ANOVA

María Isabel Chuya

ANOVA

La prueba ANOVA unidireccional, también conocida como análisis de varianza, es una técnica estadística que se utiliza para comparar las medias de tres o más grupos que se supone que provienen de poblaciones con distribuciones normales y varianzas iguales. Es una técnica de inferencia utilizada para determinar si hay diferencias significativas entre las medias de los grupos en una variable dependiente específica.

La prueba ANOVA unidireccional se denomina "unidireccional" porque solo analiza una variable independiente (factor) que divide los datos en varios grupos. El objetivo es determinar si las variaciones en las medias de los grupos son lo suficientemente grandes como para considerarse significativas o si son simplemente el resultado de variaciones aleatorias en las muestras.

El método ANOVA se basa en comparar la variabilidad dentro de los grupos (variabilidad causada por la variación aleatoria dentro de los grupos) con la variabilidad entre grupos. Se considera una diferencia significativa entre al menos uno de los grupos si la variabilidad entre los grupos es significativamente mayor que la variabilidad dentro de los grupos.

En resumen, la prueba ANOVA unidireccional se utiliza para responder preguntas como: "¿Hay alguna diferencia significativa entre las medias de tres o más grupos en una variable dependiente?" y es una herramienta importante en la investigación para comparar varios grupos al mismo tiempo y determinar si hay diferencias estadísticamente significativas entre ellos.

```
my_data <- read.csv("cancer (1).csv")
```

```
my_data <- PlantGrowth
```

Consulta tus datos

```
# Show a random sample
set.seed(1234)
dplyr::sample_n(my_data, 10)
```

```
   weight group
1    6.15  trt2
2    3.83  trt1
3    5.29  trt2
4    5.12  trt2
5    4.50  ctrl
6    4.17  trt1
7    5.87  trt1
8    5.33  ctrl
9    5.26  trt2
10   4.61  ctrl
```

```
# Show the levels
levels(my_data$group)
```

```
[1] "ctrl" "trt1" "trt2"
```

Calcule las estadísticas de resumen por grupos: conteo, media, sd:

```
library(dplyr)
```

```
Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
  group_by(my_data, group) %>%
    summarise(
      count = n(),
      mean = mean(weight, na.rm = TRUE),
      sd = sd(weight, na.rm = TRUE)
    )
```

```
# A tibble: 3 x 4
  group count  mean    sd
  <fct> <int> <dbl> <dbl>
1 ctrl     10  5.03 0.583
2 trt1     10  4.66 0.794
3 trt2     10  5.53 0.443
```

Visualiza tus datos

- Para usar gráficos base R , lea esto: Gráficos base R. Aquí, usaremos el paquete ggpubr R para una fácil visualización de datos basada en ggplot2.
- Instalar la última versión de ggpubr desde GitHub de la siguiente manera (recomendado):

```
  # Install
  if(!require(devtools)) install.packages("devtools")
```

```
Loading required package: devtools
```

```
Loading required package: usethis
```

```
  devtools::install_github("kassambara/ggpubr")
```

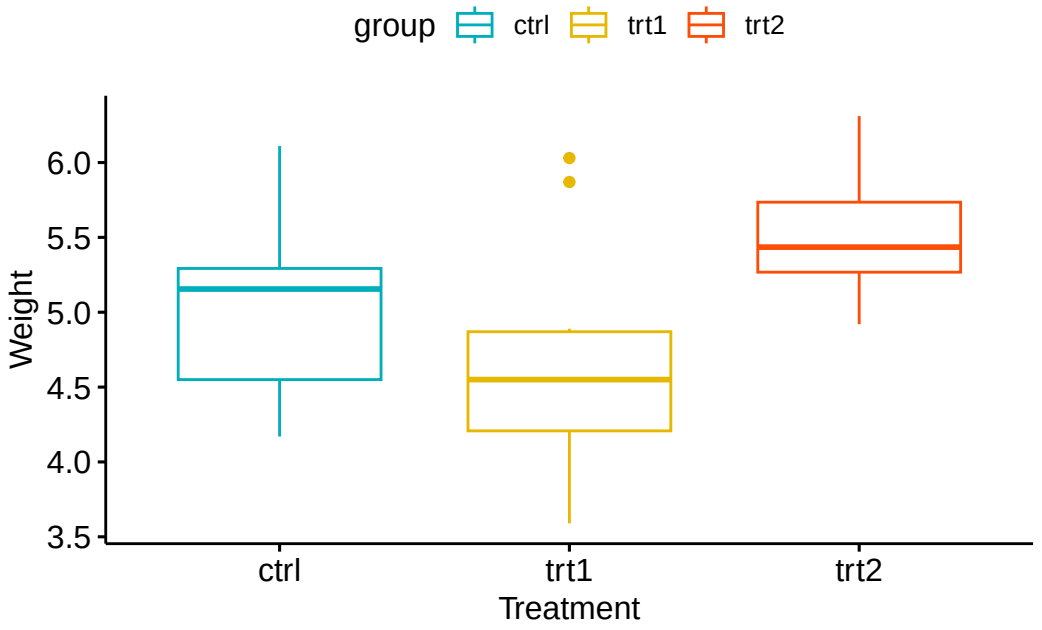
```
Skipping install of 'ggpubr' from a github remote, the SHA1 (6aeb4f70) has not changed since
  Use `force = TRUE` to force installation
```

