

Índice:

1.	PRIMER MODELO ESTUDIADO.....	3-6
2.	SEGUNDO MODELO ESTUDIADO.....	6-13
2.1.	METODO FORWARD.	
2.2.	METODO BACKWARD.	
2.3.	METODO STEPWISE.	
2.4.	METODOS BASADOS EN CRITERIOS	
2.5.	ANALIZANDO LAS OBSERVACIONES EXTREMAS	
3.	MODELO 2.1 ESTUDIADO.....	13-15
4.	MODELO 2.2 ESTUDIADO.....	15-17
5.	MODELO 2.3 ESTUDIADO.....	17-19
6.	MODELO 2.4 ESTUDIADO.....	19-22
7.	MODELO 2.5 ESTUDIADO.....	22-24
8.	TABLA RESUMEN DE LOS MODELOS ESTUDIADOS.....	24
9.	VALIDACION DEL MODELO DEFINITIVO.....	25-30
9.1.	VALIDACION CON LAS VARIABLES DEL MODELO SELECCIONADO.....	25-29
9.1.1.	MSPE	
9.1.2.	LOOCV	
9.1.3.	K-Fold Cross Validation	
9.2.	VALIDACION CON TODAS LAS VARIABLES EXPLICATIVAS....	29-30
9.2.1.	MSPE	
9.2.2.	LOOCV	
9.2.3.	K-Fold Cross Validation	
10.	PREDICCIONES.....	31-32

Breve comentario sobre las variables de importancia: Antes de empezar, detectamos en nuestro modelo 3 variables cualitativas, a saber: ShelveLoc, Urban y US. La variable cualitativa que tendrá importancia para nosotros será ShelveLoc, que tiene 3 categorías y que podremos resumir en 2: ShelveLocGood y ShelveLocMedium.

1. PRIMER MODELO ESTUDIADO:

Modelo 1: Modelo completo, Sales~::

Si realizamos un summary, tenemos los siguientes valores, que representan el rendimiento del modelo:

- El R^2 : Cuantos más predictores se incluyan en el modelo mayor es el valor de R^2 , dado que aquí incluimos todas las variables predictoras, tenemos claramente una explicabilidad del 100%.
- El error estándar residual (RSE): Queremos una estimación sobre la desviación promedio de cualquier punto (dato) respecto a la recta de regresión exacta (la verdadera). En la columna Std. Error se aprecia que son todos los errores son pequeños y que el valor de Residual standard error del modelo completo también lo es, que es una buena señal.
- El F-statistic: Si MSE es pequeño, se tendrá una F-statistic grande, de esta forma cuanto más grande sea el valor de F-statistic, mejor es el modelo. En nuestro caso el valor es altísimo.
- El p-valor: El modelo es estadísticamente significativo al ser $p - value = 2.2 \cdot e^{-16} < 0.05$.

Coefficients:					
	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	5.478e+00	2.905e-03	1885.723	< 2e-16 ***	
CompPrice	9.256e-02	1.997e-03	4634.597	< 2e-16 ***	
Income	1.579e-02	8.924e-06	1768.931	< 2e-16 ***	
Advertising	1.160e-01	5.408e-05	2144.673	< 2e-16 ***	
Population	-5.551e-06	1.794e-06	-3.095	0.002113 **	
Price	-9.530e-02	1.285e-05	-7419.057	< 2e-16 ***	
ShelveLocGood	4.836e+00	7.366e-04	6565.461	< 2e-16 ***	
ShelveLocMedium	1.952e+00	6.132e-04	3183.448	< 2e-16 ***	
Age	-4.611e-02	1.536e-05	-3001.672	< 2e-16 ***	
Education	-2.284e-02	9.533e-05	-2.396	0.017054 *	
UrbanYes	6.119e-04	5.443e-04	1.124	0.261605	
usYes	-6.711e-04	7.279e-04	-0.922	0.357107	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1					
Residual standard error: 0.004899 on 385 degrees of freedom					
Multiple R-squared: 1, Adjusted R-squared: 1					
F-statistic: 9.47e+06 on 14 and 385 DF, p-value: < 2.2e-16					

Recordemos que hay variables cualitativas, por tanto, anova() es la función adecuada para tratarlas:

