Estadística Descriptiva

Lic. Lucio José Pantazis

03/08/2023

Medidas de resumen

Datos

Supongamos que una persona registra los tiempos de su viaje en subte hasta el trabajo (en minutos), y obtiene los siguientes resultados:

```
print(x)
```

```
##  [1] 46.31477 38.36883 46.64900 46.36215 42.07321 32.30025 35.35716 38.52640
##  [9] 39.97116 52.02327 43.81797 36.00495 34.26171 38.55269 38.50392 37.94245
## [17] 41.26112 35.54039 42.17842 33.81231 38.87866 41.88698 40.66668 44.02095
## [25] 39.71447 42.51804 45.42885 36.54523 33.57700 40.23363 38.82147 37.28556
## [33] 37.83345 36.75264 43.63375 45.75956 44.96080 37.85243 46.19152 38.60327
## [41] 48.78952 42.80373 37.73608 35.83978 34.16715 34.67205 32.18109 45.78268
## [49] 44.16024 38.86336 41.33069 38.11649 52.20682 36.02330 39.72561 41.25071
## [57] 43.09122 39.13688 28.88050 33.68193 41.79364
```

Cuesta analizar los datos mirándolos uno por uno, cuesta mantener el registro, por eso se calculan medidas de resumen, que nos den una idea general de este conjunto de datos.

Media

La media se obtiene sumando todos los valores y dividiendo por la cantidad total de datos. Es decir, si x_i representan cada dato individual, la media $\overline{x}$ se obtiene del siguiente modo:

$$\overline{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

Da una idea de dónde se centran los datos. En inglés se denomina "mean" y podemos ver que en este caso coincide con el cálculo "a mano":

```
n=length(x); print(sum(x)/n)
```

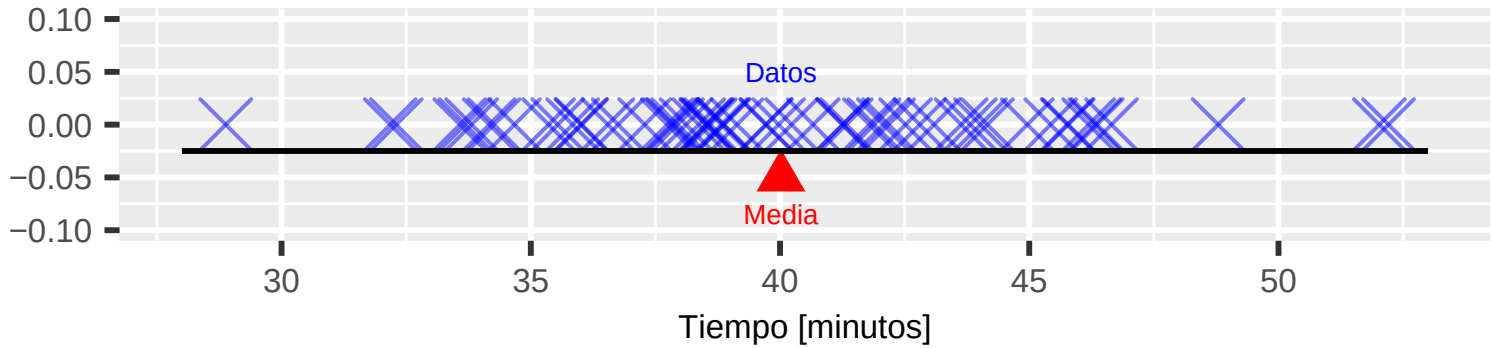
```
## [1] 40.02001
```

```
m=mean(x); print(m)
```

```
## [1] 40.02001
```

Una forma de pensarlo es que si se dispusieran los datos sobre una recta, la media busca que los datos a su izquierda y a su derecha queden "equilibrados". Mientras más alejados están los datos de la media, más influyentes son sobre el cálculo:

```
plot(GG)
```



Desvío estándard

Claro que sólo resumir los datos a partir de su media puede ser "resumirlos demasiado". Por lo tanto, se puede pensar no sólo dónde se centran los datos, sino ver cuánto se dispersan alrededor de este centro. La medida más utilizada para evaluar esto es el desvío estándard, suele nombrarse como s:

$$s = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n-1}}$$

Para estos datos, el desvío estándard da el siguiente resultado:

```
s=sqrt(sum((x-m)^2)/(n-1));print(s)
```

```
## [1] 4.792449
```

```
sd(x)
```

```
## [1] 4.792449
```

Desvío estándard

Su valor específico no siempre tiene una interpretación directa. De todos modos, sirve para comparar dos distribuciones distintas. Si dos distribuciones tienen la misma media, lo que las puede diferenciar es el valor del desvío estándard.

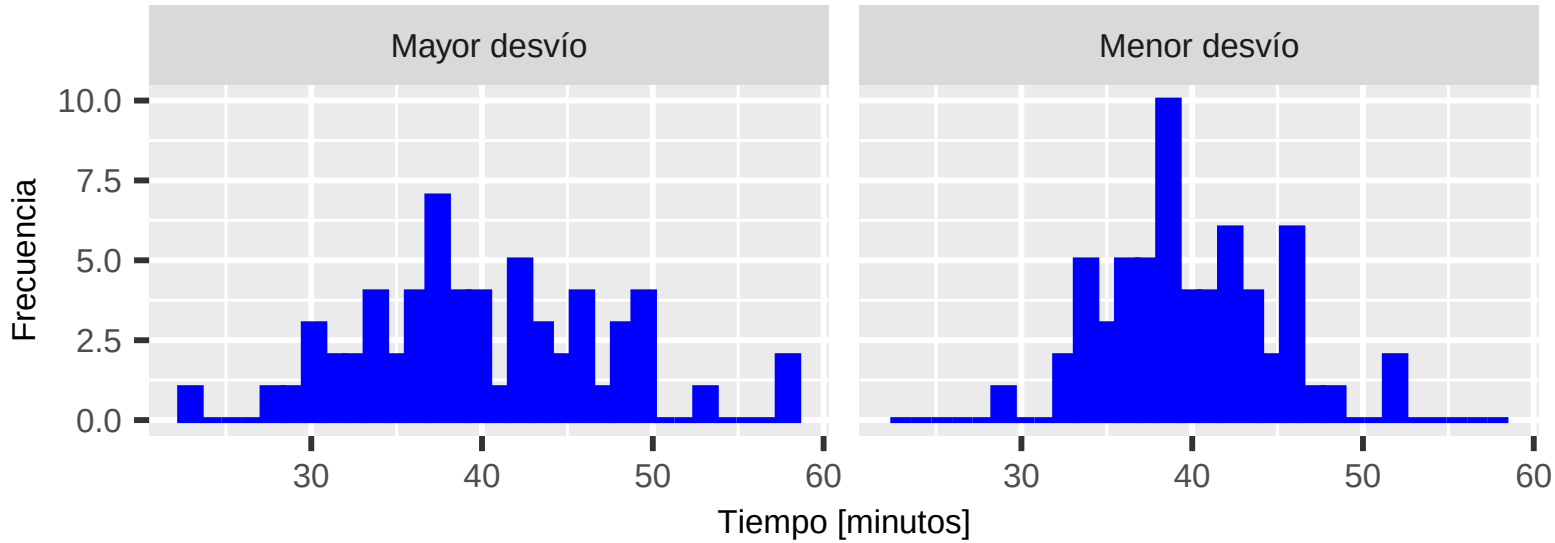
Supongamos que se comparan los datos de esta persona con alguien que se toma un cierto colectivo, y se obtiene un desvío dado por:

```
sd(y)
```

```
## [1] 7.188674
```

Comparando sus distribuciones, se observa lo siguiente:

```
plot(GG)
```



Es decir, teniendo la misma media, el valor de la dispersión no dice mucho, pero se puede ver sobre todo gráficamente que una variable tiene más dispersión que otra.

Simetría

Además de estas medidas de resumen, una cantidad que se puede evaluar es la simetría. Se denota γ y se calcula del siguiente modo:

$$\gamma = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^3}{n \cdot s^3}$$

Este valor es adimensional y se puede interpretar directamente:

- Si el valor es cercano a cero, la distribución de los datos es relativamente simétrica respecto de la media.
- Si el valor es positivo, la distribución de los datos es asimétrica a derecha.
- Si el valor es negativo, la distribución de los datos es asimétrica a izquierda.
- Mientras más alejado del cero es este valor, más asimétrica es la distribución.

Para los datos que fueron calculados, se puede ver que los datos son relativamente simétricos, porque su valor es cercano a cero:

```
g=sum((x-m)^3)/(n*s^3); print(g)
```

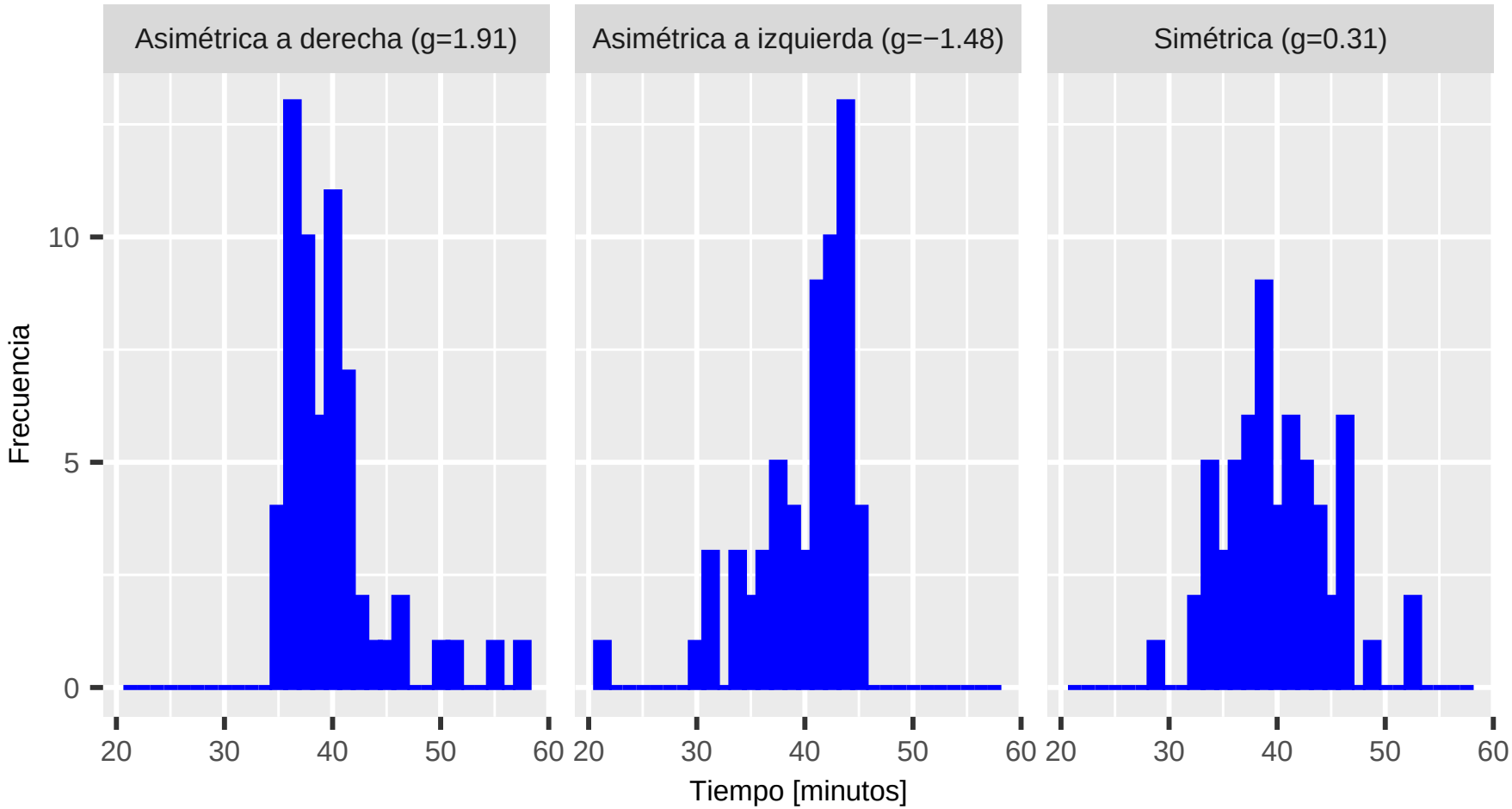
```
## [1] 0.3057749
```

Esto coincide con lo observado en el histograma.

Comparación gráfica

Gráficamente, podríamos ver en distintos histogramas cómo se visualizan estas asimetrías.

```
plot(GG)
```



Kurtosis

Además, otro valor que se puede calcular y puede diferenciar las distribuciones cuando todas las anteriores coinciden, es la kurtosis. Hace referencia a cuánto se concentran los datos y a su vez, qué peso tienen los datos más alejados de la media (lo que se suelen denominar "las colas"). Se calcula de la siguiente manera:

$$\kappa = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^4}{n \cdot s^4} - 3$$

También esta cantidad es adimensional. La resta del valor 3 es para poder interpretar la kurtosis usando como referencia la distribución normal.

- Si el valor de κ es cercano a cero, el peso de las colas es similar al de una distribución normal.
- Si el valor de κ es negativo, el peso de las colas es mayor al de una distribución normal.
- Si el valor de κ es positivo, el peso de las colas es menor al de una distribución normal.

