Medidas de Dispersión

Santiago Pérez Moncada

23/6/2020

Las **Medidas de dispersión** evalúan lo dispersos que están los datos.Algunas de las más importantes son:

- El **Rango** o **Recorrido**, que es la diferencia entre el máximo y el mínimo de las observaciones.
- El **Rango intercuartílico**, que es la diferencia entre el tercer y primer cuartil, $Q_{0.75} - Q_{0.25}$.
- La **Varianza**, a la que denotamos s^2, es la media aritmética de las diferencias al cuadrado entre los datos x_i y la media aritmetica de las observaciones, $\bar{x}$.

$$s^2 = \frac{\sum_{j=1}^{n}(x_j - \bar{x})^2}{n} = \frac{\sum_{j=1}^{k} n_j(X_j - \bar{x})^2}{n} = \sum_{j=1}^{k} f_j(X_j - \bar{x})^2$$

- La **desviación típica** es la raíz cuadrada positiva de la varianza, $std = \sqrt{s^2}$
- La **varianza muestral** es la corrección de la varianza.La denotamos por $\tilde{s}^2$

$$\tilde{s}^2 = \frac{n}{n-1}s^2 = \frac{\sum_{j=1}^{n}(x_j - \bar{x})^2}{n-1}$$

* La **desviación típica muestral**, que es la raíz cuadrada positiva de la varianza muestral, $\tilde{s} = \sqrt{\tilde{s}^2}$

Propiedades de la Varianza

- $s^2 \geq 0$ esto de debe a que, por definición, es una suma de cuadrados de números reales.
- $s^2 = 0 \Rightarrow x_j - \bar{x} = 0 \ \ \forall j = 1, 2, ..., n$.En consecuencia si $s^2 = 0$ todos los datos son iguales.
- $s^2 = \frac{\sum_{j=1}^{n} x_j^2}{n} - \bar{x}^2$. Es decir, la varianza es la media de los cuadrados de los datos menos el cuadrado de la media aritmética de estos.

Varianza y Varianza Muestral

La diferencia entre ambas definiciones viene por la interrelación entre la estadística descriptiva y la inferencial.

Por un lado, es normal medir cómo varían los datos cuantitativos mediante su varianza definida como la media aritmética de las distancias al cuadrado de los datos a su valor medio.No obstante, por otro lado, el conjunto de nuestras observaciones, por lo normal, será una muestra de una población mucho mayor y nos interesará estimar entre muchas cosas su variabilidad.

La varianza de una muestra suele dar valores más pequeños que la varianza de la población, mientra que la varianza muestral tiende a dar valores alrededor de la varianza de la población.

Esta corrección, para el caso de una muestra grande no es notable. Dividir n entre $n-1$ en el caso de que n ser grande no significa una gran diferencia y aún menos si tenemos en cuenta que lo que tratamos es de estimar la varianza de población, no de calcularla de forma exacta.

En cambio, si la muestra es relativamente pequeña digamos $n < 30$, entonces **la varianza muestral** de la muestra aproxima significativamente mejor que la **varianza poblacional**.

La diferencia entre desviación típica y desviación típica muestral es análoga.

Con `R` calcularemos la varianza y desviacion típica muestrales.Con lo cual, si queremos calcular las que no son muestrales, tendremos que multiplicarlas por $\frac{n-1}{n}$ donde n es el tamaño de la muestra. lo veremos a continuación.

Medidas de Dispersión con `R`

Medida de dispersión	Instrucción
Valor minimo y max	`range(x)`
Rango	`diff(range(x)`
Rango intercuartilico	`IQR(x,type=...)`
Varianza Muestral	`var(x)`
Desviación típica muestral	`sd(x)`
Varianza	`var(x)*(length(x)-1)/length(x)`
Desviación típica	`sd(c)*sqrt((length(x)-1)/length(x))`

Ejemplo

```
dados = sample(1:6,15, replace = TRUE)
dados
```

```
##  [1] 6 6 6 3 5 2 3 2 2 4 3 5 3 1 6
```

```
diff(range(dados)) # rango
```

```
## [1] 5
```

```
IQR(dados) #Q_0.75 - Q_0.25
```

```
## [1] 3
```

```
var(dados) #varianza muestral
```

```
## [1] 3.028571
```

```r
sd(dados) #desviacion tipica muestral
```

```
## [1] 1.740279
```

```r
n = length(dados)
var(dados)*(n-1)/n #varianza poblacional
```

```
## [1] 2.826667
```

```r
sd(dados)*sqrt((n-1)/n) #desviacion tipica poblacional
```

```
## [1] 1.681269
```

Función `summary()`

La función `summary` aplicada a un vector numérico o a una variable cuantitativa nos devuelve un resumen estadístico con los valores mínimo y máximo del vector, sus tres cuartiles y su media.

Al aplicar esta función a un data frame, esta se aplica a todas sus variables de forma simultánea.De este modo, podemos observar rápidamente si hay diferencias notables entre sus variables numéricas.