Analysis of Variance Table					
Response: Sales	Df	Sum Sq	Mean Sq	F value	Pr(>F)
CompPrice	1	13.07	13.07	5.4436e+05	< 2.2e-16 ***
Income	1	79.07	79.07	3.2942e+06	< 2.2e-16 ***
Advertising	1	219.35	219.35	9.1382e+06	< 2.2e-16 ***
Population	1	0.38	0.38	1.5931e+04	< 2.2e-16 ***
Price	1	1198.87	1198.87	4.9945e+07	< 2.2e-16 ***
ShelveLoc	2	1047.47	523.74	2.1819e+07	< 2.2e-16 ***
Age	1	217.39	217.39	9.0564e+06	< 2.2e-16 ***
Education	1	1.05	1.05	4.3757e+04	< 2.2e-16 ***
Urban	1	1.22	1.22	5.0835e+04	< 2.2e-16 ***
US	1	1.57	1.57	6.5286e+04	< 2.2e-16 ***
Residuals	385	0.01	0.00		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

En la tabla anterior sí nos dicen que las variables cualitativas Urban y US son estadísticamente significativas. El modelo completo es estadísticamente significativo.

Sin embargo, las medidas anteriores no son capaces de tratar el caso de la inclusión de variables nuevas que tengan poca significancia estadística, por ello, para la medición de la calidad del modelo y para futura selección, recurriremos al $R^2_{ajustado}$, AIC , BIC y C_p *Mallows*:

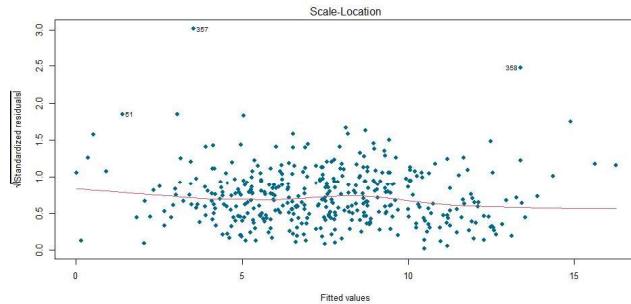
- El $R^2_{ajustado}$ da una explicabilidad de 100% como era de esperar.
- El AIC : Tiene un valor de -3103.058 . Un valor bajo es buen indicio.
- El BIC : Tiene un valor de -3039.195 . Un valor bajo es buen indicio.
- El C_p *Mallows* es una variante del AIC , devolvería 13.

Si en un principio eligiésemos este modelo como el definitivo:

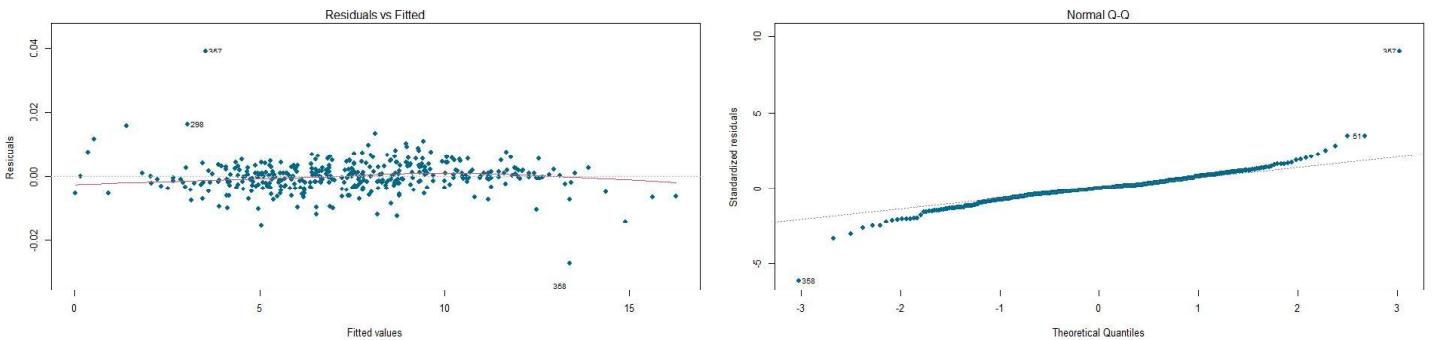
En los siguientes gráficos se tiene:

- *Residuals vs Fitted*: Los puntos se distribuyen de forma aleatoria alrededor de la línea horizontal que marca un residuo nulo, lo que es buen indicio de control sobre la linealidad, también sobre la homocedasticidad (que confirmaremos con el gráfico *Scale-Location*, la gráfica de ubicación de escala-dispersión). Sin embargo, al hacer las pruebas siguientes:
 - *Test de Braunch-Pagan*: Tenemos un p-valor menor que $2.2e^{-16} < 0.05$, luego rechazamos la hipótesis nula y aceptamos la alternativa: los residuos no tienen varianza constante.
 - *Test de varianza no constante*: El p-valor es 0.052742 , ligeramente mayor a 0.05 . Luego aceptaríamos homocedasticidad según este test.

