T3.Trees

Juan Manuel Cabrera

2023-08-05

Librerias

```
library(GGally)
library(ggplot2)
library(MASS)
library(rsq)
library(ppcor)
library(relaimpo)
library(car)
```

Datasets

```
data(trees)
attach(trees)
```

0. Objetivo

El objetivo del ejercicio es buscar un modelado que estime el volumen (Volume) de un árbol a partir de su circunferencia (Girth) y de la altura del árbol (Height).

Para entrenar el modelo vamos a tener el datasets trees que describiremos a continuación.

1. Análisis exploratorio

En este apartado se analizarán la relación entre las tres variables que componen el dataset (Girth, Height, Volume)

1.1. Dimensión del dataset

```
dim(trees)
```

```
## [1] 31  3
```

La dimensión del dataset es de 31 observaciones y 3 características.

1.2. Tipos de datos

Se comienza analizando el tipo de datos

```
str(trees)
```

```
## 'data.frame':    31 obs. of  3 variables:
##  $ Girth : num  8.3 8.6 8.8 10.5 10.7 10.8 11 11 11.1 11.2 ...
##  $ Height: num  70 65 63 72 81 83 66 75 80 75 ...
##  $ Volume: num  10.3 10.3 10.2 16.4 18.8 19.7 15.6 18.2 22.6 19.9 ...
```

Se observa que todas las variables son numéricas (doubles) por lo que no hay que realizar ninguna transformación.

1.3. Se comprueba si falta algún valor (NA)

Se va a contar el número de NA existentes en el dataframe

```
sum(is.na(trees))
```

```
## [1] 0
```

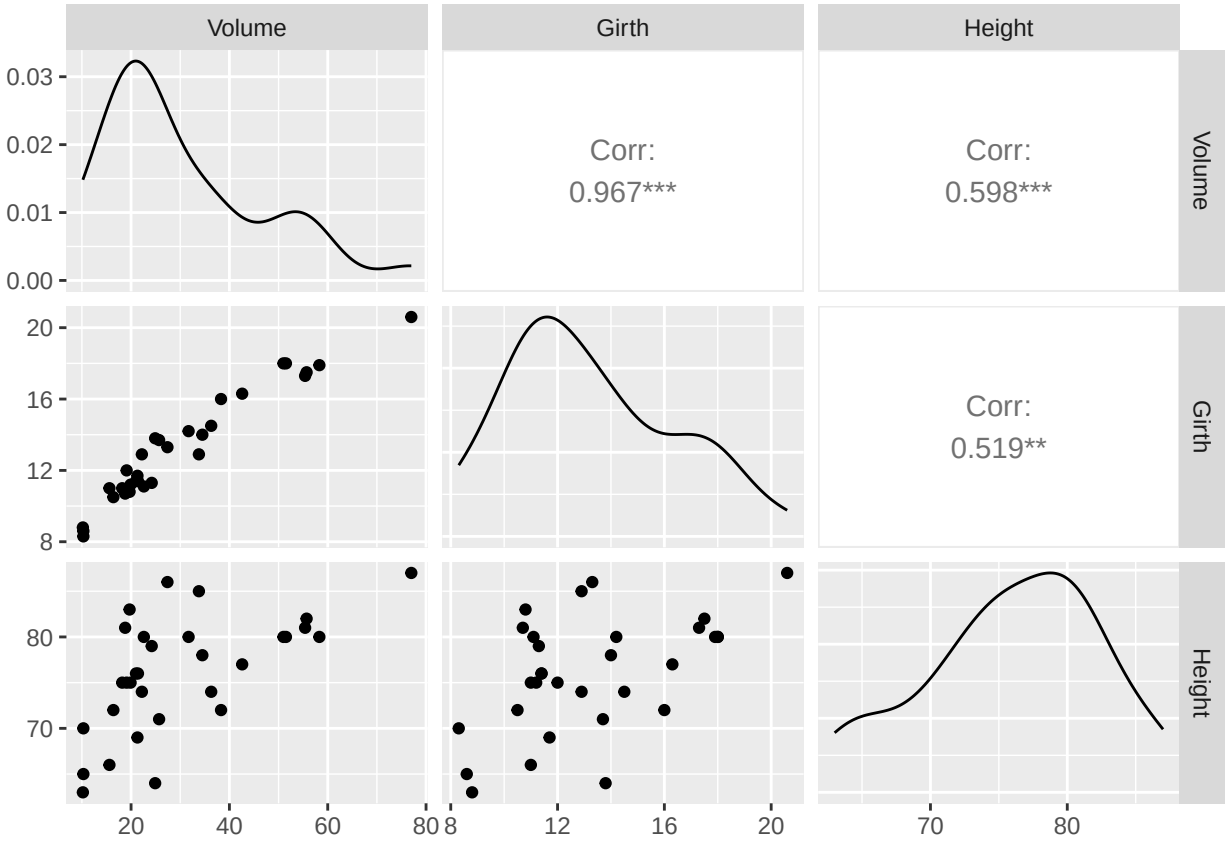
No existe ninguna périda de valor.

