Punto 2° Parcial: Ajuste de un modelo de R.L.M

Universidad Nacional de Colombia
Analisís de Regresión 2022-1S
Medellín, Colombia
2022

Daniel Villa 1005087556 Juan Pablo Vanegas 1000640165

UNIVERSIDAD NACIONAL DE COLOMBIA

Contents

1 Objetivos:

Crear un modelo ajustado de R.L.M. por el cual se pueda predecir la estatura de un individuo (discrimiando por genero) sabiendo las estaturas de los padres (madre y padre) utilizando el software estadístico *R*.

1.1 Objetivos específicos

- Plantear el modelo de R.L.M.
- Interpretar los parámetros del modelo.
- Determinar si el efecto de las estaturas de los padres sobre la estatura del sujeto es significativo.
- Interpretar nuestro R^2.
- Validar los supuestos del modelo.
- Aplicar la prueba de falta de ajuste.

2 Antecedentes Relevantes

La población encuestada, pertenece a estudiantes de la Universidad Nacional de Colombia sede Medellín de diferentes carreras, es decir, la mayoria de los sujetos de la muestra son jovenes entre los 18 y los 25 años, además decidimos que solamente aquellos que tenian la posibilidad de saber las estaturas de sus padres entraban a nuestra base de datos, ya que el porceso sería más arduo si tomamos datos donde nos faltan llenar valores en las celdas correspondientes.

3 Variables de respuesta:

En nuestro caso será la estatura del sujeto (Hombre o Mujer) para ajustar un modelo para predecir por medio de nuestras variables predictoras la esturura del sujeto.

4 Variable de Control:

En este caso tendremos 3 varibles haciendo de este un modelo de R.L.M.

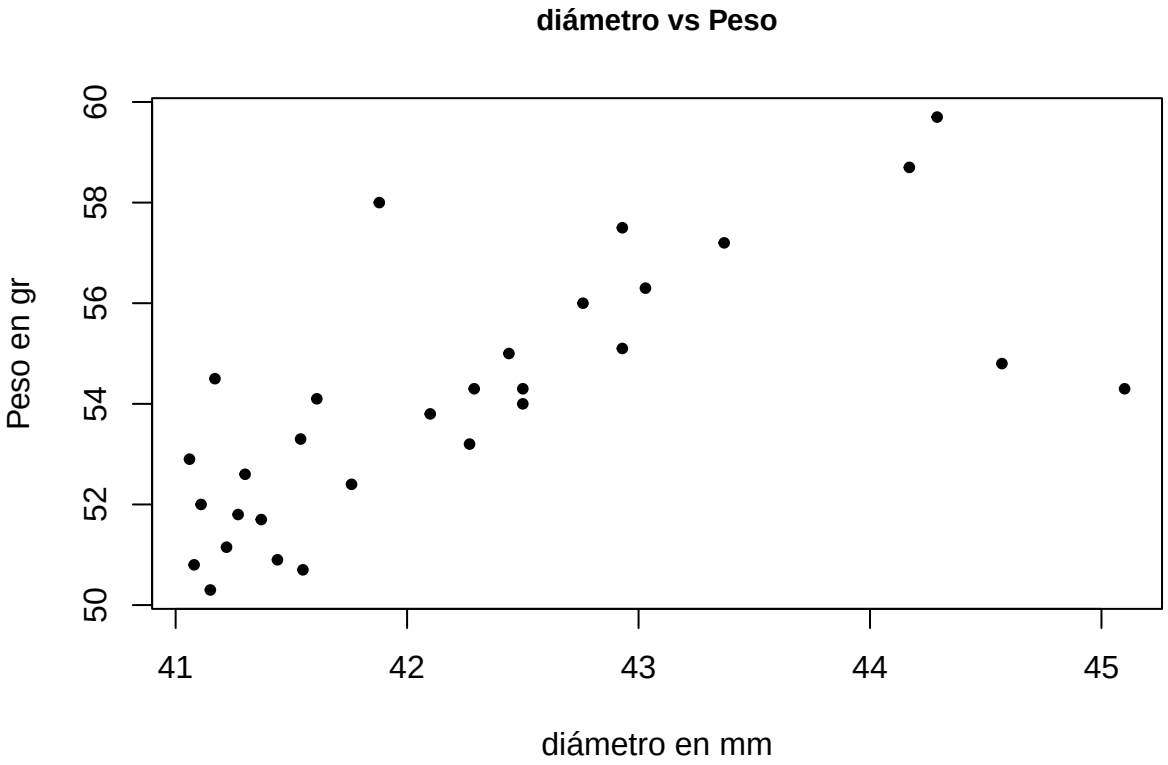
1. Estatura del Padre.
2. Estatura del Madre.
3. Genero del sujeto.

5 Ajuste del modelo

Antes de observar o crear un modelo dado unos puntos, primero haremos un test de correlación de los datos para estudiar el grado de variación conjunta entre el diámetro y peso de los huevos:

```
##
##  Pearson's product-moment correlation
##
## data:  datos$diametro and datos$peso
## t = 5.1667, df = 28, p-value = 1.758e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4521558 0.8459674
## sample estimates:
##       cor
## 0.6986211
```

Como podemos ver se rechaza H_0 ya que el $p - valor < 0.05$, es decir nuestros datos peso y el diámetro de los huevos tomados no tienen una corelación no significativa, más bien tienen una correlación positiva entre los datos, que vamos a ver por medio de un grafico de dispersión:

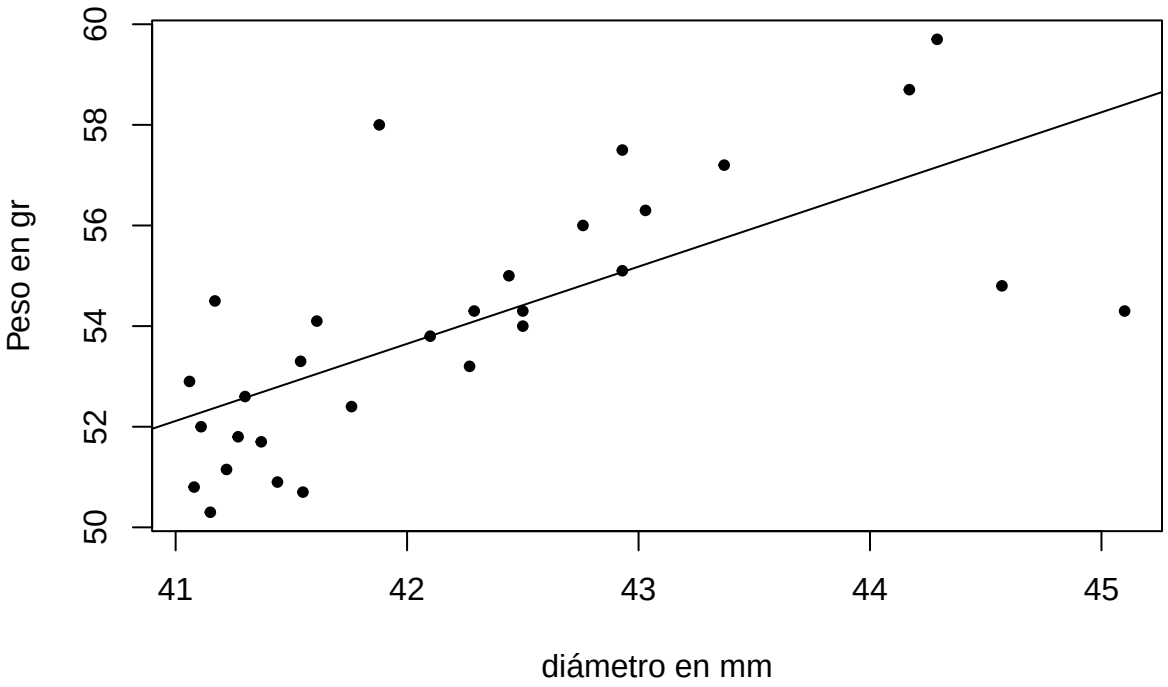


Como podemos ver los datos si tienen en algún grado una correlación positiva (mientras amuentan los valores del diámetro aumenta el peso) apriori.

Una vez visto que existe relación entre las variables pasamos a realizar el ajuste del modelo. Para ello usamos la función `lm()` que toma la forma:

```
##
## Call:
## lm(formula = peso ~ diametro, data = datos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.1037 -0.9563  0.0118  1.0663  4.5359
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -10.7807    12.5511  -0.859    0.398
## diametro      1.5340     0.2969   5.167 1.76e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.81 on 28 degrees of freedom
## Multiple R-squared:  0.4881, Adjusted R-squared:  0.4698
## F-statistic:  26.7 on 1 and 28 DF,  p-value: 1.758e-05
```

Como podemos ver nuestro intercepto dado el $p - valor > 0.05$ decimos que el intercepto tengan valor a cero, esto es logico ya que seria extraño encintrar huevos con diámetro cero y un valor de peso inicial.



En primer lugar deseamos obtener los estimadores puntuales, errores estándar y p-valores asociados con cada coeficiente

```
##              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) -10.78066 12.5510919 -0.858942 3.976665e-01
## diametro      1.53402  0.2969033  5.166734 1.758177e-05
```

El resultado del ajuste es:

`(12.5510919) (0.2969033)`

$$Peso \ = \ 0 + 1.53402 * dimetro$$

donde los valores entre paréntesis indican los errores estándar de cada coeficiente. Además, puesto que los p-valores son mayores y menores a 0.05, podemos concluir que:

1. En este caso no tiene sentido analizar el valor de la constante para diámetro = 0, ya que pertenecería a un supuesto donde el huevo (imaginariamente) exista, de ahí que el peso del huevo para diámetro = 0 sea de 0, menor que cualquiera de los datos de nuestro conjunto, en conclusión un huevo con diámetro = 0 no existe y más si su peso = 0.

2. Existen evidencias estadísticas suficientes para considerar que hay una relación lineal entre diámetro y peso. Dicha relación es positiva cuando aumenta el diámetro de un huevo dado aumente el peso del mismo. Además vemos que por cada *mm* que aumenta el diámetro de un huevo, aumenta el peso en 1.53 *gramos*.

3. El error estándar residual estimado (s) es de 3.1. Este valor es muy importante, es un medidor de la calidad (precisión) del modelo. Además nos vamos a basar en él para calcular los intervalos de confianza para el coeficiente del modelo.

5.1 Intervalos de Confianza

Obtenemos los correspondientes intervalos de confianza para el parámetro m de nuestras modelo = $Y = 0 + m * X$ con nivel significación al 95%

Parámetro	2.5 %	97.5 %
diametro	0.92584	2.142199

Interpretamos los intervalos: con una probabilidad del 95%, el efecto asociado con diametro se encuentra en el intervalo $(0.9258416, 2.142199)$.

5.2 Tabla ANOVA

Obtenemos la correspondiente tabla ANOVA donde vemos la descomposición de la variabilidad del modelo

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
diametro	1	87.49255	87.492552	26.69514	1.76e-05
Residuals	28	91.76920	3.277471	NA	NA

Observamos que la variabilidad explicada por el modelo, `SSM=87.493`, es inferior a la que queda por explicar (residuos), `SSR=91.769` y el estadístico `F=26.695`, mayor que 1. Además, volviendo a ver el resumen del modelo

```
## F-statistic: 26.695 on 1 and 28 DF, p-value: 1.758e-05
```

tenemos que el $p - valor$ asociado con el estadístico F es inferior a 0.05.

La conclusión es que hay evidencias suficientes para poder rechazar la hipótesis nula, $H_0 : F = 1$ y por tanto, resulta posible establecer un modelo de regresión lineal para explicar el comportamiento del peso de un huevo en función de su diámetro.

5.3 Coeficiente de determinación

En el `modelo1` el valor de R^2 es `Multiple R-squared: 0.4881`, alrededor del 48.81% de la variabilidad del peso es explicada por la recta ajustada.

6 Análisis de los parámetros del modelo

El test ANOVA significativo nos dice si el modelo tiene, en general, un grado de predicción significativamente bueno para la variable resultado, pero no nos dice nada sobre la contribución individual del modelo. Para encontrar los parámetros del modelo y su significación tenemos que volver a la parte `Coefficients` en el resumen del modelo.