La alternativa para desempatar sería una gráfica *Scale-Location*: Se observa una pequeña tendencia descendente, esto no es buena señal pues significaría que las varianzas disminuyen a medida que aumentamos los valores ajustados \hat{Y} , lo que sugiere una pequeña desviación en las varianzas, teniendo en el modelo algo de heterococidad.



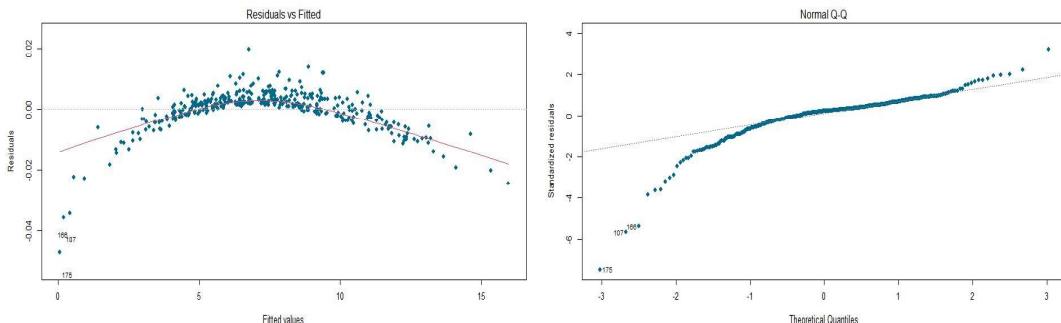
- *Normal Q-Q*: Los puntos no se encuentran ubicados sobre la recta, habría que hacer una transformación para normalizar.
- En el gráfico *Residuals vs fitted* se confirma que en los extremos hay problemas de desviación respecto a la línea horizontal que representa el valor nulo de los residuos.



- Multicolinealidad: Utilizamos el *VIF* como medidor de la colinealidad, lo más óptimo sería $0 < VIF < 1$. Se observa que CompPrice, Advertising, Price y US superan los umbrales óptimos por lo que debe de existir algo de colinealidad.

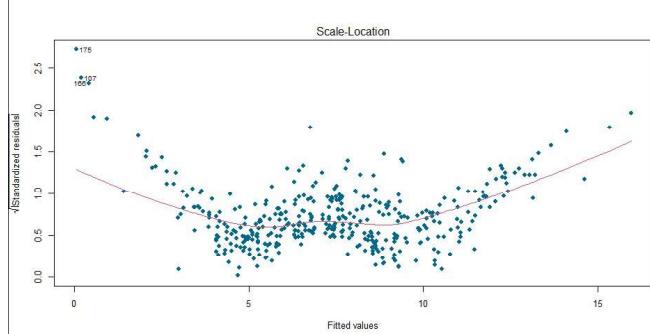
	GVIF	Df	GVIF^(1/(2*Df))
CompPrice	1.558937	1	1.248574
Income	1.036691	1	1.018180
Advertising	2.149855	1	1.466238
Population	1.161467	1	1.077714
Price	1.537637	1	1.240015
ShelveLoc	1.070062	2	1.017073
Age	1.029313	1	1.014551
Education	1.037288	1	1.018473
Urban	1.026808	1	1.013316
US	2.021606	1	1.421832

Si llevamos a cabo una transformación yjPower, no nos queda lo esperado en el *QQ-plot* (en cambio, sí mejora el resultado en los test, pero sin poder aceptar la normalidad de los residuos: Para Shapiro-Wilk nos da $p - valor = 0.1898$ pero, para Kolmogorov-Smirnov, nos mejora el p-valor de 0.0006087 a 0.00108, aunque sigue siendo insuficiente). Es más, la no linealidad y la heterocedasticidad, se hace mucho más evidente, como se refleja en *Residuals vs fifted* y *Scale-Location*:



La razón para utilizar este tipo de transformación basado en la familia yjPower es porque no todas las variables respuesta de nuestro modelo son estrictamente positivas. En concreto, la observación 175 de la columna de la variable respuesta Sales vale 0.00, de ahí el problema con el BoxCox visto en clase.

En la siguiente gráfica también se observan las tendencias ascendentes y descendentes, lo que sugiere una enorme desviación en las varianzas (heterocedasticidad):



Por tanto, este modelo presenta mucha problemática al analizar los supuestos del *ANOVA* que, al tratar de corregirlos, empeoran drásticamente. Si eliminamos outliers no influyentes, la situación no mejora lo suficiente. Además, no es necesario incluir todas las variables en el modelo (no añadir ruido) para conseguir una buena explicabilidad. Tampoco mejora la heterocedasticidad hacer una transformación logarítmica tipo: `modelo_completo_trannf <- lm(log(Sales)~., data = Carseats)`.