1.4. Visualización

Antes de proceder a realizar un análisis de correlación vamos a graficar las variables.

Vamos a mosostrar en 2D la relación existente entre las disintintas variables.

```
ggpairs(trees[, c('Volume', 'Girth', 'Height')])
```



En la relación Volume - Girth se observa que existe una correlación positiva fuerte

En la relación Volume - Height se observa una correlación positiva leve.

Además, no se observan valores outliers.

1.5. Análisis de correlación

Shapiro-Wilks

Se realiza el test de Shapiro-Wilks y comprobamos la hipótesis nula, es decir, comprobamos si las distintas variables siguen una distribución normal.

```
shapiro.test(Girth)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Girth
## W = 0.94117, p-value = 0.08893
```

```
shapiro.test(Height)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Height
## W = 0.96545, p-value = 0.4034
```

```
shapiro.test(Volume)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Volume
## W = 0.88757, p-value = 0.003579
```

De las tres variables anteriores se obtienen los siguientes resultados:

Variable	p-value	Normalidad
Girth	0.08893	no
Height	0.4034	no
Volume	0.003579	si

Se observa que unicamente *Volume* siguen una distribución normal.

Como el resto de características *Girth* y *Height* no siguien una distribución normal y al existir pocos datos aplicaremos el método Spearman para ver la correlación existente.

Correlación

En primer lugar realizamos una correlación clásica del dataset.

```
cor(trees, method="spearman")
```

```
##            Girth    Height    Volume
## Girth  1.0000000 0.4408387 0.9547151
## Height 0.4408387 1.0000000 0.5787101
## Volume 0.9547151 0.5787101 1.0000000
```

Se observa que la mayor correlación se obtiene entre *Girth* y *Volume* tal como se mostro en el gráfico anterior.

Prueba de hipótesis

Ahora se va a realizar la hipótesis para la correlación para comprobar si existe una tendencia entre las variables.

```
cor.test(Volume, Girth, method="spearman")
```

```
## Warning in cor.test.default(Volume, Girth, method = "spearman"): Cannot compute
## exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  Volume and Girth
## S = 224.61, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.9547151
```

```
cor.test(Volume, Height, method="spearman")
```

```
## Warning in cor.test.default(Volume, Height, method = "spearman"): Cannot
## compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  Volume and Height
## S = 2089.6, p-value = 0.0006484
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.5787101
```

Se obtienen los siguientes resultados:

$S(29) = 224.61$, $p < 0.001$, rs=0.9547

$S(29) = 2089.6$, $p < 0.001$, rs=0.5787

Como en ambo casos p-value < 0.05 rechazamos la hipótesis nula. Por lo tanto, existe una correlación entre las variables estadísticamente significativa, positiva y alta.

Correlación parcial

Se va a comprobar si *Volume* y *Girth* se ven afectada por *Height*.

```
pcor.test(Volume,
          Girth,
          Height,
          method = "spearman")
```

```
##    estimate      p.value statistic  n gp   Method
## 1 0.9557189 2.080355e-16  17.18489 31  1 spearman
```

S(28) = 17.185, p < 0.01, rs = 0.9557

Como p-value < 0.05, se rechaza la hipótesis nula, por lo que, para un nivel de confianza del 95%, la variable *height* es influyente.

La correlación clásica entre *Volume* y *Girth* calculada es de **0.9547**

Como la correlación clásica es practicamente igual a la correlación parcial (0.954 = 0.9557) es un indicativo que la variable confunsión (Height) es poco influyente.