β_1	Estimate	Std. Error	t value	Pr(>
diametro	1.53402	0.2969033	5.166734	1.7581e-05

Observando la tabla vemos que β_1 es la pendiente de la recta y representa el cambio en la variable dependiente (peso) asociado al cambio de una unidad en la variable predictora. Si nuestra variable predictora incrementa una unidad, nuestro modelo predice que el peso de un huevo se incrementara en 1.534 gr, pues en este caso $\beta_1 = 1.534$ Por tanto, la ecuación del modelo queda:

$$Y = 0 + 1.534 * X$$

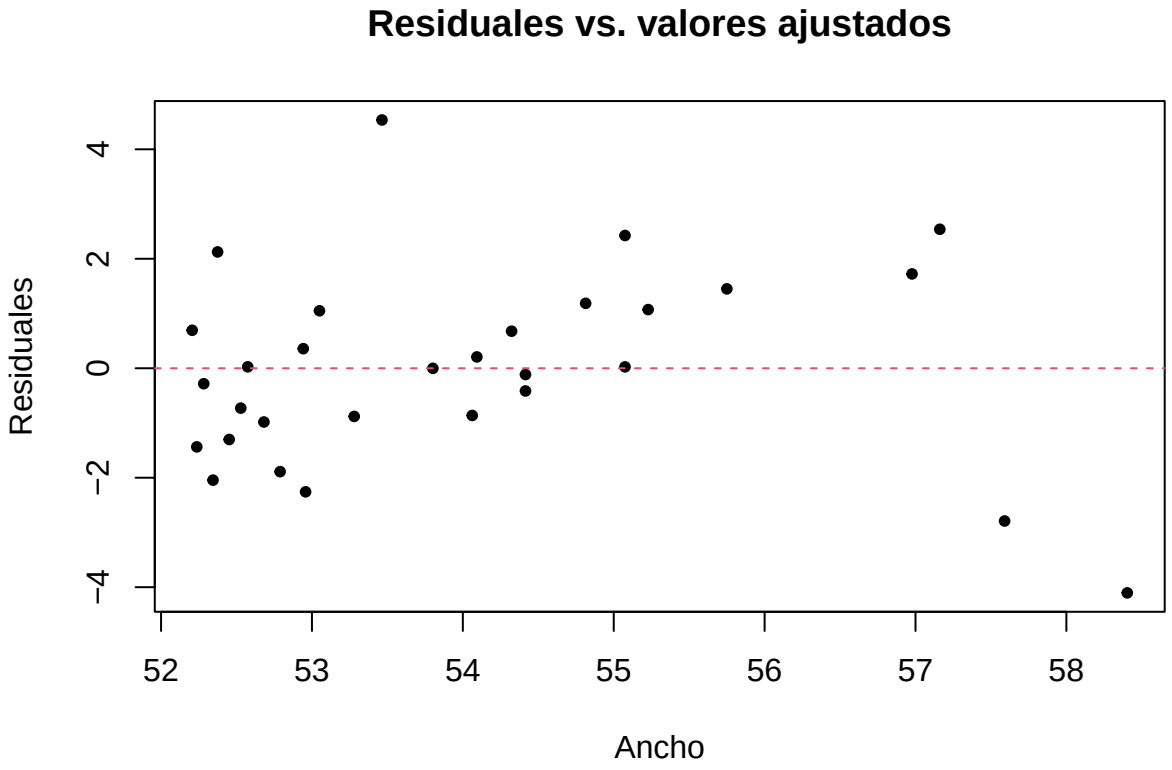
7 Diagnóstico del modelo

En este apartado hemos hecho uso tanto de J.Faraway (2009) como de Sánchez (2011) para el desarrollo del mismo.

Una vez que tenemos el modelo ajustado procedemos con su diagnóstico, que se realiza a través del **análisis de los residuos** ε_i:

- Las hipótesis de linealidad, homocedasticidad e independencia se contrastan a través de un análisis gráfico que enfrenta los valores de los residuos, ε_i, con los valores ajustados $\hat{x}_i$.
- Las hipótesis de media cero, varianza constante, incorrelación y normalidad la comprobamos analíticamente

Comenzaremos con el análisis gráfico. Los residuos deberían formar una nube de puntos sin estructura y con, aproximadamente, la misma variabilidad por todas las zonas como se muestra en el gráfico:



Continuamos ahora realizando el **diagnóstico analítico**. El primer paso es obtener los residuos, valores ajustados y estadísticos del modelo analizado para poder así estudiar si se cumplen los supuestos del mismo.

Obtención de residuos, valores ajustados y estadísticos necesarios

Para ello, añadimos los correspondientes resultados a nuestros datos a través del siguiente código:

El resultado es la creación de las siguientes variables:

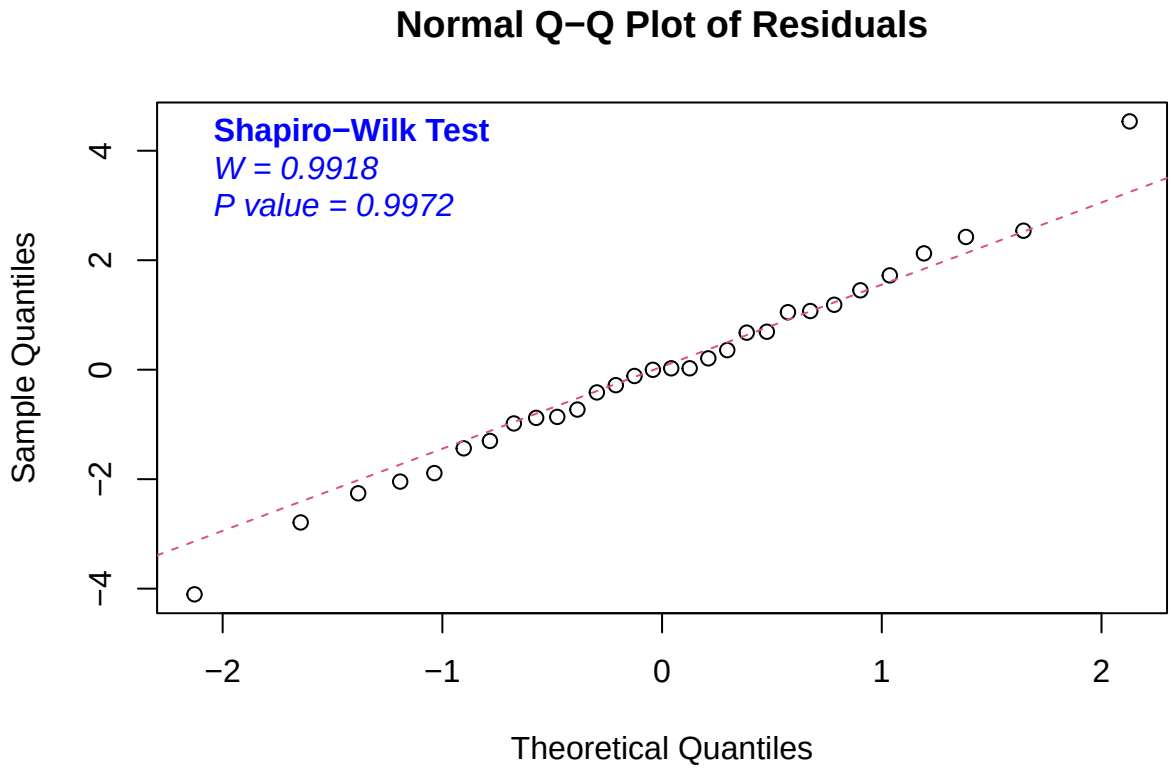
- `fitted.modelo1`: valores ajustados (valores de la variable respuesta) para las observaciones originales de la predictora.
- `residuals.modelo1`: residuos del modelo, esto es, diferencia entre valor observado de la respuesta y valor ajustado por el modelo.
- `rstudent.modelo1`: residuos estudentizados del modelo ajustado.

Vamos a utilizar todas estas variables para estudiar si nuestro modelo cumple las hipótesis.

7.1 Test de normalidad (test de Kolmogorov-Smirnov)

Empezamos el análisis con un gráfico `qqplot`, que enfrenta los valores reales a los valores que obtendríamos si la distribución fuera normal. Si los datos reales se distribuyen normalmente, estos tendrán la misma

distribución que los valores esperados y en el gráfico `qqplot` obtendremos una linea recta en la diagonal.



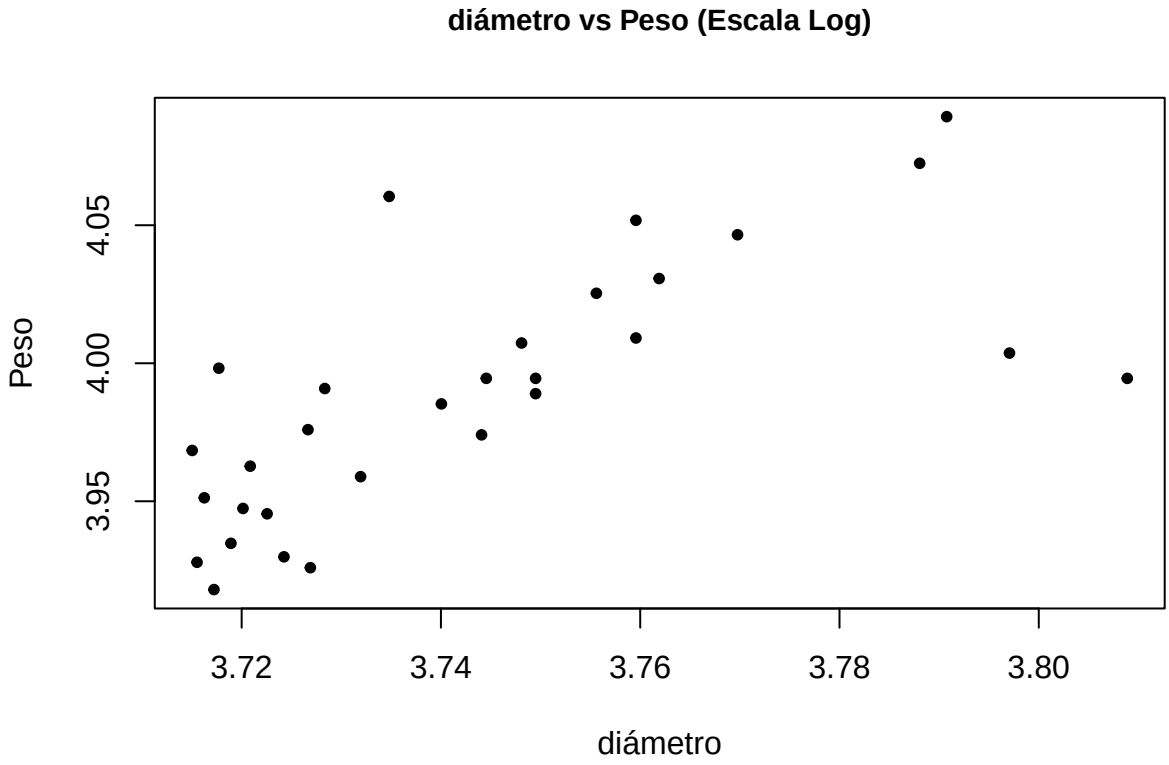
Podemos ver que nuestro $p - valor > 0.05$ por lo que no se rechaza la hipotesís nula donde los datos se distribuyen normal. además por medio de la grafica nos muestra como nuestros puntos se acomodan bien a la recta.

```
##
##  studentized Breusch-Pagan test
##
## data:  modelo1
## BP = 4.5466, df = 1, p-value = 0.03298
```

No Existe homogeneidad pues la significación es menor de 0.05, la varianza no es constante a lo largo de la muestra.

8 Transformación del Modelo

Dado que nuestra varianza no es constante, tendremos que hacer una transformación en nuestro modelo, es decir: $Y^* = log(Y)$ $X^* = log(X)$

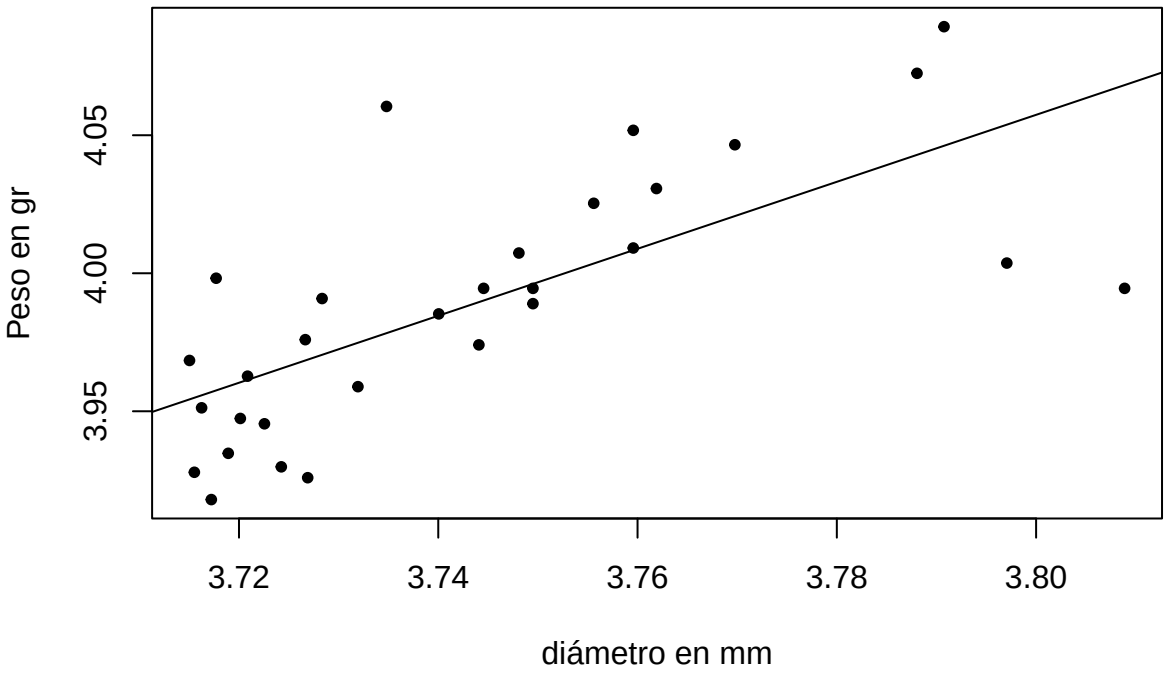


Como podemos ver los datos si tienen en algún grado una correlación positiva (mientras amuentan los valores del diámetro aumenta el peso) apriori.

