, 3, 1, 3, 3, 2, 3, 1, 3, 3, 3, 2, 3, 2, 3
## $ Name <chr> "Braund, Mr. Owen Harris", "Cumings, Mrs. John Bradley
## $ sexo <chr> "male", "female", "female", "female", "male", "male", "f
## $ edad <dbl> 22, 38, 26, 35, 35, NA, 54, 2, 27, 14, 4, 58, 20, 39, 1
## $ SibSp <int> 1, 1, 0, 1, 0, 0, 0, 3, 0, 1, 1, 0, 0, 1, 0, 0, 4, 0, 1
## $ Parch <int> 0, 0, 0, 0, 0, 0, 0, 1, 2, 0, 1, 0, 0, 5, 0, 0, 1, 0, 0
## $ Ticket <chr> "A/5 21171", "PC 17599", "STON/O2. 3101282", "113803",
## $ precio <dbl> 7.2500, 71.2833, 7.9250, 53.1000, 8.0500, 8.4583, 51.86
## $ Cabin <chr> "", "C85", "", "C123", "", "", "E46", "", "", "", "G6",
## $ embarcado <chr> "S", "C", "S", "S", "S", "Q", "S", "S", "S", "S", "C", "S",
```

# 1.3. Factores

Transformamos columnas que son factores a factor.  
**fct**: factor? Ahora si existen columnas con esta clase.

```
Titanic_datos<-Titanic_datos%>%  
  mutate(sobrevivio = as_factor(sobrevivio),  
         clase = as_factor(clase),  
         sexo = as_factor(sexo))
```

```
glimpse(Titanic_datos)
```

```
## Rows: 891  
## Columns: 12  
## $ PassengerId <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,  
## $ sobrevivio <fct> 0, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0  
## $ clase <fct> 3, 1, 3, 1, 3, 3, 1, 3, 3, 2, 3, 1, 3, 3, 3, 2, 3, 2, 3  
## $ Name <chr> "Braund, Mr. Owen Harris", "Cumings, Mrs. John Bradley  
## $ sexo <fct> male, female, female, female, male, male, male, male, f  
## $ edad <dbl> 22, 38, 26, 35, 35, NA, 54, 2, 27, 14, 4, 58, 20, 39, 1  
## $ SibSp <int> 1, 1, 0, 1, 0, 0, 0, 3, 0, 1, 1, 0, 0, 1, 0, 0, 4, 0, 1  
## $ Parch <int> 0, 0, 0, 0, 0, 0, 0, 1, 2, 0, 1, 0, 0, 5, 0, 0, 1, 0, 0  
## $ Ticket <chr> "A/5 21171", "PC 17599", "STON/O2. 3101282", "113803",  
## $ precio <dbl> 7.2500, 71.2833, 7.9250, 53.1000, 8.0500, 8.4583, 51.86
```



# 1.4. Supervivencia

¿Cuál fue el número de supervivientes?

Vamos a crear un objeto con esa tabla.

```
Sobrevivientes<-Titanic_datos %>%  
  group_by(sobrevivio) %>%  
  count()
```

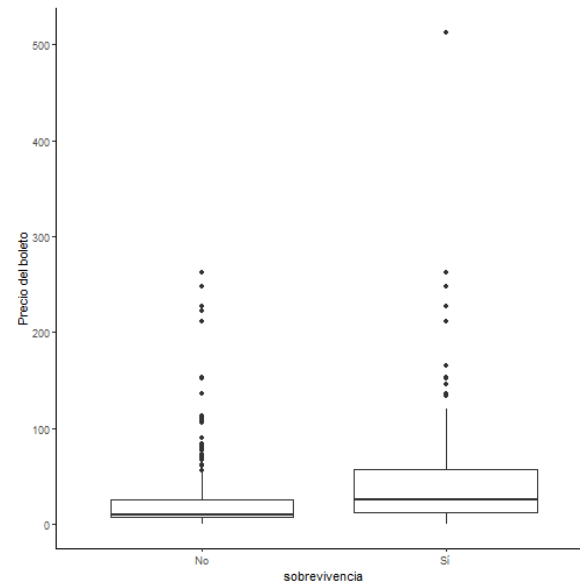
Usaremos la información de la tabla que creamos para agregar el número de supervivientes.