2. Modelado de regresión múltiple sin iteracción

El modelo matemático de la función es:

$Volume = \beta_0 + \beta_1 \cdot Girth + \beta_2 \cdot Height$

```
model_1 <- lm(Volume ~ Girth + Height, data=trees)

summary(model_1)
```

```
##
## Call:
## lm(formula = Volume ~ Girth + Height, data = trees)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.4065 -2.6493 -0.2876  2.2003  8.4847
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -57.9877     8.6382  -6.713 2.75e-07 ***
## Girth         4.7082     0.2643  17.816  < 2e-16 ***
## Height        0.3393     0.1302   2.607   0.0145 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.882 on 28 degrees of freedom
## Multiple R-squared:  0.948,  Adjusted R-squared:  0.9442
## F-statistic:   255 on 2 and 28 DF,  p-value: < 2.2e-16
```

2.1. Bondad de ajuste

- Prueba F global F(2,28) = 255, p < 0.001.

Como p-value < 0.05 se rechaza la hipótesis nula, por lo que alguno de los coeficientes de pendiente β_j será distinto de cero.

- R2 ajustado = 0.9442

El modelo interpreta el **94.42%** de la variabilidad de la respuesta, es decir, el modelo se ajusta bien a los datos.

```
rsq.partial(model_1)
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded

## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded

## $adjustment
## [1] FALSE
##
## $variable
## [1] "Girth"  "Height"
##
## $partial.rsq
## [1] 0.9189376 0.1952712
```

- $R2_(Girth) = 0.9189$
- $R2_(Height) = 0.1953$

Analizando los R2 parcial se observa que la variable Girth es mucho más influyente que la variable Height. Hecho que se observó en el apartado anterior cuando se analizó la correlación clásica y parcial.

- RSE = 3.882

Indica que hay un error de 3.882 pies cúbicos (109.9m3) en el volumen del árbol.

Tasa de error

```
sigma(model_1)/mean(Volume)*100
```

```
## [1] 12.86612
```

- Tasa de error = 12.866%

El modelo tiene una tasa de error del 12.866%.

2.2 Coeficientes

El intercepto (β_0) vale -57.99, este no tiene sentido cuando extrapolamos a un árbol cuyo Girth y/o Height es 0 ya que un árbol no puede tener un volumen negativo (ni circunferencia ni altura).

El coeficiente de regresión (β_1) para el predictor Girth vale 4.71 y representa el cambio del volumen cuando el arbol es más/menos ancho y la altura es constante. Tiene pendiente positiva, hecho lógico ya que a más circunferencia, más volumen.

El coeficiente de regresión (β_2) para el predictor Height vale 0.334 y representa el cambio de volumen cuando el árbol es más/menos alto.

Como para cada coeficiente de regresión el p-value es menor 0.05. Se concluye que todas las variables contribuyen al modelo.

2.3 Intervalos de confianza

El IC permitirá determinar, con un 95% de probabilidad, el rango del coeficiente de regresión de cada predictor.

```
confint(model_1)
```

```
##                    2.5 %      97.5 %
## (Intercept) -75.68226247 -40.2930554
## Girth         4.16683899   5.2494820
## Height        0.07264863   0.6058538
```

Existe un 95% de probabilidad de que el intervalo [4.167 - 5.249] contenga el valor verdadero de la pendiente de *Girth*.

Existe un 95% de probabilidad de que el intervalo [0.073 - 0.606] contenga el valor verdadero de la pendiente de *Height*.

2.4. Importancia de los predictores

A continuación se van a determinar cuál es la contribución de cada predictor al modelo.

```
crlm <- calc.relimp(model_1,
            type = c("lmg"),
            rela = T)
crlm
```