Una vez visto que existe relación entre las variables pasamos a realizar el ajuste del modelo. Para ello usamos la función `lm(log())` *(log() <- logaritmo neperiano)* que toma la forma:

```
##
## Call:
## lm(formula = log(peso) ~ log(diametro), data = datos)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.073651 -0.017460  0.000711  0.020168  0.082143
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.5532     0.8645  -0.640    0.527
## log(diametro)   1.2133     0.2309   5.254 1.38e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03292 on 28 degrees of freedom
## Multiple R-squared:  0.4965, Adjusted R-squared:  0.4785
## F-statistic: 27.61 on 1 and 28 DF,  p-value: 1.385e-05
```

Como podemos ver nuestro intercepto dado el $p - valor > 0.05$ decimos que el intercepto tengan valor a cero, esto es logico ya que seria extraño encintrar huevos con diametro cero y un valor de peso inicial.



En primer lugar deseamos obtener los estimadores puntuales, errores estándar y p-valores asociados con cada coeficiente

```
##                 Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)   -0.5531989  0.8644818 -0.6399197 5.274268e-01
## log(diametro)  1.2133149  0.2309251  5.2541481 1.384744e-05
```

El resultado del ajuste es:

`(0.8644818) (0.2309251)`

$$Peso \ = \ 0 + 1.2133149 * dimetro$$

donde los valores entre paréntesis indican los errores estándar de cada coeficiente. Además, puesto que los p-valores son mayores y menores a 0.05, podemos concluir que:

1. En este caso no tiene sentido analizar el valor de la constante para diámetro = 0, ya que pertenecería a un supuesto donde el huevo (imaginariamente) exista, de ahí que el peso del huevo para diámetro = 0 sea de 0, menor que cualquiera de los datos de nuestro conjunto, en conclusión un huevo con diámetro = 0 no existe y más si su peso = 0.
2. Existen evidencias estadísticas suficientes para considerar que hay una relación lineal entre diámetro y peso. Dicha relación es positiva cuando aumenta el diámetro de un huevo dado aumente el peso del mismo. Además vemos que por cada *mm* que aumenta el diámetro de un huevo, aumenta el peso en 1.21 *gramos*.
3. El error estándar residual estimado (s) es de 0.001. Este valor es muy importante, es un medidor de la calidad (precisión) del modelo. Además nos vamos a basar en él para calcular los intervalos de confianza para el coeficiente del modelo.

8.1 Intervalos de Confianza

Obtenemos los correspondientes intervalos de confianza para el parámetro m de nuestras modelo = $Y^* = 0 + m * X^*$ con nivel significación al 95%

Parámetro	2.5 %	97.5 %
diametro	0.74028	1.686344

Interpretamos los intervalos: con una probabilidad del 95%, el efecto asociado con diametro se encuentra en el intervalo $(0.7402862, 1.686344)$.

8.2 Tabla ANOVA

Obtenemos la correspondiente tabla ANOVA donde vemos la descomposición de la variabilidad del modelo

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
log(diametro)	1	0.0299145	0.0299145	27.60607	1.38e-05
Residuals	28	0.0303414	0.0010836	NA	NA

Observamos que la variabilidad explicada por el modelo, `SSM=0.029915`, es inferior a la que queda por explicar (residuos), `SSR=0.030341` y el estadístico `F=27.606`, mayor que 1. Además, volviendo a ver el resumen del modelo

```
## F-statistic: 27.606 on 1 and 28 DF, p-value: 1.385e-05
```

tenemos que el $p - valor$ asociado con el estadístico F es inferior a 0.05.

La conclusión es que hay evidencias suficientes para poder rechazar la hipótesis nula, $H_0 : F = 1$ y por tanto, resulta posible establecer un modelo de regresión lineal para explicar el comportamiento del peso de un huevo en función de su diámetro.

8.3 Coeficiente de determinación

En el `modelo1` el valor de R^2 es `Multiple R-squared: 0.4965`, alrededor del 49.65% de la variabilidad del peso es explicada por la recta ajustada.

9 Análisis de los parámetros del modelo

El test ANOVA significativo nos dice si el modelo tiene, en general, un grado de predicción significativamente bueno para la variable resultado, pero no nos dice nada sobre la contribución individual del modelo. Para encontrar los parámetros del modelo y su significación tenemos que volver a la parte `Coefficients` en el resumen del modelo.

β_1	Estimate	Std. Error	t value	Pr(>
log(diametro)	1.2133149	0.2309251	5.2541481	1.384744e-05

Observando la tabla vemos que β_1 es la pendiente de la recta y representa el cambio en la variable dependiente (peso) asociado al cambio de una unidad en la variable predictora. Si nuestra variable predictora incrementa una unidad, nuestro modelo predice que el peso de un huevo se incrementara en 1.213 gr, pues en este caso $\beta_1 = 1.213$ Por tanto, la ecuación del modelo queda:

$$Y = 0 + 1.213 * X$$

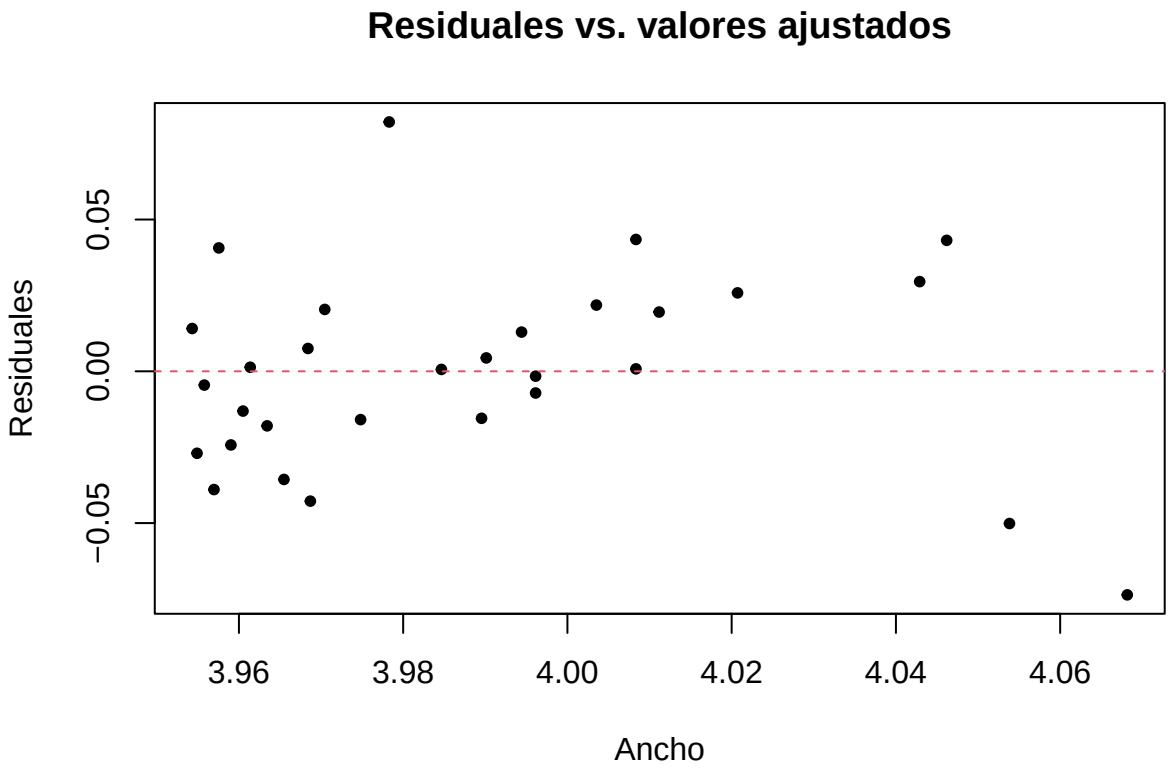
10 Diagnóstico del modelo

En este apartado hemos hecho uso tanto de *J.Faraway (2009)* como de *Sánchez (2011)* para el desarrollo del mismo.

Una vez que tenemos el modelo ajustado procedemos con su diagnóstico, que se realiza a través del **análisis de los residuos** ε_i:

- Las hipótesis de linealidad, homocedasticidad e independencia se contrastan a través de un análisis gráfico que enfrenta los valores de los residuos, ε_i, con los valores ajustados $\hat{x}_i$.
- Las hipótesis de media cero, varianza constante, incorrelación y normalidad la comprobamos analíticamente

Comenzaremos con el análisis gráfico. Los residuos deberían formar una nube de puntos sin estructura y con, aproximadamente, la misma variabilidad por todas las zonas como se muestra en el gráfico:



Continuamos ahora realizando el **diagnóstico analítico**. El primer paso es obtener los residuos, valores ajustados y estadísticos del modelo analizado para poder así estudiar si se cumplen los supuestos del mismo.

Obtención de residuos, valores ajustados y estadísticos necesarios

Para ello, añadimos los correspondientes resultados a nuestros datos a través del siguiente código:

El resultado es la creación de las siguientes variables:

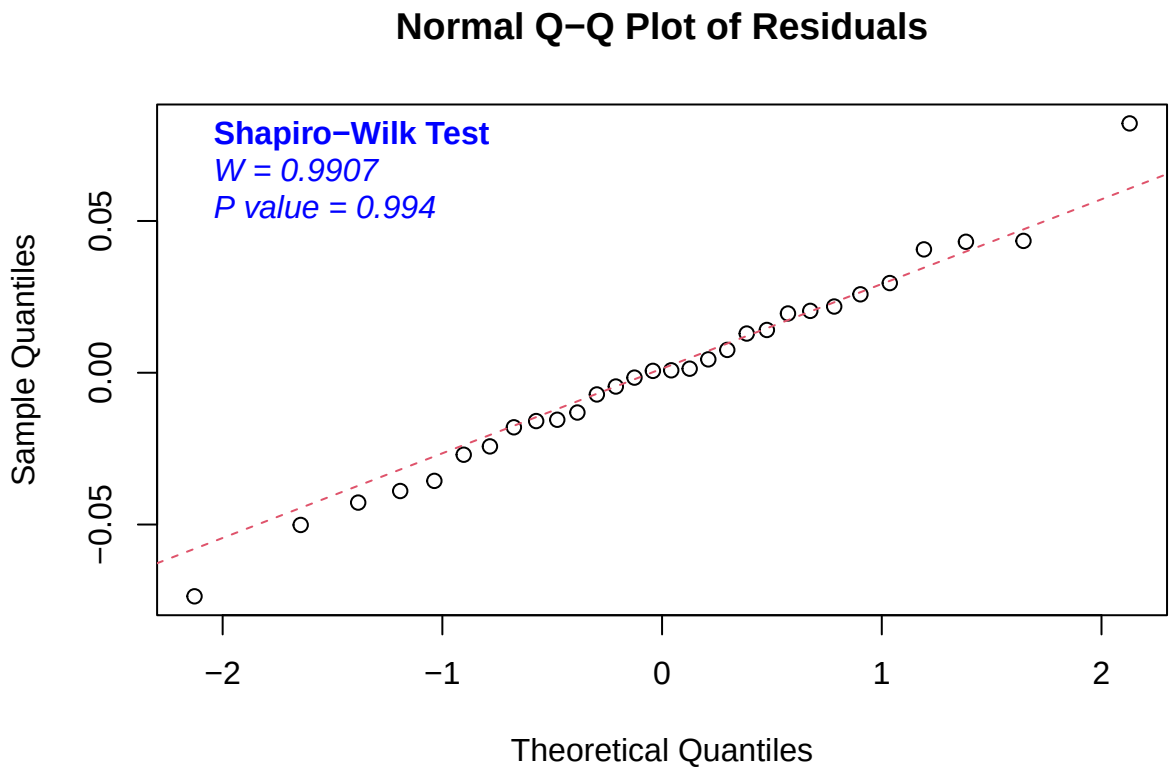
- `fitted.modelo2`: valores ajustados (valores de la variable respuesta) para las observaciones originales de la predictora.
- `residuals.modelo2`: residuos del modelo, esto es, diferencia entre valor observado de la respuesta y valor ajustado por el modelo.
- `rstudent.modelo2`: residuos estudentizados del modelo ajustado.

Vamos a utilizar todas estas variables para estudiar si nuestro modelo cumple las hipótesis.

10.1 Test de normalidad (test de Kolmogorov-Smirnov)

Empezamos el análisis con un gráfico `qqplot`, que enfrenta los valores reales a los valores que obtendríamos si la distribución fuera normal. Si los datos reales se distribuyen normalmente, estos tendrán la misma

distribución que los valores esperados y en el gráfico `qqplot` obtendremos una linea recta en la diagonal.