```
g1 <- ggplot(Titanic_datos,  
             aes(sobrevivio)) +  
  geom_bar()+  
  geom_text(data = Sobrevivientes  
            aes(sobrevivio,  
                y=25,  
                label=n),  
            color="white")+  
  xlab("supervivencia")+  
  ylab("Frecuencia")+  
  theme_classic()
```

# 1.4. Precio

Podemos explorar si los que tenían un boleto mas caro, tenían mas posibilidades de sobrevivir.

```
p1 <- ggplot(Titanic_datos,
             aes(x=sobrevivio,
                 y=precio)) +
  geom_boxplot()+
  xlab("sobrevivencia")+
  ylab("Precio del boleto")+
  theme_classic()+
  scale_x_discrete(breaks=c("0", "1"),
                  labels=c("No", "Sí"))
```



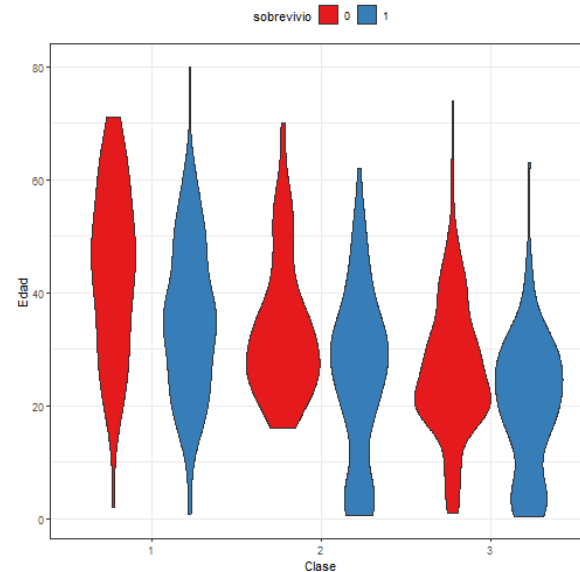
## Nuevos conceptos

- Podemos usar *scale\_x\_discrete* para cambiar la etiqueta en el eje x.

# 1.5. Clase y edad

Podemos explorar si los que tenían la edad tenía un efecto, además de la clase, en las posibilidades de sobrevivir.

```
c1<-ggplot(Titanic_datos,  
           aes(x= as.factor(clase), edad, fill=sobrevivio))+  
  geom_violin()+  
  xlab("Clase")+  
  ylab("Edad")+  
  theme_bw()+  
  theme(legend.position='top')+  
  scale_fill_brewer(palette = "Set1")
```



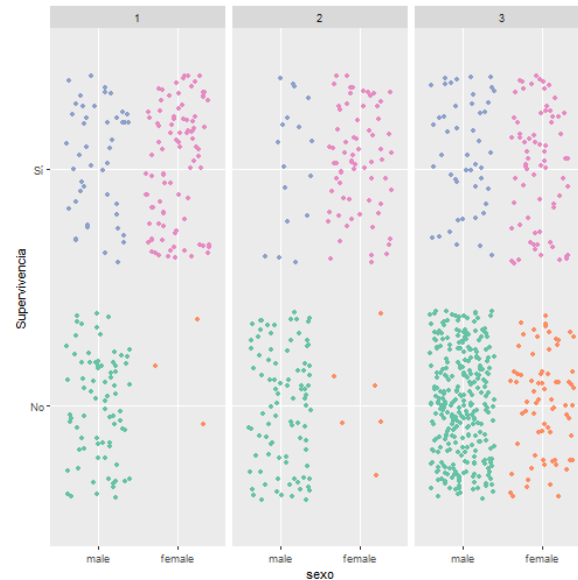
## Nuevos conceptos

- Podemos usar `scale_fill_brewer` para cambiar los colores de relleno.

# 1.6. Sexo y clase

Podemos explorar los datos separando por sexo, por clase y por si sobrevivieron o no.