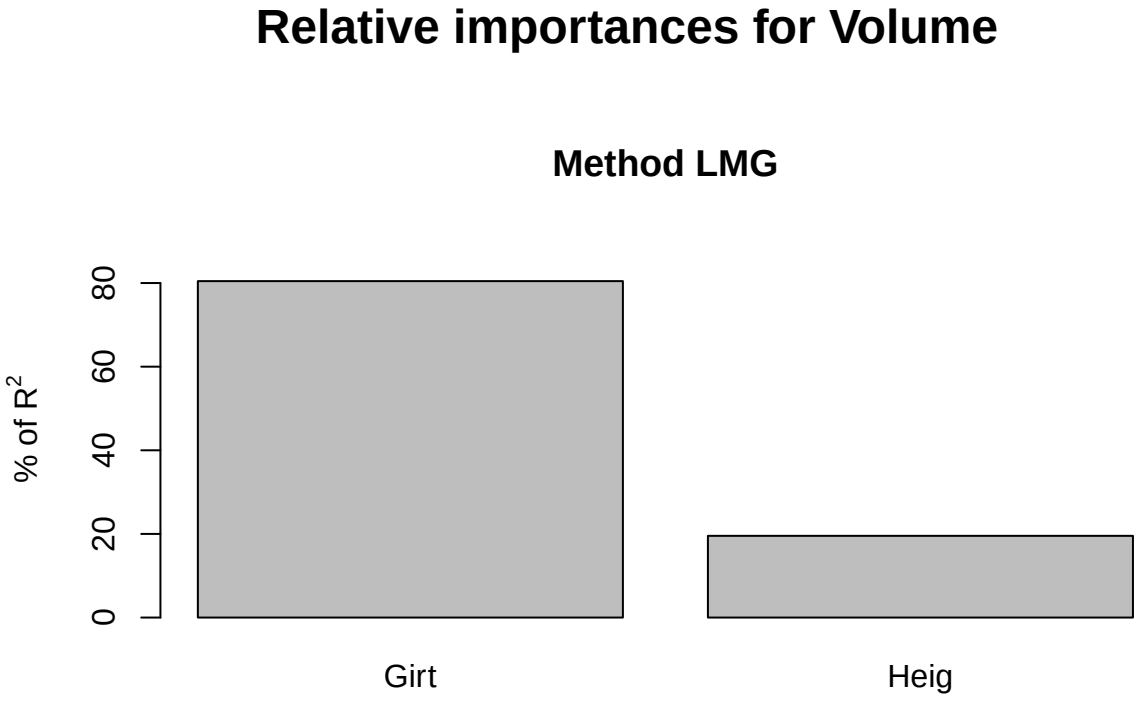
```
## Response variable: Volume
## Total response variance: 270.2028
## Analysis based on 31 observations
##
## 2 Regressors:
## Girth Height
## Proportion of variance explained by model: 94.8%
## Metrics are normalized to sum to 100% (rela=TRUE).
##
## Relative importance metrics:
##
##             lmg
## Girth  0.804561
## Height 0.195439
##
## Average coefficients for different model sizes:
##
##              1X       2Xs
## Girth  5.065856 4.7081605
## Height 1.543350 0.3392512
```

La importancia de los predictores da como resultado:

- *Girth* tiene una importancia del 80.456%
- *Height* tiene una importancia del 19.544%

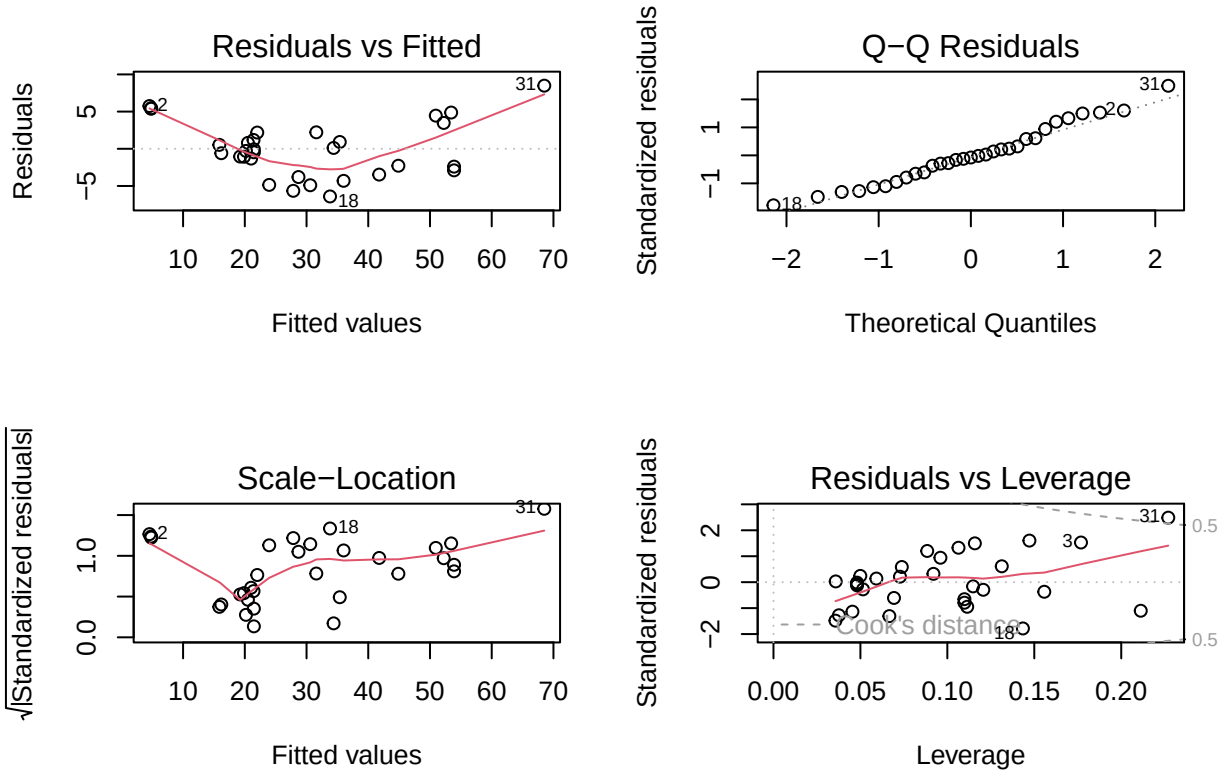
En el siguiente gráfico de barras se puede observar de una forma más clara que el volumen del árbol depende más de su circunferencia (Girth) que de la altura (Height).

```
plot(crlm)
```