Con los datos provistos, la kurtosis da el siguiente resultado:

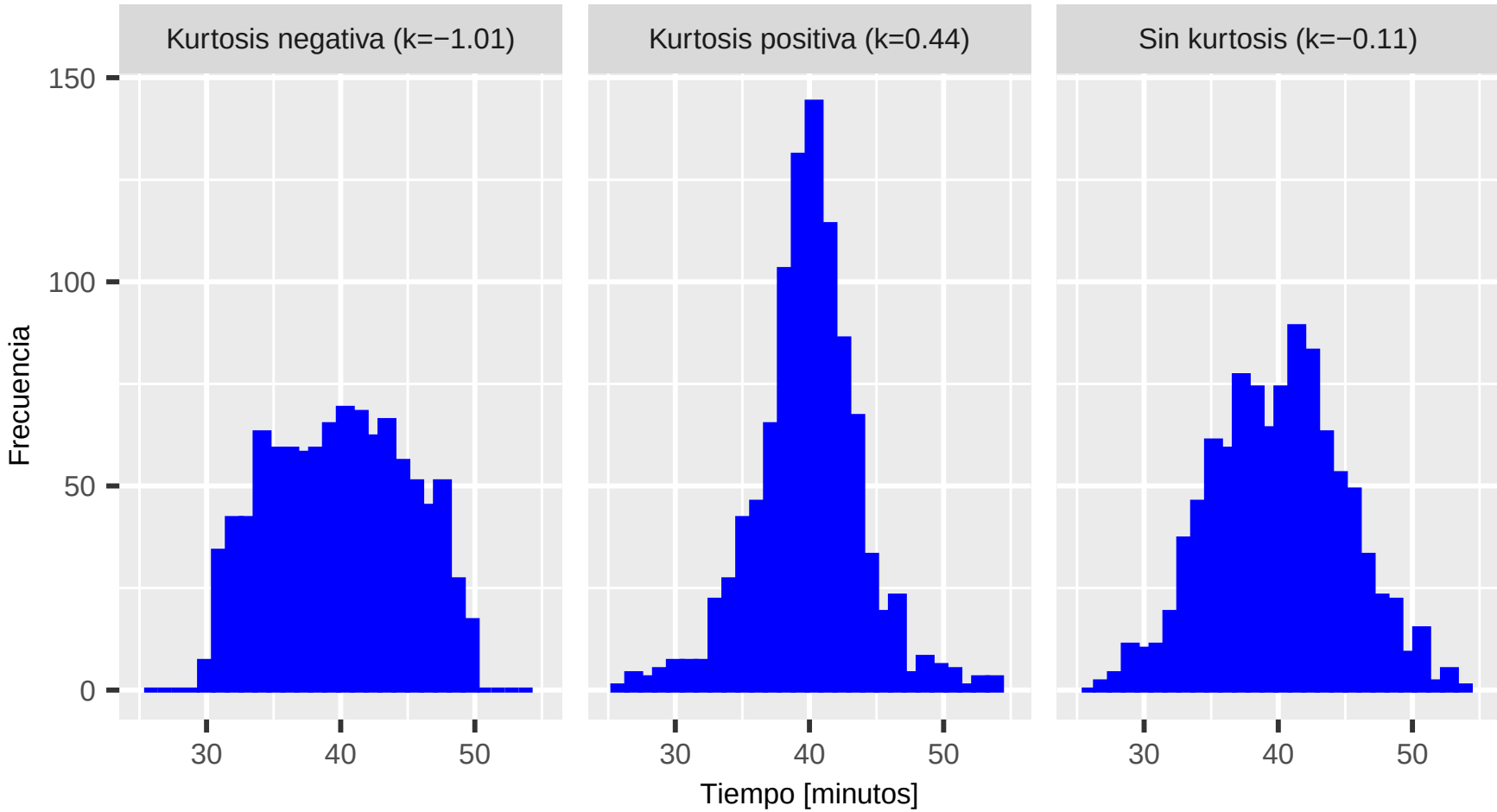
```
k=sum((x-m)^4)/(n*s^4)-3;print(k)
```

```
## [1] -0.1125623
```

Comparación gráfica

Nuevamente, todas estas magnitudes se observan mejor gráficamente ya que hacen referencia a la forma de la distribución.

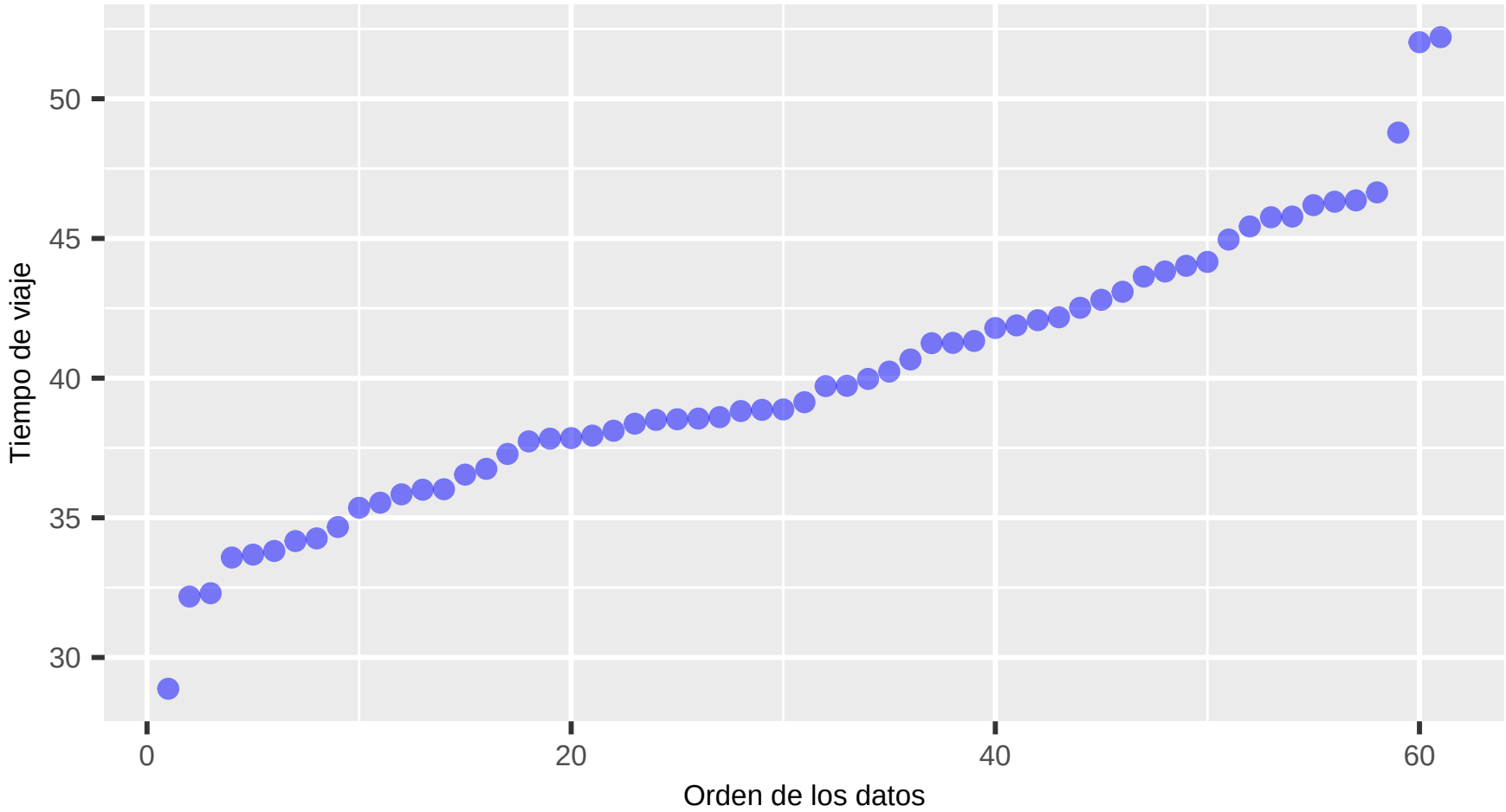
```
plot(GG)
```



Mediana

Otra forma de considerar el centro de los datos es con la que se llama la mediana. La idea detrás de su cálculo es primero ordenando los datos, y buscar el valor que divide a la mitad de los datos "menores" de los datos "mayores". Vamos a verlo primero gráficamente con los datos de los tiempos de viaje:

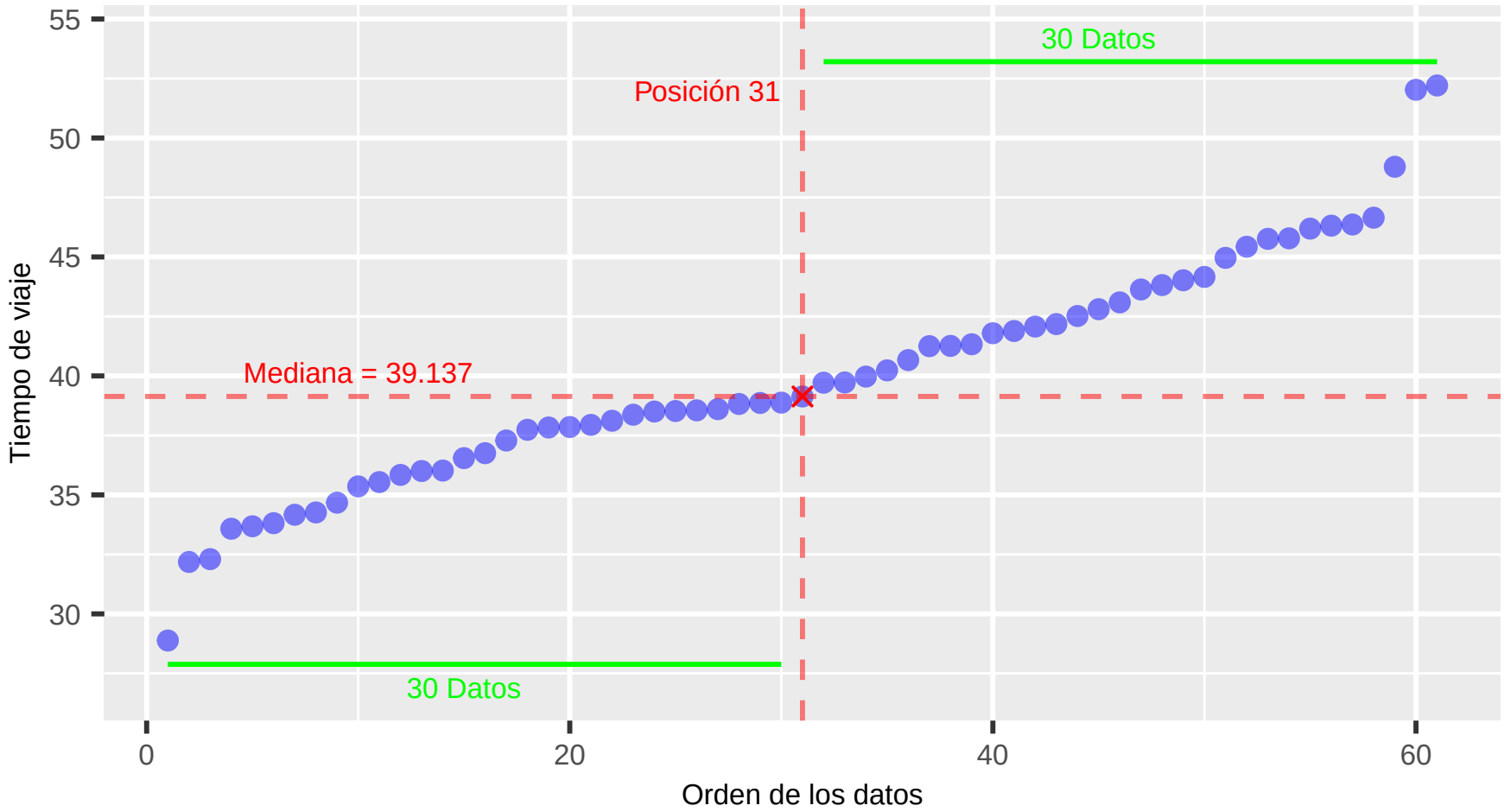
```
plot(GG)
```



Mediana

En este caso, tenemos 61 datos. Por lo tanto, para separar los datos a la mitad, deberíamos buscar el dato número 31 de esta lista ordenada, ya que deja 30 datos por debajo y 30 datos por arriba, y por lo tanto, los divide equitativamente:

```
plot(GG_N)
```



Cálculo

Es decir, en este conjunto de 61, para calcular la mediana, basta ordenar los datos y luego tomar la posición intermedia 31. Podemos ver que esto coincide con lo que se obtiene por software:

```
med=xS[31];print(med)
```

```
## [1] 39.13688
```

```
median(x)
```