- Visualiza tus datos con ggpubr:

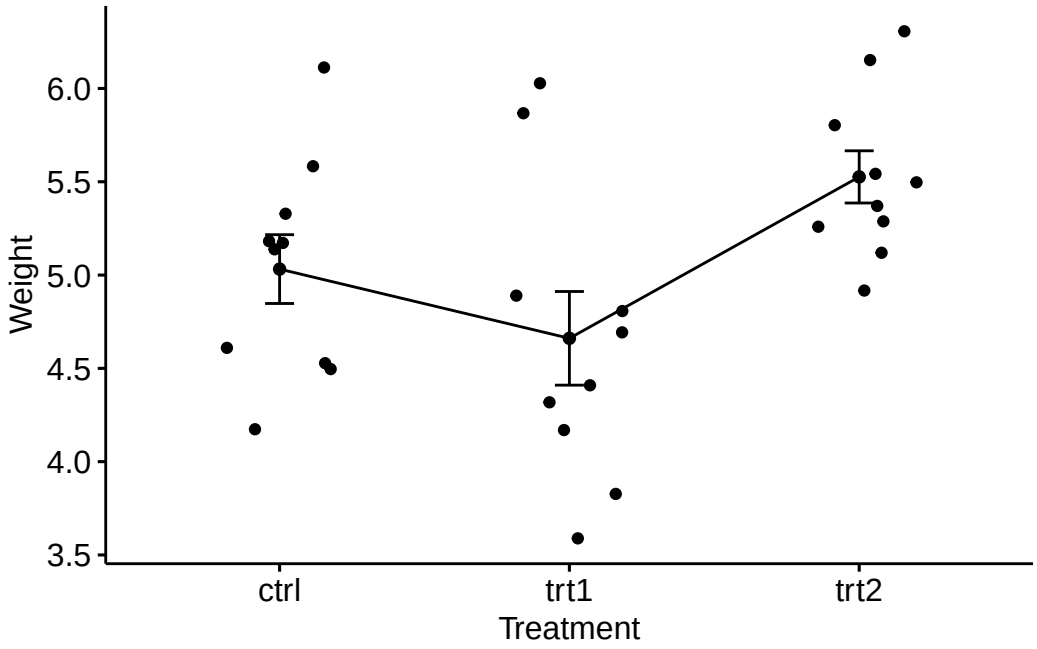
```
  # Box plots
  # ++++++++++++++++++++
  # Plot weight by group and color by group
  library("ggpubr")
```

```
Loading required package: ggplot2
```

```r
ggboxplot(my_data, x = "group", y = "weight",
          color = "group", palette = c("#00AFBB", "#E7B800", "#FC4E07"),
          order = c("ctrl", "trt1", "trt2"),
          ylab = "Weight", xlab = "Treatment")
```