2.5. Evaluación de los supuestos del modelo

```
par(mfrow = c(2,2))
plot(model_1)
```



A continuación analizaremos los 4 gráficos.

1. Residuals vs Fitted: representa la linealidad del modelo.

Se observa que los residuos no siguen una tendencia lineal.

2. Q-Q Residuals: representa la distribución normal del modelo.

Los residuos se separan lijeramente de la distribución normal teórica (línea discontinua), pero podemos concluir que el modelo sigue una cierta distribución normal.

3. Scale-Location: permite evaluar el supuesto de homocedasticidad.

Se observa que la línea roja no es horizontal por lo que no se cumple el supuesto de homocedasticidad. Por lo tanto podríamos decir que el modelo presenta heterocedasticidad.

4. Residuals vs Leverage: permite identificar valores inusuales o influyentes sobre el modelo.

No se observan outliers si bien la observación nº 31 supera la distancia de Cook, lo que indica que es una observación influyente.

3. Modelo de regresión con iteracción

Ahora realizaremos el modelado teniendo en cuenta que va a existir una iteracción entre Girth y Height

$Volume = \beta_0 + \beta_1 \cdot Girth + \beta_1 \cdot Height + \beta_3 \cdot Girth \cdot Height$

```r
model_2 <- lm(Volume ~ Girth*Height, data=trees)
summary(model_2)
```

```
##
## Call:
## lm(formula = Volume ~ Girth * Height, data = trees)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.5821 -1.0673  0.3026  1.5641  4.6649
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  69.39632   23.83575   2.911  0.00713 **
## Girth        -5.85585    1.92134  -3.048  0.00511 **
## Height       -1.29708    0.30984  -4.186  0.00027 ***
## Girth:Height  0.13465    0.02438   5.524 7.48e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.709 on 27 degrees of freedom
## Multiple R-squared:  0.9756, Adjusted R-squared:  0.9728
## F-statistic: 359.3 on 3 and 27 DF,  p-value: < 2.2e-16
```

3.1. Coeficientes

El intercepto (β_0) vale 69.396.

El coeficiente de regresión (β_1) para el predictor Girth vale -5.866. Que la pendiente sea negativa no tiene sentido, ya que el volumen del árbol no puede disminuir cuando aumenta su grosor y mantenemos constante la altura.

El coeficiente de regresión (β_2) para el predictor Height vale -1.2971. Pasa lo mismo que en el caso anterior, el volumen del árbol no puede disminuir cuando aumenta la altura.

El coeficiente de regresión (β_3) para la iteracción Girth*Height vale 0.135

Multicolinealidad

A continuación se va a analizar si existe multicolinealidad con las variables independientes.

```r
vif(model_2)
```

```
## there are higher-order terms (interactions) in this model
## consider setting type = 'predictor'; see ?vif

##        Girth       Height Girth:Height
##    148.66145     15.93884    210.97302
```

Se observa que existe problema de multicolinealidad, los valores VIF son mayores a 5.