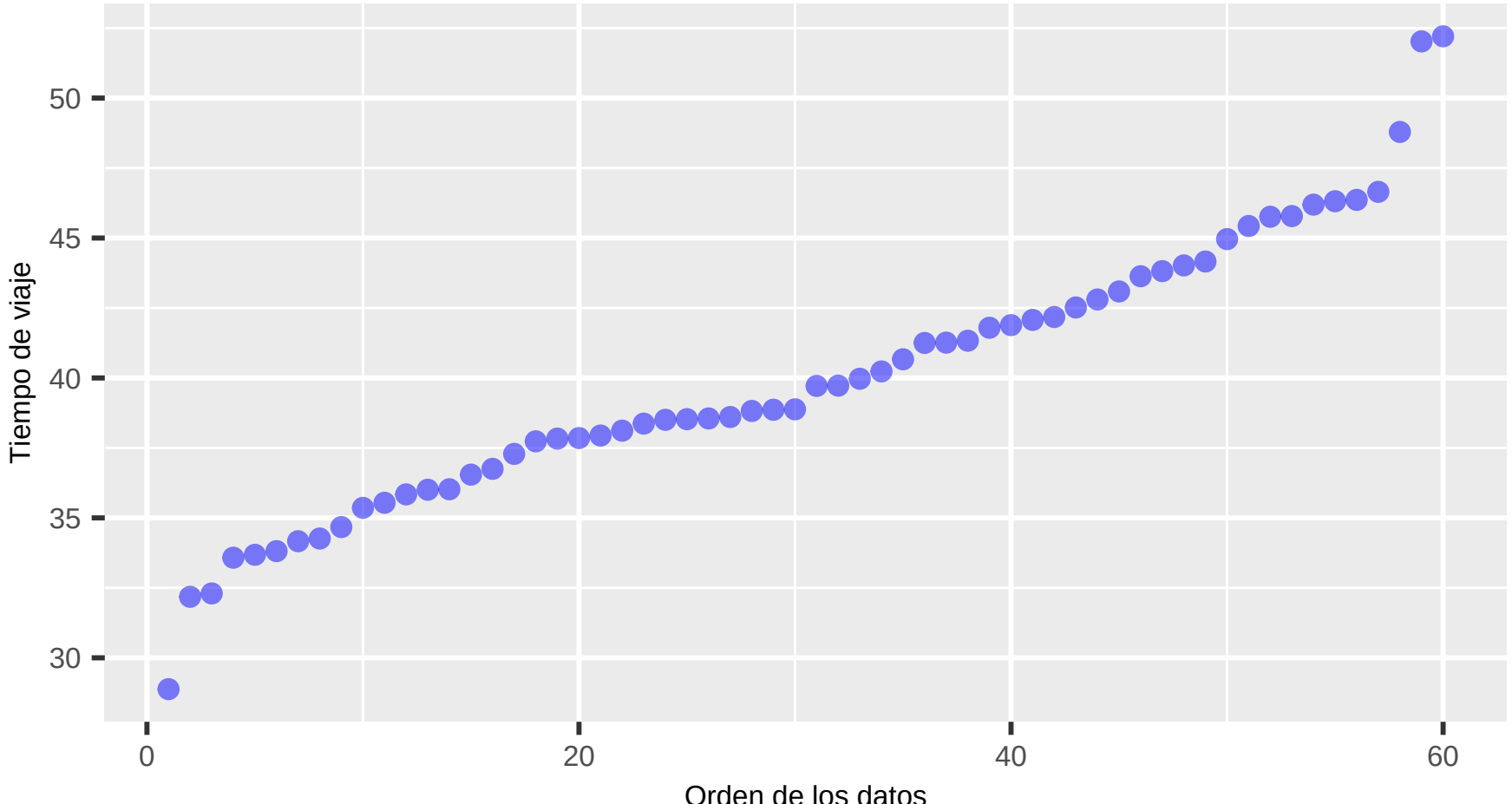
```
## [1] 39.13688
```

Caso de paridad

¿Qué pasa cuando no hay una observación que "divide a los datos a la mitad"? Por ejemplo, si los datos son pares, no hay ninguna observación que deje la mitad de los datos a su izquierda y la mitad de los datos a su derecha.

Supongamos que en vez de los 61 datos anteriores, tenemos 60 y ya están ordenados.

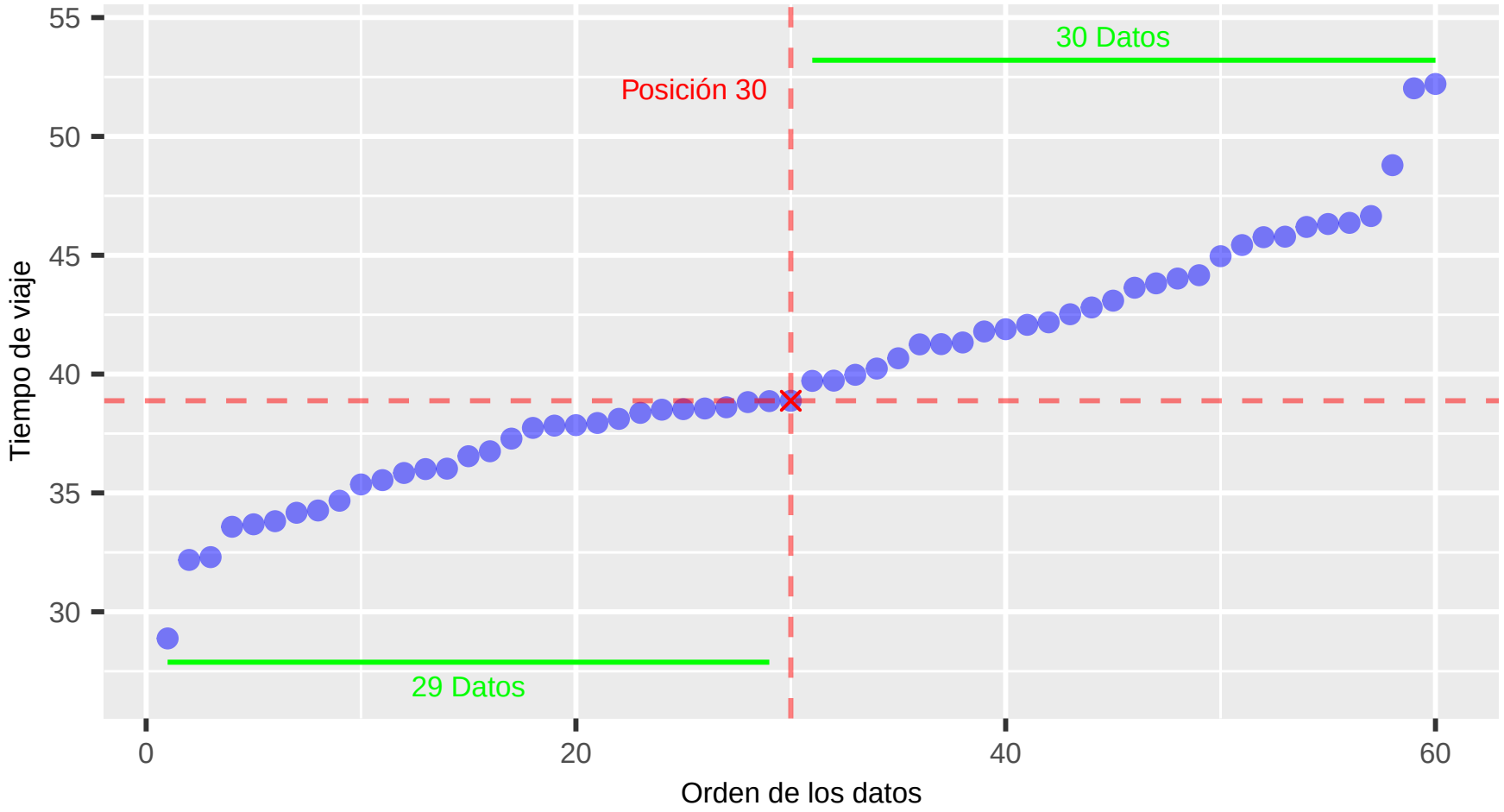
```
plot(GG)
```



Caso de paridad

Si tomamos la posición número 30, tiene 29 valores por debajo y 30 valores por encima.

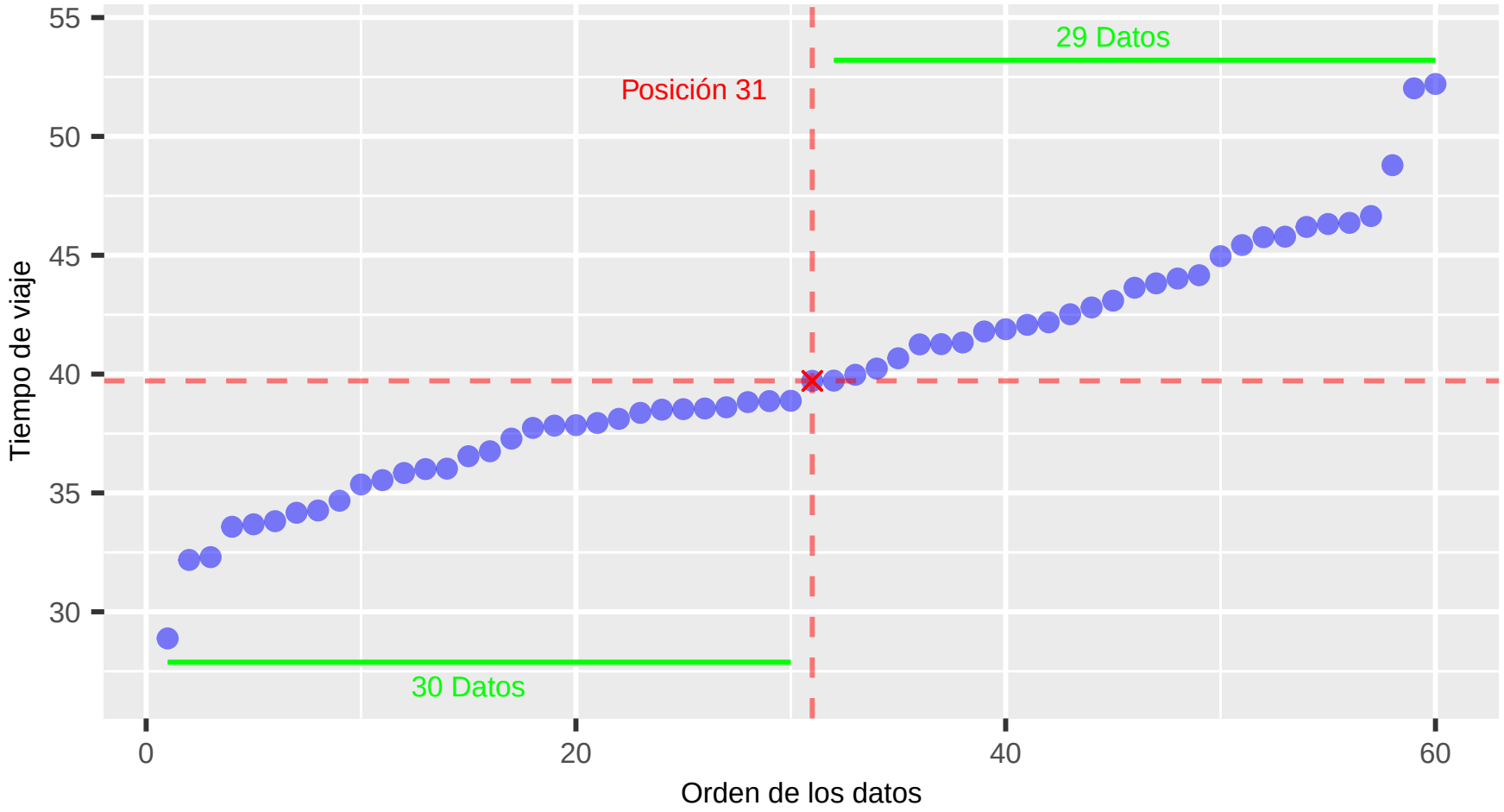
```
plot(GG_N)
```



Caso de paridad

Si tomamos la posición número 31, tiene 30 valores por debajo y 29 valores por encima.