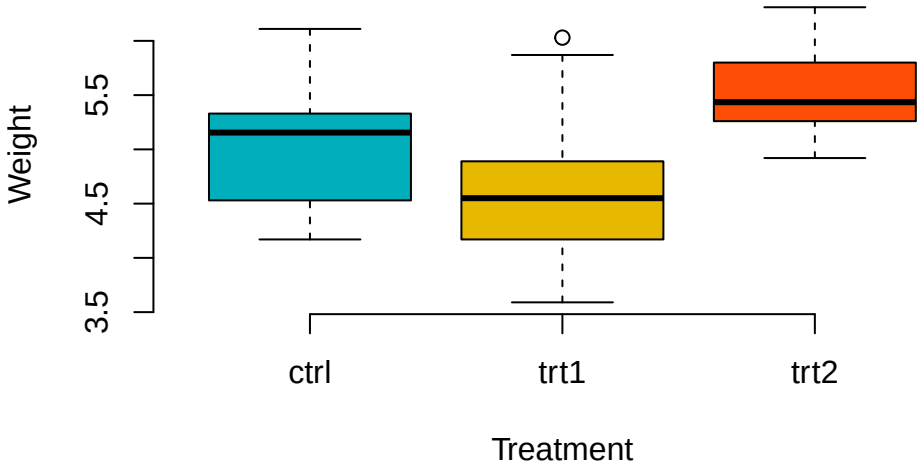



```r
# Mean plots
# ++++++++++++++++++++
# Plot weight by group
# Add error bars: mean_se
# (other values include: mean_sd, mean_ci, median_iqr, ....)
library("ggpubr")
ggline(my_data, x = "group", y = "weight",
       add = c("mean_se", "jitter"),
       order = c("ctrl", "trt1", "trt2"),
       ylab = "Weight", xlab = "Treatment")
```



Si aún desea usar gráficos base R, escriba los siguientes scripts:

```
# Box plot
boxplot(weight ~ group, data = my_data,
        xlab = "Treatment", ylab = "Weight",
        frame = FALSE, col = c("#00AFBB", "#E7B800", "#FC4E07"))
```




```
# plotmeans
library("gplots")
```

```
Attaching package: 'gplots'
```

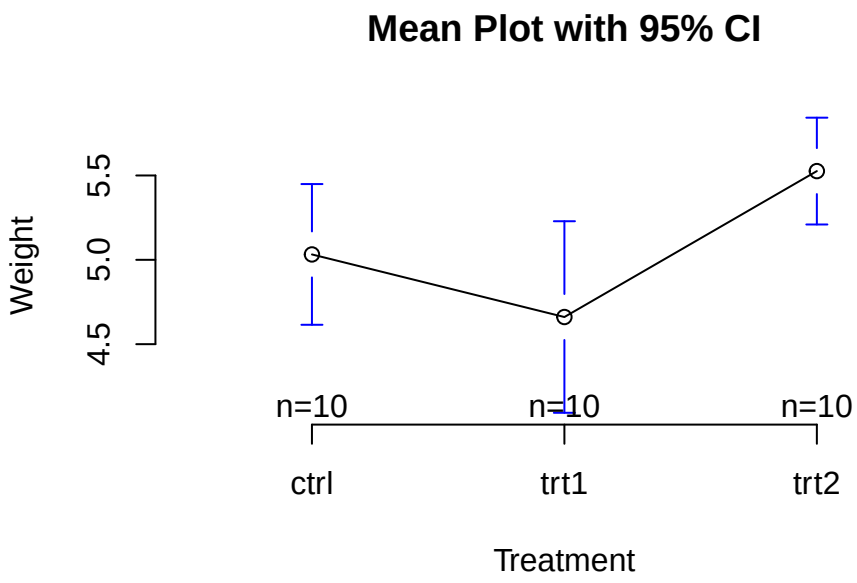
```
The following object is masked from 'package:stats':

    lowess
```

```
plotmeans(weight ~ group, data = my_data, frame = FALSE,
          xlab = "Treatment", ylab = "Weight",
          main="Mean Plot with 95% CI")
```

```
Warning in plot.xy(xy.coords(x, y), type = type, ...): "frame" is not a
graphical parameter
```

```
Warning in axis(1, at = 1:length(means), labels = legends, ...): "frame" is not
a graphical parameter
```

```
Warning in plot.xy(xy.coords(x, y), type = type, ...): "frame" is not a
graphical parameter
```