Esto significa que las variables independientes estan teniendo una fuerte correlación entre sí, es decir, no se puede determinar los coeficientes de regresión del modelo de forma fiable ya que no se pueden aislar los efectos de las variables independientes.

Centraremos las variables para intentar resolver el problema de multicolinealidad

```r
#A través de la función c() transformamos la matriz en un vamos numérico
#Necesario para posteriormente trabajar con los valores Girth2 y Height2 de forma más sencilla.

Girth2 <-  c(scale(Girth, center=T, scale=F))
Height2 <-  c(scale(Height, center=T, scale=F))

model_3 <- lm(Volume ~ Girth2*Height2, data = trees)

vif(model_3)
```

```
## there are higher-order terms (interactions) in this model
## consider setting type = 'predictor'; see ?vif
```

```
##         Girth2        Height2 Girth2:Height2
##       1.513180       1.487784       1.126849
```

Al centrar las variables se observa que el valor VIF es menor que 5, o lo que es lo mismo, se ha resuelto los problemas de multicolinealidad.

Nuevo modelo

Se genera el nuevo modelo con las variables centradas.

```r
summary(model_3)
```

```
##
## Call:
## lm(formula = Volume ~ Girth2 * Height2, data = trees)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.5821 -1.0673  0.3026  1.5641  4.6649
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    28.81791    0.54466  52.910  < 2e-16 ***
## Girth2          4.37789    0.19384  22.585  < 2e-16 ***
## Height2         0.48687    0.09466   5.143 2.07e-05 ***
## Girth2:Height2  0.13465    0.02438   5.524 7.48e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.709 on 27 degrees of freedom
## Multiple R-squared:  0.9756, Adjusted R-squared:  0.9728
## F-statistic: 359.3 on 3 and 27 DF,  p-value: < 2.2e-16
```

3.2. Coeficientes

El intercepto (β_0) vale 28.82.

El coeficiente de regresión (β_1) para el predictor Girth vale 4.378. Ahora el valor si tiene sentido, el volumen del árbol aumenta en función de su circunferencia.

El coeficiente de regresión (β_2) para el predictor Height vale 0.487. La altura aumenta en función de la altura

El coeficiente de regresión (β_3) para la iteracción Girth*Height vale 0.135

3.3 Ecuación del modelo

La función que define el modelo es:

$Volume = 28.818 + 4.378 \cdot Girth + 0.487 \cdot Height + 0.135 \cdot Girth \cdot Height$

3.4. Annova

A través de la función ANOVA vamos a comprobar si la iteracción del modelo es estadisticamente significativo.

```
Anova(model_3)
```

```
## Anova Table (Type II tests)
##
## Response: Volume
##                 Sum Sq Df F value    Pr(>F)
## Girth2          4783.0  1 651.965 < 2.2e-16 ***
## Height2          102.4  1  13.956 0.0008867 ***
## Girth2:Height2   223.8  1  30.512 7.484e-06 ***
## Residuals        198.1 27
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

$F(1,30) = 30.512$, $p < 0.001$

Se observa que la iteracción es significativa, por lo tanto hay que tenerla en cuenta esta iteracción en el modelo final.

3.5. Predicciones

Vamos a comprobar algunas predicciones del modelo

Como model_3 está centrado, vamos a centrar los nuevos valores a predecir.

El centrado sigue la siguiente función:

$V_c = V - \bar{V}$

Donde:

V_c es la variable centrada

V es la variable sin centrar

$\bar{V}$ es la media del conjunto

```
#Determinamos la media de Girth y Height
media_girth = mean(trees$Girth)
media_height = mean(trees$Height)

#Nuevos valores a predecir
newGirth = c(10.8, 12.9, 20)
newHeight = c(83, 85, 90)

#Aplicamos el centrado a los valores a predecir
newGirth_centered =  newGirth - media_girth
newHeight_centeres = newHeight - media_height

#Generamos el dataframe con los valores a predecir
```

```r
new <- data.frame(Girth2 = newGirth_centered, Height2 = newHeight_centeres)

predictions <- predict(model_3, newdata = new, interval="prediction")

predictions
```

```
##        fit      lwr      upr
## 1 19.19944 13.18673 25.21214
## 2 31.25233 25.33994 37.16473
## 3 77.91976 70.46084 85.37868
```

Se observa que los valor obtenidos son:

Girth	Height	Volume (fit)
10.8	83	19.199
12.9	85	31.252
20	90	77.920

También podemos aplicar la ecuación del modelo directamente tal como se muestra a continuación.