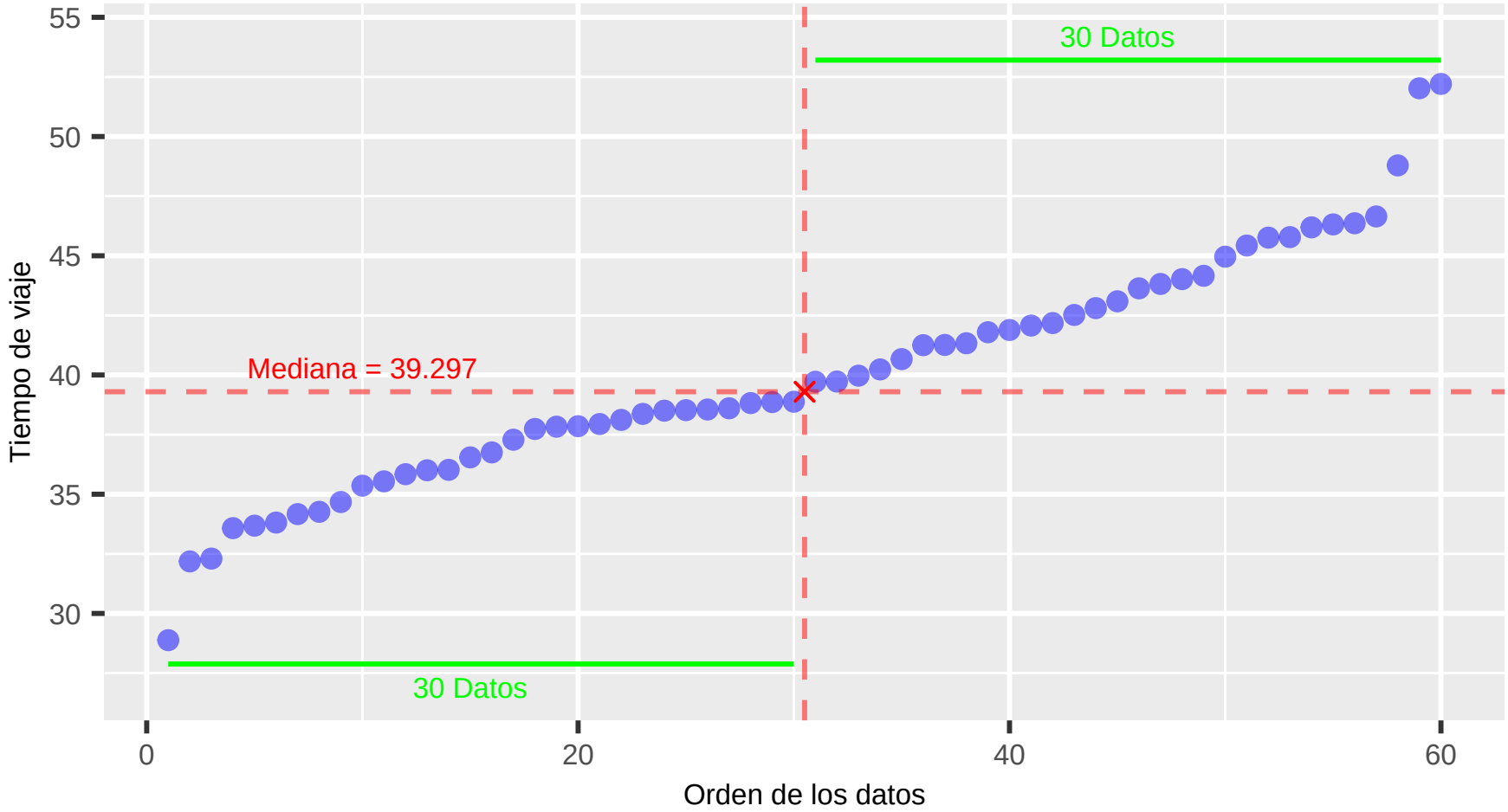
```
plot(GG_N)
```



Caso de paridad

Por lo tanto, ninguno de los dos datos consigue dividir equitativamente las observaciones. Por eso, se busca un valor que no sea específicamente alguna de las observaciones, por ejemplo, el promedio entre estos dos valores en disputa:

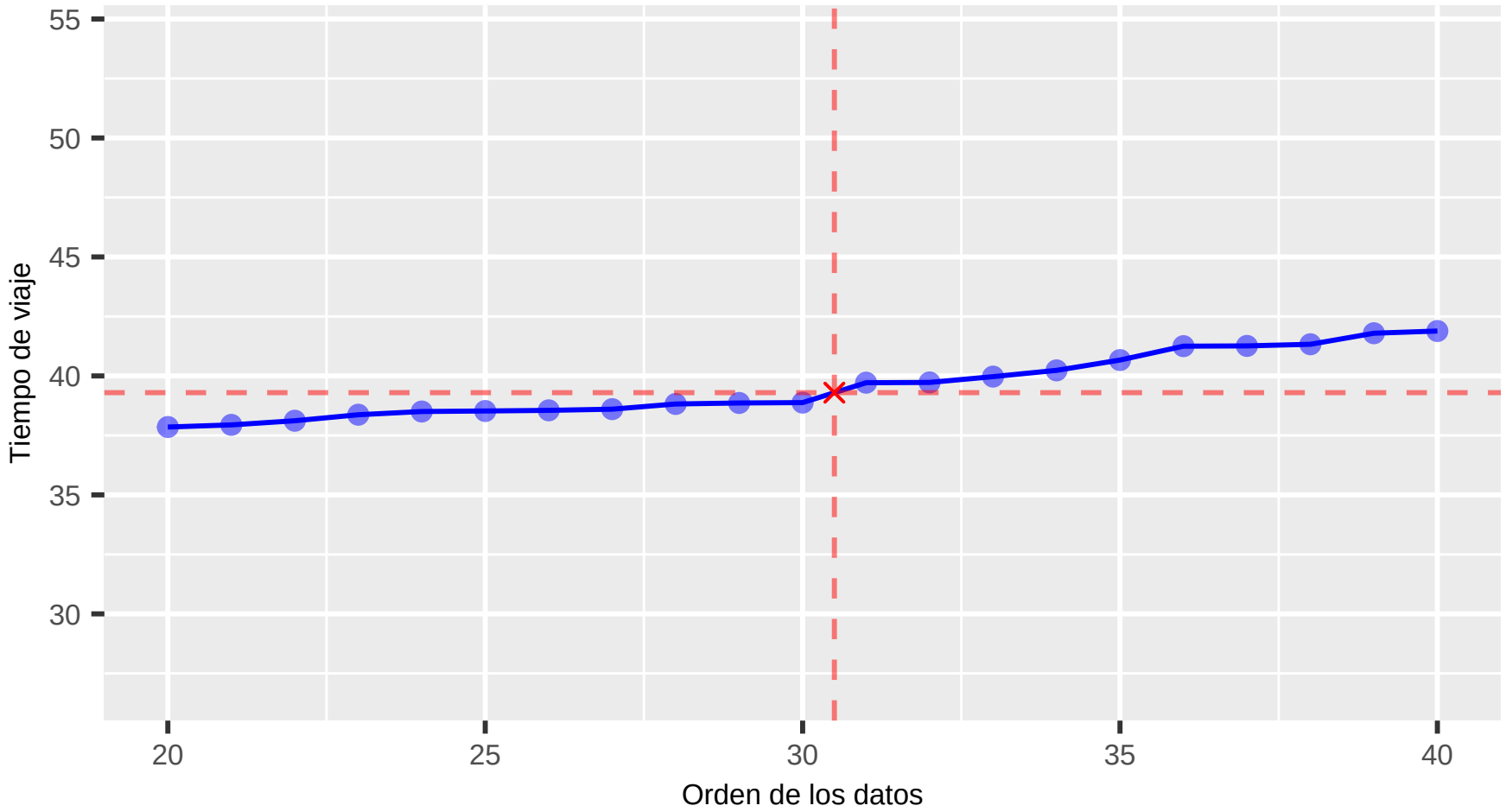
```
plot(GG_N)
```



Caso de paridad

Otra forma de verlo es que se toma la línea que une ambos valores en disputa, y se elige el valor intermedio de la recta que los une. Esta interpretación sirve para cuando hay cálculos similares más complejos:

```
plot(GG_N)
```



Caso de paridad

Vemos que coincide con lo obtenido por software:

```
med=(xSnew[31]+xSnew[30])/2;print(med)
```

```
## [1] 39.29656
```

```
median(xSnew)
```

```
## [1] 39.29656
```

Sin embargo, como en este caso puede tomarse **cualquier** valor intermedio, hay varias formas de calcular la mediana en el caso en que haya "disputa". Eso lo podemos ver usando el comando `quantile` y las distintas opciones para el argumento `type`:

```
quantile(xSnew,0.5,type=1)
```

```
##      50%
## 38.87866
```

```
quantile(xSnew,0.5,type=2)
```

```
##      50%
## 39.29656
```

```
quantile(xSnew,0.5,type=3)
```

```
##      50%
## 38.87866
```

```
quantile(xSnew,0.5,type=4)
```

```
##      50%
## 38.87866
```

```
quantile(xSnew,0.5,type=5)
```

```
##      50%
## 39.29656
```

Cuartiles

Así como la mediana divide a los datos a la mitad, los cuartiles dividen a los datos en "cuartos". Es decir,

- el primer cuartil (se suele llamar q_1) deja el 25% de los datos a su izquierda y 75% a su derecha
- el segundo cuartil es la mediana, ya que deja el 50% de los datos a su izquierda y 50% a su derecha
- el tercer cuartil (se suele llamar q_3) deja el 75% de los datos a su izquierda y 25% a su derecha

Sus cálculos son similares a los de la mediana, puede tener que tomarse un compromiso en caso de que no se puedan acumular **exactamente** la cantidad de datos requerida a izquierda y derecha.

Primer cuartil

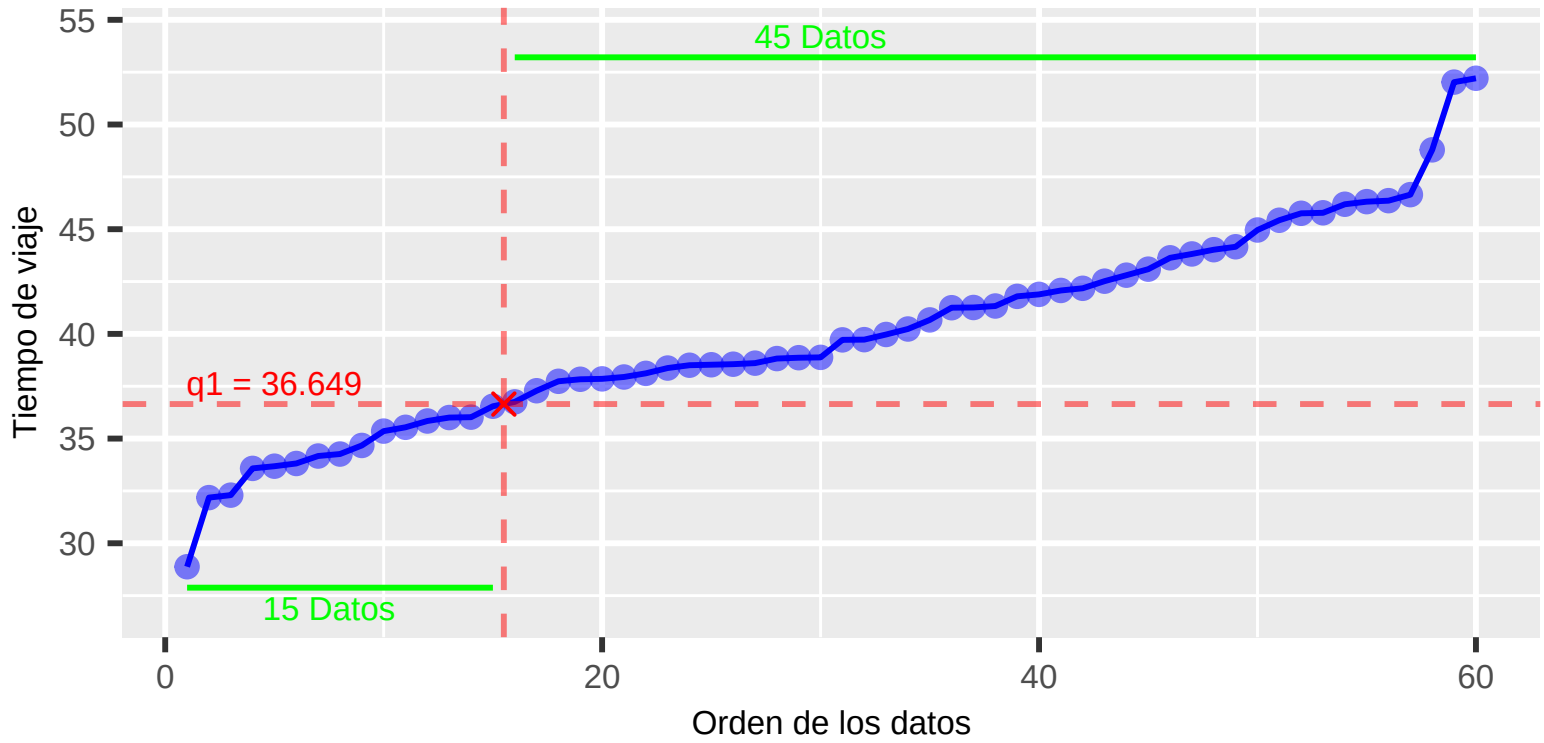
Para el ejemplo, el 25% de los 60 datos totales son 15 datos. Por lo tanto, calculamos el primer cuartil como el promedio entre la observación 15 y 16 de los datos ordenados:

```
q1=(xSnew[15]+xSnew[16])/2;print(q1)
```

```
## [1] 36.64894
```

Gráficamente, veamos que el cálculo del primer cuartil cumple lo pedido.

```
plot(GG_N)
```