Calcule la prueba ANOVA unidireccional

La función **summary.aov** () se utiliza para resumir el modelo de análisis de varianza.

```r
# Compute the analysis of variance
res.aov <- aov(weight ~ group, data = my_data)
# Summary of the analysis
summary(res.aov)
```

```
            Df Sum Sq Mean Sq F value Pr(>F)
group        2  3.766  1.8832   4.846 0.0159 *
Residuals   27 10.492  0.3886
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretación de la información

Podemos concluir que existen diferencias significativas entre los grupos resaltados con "*" en el resumen del modelo si el valor p es menor que el nivel de significación 0,05. Tukey comparaciones por pares múltiples La función TukeyHD () utiliza el ANOVA ajustado como argumento.

```
TukeyHSD(res.aov)
```

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = weight ~ group, data = my_data)

$group
            diff        lwr       upr     p adj
trt1-ctrl -0.371 -1.0622161 0.3202161 0.3908711
trt2-ctrl  0.494 -0.1972161 1.1852161 0.1979960
trt2-trt1  0.865  0.1737839 1.5562161 0.0120064
```

Diferencia entre las medias de dos grupos:

El término "diff" se refiere a la diferencia entre las medias de dos grupos, que es una medida utilizada para comparar el valor promedio de una variable entre dos conjuntos distintos.

Intervalo de confianza al 95%:

Los valores "lwr" (punto final inferior) y "upr" (punto final superior) representan el intervalo de confianza al 95%, que es un rango estimado donde se espera que se encuentre el verdadero valor poblacional con un nivel de confianza del 95%. Este intervalo proporciona una estimación de la incertidumbre asociada con la estimación de la media.

Valor p ajustado:

El valor "p adj" se refiere al valor p después del ajuste para las comparaciones múltiples. Cuando se realizan múltiples pruebas o comparaciones en un mismo conjunto de datos, es común ajustar los valores p para controlar el error de tipo I y reducir las falsas alarmas. El ajuste del valor p ayuda a mantener un nivel adecuado de significancia estadística en estas comparaciones múltiples.

Múltiples comparaciones utilizando el paquete "mulcomp":

El paquete "mulcomp" proporciona herramientas para realizar comparaciones múltiples en análisis estadísticos, lo que permite evaluar diferencias entre varios grupos o condiciones de manera simultánea y corregir los valores p para controlar el error de tipo I.

Modelo y lincft():

El término “modelo” se refiere a un modelo estadístico ajustado, como un objeto devuelto por la función “aov()”. “lincft()” especifica las hipótesis lineales a probar en modelos ANOVA y se utiliza junto con el paquete “mcp()” para realizar comparaciones múltiples.

Paquete “mvtnorm”:

El paquete “mvtnorm” proporciona funciones relacionadas con las distribuciones t y normales multivariadas, lo que permite realizar análisis estadísticos más complejos que involucran múltiples variables.

Paquete “survival”:

El paquete “survival” se utiliza para el análisis de supervivencia, lo que implica analizar datos de tiempo hasta un evento específico. Este paquete ofrece funciones y herramientas para evaluar la probabilidad de supervivencia y realizar análisis relacionados.

Paquete “TH.data”:

El paquete “TH.data” proporciona varios conjuntos de datos comúnmente utilizados para la enseñanza y demostraciones en el ámbito estadístico, lo que facilita el acceso a datos relevantes para el aprendizaje y la práctica.

Paquete “MASS”:

El paquete “MASS” contiene funciones y conjuntos de datos para el modelado estadístico práctico. Este paquete ofrece herramientas para diversos análisis estadísticos y se utiliza ampliamente en aplicaciones prácticas.

```
library(multcomp)
```

```
Loading required package: mvtnorm
```

```
Loading required package: survival
```

```
Loading required package: TH.data
```

```
Loading required package: MASS
```

```

Attaching package: 'MASS'
```

```
The following object is masked from 'package:dplyr':

    select
```

```
Attaching package: 'TH.data'

The following object is masked from 'package:MASS':

    geyser
```

```r
library(multcomp)
summary(glht(res.aov, linfct = mcp(group = "Tukey")))
```

```
     Simultaneous Tests for General Linear Hypotheses

Multiple Comparisons of Means: Tukey Contrasts


Fit: aov(formula = weight ~ group, data = my_data)

Linear Hypotheses:
                 Estimate Std. Error t value Pr(>|t|)
trt1 - ctrl == 0  -0.3710     0.2788  -1.331   0.3908
trt2 - ctrl == 0   0.4940     0.2788   1.772   0.1979
trt2 - trt1 == 0   0.8650     0.2788   3.103   0.0121 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
(Adjusted p values reported -- single-step method)
```