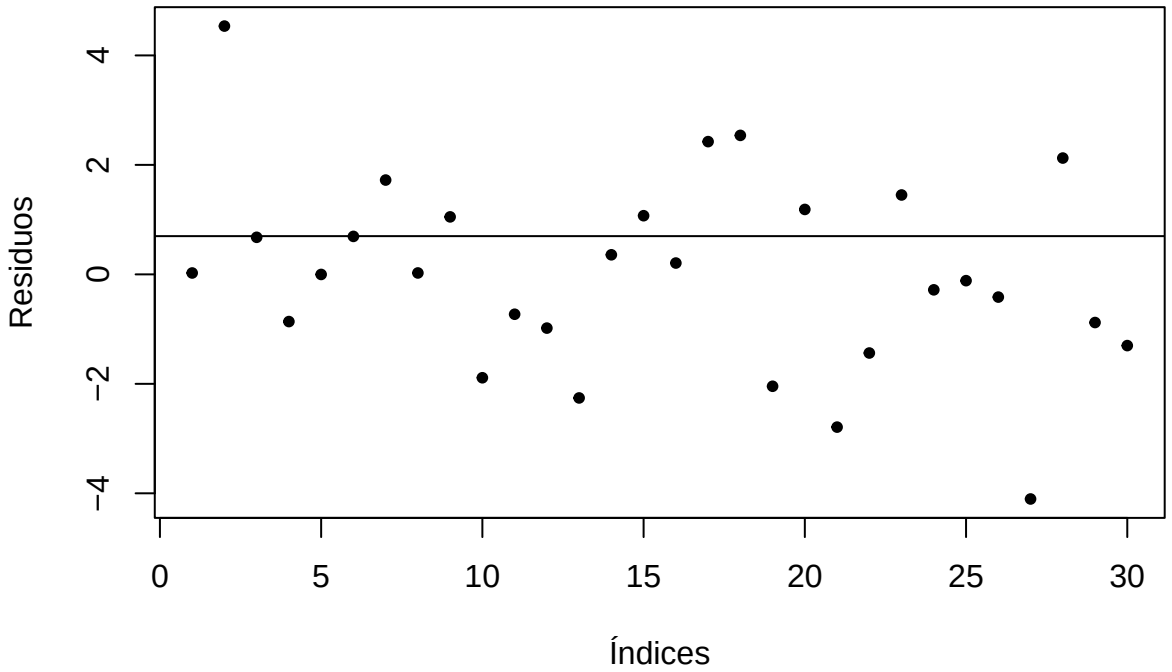


Aqui podemos ver que tambien se cumple el supuesto de normalidad, dejandonos con la siguiente pregunta:
¿ya que nuestros datos están transformados será que nuestra varianza será constante?

```
##
##  studentized Breusch-Pagan test
##
## data:  modelo2
## BP = 3.7084, df = 1, p-value = 0.05414
```

Existe homogeneidad pues la significación es mayor de 0.05, la varianza es constante a lo largo de la muestra.

10.2 Autocorrelación (test de Durbin-Watson)



Si hubiera una correlación seria, veríamos picos más largos de residuos por encima y por debajo de la línea de correlación. A menos que estos efectos sean fuertes, puede ser difícil de detectar la autocorrelación, por ello realizamos el contraste de Durbin-Watson.

```
##
##  Durbin-Watson test
##
## data:  peso ~ diametro
## DW = 2.0779, p-value = 0.7969
## alternative hypothesis: true autocorrelation is not 0
```

En el contraste de autocorrelación también aceptamos la hipótesis nula de que no existe correlación entre los residuos con un $p - valor$ superior a 0.05.

11 Predicción

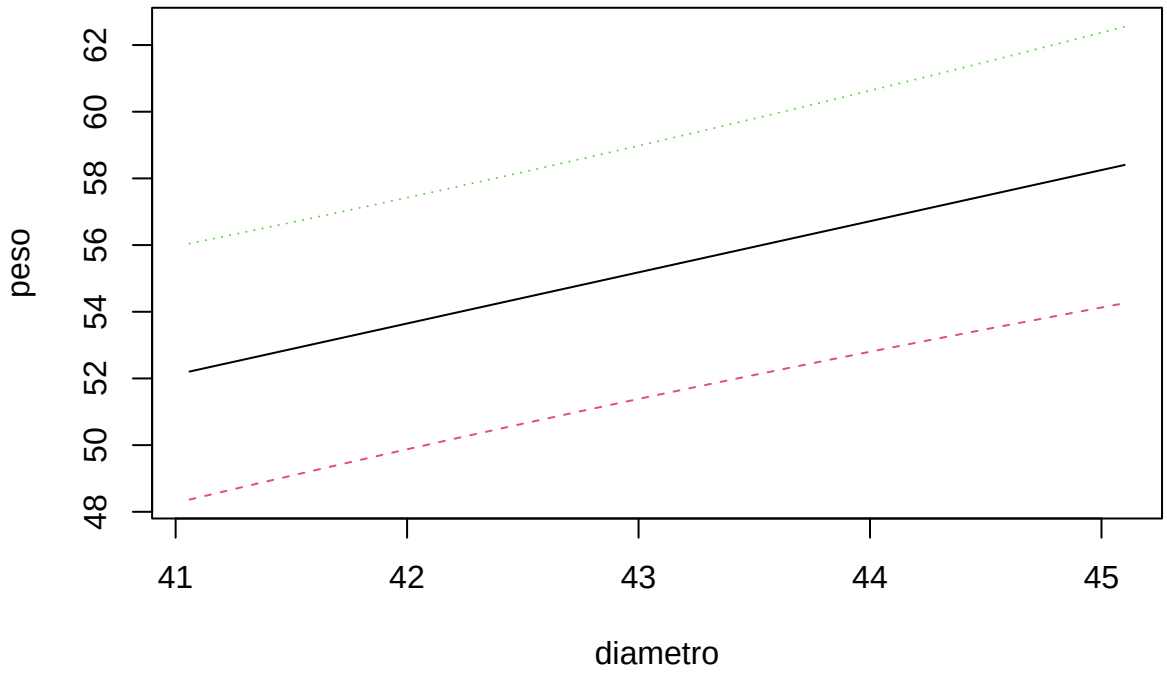
Tenemos un modelo de regresión con la capacidad de relacionar la variable predictora y la variable dependiente. Podemos utilizarlo ahora para predecir eventos futuros de la variable dependiente a través de nuevos valores de la variable predictora.

Para ello debe verificarse alguna de las siguientes condiciones:

- el valor de la predictora está dentro del rango de la variable original.
- si el valor de la predictora está fuera del rango de la original, debemos asegurar que los valores futuros mantendrán el modelo lineal propuesto.

11.1 Predicción de nuevas observaciones

Dibujamos las bandas de predicción, que reflejan la incertidumbre sobre futuras observaciones:



Como nuestros datos estan escaladaos a el `log()` vamos a destransformarlos para obtener nuestras predicciones:

aqui podemos ver en `fit` nuestro diametro ajustado y `lwr` & `upr` nuestros intervalos de predicción para estos anchos de un huevo dado.

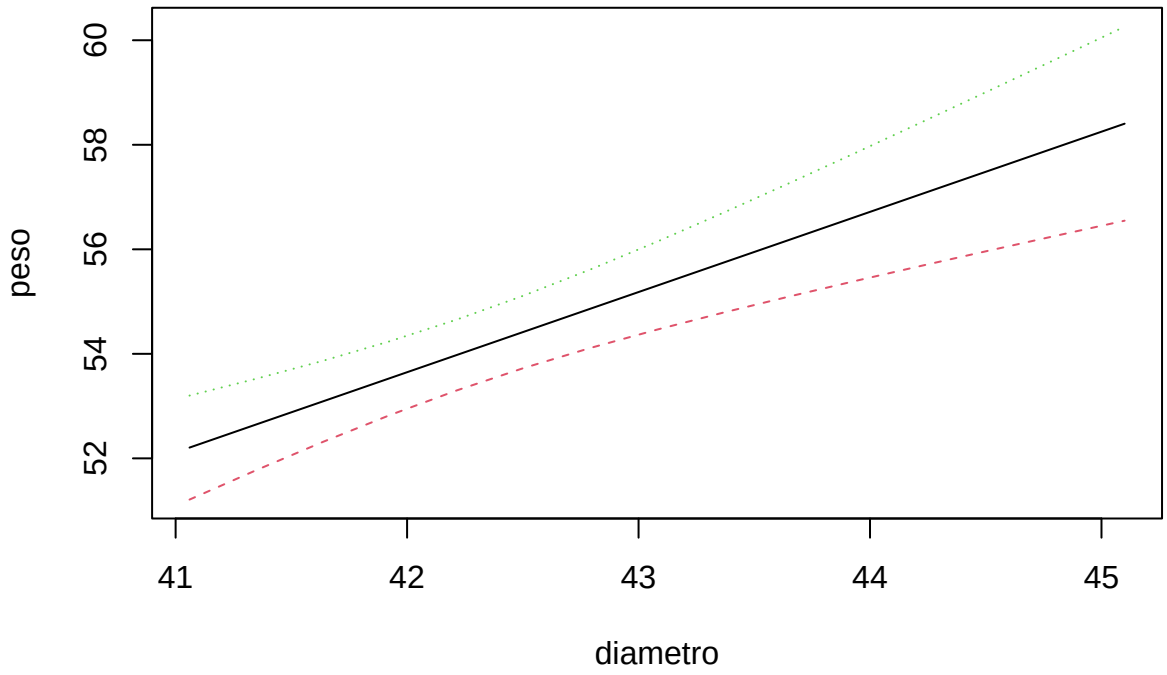
11.2 Intervalos de confianza para los predictores:

Dado un nuevo conjunto de predictores, x0, debemos evaluar la incertidumbre en esta predicción. Para tomar decisiones racionales necesitamos algo más que puntos estimados. Si la predicción tiene intervalo de confianza ancho entonces entonces los resultados estarán lejos de la estimación puntual.

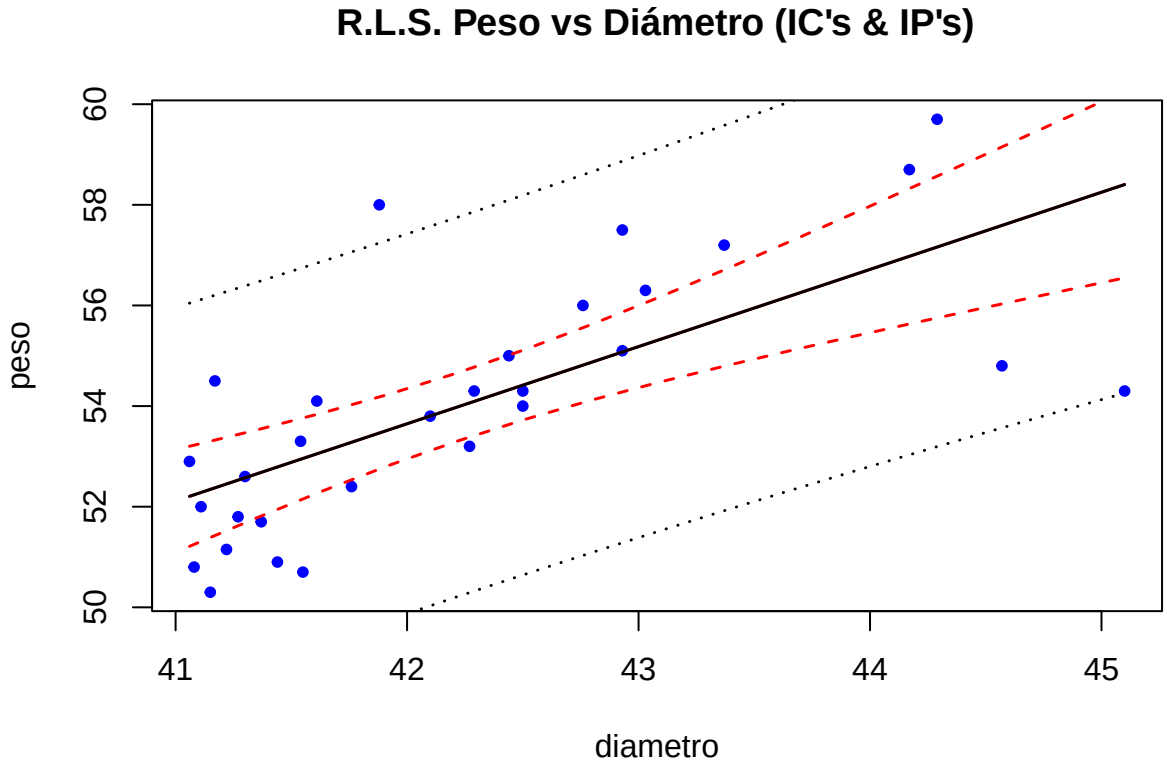
> Nota: Las bandas de confianza reflejan la incertidumbre en la línea de regresión (lo bien que la línea está calculada).

```
##        fit      lwr      upr
## 1 52.20622 51.21131 53.20113
## 2 52.64890 51.77438 53.52341
## 3 53.09157 52.31614 53.86700
## 4 53.53424 52.82755 54.24093
## 5 53.97692 53.29932 54.65451
## 6 54.41959 53.72644 55.11275
```

Dibujamos las bandas de confianza, que además reflejan la incertidumbre sobre futuras observaciones:



Por último podemos hacer un gráfico con la nube de puntos y los dos bandas, la de confianza y la de predicción *(Ferrari & Head, 2010)*.