Tercer cuartil

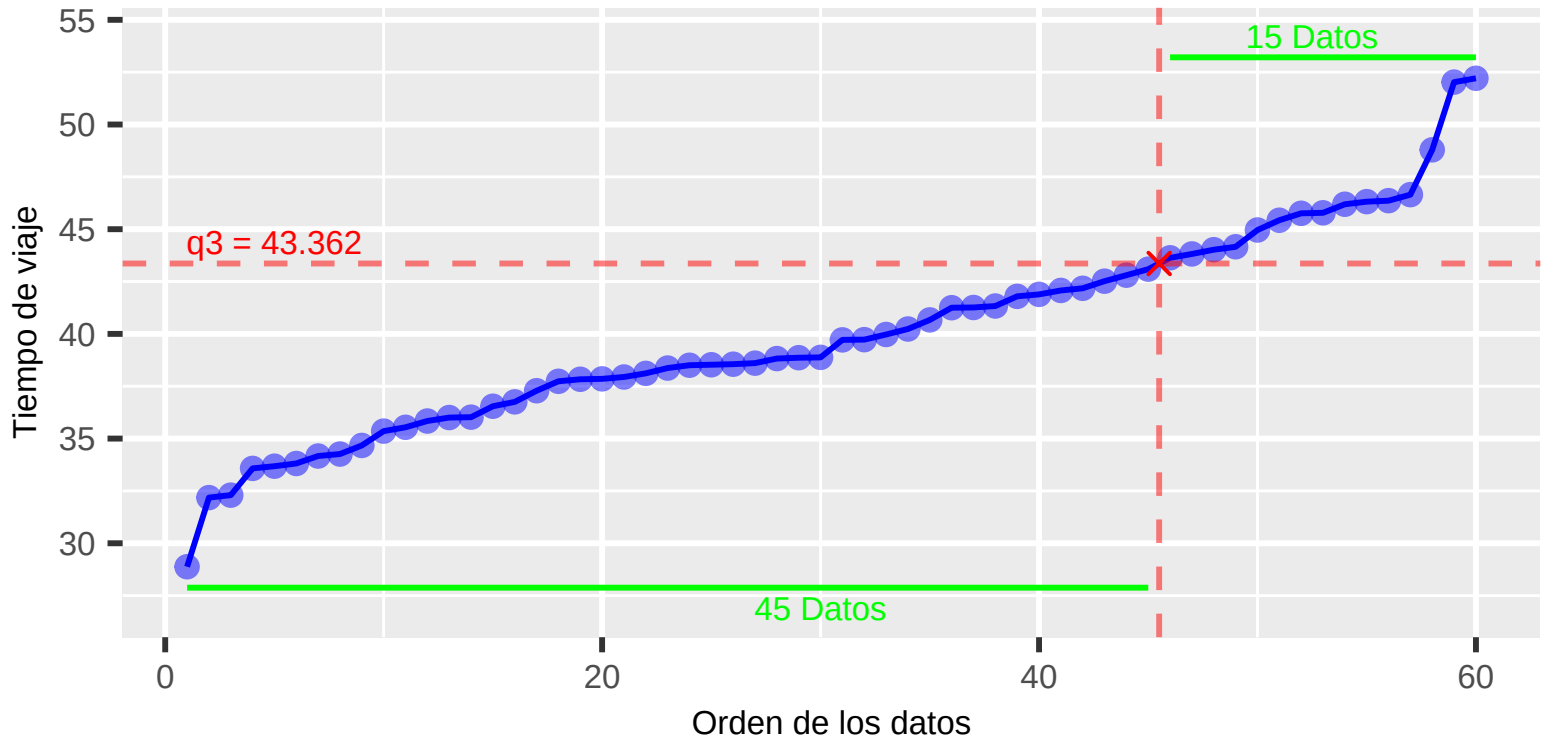
Para el ejemplo, el 75% de los 60 datos totales son 45 datos. Por lo tanto, calculamos el tercer cuartil como el promedio entre la observación 45 y 46 de los datos ordenados:

```
q3=(xSnew[45]+xSnew[46])/2;print(q1)
```

```
## [1] 36.64894
```

Gráficamente, veamos que el cálculo del tercer cuartil cumple lo pedido.

```
plot(GG_N)
```

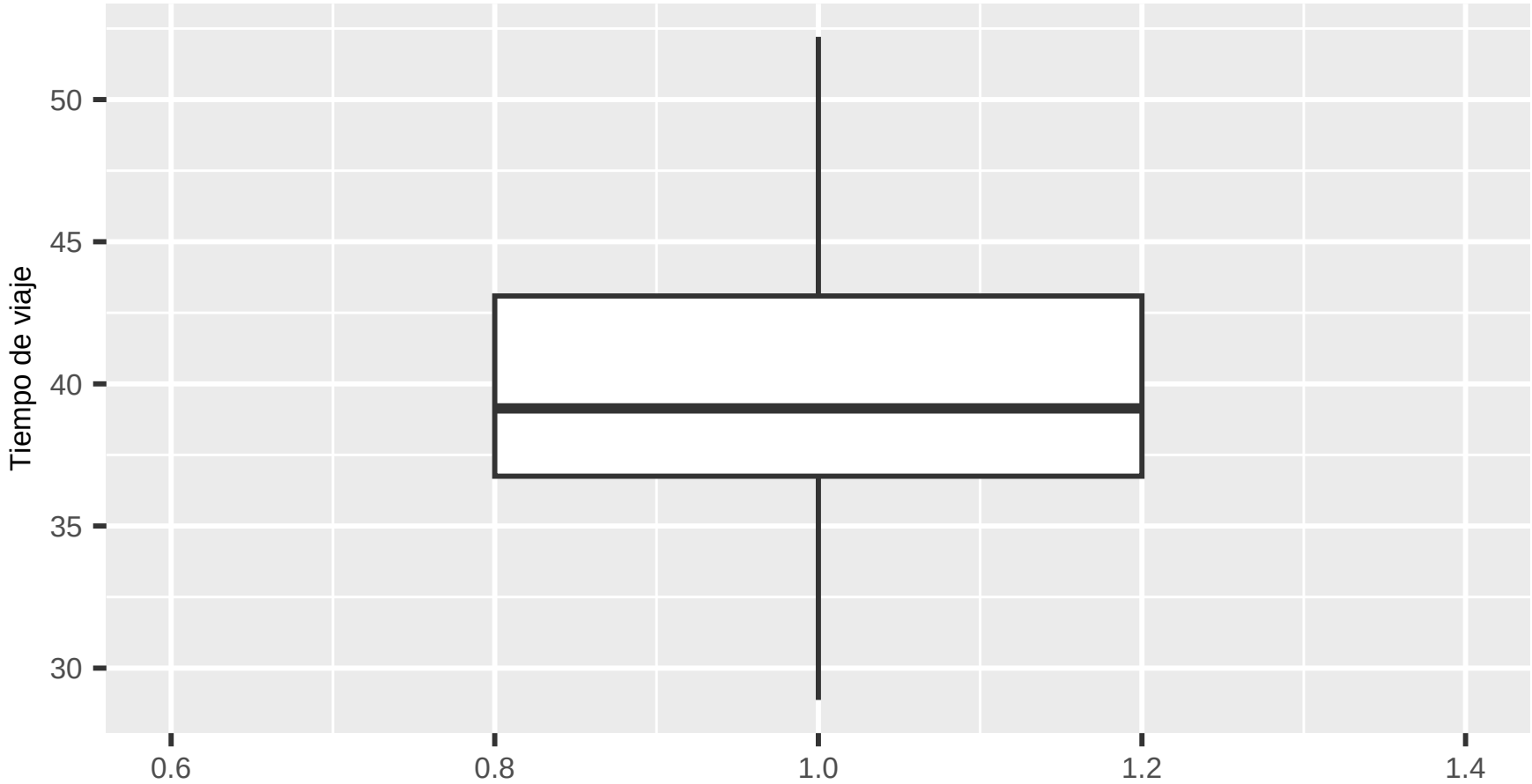


Boxplots

Boxplots

Con estos cálculos, hay un gráfico muy útil que permite ver la distribución de los datos de forma resumida, ya que en vez de graficar todas las observaciones, se basa en los cuartiles. El gráfico se denomina "boxplot"

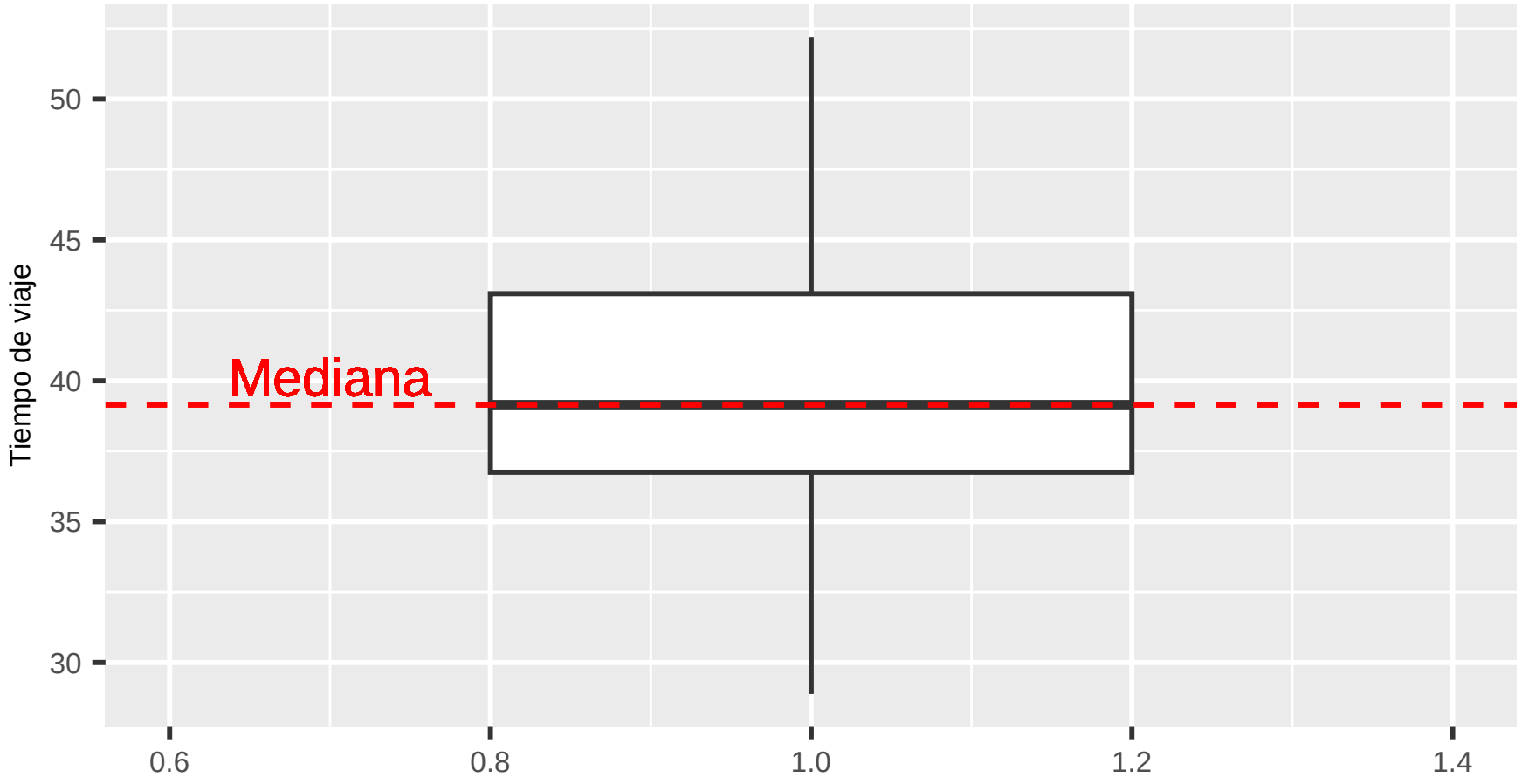
```
plot(GG)
```



Boxplots

La línea central de la "caja" representa la mediana:

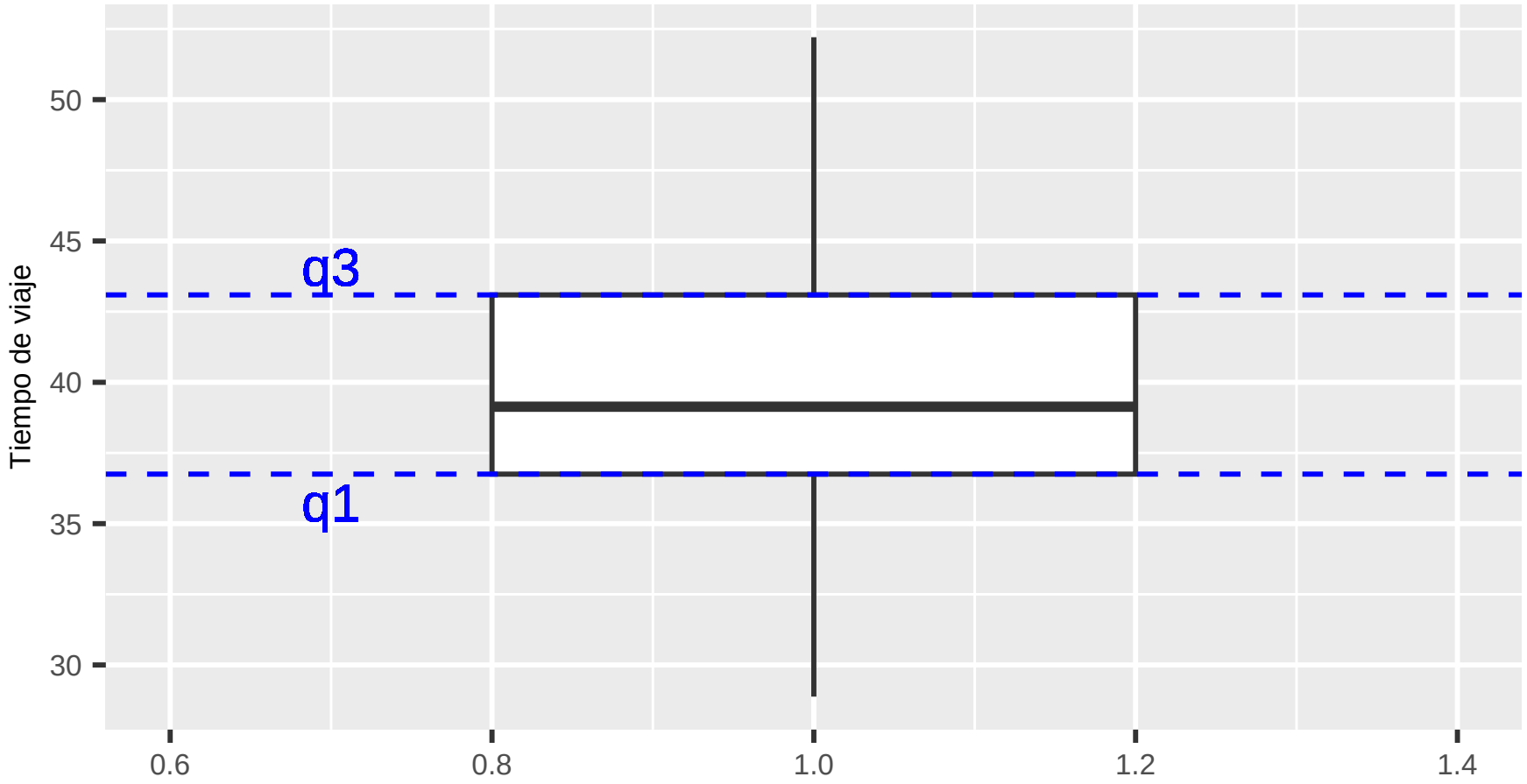
```
plot(GG_N)
```



Boxplots

La los límites de la "caja" son los cuartiles, por lo tanto, la "caja" representa el 50% **central** de los datos:

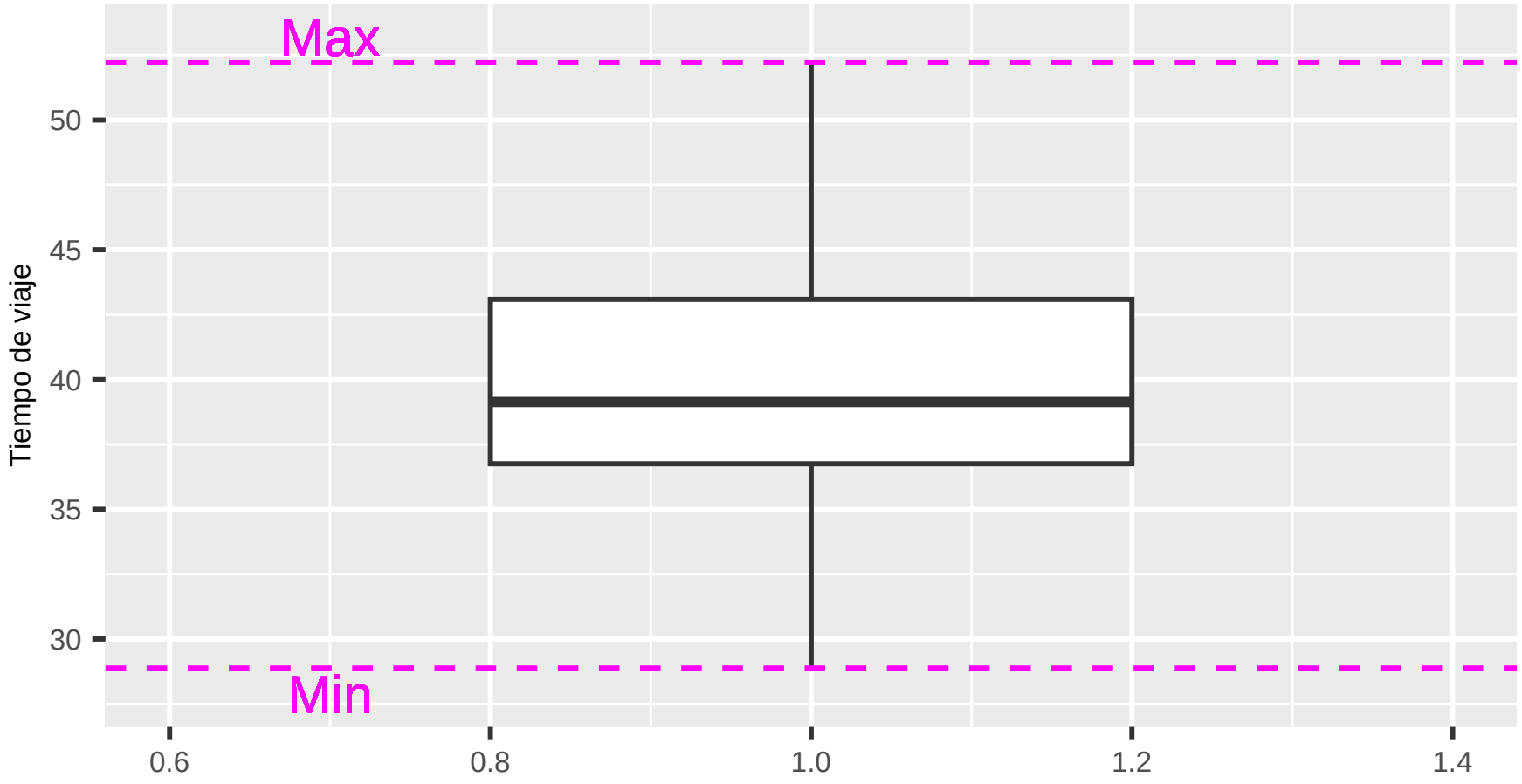
```
plot(GG_N)
```



Boxplots

Por último, el gráfico se mueve entre el mínimo y el máximo de los valores observados

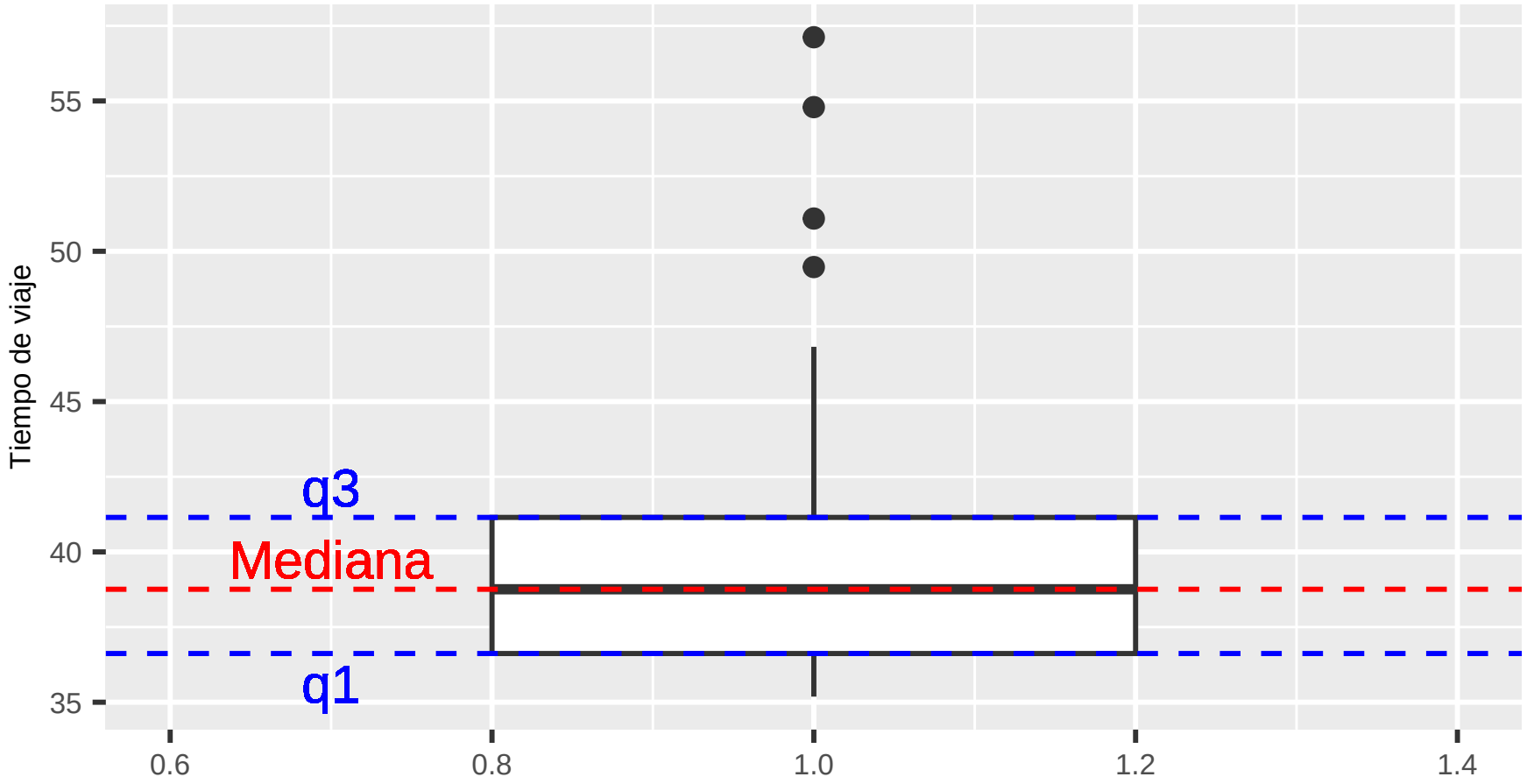
```
plot(GG_N)
```



Asimetría

Con este gráfico también se puede ver la asimetría, notamos aquí diferencias entre la distancia de ambos cuartiles a la mediana.

```
plot(GG_N)
```



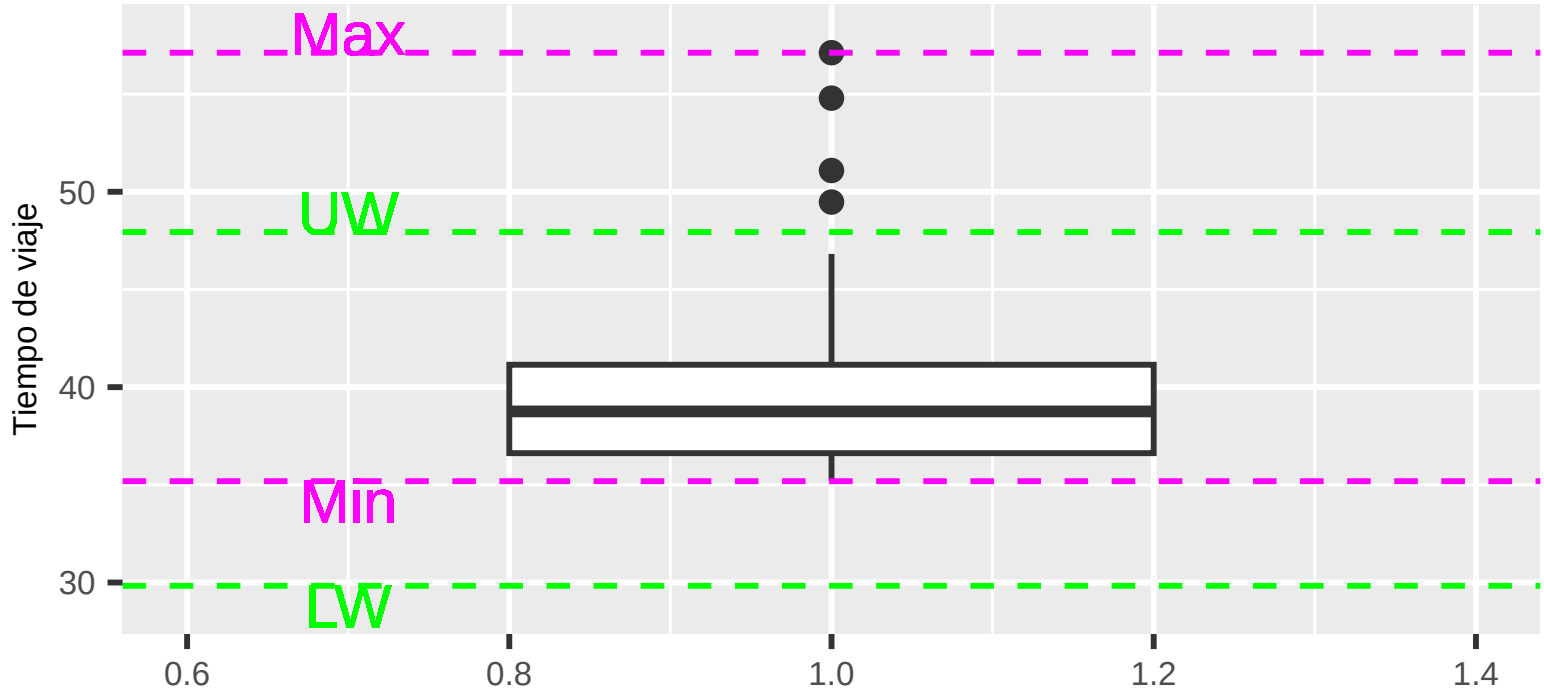
Outliers

Además, hay observaciones que se grafican con puntos que se denominan "outliers". Estos datos se denominan así porque se consideran atípicos, ya que se consideran suficientemente alejados del 50% central. Los límites de lo que se considera "normal", según la regla establecida por un matemático llamado Tukey, son los siguientes:

- $L_W = q_1 - 1.5 \cdot (q_3 - q_1) = q_1 - 1.5 \cdot IQR$
- $U_W = q_3 + 1.5 \cdot (q_3 - q_1) = q_3 + 1.5 \cdot IQR$

Es decir, se basa en el Interquartilic Range (IQR) que representa la longitud de la caja central. Esa medida da una noción de dispersión normal alrededor de la mediana. Por lo tanto, los datos que excedan esos valores se consideran outliers:

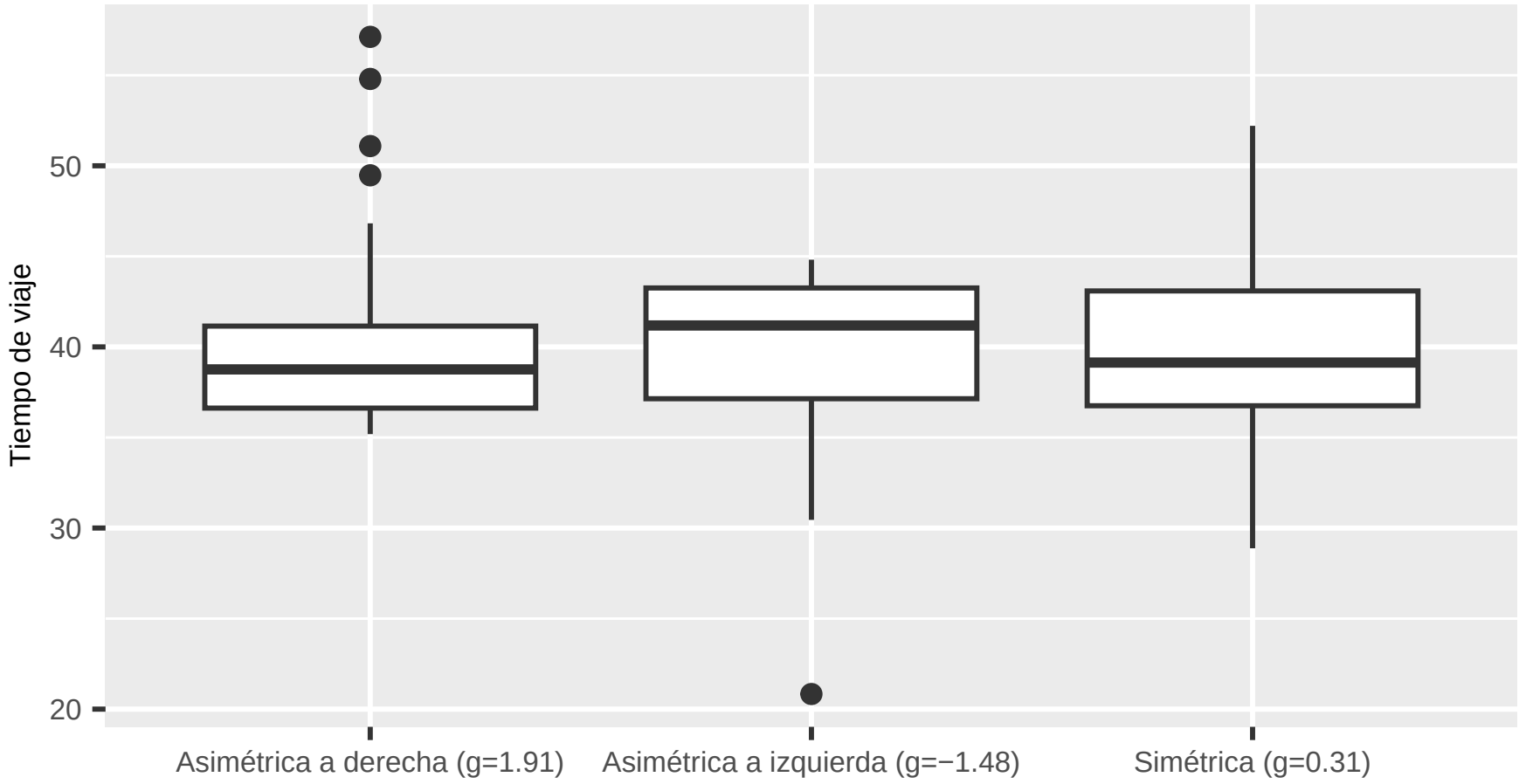
```
plot(GG_N)
```



Comparación de boxplots

Para comparar boxplots, deben disponerse en paralelo y en el mismo gráfico:

```
plot(GG)
```



Datos agrupados

Datos agrupados

A veces, los datos no vienen individualizados, sino que se disponen rangos de valores (entre un límite inferior L_i y un límite superior L_s) y se calcula cuántos datos se observaron en cada rango. La cantidad de datos en cada intervalo se denomina frecuencia y se suele notar con la letra f_i.

```
print(dfI)
```

```
##   Li Ls fi
## 1 27 30  1
## 2 30 33  2
## 3 33 36  9
## 4 36 39 18
## 5 39 42 10
## 6 42 45 10
## 7 45 48  7
## 8 48 51  1
## 9 51 54  2
```

A partir de estos datos, se entiende que, por ejemplo:

- Hay 1 dato entre 27 y 30 minutos.
- Hay 2 datos entre 30 y 33 minutos.
- Hay 9 datos entre 33 y 36 minutos.
- Hay 18 datos entre 36 y 39 minutos.
- Hay 10 datos entre 39 y 42 minutos.

Es decir, ya no sabemos los valores específicos observados, sino que hemos perdido información.

Media con datos agrupados

Ante esta pérdida de información, es importante tratar de que los cálculos puedan adaptarse a esta adversidad. Por ejemplo, para hallar la media (que representaría un centro para los datos), habría que realizar un promedio.

El primer problema, es que en un rango de valores no hay ningún valor específico para sumar. Por lo tanto, se puede considerar un representante de cada intervalo denominado "marca de clase" y que se nota con la letra x_i:

```
print(dfI)
```

```
##   Li Ls fi   xi
## 1 27 30  1 28.5
## 2 30 33  2 31.5
## 3 33 36  9 34.5
## 4 36 39 18 37.5
## 5 39 42 10 40.5
## 6 42 45 10 43.5
## 7 45 48  7 46.5
## 8 48 51  1 49.5
## 9 51 54  2 52.5
```

Por lo tanto, se podrían promediar estas marcas de clase y obtener un centro. Es decir, siendo L el número de intervalos:

$$\frac{\sum_{i=1}^{L} x_i}{L} = \frac{28.5 + 31.5 + 34.5 + 37.5 + 40.5 + 43.5 + 46.5 + 49.5 + 52.5}{9} = 40.5$$

Media con datos agrupados

Sin embargo, la cuenta anterior en la que sólo se promedian las marcas de clase, todos los intervalos tienen la misma influencia. Como se quiere buscar un centro para los datos, tiene sentido que los intervalos de mayor frecuencia tengan mayor influencia. Por lo tanto, la media con los datos agrupados se calcula del siguiente modo:

$$\overline{x}_{Ag} = \frac{\sum_{i=1}^{L} x_i \cdot f_i}{n}$$

```
dfI
```

```
##   Li Ls fi   xi xi.fi
## 1 27 30  1 28.5  28.5
## 2 30 33  2 31.5  63.0
## 3 33 36  9 34.5 310.5
## 4 36 39 18 37.5 675.0
## 5 39 42 10 40.5 405.0
## 6 42 45 10 43.5 435.0
## 7 45 48  7 46.5 325.5
## 8 48 51  1 49.5  49.5
## 9 51 54  2 52.5 105.0
```

Es decir, la media con los datos agrupados tiene el siguiente resultado:

$$\frac{\sum_{i=1}^{L} x_i \cdot f_i}{n} = \frac{28.5 + 63 + 310.5 + 675 + 405 + 435 + 325.5 + 49.5 + 105}{60} = 39.95$$

Vemos que es similar al resultado con los datos sin agrupar:

```
mean(xSnew) # Valor real
```

```
## [1] 40.03473
```

Desvío estándard con datos agrupados

De manera similar, para el cálculo del desvío con los datos agrupados, se realiza una suma similar a la de los datos sin agrupar sólo que se incorporan las frecuencias al cálculo:

$$s_{Ag} = \sqrt{\frac{\sum_{i=1}^{L} (x_i - \overline{x}_{Ag})^2 \cdot f_i}{n-1}}$$

Del mismo modo que antes, se puede calcular el valor de la suma correspondiente a cada intervalo:

```
dfI
```

```
##   Li Ls fi   xi DifCuad.fi
## 1 27 30  1 28.5   131.1025
## 2 30 33  2 31.5   142.8050
## 3 33 36  9 34.5   267.3225
## 4 36 39 18 37.5   108.0450
## 5 39 42 10 40.5     3.0250
## 6 42 45 10 43.5   126.0250
## 7 45 48  7 46.5   300.3175
## 8 48 51  1 49.5    91.2025
## 9 51 54  2 52.5   315.0050
```

Por lo tanto, el desvío con estos datos se calcula del siguiente modo, que es similar al obtenido por los datos sin agrupar:

```
sAg=sqrt(sum(dfI$DifCuad.fi)/(n-1));print(sAg)
```

```
## [1] 5.016667
```

```
print(sd(xSnew)) # Valor real
```

```
## [1] 4.831502
```

Medidas de resumen con datos agrupados

Obviamente no dan exactamente igual, ya que con la pérdida de información al agrupar los datos se pierde precisión también. De todas formas deben dar similares.

Notar que la única diferencia entre datos agrupados y datos sin agrupar, es que se incluye la frecuencia en las sumas. Por lo tanto, podemos hacer el mismo paralelismo para los cálculos de la simetría y la kurtosis:

	Datos sin agrupar (n datos)		**Datos agrupados (n datos, L intervalos)**
Media:	$\overline{x} = \frac{\sum_{i=1}^{n} x_i}{n}$	$\Rightarrow$	$\overline{x}_{Ag} = \frac{\sum_{i=1}^{L} x_i \cdot f_i}{n}$
Desvío:	$s = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \overline{x})^2}{n-1}}$	$\Rightarrow$	$s_{Ag} = \sqrt{\frac{\sum_{i=1}^{L} (x_i - \overline{x}_{Ag})^2 \cdot f_i}{n-1}}$
Simetría:	$\gamma = \frac{\sum_{i=1}^{n} (x_i - \overline{x})^3}{n \cdot s^3}$	$\Rightarrow$	$\gamma_{Ag} = \frac{\sum_{i=1}^{L} (x_i - \overline{x}_{Ag})^3 \cdot f_i}{n \cdot s_{Ag}^3}$
Kurtosis:	$\kappa = \frac{\sum_{i=1}^{n} (x_i - \overline{x})^4}{n \cdot s^4} - 3$	$\Rightarrow$	$\kappa_{Ag} = \frac{\sum_{i=1}^{L} (x_i - \overline{x}_{Ag})^4 \cdot f_i}{n \cdot s_{Ag}^4} - 3$

Mediana con datos agrupados

Para calcular percentiles como la mediana, la lógica es distinta porque el cálculo no se basa en sumas, sino que se basa en cantidad de datos acumulados. Por lo tanto, ya no se usan las marcas de clase y se focaliza en cuántos datos fueron acumulados hasta cada límite superior del intervalo. Es decir, se utiliza la frecuencia **acumulada** que se denota F_i:

```
print(dfI)
```

```
##   Li Ls fi Fi
## 1 27 30  1  1
## 2 30 33  2  3
## 3 33 36  9 12
## 4 36 39 18 30
## 5 39 42 10 40
## 6 42 45 10 50
## 7 45 48  7 57
## 8 48 51  1 58
## 9 51 54  2 60
```

Estos datos se leen del siguiente modo. Por ejemplo:

- Hay 1 dato con tiempos menores a 30 minutos
- Hay 3 datos con tiempos menores a 33 minutos.
- Hay 12 datos con tiempos menores a 36 minutos.
- Hay 30 datos con tiempos menores a 39 minutos.

Mediana con datos agrupados

La lógica de la mediana es que acumula el 50% de los datos. Como en este caso los datos totales son 60, los datos que deben acumularse son 30. Mirando la columns `Fi`, se ve que esos datos justo se acumulan a los 39 minutos:

```
print(dfI)
```

```
##   Li Ls fi Fi
## 1 27 30  1  1
## 2 30 33  2  3
## 3 33 36  9 12
## 4 36 39 18 30
## 5 39 42 10 40
## 6 42 45 10 50
## 7 45 48  7 57
## 8 48 51  1 58
## 9 51 54  2 60
```

Es decir, la mediana para estos datos tiene un valor de 39 minutos. Es similar al valor real obtenido con los datos sin agrupar:

```
median(xSnew) # Valor real
```

```
## [1] 39.29656
```

Cuartiles con datos agrupados

Claro que no siempre es tan sencillo. Para la mediana encontramos el valor que acumula exactamente los datos deseados, pero casi nunca es el caso.