Prueba t por pares

La función **pairewise.t.test** () también se puede utilizar para calcular comparaciones por pares entre niveles de grupo con correcciones para pruebas múltiples.

```r
pairwise.t.test(my_data$weight, my_data$group,
                  p.adjust.method = "BH")
```

```
    Pairwise comparisons using t tests with pooled SD

data:  my_data$weight and my_data$group
```

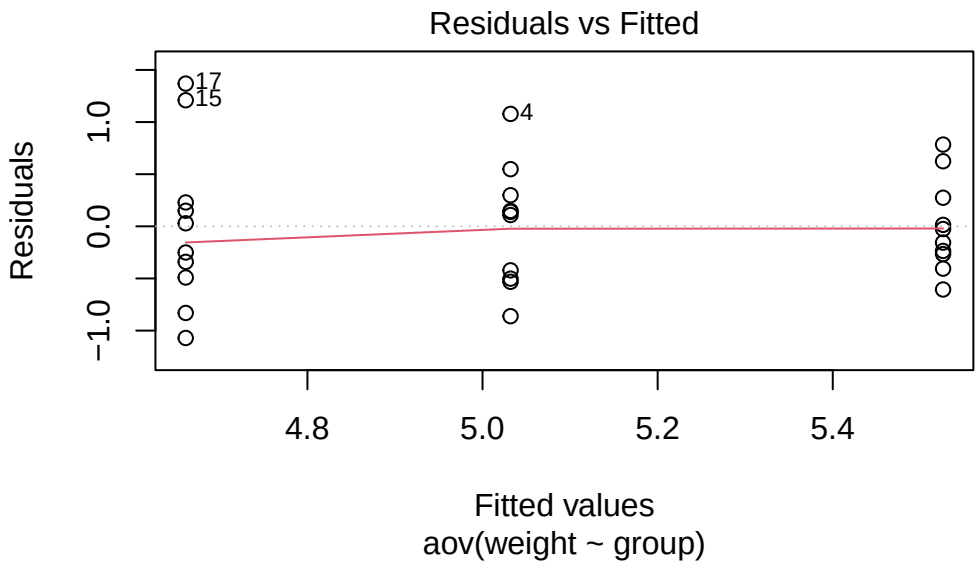
```
     ctrl  trt1
trt1 0.194 -
trt2 0.132 0.013

P value adjustment method: BH
```

La tabla de valores p para las comparaciones por pares se produce como resultado. Aquí, el método de Benjamini-Hochberg se utilizó para ajustar los valores p. Verifique la uniformidad del supuesto de varianza.

Se puede verificar la homogeneidad de las varianzas mediante la gráfica de desechos versus ajustes.

Es positivo que no exista una conexión evidente entre los desechos y los valores ajustados (la media de cada grupo), lo cual es beneficioso. Por lo tanto, podemos asumir que las varianzas son homogéneas.

```
# 1. Homogeneity of variances
plot(res.aov, 1)
```




```
library(car)
```

```
Loading required package: carData


Attaching package: 'car'

The following object is masked from 'package:dplyr':

    recode
```

```
  leveneTest(weight ~ group, data = my_data)
```

```
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  2  1.1192 0.3412
      27
```

```
  #install.packages("carData")
```

El carDatapaquete proporciona una variedad de conjuntos de datos que se utilizan con frecuencia en ejemplos y ejercicios relacionados con el carpaquete. Estos conjuntos de datos se utilizan tanto para modelos de regresión lineales como no lineales. El carDatapaquete se carga automáticamente cuando se carga el carpaquete, que podría haber sido cargado anteriormente, para acceder a los conjuntos de datos proporcionados por carData.