Donde:

- Las lineas rojas son Intervalos de Confianza.
- Los puntos azules, son valores predichos.
- y las lineas punteadas negras son el Intervalo de predicción, junto con la linea del modelo quitando la

transformación del `log()`.

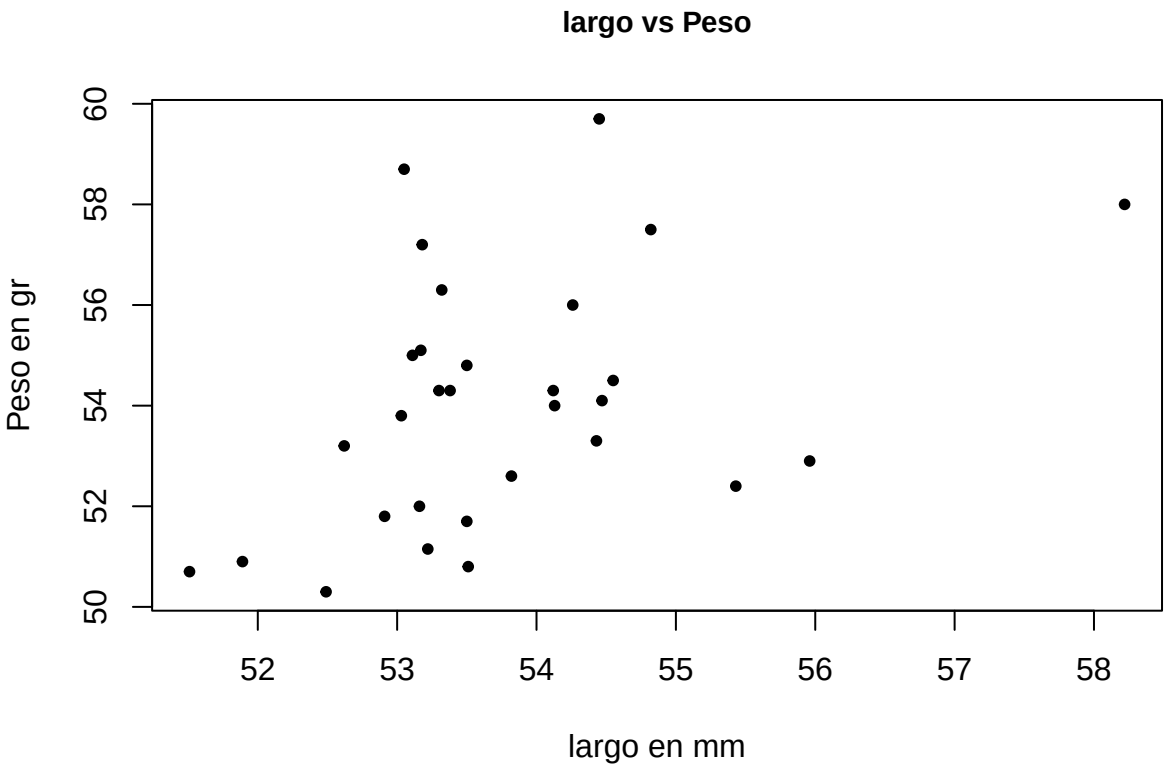
12 Modelo N°2 (Peso~Altura)

12.1 Ajuste del modelo:

Antes de observar o crear un modelo dado unos puntos, primero haremos un test de correlación de los datos para estudiar el grado de variación conjunta entre el diámetro y peso de los huevos:

```
##
##  Pearson's product-moment correlation
##
## data:  datos$largo and datos$peso
## t = 2.4424, df = 28, p-value = 0.02116
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.06926414 0.67711427
## sample estimates:
##       cor
## 0.4190759
```

Como podemos ver se rechaza H_0 ya que el $p - valor < 0.05$, es decir nuestros datos peso y el largo de los huevos tomados tienen una corelación significativa, es decir, una correlación positiva entre los datos, que vamos a ver por medio de un grafico de dispersión:



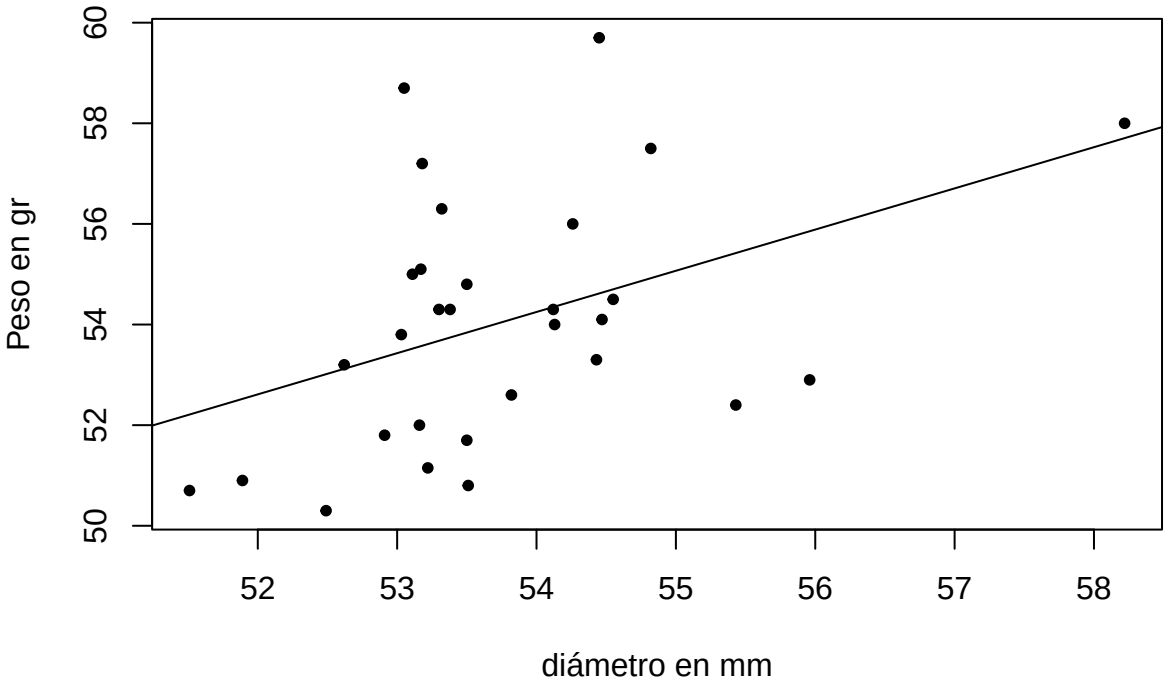
Como podemos ver los datos si tienen en algún grado una correlación positiva (mientras amuentan los valores del diámetro aumenta el peso) apriori.

Una vez visto que existe relación entre las variables pasamos a realizar el ajuste del modelo. Para ello usamos la función `lm()` que toma la forma:

```
##
```

```
## Call:
## lm(formula = peso ~ largo, data = datos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0482 -1.5604 -0.1238  1.3495  5.2285
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  10.0310    18.0260   0.556   0.5823
## largo         0.8189     0.3353   2.442   0.0212 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.297 on 28 degrees of freedom
## Multiple R-squared:  0.1756, Adjusted R-squared:  0.1462
## F-statistic: 5.965 on 1 and 28 DF,  p-value: 0.02116
```

Como podemos ver nuestro intercepto dado el $p - valor > 0.05$ decimos que el intercepto tengan valor a cero, esto es logico ya que seria extraño encintrar huevos con altura = cero y un valor de peso inicial.



En primer lugar deseamos obtener los estimadores puntuales, errores estándar y p-valores asociados con cada coeficiente

```
##               Estimate Std. Error   t value   Pr(>|t|)
## (Intercept) 10.0309793 18.0260060 0.5564726 0.58230836
## largo        0.8188604  0.3352747 2.4423569 0.02116101
```

El resultado del ajuste es:

(18.0260060) (0.3352747)

$$Peso \ = \ 0 + 0.8188604 * largo$$

donde los valores entre paréntesis indican los errores estándar de cada coeficiente. Además, puesto que los

p-valores son mayores y menores a 0.05, podemos concluir que:

1. En este caso no tiene sentido analizar el valor de la constante para largo = 0, ya que pertenecería a un supuesto donde el huevo (imaginariamente) exista, de ahí que el peso del huevo para largo = 0 sea de 0, menor que cualquiera de los datos de nuestro conjunto, en conclusión un huevo con largo = 0 no existe y más si su peso = 0.
2. Existen evidencias estadísticas suficientes para considerar que hay una relación lineal entre largo y peso. Dicha relación es positiva cuando aumenta el largo de un huevo dado aumente el peso del mismo. Además vemos que por cada *mm* que aumenta el diámetro de un huevo, aumenta el peso en 0.82 *gramos*.
3. El error estándar residual estimado (s) es de 4.92. Este valor es muy importante, es un medidor de la calidad (precisión) del modelo. Además nos vamos a basar en él para calcular los intervalos de confianza para el coeficiente del modelo.

12.2 Intervalos de Confianza

Obtenemos los correspondientes intervalos de confianza para el parámetro m de nuestras modelo = $Y = 0 + m * X$ con nivel significación al 95%

Parámetro	2.5 %	97.5 %
diametro	0.13208	1.505639

Interpretamos los intervalos: con una probabilidad del 95%, el efecto asociado con diametro se encuentra en el intervalo $(0.1320814, 1.505639)$.

12.3 Tabla ANOVA

Obtenemos la correspondiente tabla ANOVA donde vemos la descomposición de la variabilidad del modelo

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
largo	1	31.48277	31.482768	5.965107	0.021161
Residuals	28	147.77898	5.277821	NA	NA

Observamos que la variabilidad explicada por el modelo, `SSM=31.4828`, es inferior a la que queda por explicar (residuos), `SSR=5.2778` y el estadístico `F=5.9651`, mayor que 1. Además, volviendo a ver el resumen del modelo

```
## F-statistic: 5.9651 on 1 and 28 DF, p-value: 0.02116
```

tenemos que el $p - valor$ asociado con el estadístico F es inferior a 0.05.

La conclusión es que hay evidencias suficientes para poder rechazar la hipótesis nula, $H_0 : F = 1$ y por tanto, resulta posible establecer un modelo de regresión lineal para explicar el comportamiento del peso de un huevo en función de su diámetro.

12.4 Coeficiente de determinación

En el `modelo3` el valor de R^2 es `Multiple R-squared: 0.1756`, alrededor del 17.56% de la variabilidad del peso es explicada por la recta ajustada.

13 Análisis de los parámetros del modelo

El test ANOVA significativo nos dice si el modelo tiene, en general, un grado de predicción significativamente bueno para la variable resultado, pero no nos dice nada sobre la contribución individual

del modelo. Para encontrar los parámetros del modelo y su significación tenemos que volver a la parte `Coefficients` en el resumen del modelo.

β_1	Estimate	Std. Error	t value	Pr(>
largo	0.8188604	0.3352747	2.4423569	0.02116101

Observando la tabla vemos que β_1 es la pendiente de la recta y representa el cambio en la variable dependiente (peso) asociado al cambio de una unidad en la variable predictora. Si nuestra variable predictora incrementa una unidad, nuestro modelo predice que el peso de un huevo se incrementara en 0.82 gr, pues en este caso $\beta_1 = 0.82$ Por tanto, la ecuación del modelo queda:

$$Y = 0 + 0.82 * X$$

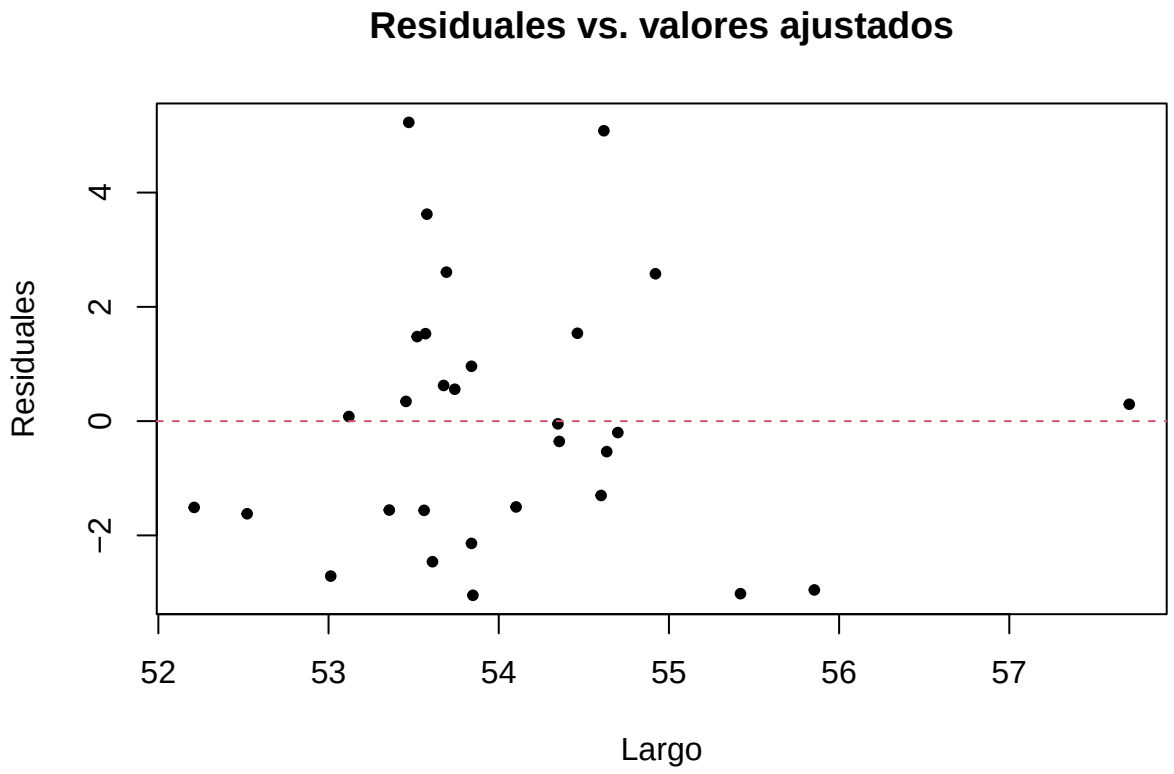
14 Diagnóstico del modelo

En este apartado hemos hecho uso tanto de J.Faraway (2009) como de Sánchez (2011) para el desarrollo del mismo.