```
p3<-ggplot(Titanic_datos,  
           aes(sexo, supervivio))+  
  geom_jitter(aes(color=interacti  
  
  facet_wrap(~clase)+  
  ylab("Supervivencia")+  
  scale_y_discrete(breaks=c("0", "  
    labels=c("No", "Sí"))+  
  theme(legend.position = "none")  
  scale_color_brewer(palette = "S
```



## Nuevos conceptos

- Podemos usar *geom\_jitter* para mover los puntos.
- Podemos usar *scale\_color\_brewer* para cambiar los colores de los puntos.

# Ejercicios

- Cargar **paquetes** de titanic y tidyverse
- **Renombrar** objeto, columnas y transformar espacios vacíos a NA
- Crear tres **gráficos** y cambiar clase de columnas a factor

---

Paquetes

Renombrar

NAs

Precio

Clase y edad

Clase y sexo

```
#install.packages('titanic')  
library(titanic)  
library(tidyverse)
```



**ANOVAS**

# 2.1. Teoría

## Teoría

El análisis de la varianza (ANOVA) se utiliza de forma intensiva en el análisis y diseño de experimentos para evaluar el efecto de tratamientos en la variabilidad de la variable respuesta.

Un análisis de la varianza permite determinar, por ejemplo, si diferentes tratamientos (es decir, un grupo de más de dos tratamientos) muestran diferencias significativas en sus resultados o si por el contrario puede suponerse que sus medias poblacionales no difieren.

## 2.2. Insectos

### Ejercicio

Estamos interesados en conocer si hay colores más atractivos para los insectos. Para ello se diseñaron trampas con los siguientes colores: amarillo, azul, blanco y verde.

Se cuantificó el número de insectos que quedaban atrapados.

Generemos los datos.

```
insectos <- c(16, 11, 20, 21, 14, 7, 37, 32, 15, 25, 39,  
             41, 21, 12, 14, 17, 13, 17, 45, 59, 48, 46, 38, 47)  
colores <- as.factor(c(rep(c("azul", "verde", "blanco", "amarillo"),  
                          each=6)))
```

**Nuevo concepto** *rep* es repetir ese factor, *each* seis veces.

Crear data frame.

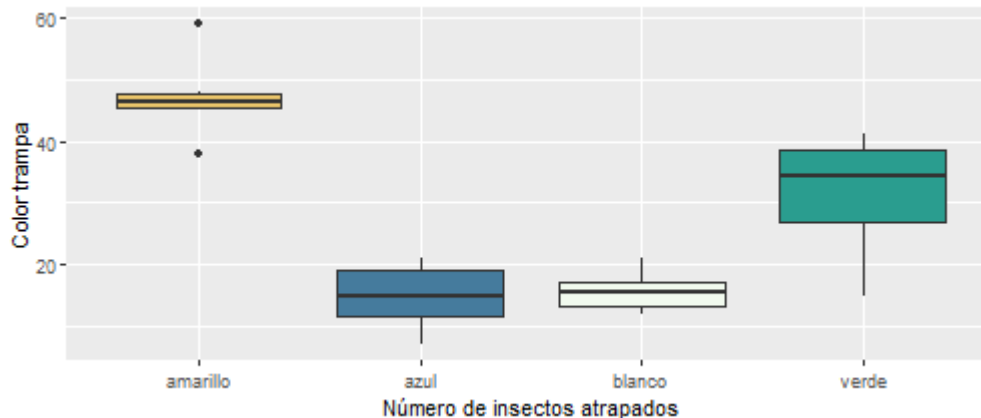
```
Insectos_df <- data.frame(insectos=insectos, colores=colores)
```



## 2.2. Insectos

Exploramos los datos

```
Trampas_Fig<-ggplot(Insectos_df,  
  aes(x=colores,y=insectos)) +  
  geom_boxplot(aes(fill=colores))+  
  xlab("Número de insectos atrapados")+  
  ylab("Color trampa")+  
  theme(legend.position = 'none')+  
  scale_fill_manual(values=c("#e9c46a", "#457b9d", "#f1faee", "#2a9d8f"))  
Trampas_Fig
```



**Nuevo concepto** `scale_fill_manual` para especificar los colores a usar.

## 2.3. ANOVA

Esta es la forma de pedir un ANOVA en R:

```
Anova_insectos<-aov(lm(insectos ~ colores))
```

Elementos generados en el ANOVA:

```
names(Anova_insectos)
```

```
## [1] "coefficients" "residuals" "effects" "rank"  
## [5] "fitted.values" "assign" "qr" "df.residual"  
## [9] "contrasts" "xlevels" "call" "terms"  
## [13] "model"
```

Igual que con los modelos lineales, pedimos un resumen de la tabla del ANOVA