Interpretacion Como se puede ver de los resultados anteriores, el valor p no es menor que el nivel de significancia de 0.05. Esto significa que no hay pruebas estadísticas que sugieran que la varianza entre los grupos sea significativamente diferente. Por lo tanto, podemos asumir que las variaciones en los diferentes grupos de tratamiento son iguales. El supuesto de homogeneidad de la varianza se ha relajado.

n la función oneway.test () se ha implementado un procedimiento alternativo (es decir: **prueba unidireccional de Welch**), que no requiere esa suposición .

- **Prueba ANOVA sin suposición de varianzas iguales**

```
  oneway.test(weight ~ group, data = my_data)
```

```
    One-way analysis of means (not assuming equal variances)

data:  weight and group
F = 5.181, num df = 2.000, denom df = 17.128, p-value = 0.01739
```

- **Pruebas t por pares sin suposición de varianzas iguales**

```r
pairwise.t.test(my_data$weight, my_data$group,
                p.adjust.method = "BH", pool.sd = FALSE)
```

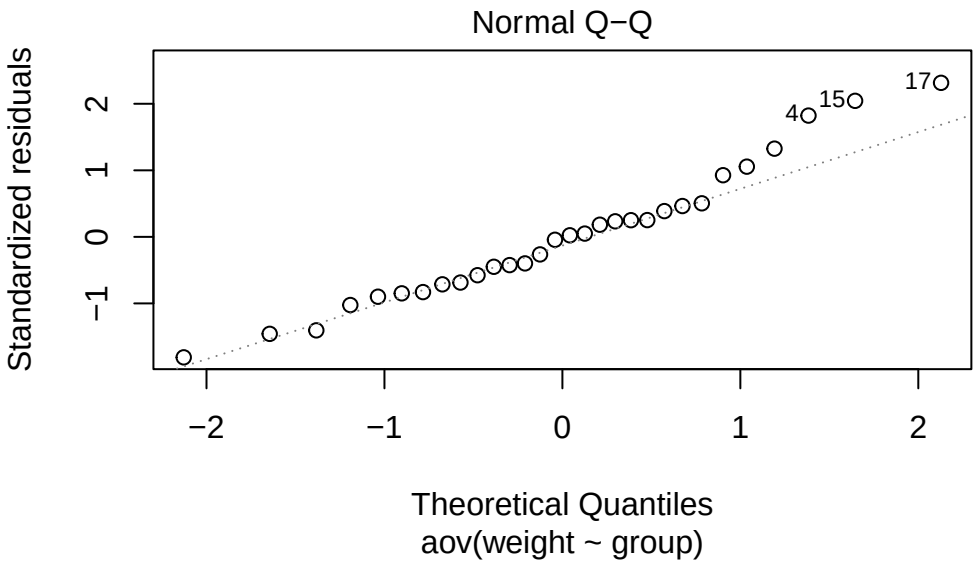
```
    Pairwise comparisons using t tests with non-pooled SD

data:  my_data$weight and my_data$group

     ctrl  trt1
trt1 0.250 -
trt2 0.072 0.028

P value adjustment method: BH
```

Comprobar el supuesto de normalidad

La gráfica muestra la normalidad de los desechos. El siguiente gráfico muestra los cuantiles de desecho frente a los cuantiles de distribución normal. Además, se traza una línea de referencia de 45°.

La suposición de que los desechos se distribuyen normalmente se verifica utilizando la gráfica de probabilidad normal de desechos. Debe seguir una línea recta.

```r
# 2. Normality
plot(res.aov, 2)
```



La conclusión anterior está respaldada por la **prueba de Shapiro-Wilk** sobre los residuos de ANOVA ($W = 0{,}96$, $p = 0{,}6$) que no encuentra indicios de que se haya violado la normalidad.

```
# Extract the residuals
aov_residuals <- residuals(object = res.aov )
# Run Shapiro-Wilk test
shapiro.test(x = aov_residuals )
```

```
    Shapiro-Wilk normality test

data:  aov_residuals
W = 0.96607, p-value = 0.4379
```

Alternativa no paramétrica a la prueba ANOVA unidireccional

```
kruskal.test(weight ~ group, data = my_data)
```

```
    Kruskal-Wallis rank sum test
```

```
data:  weight by group
Kruskal-Wallis chi-squared = 7.9882, df = 2, p-value = 0.01842
```