Una vez que tenemos el modelo ajustado procedemos con su diagnóstico, que se realiza a través del **análisis de los residuos** ε_i:

- Las hipótesis de linealidad, homocedasticidad e independencia se contrastan a través de un análisis gráfico que enfrenta los valores de los residuos, ε_i, con los valores ajustados $\hat{x}_i$.
- Las hipótesis de media cero, varianza constante, incorrelación y normalidad la comprobamos analíticamente

Comenzaremos con el análisis gráfico. Los residuos deberían formar una nube de puntos sin estructura, para esto evaluaremos el siguiente grafico:



Continuamos ahora realizando el **diagnóstico analítico**. El primer paso es obtener los residuos, valores ajustados y estadísticos del modelo analizado para poder así estudiar si se cumplen los supuestos del mismo.

Obtención de residuos, valores ajustados y estadísticos necesarios

Para ello, añadimos los correspondientes resultados a nuestros datos a través del siguiente código:

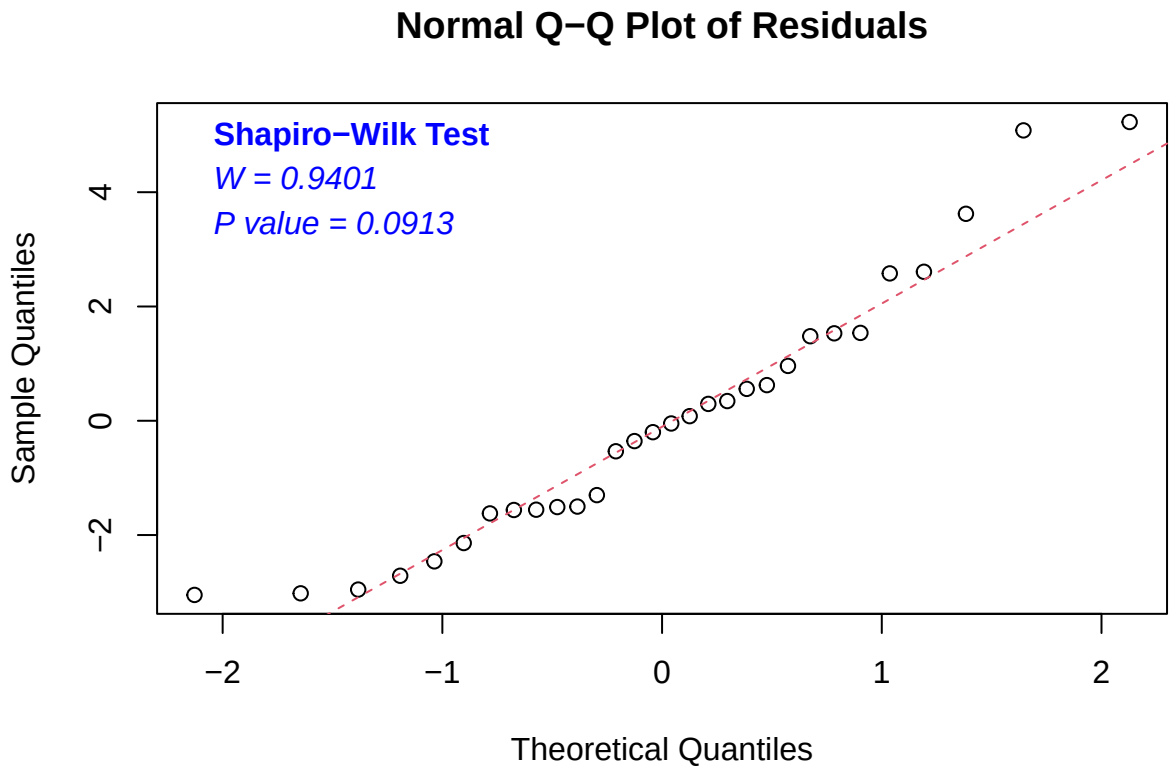
El resultado es la creación de las siguientes variables:

- `fitted.modelo3`: valores ajustados (valores de la variable respuesta) para las observaciones originales de la predictora.
- `residuals.modelo3`: residuos del modelo, esto es, diferencia entre valor observado de la respuesta y valor ajustado por el modelo.
- `rstudent.modelo3`: residuos estudentizados del modelo ajustado.

Vamos a utilizar todas estas variables para estudiar si nuestro modelo cumple las hipótesis.

14.1 Test de normalidad (test de Kolmogorov-Smirnov)

Empezamos el análisis con un gráfico `qqplot`, que enfrenta los valores reales a los valores que obtendríamos si la distribución fuera normal. Si los datos reales se distribuyen normalmente, estos tendrán la misma distribución que los valores esperados y en el gráfico `qqplot` obtendremos una linea recta en la diagonal.



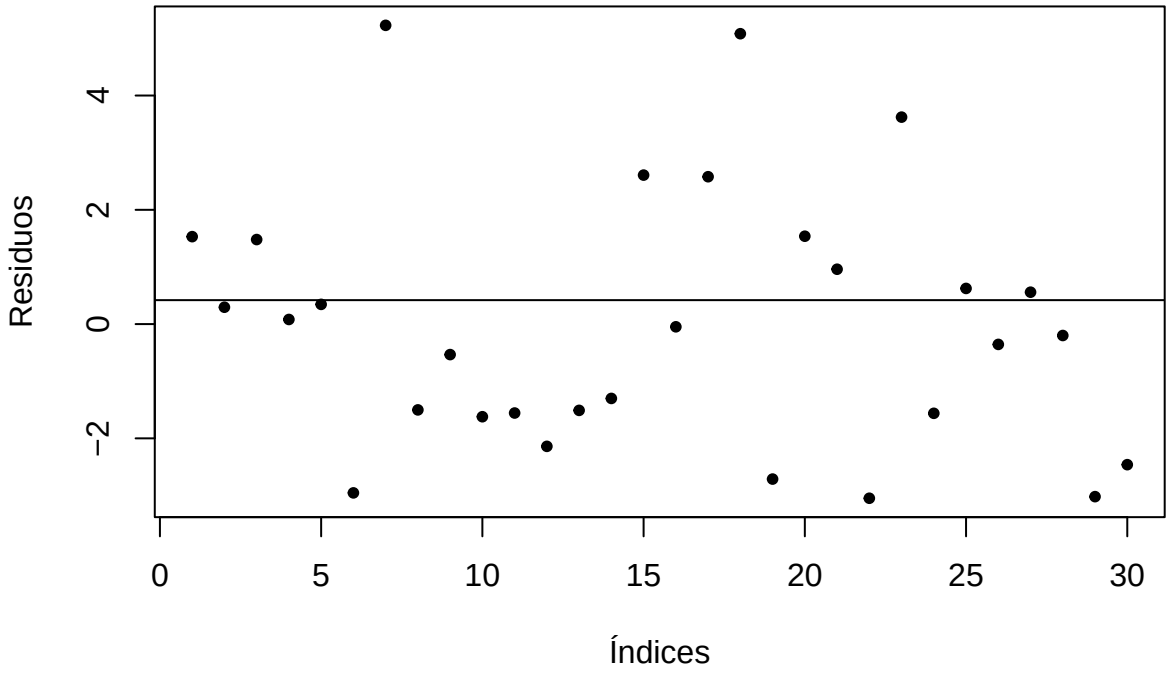
Podemos ver que nuestro $p - valor > 0.05$ por lo que no se rechaza la hipotesís nula donde los datos se distribuyen normal. además por medio de la grafica nos muestra como nuestros puntos se acomodan bien a la recta.

```
##
##  studentized Breusch-Pagan test
##
## data:  modelo3
## BP = 0.00023802, df = 1, p-value = 0.9877
```

Existe homogeneidad pues la significación es mayor de 0.05, la varianza es constante a lo largo de la muestra.

14.2 Autocorrelación (test de Durbin-Watson)

Hemos asumido que los residuos son incorrelados, vamos a comprobarlo.



Si hubiera una correlación seria, veríamos picos más largos de residuos por encima y por debajo de la línea de correlación. A menos que estos efectos sean fuertes, puede ser difícil de detectar la autocorrelación, por ello realizamos el *contraste de Durbin-Watson*.

```
##
##  Durbin-Watson test
##
## data:  peso ~ largo
## DW = 2.3583, p-value = 0.3051
## alternative hypothesis: true autocorrelation is not 0
```

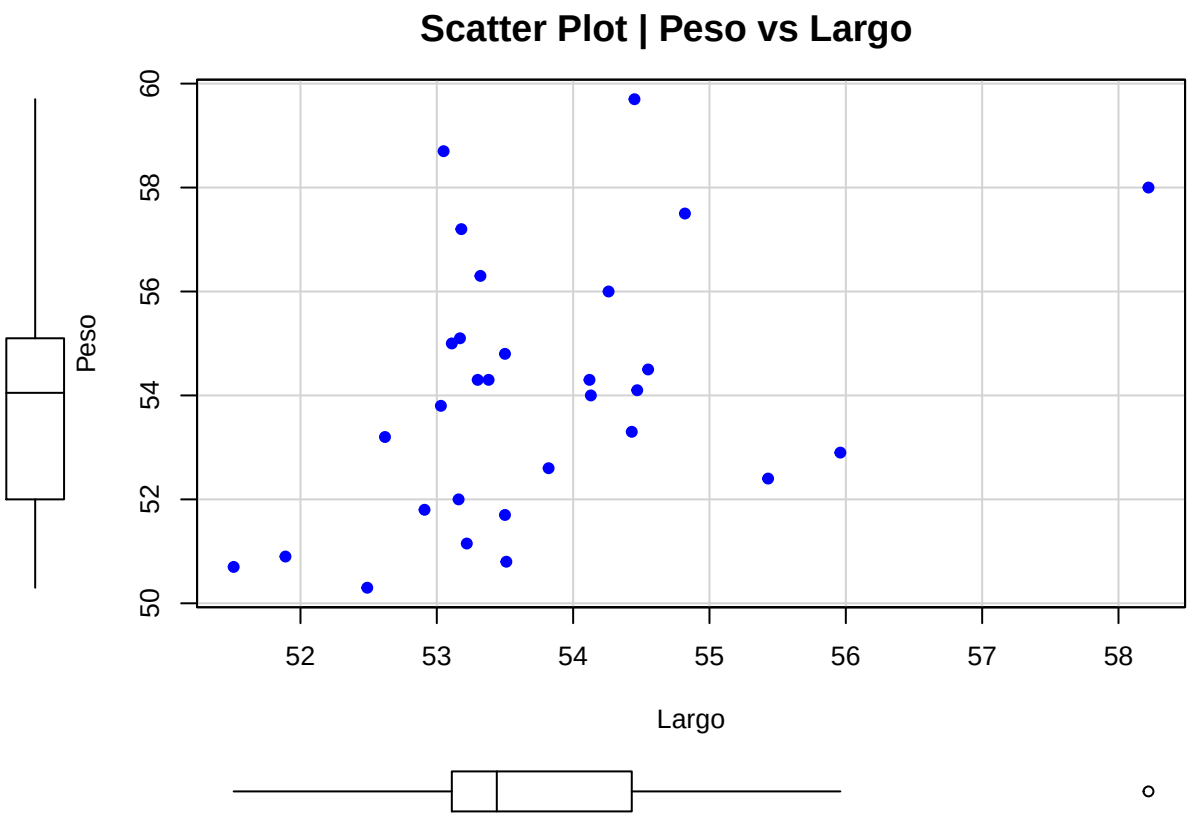
En el contraste de autocorrelación también aceptamos la hipótesis nula de que no existe correlación entre los residuos con un $p - valor$ superior a 0.05

14.3 Valores atípicos:

Para observar si hay datos atípicos que hacen que nuestro modelo no sea el más optimo. Vamos a realizar un test de valores atípicos (Bonferroni).