```
summary(Anova_insectos)
```

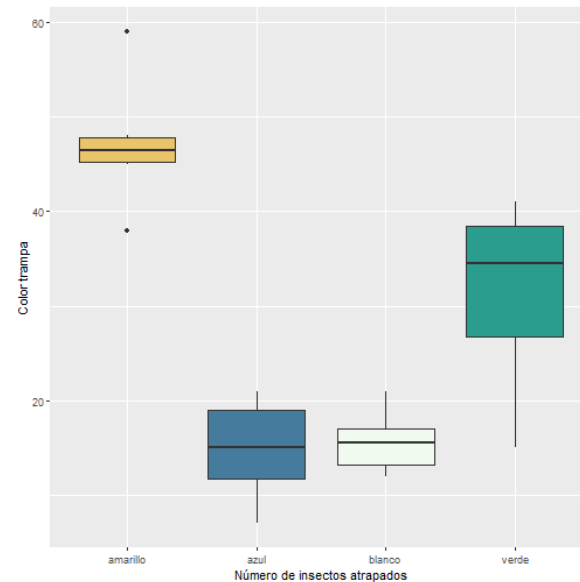
```
##           Df Sum Sq Mean Sq F value    Pr(>F)  
## colores      3   4218    1406  30.55 1.15e-07 ***  
## Residuals   20     921      46  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## 2.3. TukeyHSD

Si hemos detectado diferencias significativas entre las medias de las poblaciones. ¿Sería posible saber cuáles son los grupos que generan estas diferencias?

```
Insectos_df %>%  
  group_by(colores) %>%  
  summarise(promedio=mean(insecto
```

```
## # A tibble: 4 x 2  
##   colores promedio  
##   <fct>      <dbl>  
## 1 amarillo    47.2  
## 2 azul        14.8  
## 3 blanco     15.7  
## 4 verde      31.5
```



## 2.3. TukeyHSD

Prueba post-hoc.

```
intervalos<-TukeyHSD(Anova_insectos)
intervalos
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = lm(insectos ~ colores))
##
## $colores
##
```

	diff	lwr	upr	p adj
## azul-amarillo	-32.3333333	-43.296330	-21.37034	0.0000004
## blanco-amarillo	-31.5000000	-42.462996	-20.53700	0.0000006
## verde-amarillo	-15.6666667	-26.629663	-4.70367	0.0036170
## blanco-azul	0.8333333	-10.129663	11.79633	0.9964823
## verde-azul	16.6666667	5.703670	27.62966	0.0020222
## verde-blanco	15.8333333	4.870337	26.79633	0.0032835

## 2.4. Validación del modelo

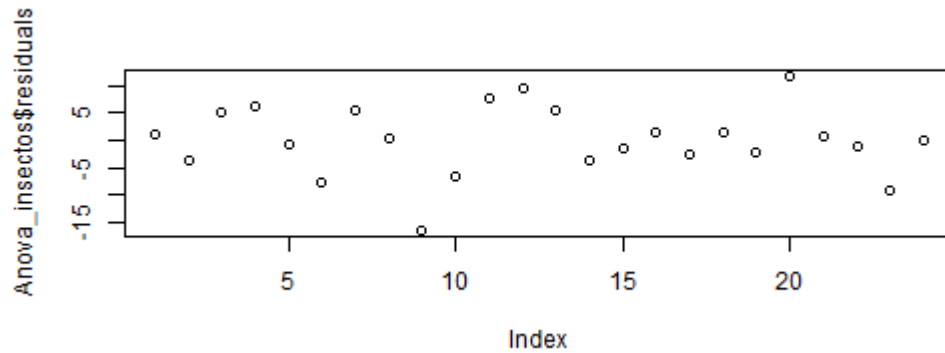
Los supuestos que se deben cumplir son tres:

- Independencia,
- homocedasticidad y
- normalidad.