```r
28.818 + 4.378*newGirth_centered + 0.487*newHeight_centeres + 0.135*newGirth_centered*newHeight_centeres
```

```
## [1] 19.19424 31.25247 77.95511
```

Se comprueba que los valores determinados son practicamente los mismos.

Graficamos

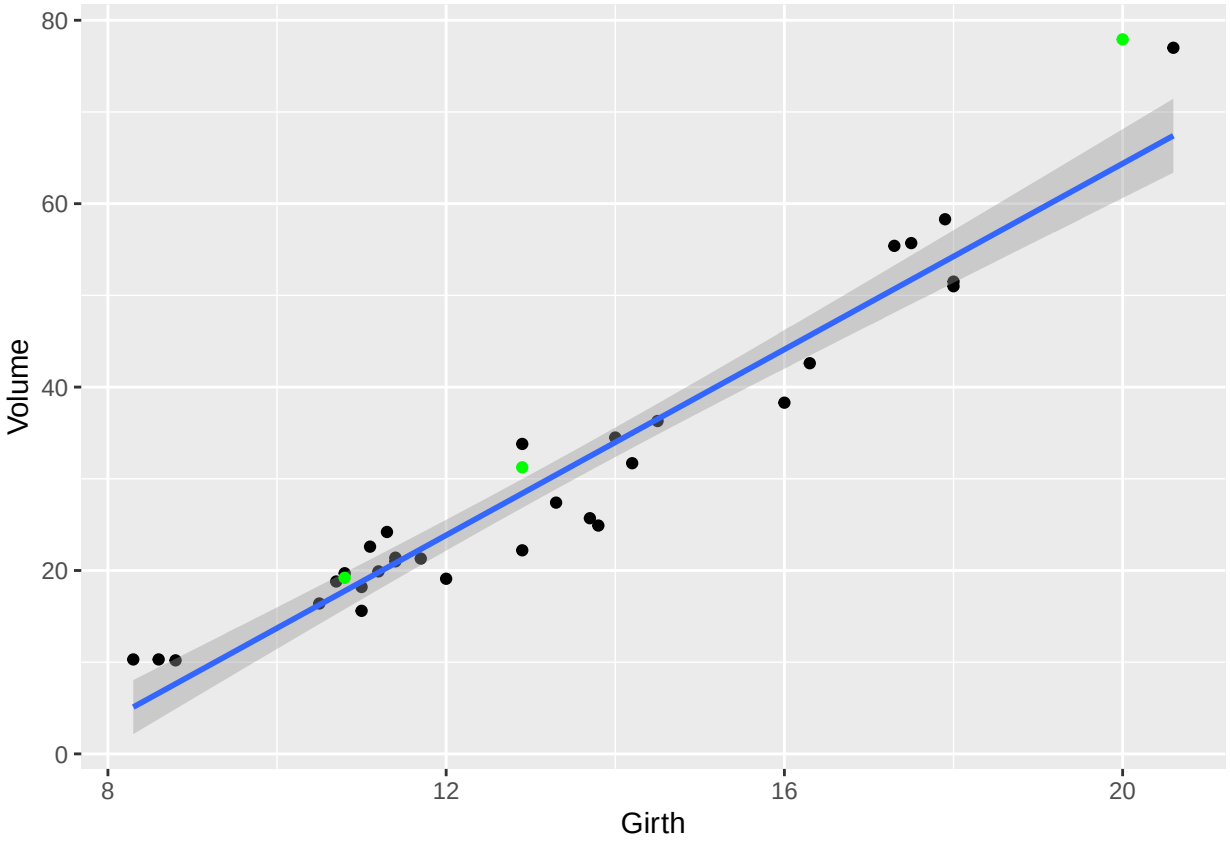
Mostramos en la gráfica los puntos predichos

```r
volume_df <- as.data.frame(predictions)
df <- data.frame(newGirth,volume_df['fit'])

p <- ggplot(trees, aes(Girth, Volume)) +
  geom_point() +
  stat_smooth(method = lm)

p + geom_point(data = df, mapping = aes(newGirth, fit), color="green")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```


80
60
40
20
0
Volume
8
12
16
20
Girth