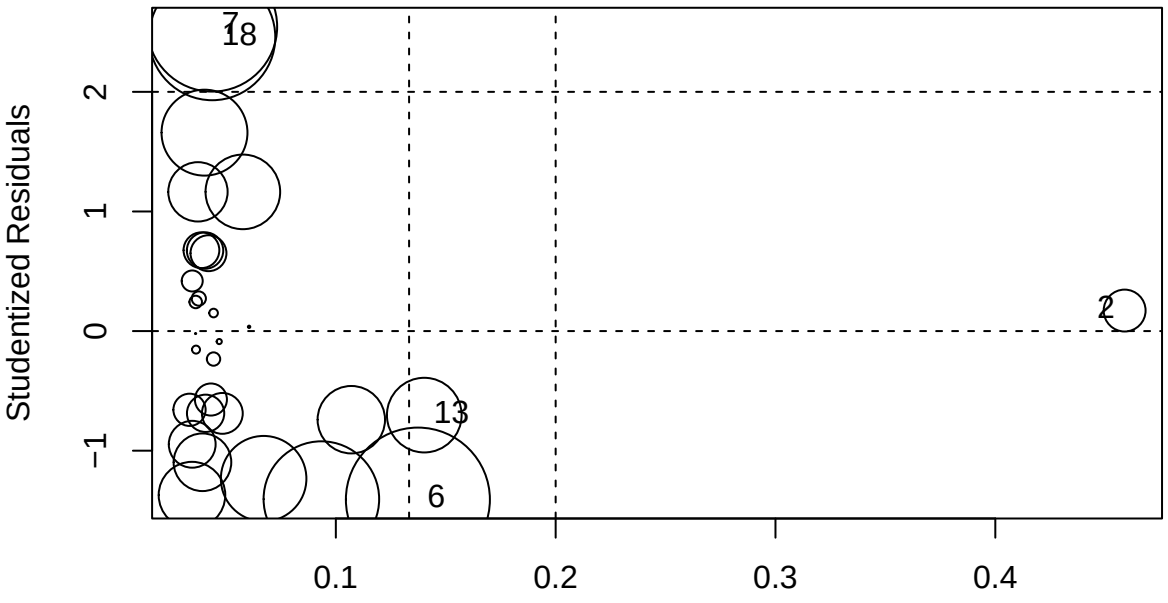
En el caso de observar valores atípicos los pasos a seguir son:

- Descartar que sea un error.
- Analizar si es un caso influyente.
- En caso de ser influyente calcular las rectas de regresión incluyéndolo y excluyéndolo, y elegir la que mejor se adapte al problema y a las observaciones futuras.



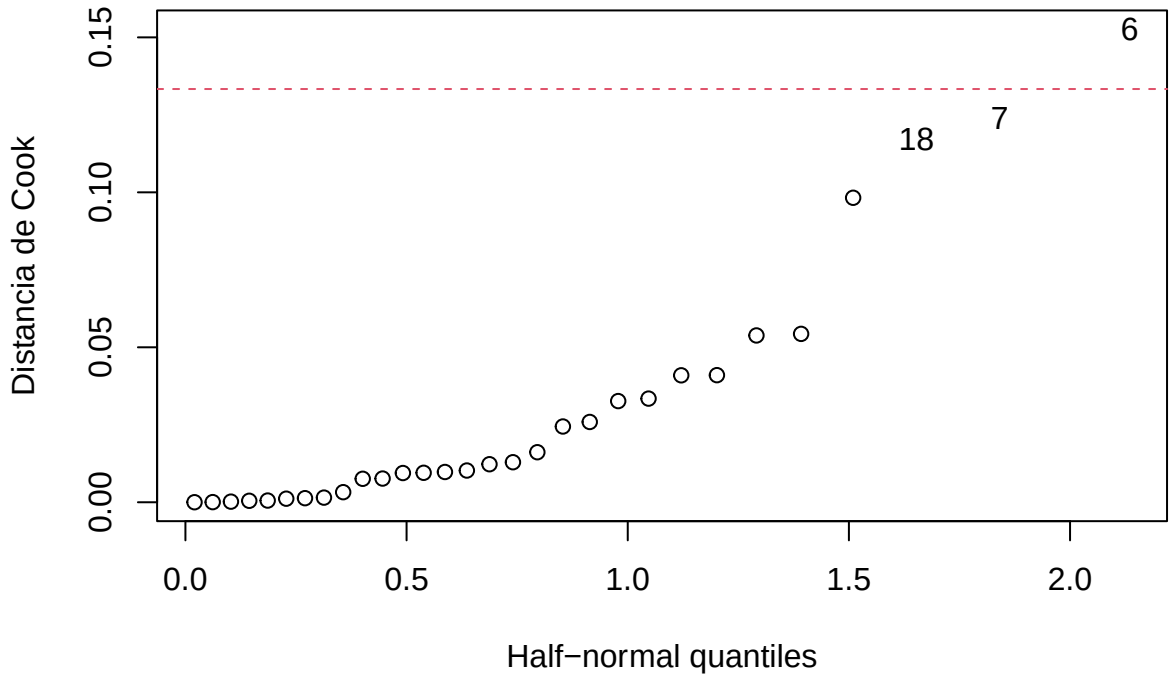
En la variable `largo` vemos que la mediana no está centrada en la media y contiene un valor atípico en la muestra, los datos no son uniformes. Con la variable `peso` ocurre casí mismo a excepción de ese dato atípico.

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##   rstudent unadjusted p-value Bonferroni p
## 7 2.544821           0.016962      0.50886
```



	StudRes	Hat	CookD
2	0.1714820	0.4588309	0.0129136
6	-1.4087161	0.1373254	0.1525856
7	2.5448208	0.0437795	0.1239995
13	-0.7026427	0.1402320	0.0410043
18	2.4573114	0.0437596	0.1170944

El gráfico nos indican que la observación número **6** es un valor influyente dado que $D_i > \frac{4}{n}$ donde D_i es la distancica de cook y n es el tamaño de la muestra. Las observaciones **6**,**13** y **18** que vemos en el gráfico son medidas influyentes pero solamente se observa en el test que no hay datos atipicos pero por el $|x|$ de los residuales estandarizados se sabe que la observación **7** podria serlo, con un grafico de las distancias de cook miraremos si se cumple lo anterior.



La linea roja representa el corte donde se puede observar si un dato influye o no dentro del modelo, pero con el grafico de influencia nos muestra que la observación #6 no esta alejada de los datos, por lo cual concluimos que no hay datos extraños o atípicos que afecten el modelo.

15 Predicción

Tenemos un modelo de regresión con la capacidad de relacionar la variable predictora y la variable dependiente. Podemos utilizarlo ahora para predecir eventos futuros de la variable dependiente a través de nuevos valores de la variable predictora.

Para ello debe verificarse alguna de las siguientes condiciones:

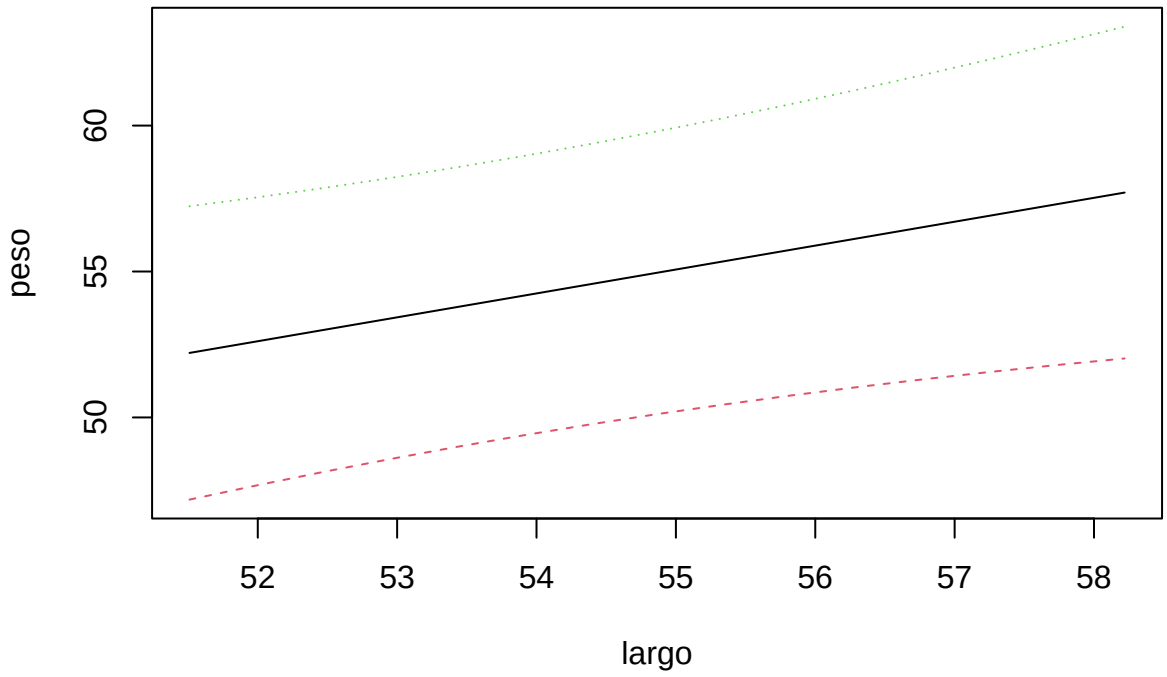
1. el valor de la predictora está dentro del rango de la variable original.
2. si el valor de la predictora está fuera del rango de la original, debemos asegurar que los valores futuros mantendrán el modelo lineal propuesto.

15.1 Predicción de nuevas observaciones

```
##        fit      lwr      upr
## 1 52.21048 47.18543 57.23553
## 2 52.60295 47.66873 57.53717
## 3 52.99542 48.13140 57.85943
## 4 53.38788 48.57254 58.20323
## 5 53.78035 48.99151 58.56920
## 6 54.17282 49.38792 58.95772
```

aqui podemos ver según un largo dado en mm de un huevo caul sera su intervalod e predicción de su peso en las dos columnas lower & upper.

Dibujamos las bandas de predicción, que reflejan la incertidumbre sobre futuras observaciones:

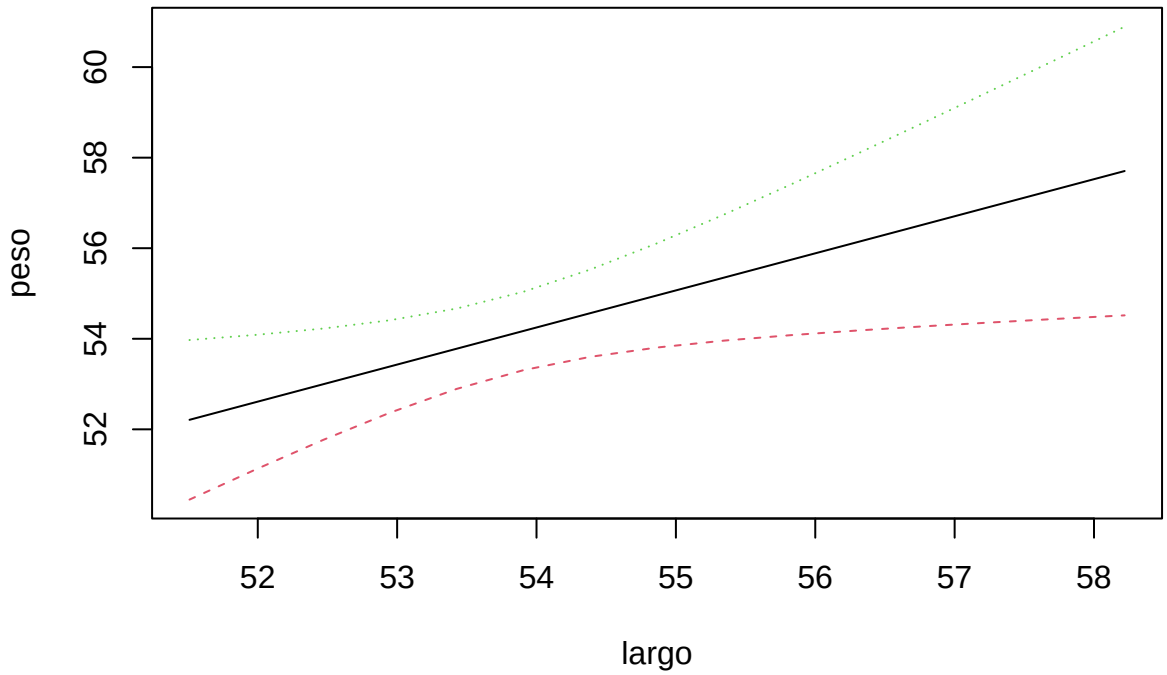


15.2 Intervalos de confianza para los predictores

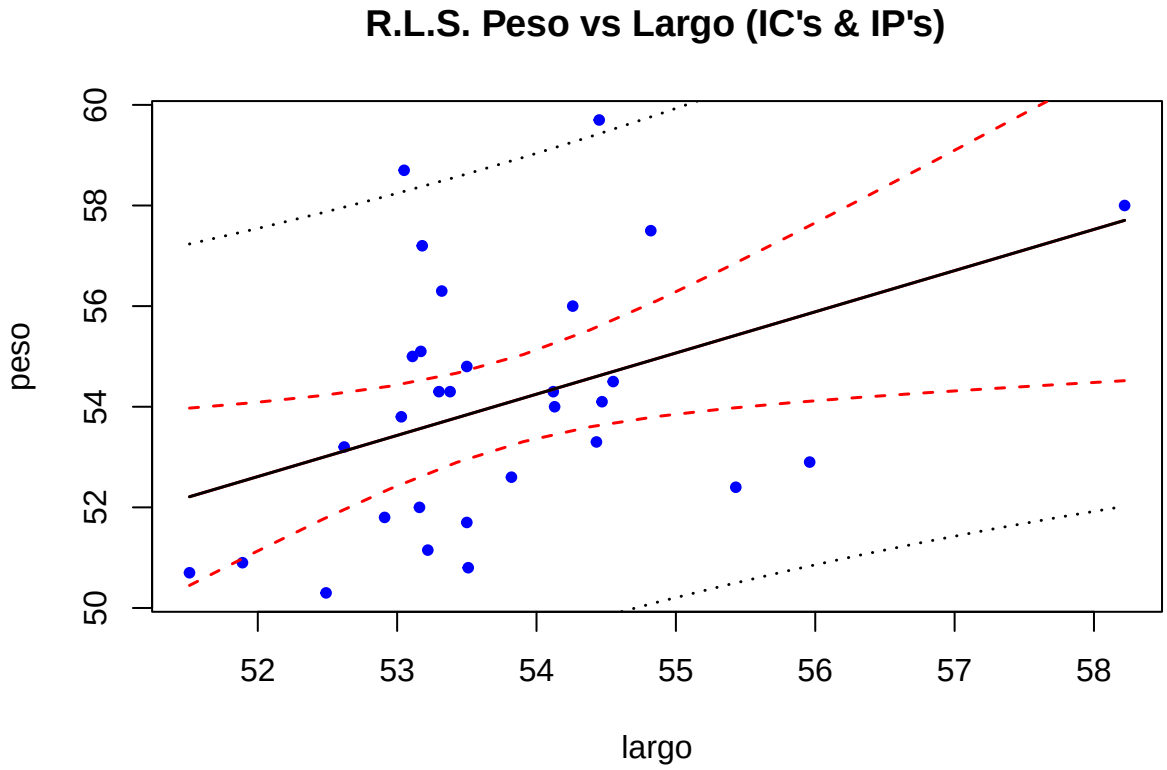
Además nosotros podemos tener un IC para nuestras `x` en este caso el largo, es decir, cuanto puede variar el largo de un huevo y según eso dar una estimación de su peso.