Por ejemplo, si quisiéramos calcular el primer cuartil, sería aquel valor que acumula el 25% de 60 datos, serían 15 datos. En este caso, no hay un intervalo que acumule esos valores exactamente:

```
print(dfI)
```

```
##   Li Ls fi Fi
## 1 27 30  1  1
## 2 30 33  2  3
## 3 33 36  9 12
## 4 36 39 18 30
## 5 39 42 10 40
## 6 42 45 10 50
## 7 45 48  7 57
## 8 48 51  1 58
## 9 51 54  2 60
```

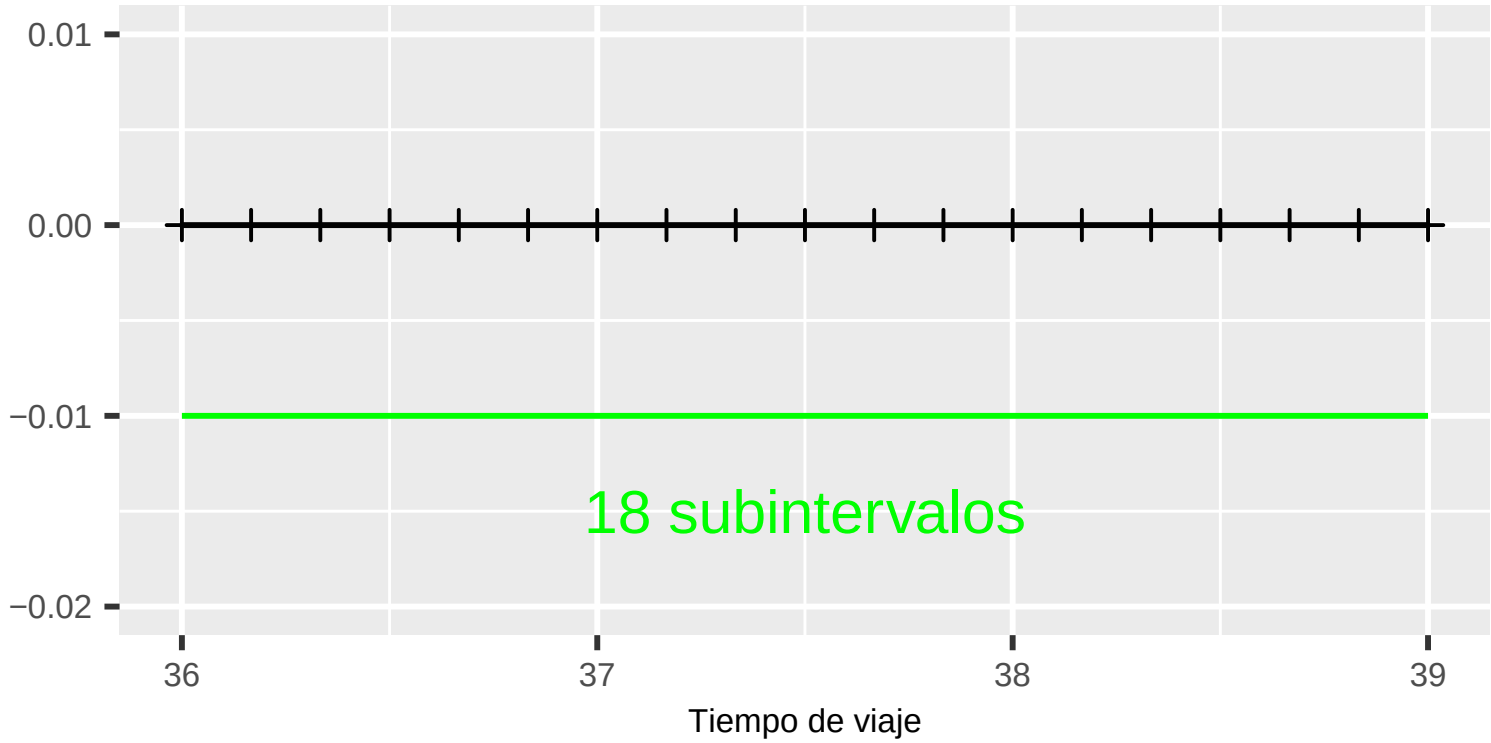
- Vemos en estos datos que hasta los 36 minutos se acumulan 12 datos y hasta los 39 minutos se acumulan 30.
- Por lo tanto, si bien no sabemos cuál es el primer cuartil, intuitivamente vemos que este valor debería estar entre 36 y 39 minutos.
- Además, intuitivamente, por la cantidad de datos a acumular (15), debería dar más cercano a 36 (12 datos acumulados) que a 39 (30 datos acumulados)

Cuartiles con datos agrupados

Como hemos perdido información, tendremos que pagar un costo de precisión. Sin embargo, buscaremos utilizar toda la información disponible.

Por ejemplo, sabemos que en el intervalo de 36 a 39 donde se encuentra el primer cuartil, hay 18 datos. Por lo tanto, la mejor aproximación que podemos hacer es asumir que esos 18 datos se distribuyen equitativamente en ese intervalo:

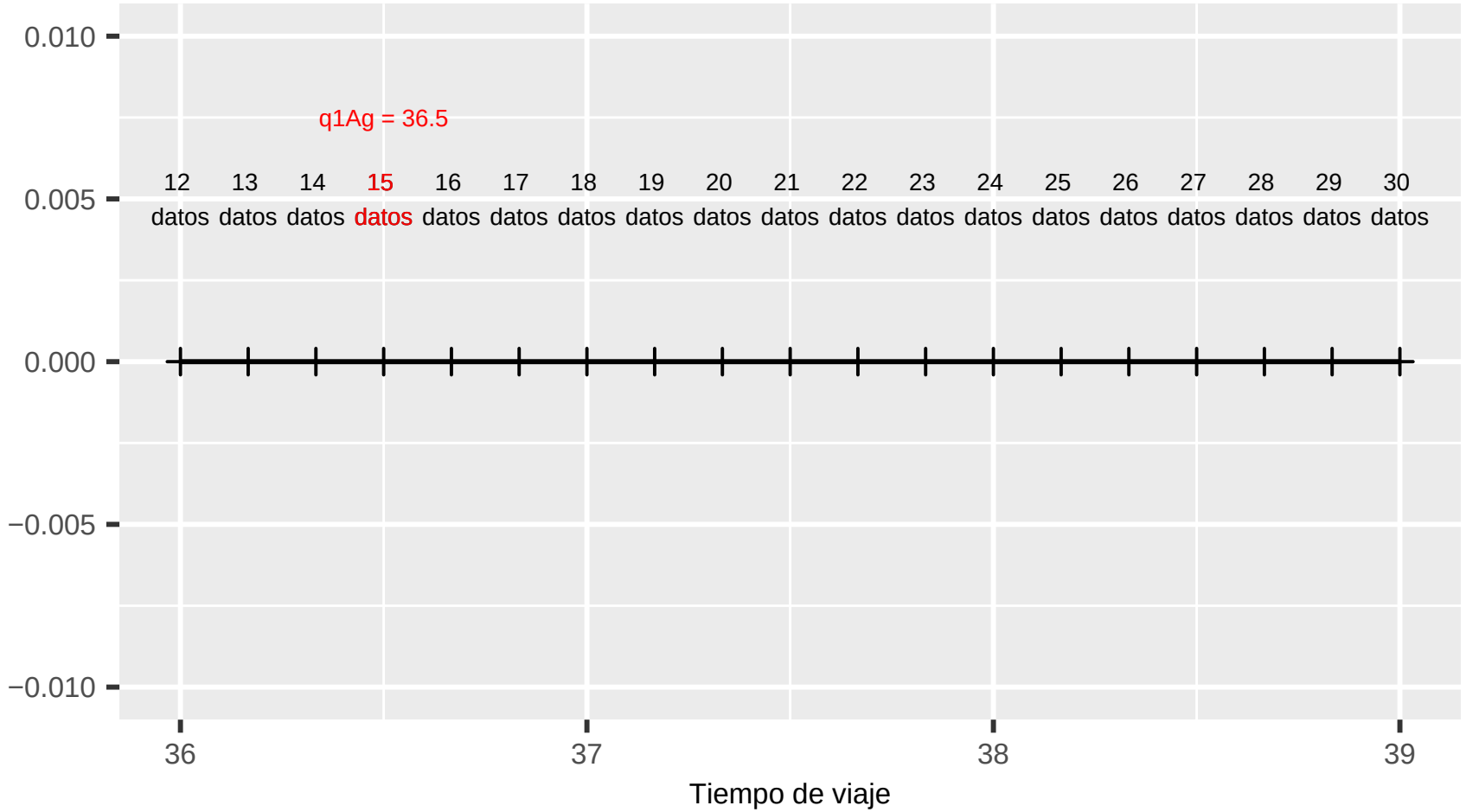
```
print(GG_N)
```



Cuartiles con datos agrupados

Por lo tanto, si asumimos esta distribución, tendríamos que considerar el límite del tercer subintervalo ya que al comienzo del intervalo total se acumularon 12 datos y restan acumular 3.

```
print(GG_N)
```



Cuartiles con datos agrupados

Analíticamente, el cuartil se calcula del siguiente modo:

$$q1_{Ag} = Li_i + \frac{0.25 \cdot n - F_{i-1}}{f_i} \cdot (Ls_i - Li_i) = 36 + \frac{15 - 12}{18} \cdot 3 = 36.5$$

donde i es el intervalo en el que identificamos donde está el cuartil.

Vemos que da similar al valor real:

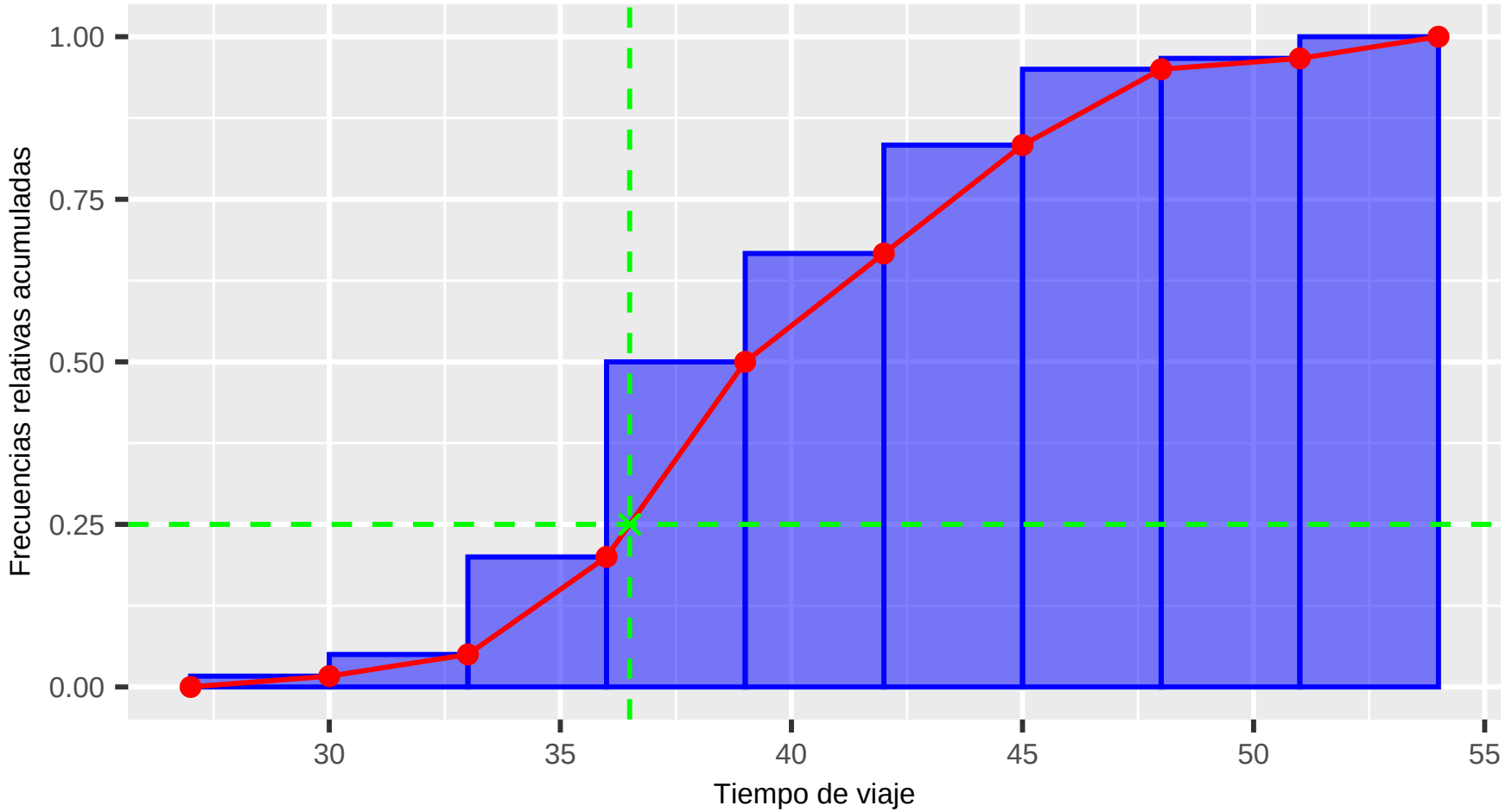
```
print(quantile(xSnew,0.25)) # Valor real
```

```
##       25%
## 36.70079
```

Interpretación gráfica

Gráficamente, se puede interpretar el valor como la abscisa del polígono de frecuencias acumuladas relativas que logra una ordenada de 25%:

```
print(GG)
```



Ejercicio

Calcular el tercer cuartil con estos datos agrupados