Ejemplo

```r
cangrejos = read.table("../data/datacrab.txt",header = TRUE) #CARGAMOS DATOS
cangrejos = cangrejos[-1]#eliminamos la primer columna
summary(cangrejos)
```

```
##      color           spine           width          satell           weight
##  Min.   :2.000   Min.   :1.000   Min.   :21.0   Min.   : 0.000   Min.   :1200
##  1st Qu.:3.000   1st Qu.:2.000   1st Qu.:24.9   1st Qu.: 0.000   1st Qu.:2000
##  Median :3.000   Median :3.000   Median :26.1   Median : 2.000   Median :2350
##  Mean   :3.439   Mean   :2.486   Mean   :26.3   Mean   : 2.919   Mean   :2437
##  3rd Qu.:4.000   3rd Qu.:3.000   3rd Qu.:27.7   3rd Qu.: 5.000   3rd Qu.:2850
##  Max.   :5.000   Max.   :3.000   Max.   :33.5   Max.   :15.000   Max.   :5200
```

Si nos interesa comparar numéricamente los pesos y las anchuras de los cangrejos con 3 colores con los que tienen 5 colores, utilizamos las siguientes instrucciones:

```r
summary(subset(cangrejos, color == 3, c("weight","width")))
```

```
##      weight         width
##  Min.   :1300   Min.   :22.5
##  1st Qu.:2100   1st Qu.:25.1
##  Median :2500   Median :26.5
##  Mean   :2538   Mean   :26.7
##  3rd Qu.:3000   3rd Qu.:28.2
##  Max.   :5200   Max.   :33.5
```

```r
summary(subset(cangrejos, color == 5, c("weight","width")))
```

```
##      weight         width
##  Min.   :1300   Min.   :21.00
##  1st Qu.:1900   1st Qu.:23.90
##  Median :2125   Median :25.50
##  Mean   :2174   Mean   :25.28
##  3rd Qu.:2400   3rd Qu.:26.57
##  Max.   :3225   Max.   :29.30
```

Y deducimos que los cangrejos con 5 colores pesan ligeramente menos y tienen menos anchura que los que tienen 3 colores.

Funcion `by()`

La función `by()` se utiliza para aplicar una determinada función a algunas de las columnas de un data frame segmentándolas según los niveles de un factor.

La sintaxis de esta función es `by(columnas, factor, FUN = función)`.

Con lo cual, haciendo uso de la función `by` y especificando `FUN = summary`, podemos calcular el resumen estadistico anteriormente comentado a subpoblaciones definidas por los niveles de un factor.

Ejemplo

Para este ejemplo, haremos uso del famoso dataset iris.

Si nos interesa calcular de forma rápida y sencilla las longuitudes de los sépalos y petalos en función de la especie, necesitaríamos hacer uso de la instrucción mostrada a continuación.

```r
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

```r
by(iris[,c(1,3)], iris$Species, FUN = summary)
```

```
## iris$Species: setosa
##   Sepal.Length    Petal.Length
##  Min.   :4.300   Min.   :1.000
##  1st Qu.:4.800   1st Qu.:1.400
##  Median :5.000   Median :1.500
##  Mean   :5.006   Mean   :1.462
##  3rd Qu.:5.200   3rd Qu.:1.575
##  Max.   :5.800   Max.   :1.900
## ------------------------------------------------------------
```

```
## iris$Species: versicolor
##   Sepal.Length    Petal.Length
##  Min.   :4.900   Min.   :3.00
##  1st Qu.:5.600   1st Qu.:4.00
##  Median :5.900   Median :4.35
##  Mean   :5.936   Mean   :4.26
##  3rd Qu.:6.300   3rd Qu.:4.60
##  Max.   :7.000   Max.   :5.10
## ------------------------------------------------------------
## iris$Species: virginica
##   Sepal.Length    Petal.Length
##  Min.   :4.900   Min.   :4.500
##  1st Qu.:6.225   1st Qu.:5.100
##  Median :6.500   Median :5.550
##  Mean   :6.588   Mean   :5.552
##  3rd Qu.:6.900   3rd Qu.:5.875
##  Max.   :7.900   Max.   :6.900
```

Tanto la función `by` como la función `aggregate` son equivalentes. No obstante, los resultados se muestran de forma diferente en función de cual utilicemos.

En el caso del ejemplo anterior, convenía más hacer uso de la función `by`.

Podemos comprobarlo introduciendo la siguiente instruccion.

```
aggregate(cbind(Sepal.Length,Petal.Length)~Species, data = iris, FUN = summary)
```

```
##      Species Sepal.Length.Min. Sepal.Length.1st Qu. Sepal.Length.Median
## 1     setosa             4.300                4.800               5.000
## 2 versicolor             4.900                5.600               5.900
## 3  virginica             4.900                6.225               6.500
##   Sepal.Length.Mean Sepal.Length.3rd Qu. Sepal.Length.Max. Petal.Length.Min.
## 1             5.006                5.200             5.800             1.000
## 2             5.936                6.300             7.000             3.000
## 3             6.588                6.900             7.900             4.500
##   Petal.Length.1st Qu. Petal.Length.Median Petal.Length.Mean
## 1                1.400               1.500              1.462
## 2                4.000               4.350              4.260
## 3                5.100               5.550              5.552
##   Petal.Length.3rd Qu. Petal.Length.Max.
## 1                1.575             1.900
## 2                4.600             5.100
## 3                5.875             6.900
```

Valores NA

La mayoria de las funciones vistas a lo largo de este tema no funcionan bien con los valores **NA**.

Para no tenerlos en cuenta a la hora de aplicar estas funciones, hay que especificar el parámetro `na.rm = TRUE` en el argumento de la función.