```
##        fit      lwr      upr
## 1 52.21048 50.44823 53.97273
## 2 52.60295 51.11939 54.08651
## 3 52.99542 51.76534 54.22549
## 4 53.38788 52.36714 54.40863
## 5 53.78035 52.89297 54.66774
## 6 54.17282 53.30698 55.03866
```

Dibujamos las bandas de confianza, que además reflejan la incertidumbre sobre futuras observaciones:



Por último podemos hacer un gráfico con la nube de puntos y los dos bandas, la de confianza y la de predicción (Ferrari & Head, 2010).



16 Apéndice

16.1 Lista de figuras

Figura:

16.2 Codigo:

```
## ----message=FALSE, warning=FALSE, include=FALSE-----------------------
library(readr)
library(tidyverse)
library(kableExtra)
library(magrittr)
library(ggExtra)
library(GGally)
library(janitor)
library(tidystats)
library(car)
library(faraway)
library(lmtest)
library(graphics)
datos <- read_delim("eggs.csv", delim = ";") %>% clean_names()
names(datos)[3] <- "diametro"

myQQnorm <- function(modelo, student = F, ...){
  if(student){
    res <- rstandard(modelo)
    lab.plot <- "Normal Q-Q Plot of Studentized Residuals"
  } else {
    res <- residuals(modelo)
    lab.plot <- "Normal Q-Q Plot of Residuals"
  }
  shapiro <- shapiro.test(res)
  shapvalue <- ifelse(shapiro$p.value < 0.001, "P value < 0.001",
  paste("P value = ", round(shapiro$p.value, 4), sep = ""))
  shapstat <- paste("W = ", round(shapiro$statistic, 4), sep = "")
  q <- qqnorm(res, plot.it = FALSE)
  qqnorm(res, main = lab.plot, ...)
  qqline(res, lty = 2, col = 2)
  text(min(q$x, na.rm = TRUE), max(q$y, na.rm = TRUE)*0.95,
  pos = 4, 'Shapiro-Wilk Test', col = "blue", font = 2)
  text(min(q$x, na.rm = TRUE), max(q$y, na.rm = TRUE)*0.80,
  pos = 4, shapstat, col = "blue", font = 3)
  text(min(q$x, na.rm = TRUE), max(q$y, na.rm = TRUE)*0.65,
  pos = 4, shapvalue, col = "blue", font = 3)
}


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
cor.test(datos$diametro, datos$peso)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
plot(datos$diametro, datos$peso, xlab = "diámetro en mm",
     ylab = "Peso en gr", main = "diámetro vs Peso",
     cex.main = 0.95, pch=20)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
modelo1 <- lm(peso~diametro, data=datos)
```

```
summary(modelo1)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
plot(datos$diametro, datos$peso, xlab = "diámetro en mm",
     ylab = "Peso en gr", pch=20)
abline(modelo1)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
summary(modelo1)$coefficients


## ----message=FALSE, warning=FALSE, include=FALSE-----------------------
MSR.ancho <- mean(summary(modelo1)$residuals^2)


## ----message=FALSE, warning=FALSE, include=FALSE-----------------------
confint(modelo1, level = 0.95)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
anova(modelo1)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
plot(fitted(modelo1), residuals(modelo1), xlab = "Ancho",
ylab = "Residuales", main = "Residuales vs. valores ajustados",pch=20)
abline(h = 0, lty = 2, col = 2)


## ----message=FALSE, warning=FALSE, include=FALSE-----------------------
datos$fitted.modelo1 <- fitted(modelo1)
datos$residuals.modelo1 <- residuals(modelo1)
datos$rstudent.modelo1 <- rstudent(modelo1)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
modelo1 %>% myQQnorm()


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
bptest(modelo1)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
plot(log(datos$diametro), log(datos$peso), xlab = "diámetro",
     ylab = "Peso", main = "diámetro vs Peso (Escala Log)",
     cex.main = 0.95, pch=20)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
modelo2 <- lm(log(peso)~log(diametro), data=datos)
summary(modelo2)
```

```
## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
plot(log(datos$diametro), log(datos$peso), xlab = "diámetro en mm",
     ylab = "Peso en gr", pch=20)
abline(modelo2)


## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
summary(modelo2)$coefficients


## ----message=FALSE, warning=FALSE, include=FALSE----------------------
MSR.log.ancho <- mean(summary(modelo2)$residuals^2)


## ----message=FALSE, warning=FALSE, include=FALSE----------------------
confint(modelo2, level = 0.95)


## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
anova(modelo2)


## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
plot(fitted(modelo2), residuals(modelo2), xlab = "Ancho",
ylab = "Residuales", main = "Residuales vs. valores ajustados",pch=20)
abline(h = 0, lty = 2, col = 2)


## ----message=FALSE, warning=FALSE, include=FALSE----------------------
datos$fitted.modelo2 <- fitted(modelo2)
datos$residuals.modelo2 <- residuals(modelo2)
datos$rstudent.modelo2 <- rstudent(modelo2)


## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
modelo2 %>% myQQnorm()


## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
bptest(modelo2)


## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
plot(datos$residuals.modelo1, pch = 20,
     ylab = "Residuos", xlab = "Índices")
abline(h = cor(datos$peso, datos$diametro))


## ---------------------------------------------------------------------
dwtest(peso~diametro, alternative = "two.sided", data = datos)
```

```
## ----message=FALSE, warning=FALSE, include=FALSE-----------------------
x0 <- seq(min(datos$diametro), max(datos$diametro), length = 15)
dfp <- data.frame(diametro = x0)
pred.ip <- predict(modelo1, dfp, interval = "prediction", se.fit = TRUE, data = datos)
head(pred.ip$fit)


## ----------------------------------------------------------------------
matplot(x0, pred.ip$fit, type = "l", xlab = "diametro", ylab = "peso")


## ----message=FALSE, warning=FALSE, include=FALSE-----------------------
x0 <- seq(min(datos$diametro), max(datos$diametro), length = 15)
dfp <- data.frame(diametro = x0)
pred.ip <- predict(modelo2, dfp, interval = "prediction",
se.fit = TRUE, data = datos)
pred.ip1 <- predict(modelo1, dfp, interval = "prediction",
se.fit = TRUE, data = datos)
head(pred.ip$fit)
newpred <- exp(pred.ip$fit)
head(newpred)


## ----------------------------------------------------------------------
pred.ic <- predict(modelo1, dfp, interval = "confidence",
se.fit = TRUE, data = datos)
head(pred.ic$fit)


## ----------------------------------------------------------------------
matplot(x0, pred.ic$fit, type = "l", xlab = "diametro", ylab = "peso")


## ----------------------------------------------------------------------
plot(datos$diametro, datos$peso, pch = 20, xlab = "diametro",
ylab = "peso", col="blue")

# Añadimos las bandas
matlines(dfp$diametro, pred.ic$fit, lty = c(1, 2, 2),
         lwd = 1.5, col = "red")

matlines(dfp$diametro, pred.ip1$fit, lty = c(1, 3, 3),
         lwd = 1.5, col= "black")
title(main= "R.L.S. Peso vs Diámetro (IC's & IP's)")


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
cor.test(datos$largo, datos$peso)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
plot(datos$largo, datos$peso, xlab = "largo en mm",
     ylab = "Peso en gr", main = "largo vs Peso",
     cex.main = 0.95, pch=20)
```

```
## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
modelo3 <- lm(peso~largo, data=datos)
summary(modelo3)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
plot(datos$largo, datos$peso, xlab = "diámetro en mm",
     ylab = "Peso en gr", pch=20)
abline(modelo3)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
summary(modelo3)$coefficients


## ----message=FALSE, warning=FALSE, include=FALSE-----------------------
MSR.largo <- mean(summary(modelo3)$residuals^2)


## ----message=FALSE, warning=FALSE, include=FALSE-----------------------
confint(modelo3, level = 0.95)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
anova(modelo3)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
plot(fitted(modelo3), residuals(modelo3), xlab = "Largo",
ylab = "Residuales", main = "Residuales vs. valores ajustados",pch=20)
abline(h = 0, lty = 2, col = 2)


## ----message=FALSE, warning=FALSE, include=FALSE-----------------------
datos$fitted.modelo3 <- fitted(modelo3)
datos$residuals.modelo3 <- residuals(modelo3)
datos$rstudent.modelo3 <- rstudent(modelo3)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
modelo3 %>% myQQnorm()


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
bptest(modelo3)


## ----echo=FALSE, message=FALSE, warning=FALSE--------------------------
plot(datos$residuals.modelo3, pch = 20, ylab = "Residuos", xlab = "Índices")
abline(h = cor(datos$peso , datos$largo))
```

```
## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
dwtest(peso~largo, alternative = "two.sided", data = datos)


## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
scatterplot(peso~largo,data = datos,smooth = F, pch=19,
            regLine = F, xlab = "Largo", ylab = "Peso")
title(main = "Scatter Plot | Peso vs Largo")


## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
outlierTest(modelo3, cutoff = 0.05, n.max = 10, order = TRUE)
influencePlot(modelo3, id.n = 2)


## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
cook3 <- cooks.distance(modelo3)
labels3 <- rownames(datos)
halfnorm(cook, 3, labs = labels, ylab = "Distancia de Cook")
abline(h=4/30, lty = 2, col = 2)


## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
x0.l <- seq(min(datos$largo), max(datos$largo), length = 15)
dfp.l <- data.frame(largo = x0.l)
pred.ip.l <- predict(modelo3, dfp.l, interval = "prediction",
                     se.fit =TRUE, data = datos)
head(pred.ip.l$fit)


## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
matplot(x0.l, pred.ip.l$fit, type = "l", xlab = "largo", ylab = "peso")


## ----echo=FALSE, message=FALSE, warning=FALSE-------------------------
pred.ic.l <- predict(modelo3, dfp.l, interval = "confidence",
se.fit = TRUE, data = datos)
head(pred.ic.l$fit)


## ---------------------------------------------------------------------
matplot(x0.l, pred.ic.l$fit, type = "l", xlab = "largo", ylab = "peso")


## ---------------------------------------------------------------------
plot(datos$largo, datos$peso, pch = 20, xlab = "largo",
ylab = "peso", col="blue")

# Añadimos las bandas
matlines(dfp.l$largo, pred.ic.l$fit, lty = c(1, 2, 2),
         lwd = 1.5, col = "red")

matlines(dfp.l$largo, pred.ip.l$fit, lty = c(1, 3, 3),
         lwd = 1.5, col= "black")
```

```
title(main= "R.L.S. Peso vs Largo (IC's & IP's)")
```