# 2.5. Independencia

Los valores deben ser independientes.

```
plot(Anova_insectos$residuals)
```



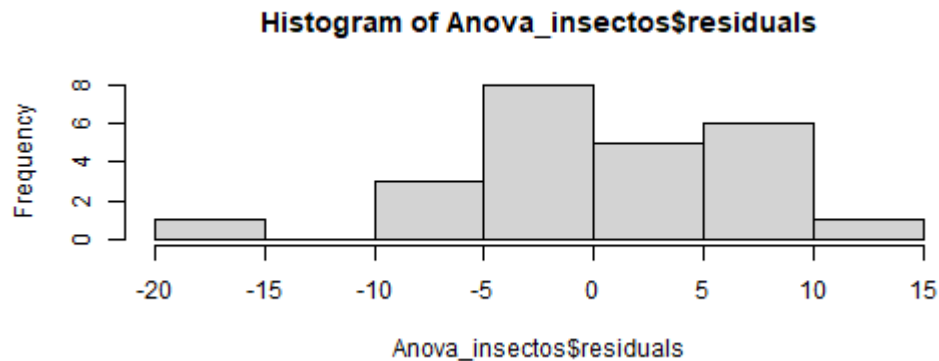
## 2.6. Normalidad

El test de Shapiro-Wilk indica que no tenemos evidencia suficiente para rechazar la hipótesis nula (normalidad de los residuos)

```
shapiro.test(Anova_insectos$residuals)
```

```
##  
##      Shapiro-Wilk normality test  
##  
## data:  Anova_insectos$residuals  
## W = 0.97337, p-value = 0.75
```

```
hist(Anova_insectos$residuals)
```



## 2.7. Homocedasticidad

El test de Bartlett indica que no tenemos evidencia suficiente para rechazar la hipótesis nula (las varianzas son iguales)

```
bartlett.test(Anova_insectos$residuals ~ colores)
```

```
##  
##      Bartlett test of homogeneity of variances  
##  
## data:  Anova_insectos$residuals by colores  
## Bartlett's K-squared = 5.2628, df = 3, p-value = 0.1535
```



## 2.8. Anova con dos factores

Crear nueva columna con factor de tamaño.

```
Insectos_df$tamano <- as.factor(c(rep(c("grande", "mediana", "chica"),  
                                each=2)))
```

Crear nuevo modelo.

```
Anova_insectos_2<-aov((insectos ~ colores + tamano),data=Insectos_df)
```

Resultados del modelo.

```
summary(Anova_insectos_2)
```

```
##           Df Sum Sq Mean Sq F value    Pr(>F)
## colores     3   4218  1406.2   28.96 4.18e-07 ***
## tamano      2     47    23.3    0.48  0.627
## Residuals  18    874    48.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## 2.8. Anovas con interaccion

Agregar interacción en el modelo.

```
Anova_insectos_3<-aov((insectos ~ colores * tamaño),data=Insectos_df)
```

Otra manera de escribir el mismo modelo.

```
Anova_insectos_4<-aov((insectos ~ colores + tamaño + colores : tamaño),
```

## 2.8. Anovas con interaccion

Mismos resultados.

```
summary(Anova_insectos_3)
```

```
##              Df Sum Sq Mean Sq F value  Pr(>F)
## colores          3   4218   1406.2   57.103 2.24e-07 ***
## tamaño           2     47     23.3    0.946  0.4155
## colores:tamaño   6     578     96.4    3.915  0.0212 *
## Residuals       12     296     24.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(Anova_insectos_4)
```

```
##              Df Sum Sq Mean Sq F value  Pr(>F)
## colores          3   4218   1406.2   57.103 2.24e-07 ***
## tamaño           2     47     23.3    0.946  0.4155
## colores:tamaño   6     578     96.4    3.915  0.0212 *
## Residuals       12     296     24.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# Ejercicios

- Generar **datos y modelo**
- Verificar algunos **supuestos**
- Analisis **post-hoc**
- Crear anova de **dos factores**
- Crear anova con **interacción**

---

Datos y modelo

Supuestos

Post-hoc

Dos factores

Interacciones

Generar datos y modelo.

```
insectos <- c(16, 11, 20, 21, 14, 7, 37, 32, 15, 25, 39, 41, 21, 12, 14, 17, 13, 17, 45, 59)
colores <- as.factor(c(rep(c("azul", "verde", "blanco", "amarillo"), each = 5)))
```

```
Anova_insectos <- aov(lm(insectos ~ colores))
```

```
summary>Anova_insectos)
```

# Contacto

- Factores
- Analisis de varianza

Para dudas, comentarios y sugerencias:  
Escríbeme a [miriamjlerma@gmail.com](mailto:miriamjlerma@gmail.com)

Este material esta accesible y se encuentra en  
mi [github](#) y mi [página www.miriam-lerma.com](http://www.miriam-lerma.com)