2. **MODELO 2 ESTUDIADO:**

Modelo 2:

Vamos a utilizar los métodos forward, backward y stepwise para tratar el modelo más adecuado y, una vez tengamos el modelo definitivo, analizar a fondo su explicabilidad y capacidad predictiva.

2.1Método forward:

Inicia con el modelo vacío y busca el mejor modelo con menor *AIC* con una, luego dos, luego tres variables y así sucesivamente. La desventaja de este método es que nos recomienda un modelo con una variable NO significativa, aunque no es el caso:

```

call:
lm(formula = Sales ~ ShelveLoc + Price + CompPrice + Advertising +
    Age + Income, data = Carseats)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.7728 -0.6954  0.0282  0.6732  3.3292 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 5.475226  0.505005 10.84 <2e-16 ***
ShelveLocGood 4.835675  0.152499 31.71 <2e-16 ***
ShelveLocMedium 1.951993  0.125375 15.57 <2e-16 ***
Price        -0.095319  0.002670 -35.70 <2e-16 ***
CompPrice     0.092571  0.004123 22.45 <2e-16 ***
Advertising   0.115903  0.007724 15.01 <2e-16 ***
Age          -0.046128  0.003177 -14.52 <2e-16 ***
Income        0.015785  0.001838   8.59 <2e-16 ***  
---
signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.019 on 392 degrees of freedom
Multiple R-squared:  0.872, Adjusted R-squared:  0.8697 
F-statistic: 381.4 on 7 and 392 DF,  p-value: < 2.2e-16

```

El método que nos recomienda es: `Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age + Income`.

2.2Método backward:

Inicia con el modelo completo, procede igual que antes.

```

Call:
lm(formula = Sales ~ CompPrice + Income + Advertising + Price +
    ShelveLoc + Age, data = Carseats)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.7728 -0.6954  0.0282  0.6732  3.3292 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 5.475226  0.505005 10.84   <2e-16 ***
CompPrice   0.092571  0.004123 22.45   <2e-16 ***
Income      0.015785  0.001838  8.59   <2e-16 ***
Advertising 0.115903  0.007724 15.01   <2e-16 ***
Price       -0.095319  0.002670 -35.70   <2e-16 ***
ShelveLocGood 4.835675  0.152499 31.71   <2e-16 ***
ShelveLocMedium 1.951993  0.125375 15.57   <2e-16 ***
Age        -0.046128  0.003177 -14.52   <2e-16 ***  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.019 on 392 degrees of freedom
Multiple R-squared:  0.872, Adjusted R-squared:  0.8697 
F-statistic: 381.4 on 7 and 392 DF,  p-value: < 2.2e-16

```

El método que nos recomienda es: Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age + Income.

2.3Método stepwise:

Este inicia con el modelo vacío. Este método nos quitaría las variables no significativas (a diferencia de los otros dos métodos):

```

Call:
lm(formula = Sales ~ ShelveLoc + Price + CompPrice + Advertising +
    Age + Income, data = Carseats)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.7728 -0.6954  0.0282  0.6732  3.3292 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 5.475226  0.505005 10.84   <2e-16 ***
ShelveLocGood 4.835675  0.152499 31.71   <2e-16 ***
ShelveLocMedium 1.951993  0.125375 15.57   <2e-16 ***
Price       -0.095319  0.002670 -35.70   <2e-16 ***
CompPrice   0.092571  0.004123 22.45   <2e-16 ***
Advertising 0.115903  0.007724 15.01   <2e-16 ***
Age        -0.046128  0.003177 -14.52   <2e-16 ***  
Income      0.015785  0.001838  8.59   <2e-16 ***  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.019 on 392 degrees of freedom
Multiple R-squared:  0.872, Adjusted R-squared:  0.8697 
F-statistic: 381.4 on 7 and 392 DF,  p-value: < 2.2e-16

```

El método que nos recomienda es: Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age + Income. Igual que antes.

Todas sus variables son significativas al tener su p-valor menor que 0.05, por tanto, analizamos ahora el rendimiento del modelo:

- El R^2 : Tenemos una explicabilidad del 86.97%. Un gran dato y sin necesidad de utilizar las 13 variables, ¡con solo 6!
- El error estándar residual (RSE): En la columna Std. Error se aprecia que son todos los errores son relativamente pequeños y que el valor de Residual standard error del modelo también lo es (1.019), que es una buena señal.
- El F-statistic: Si MSE es pequeño, se tendrá una F-statistic grande, de esta forma cuanto más grande sea el valor de F-statistic, mejor es el modelo. En nuestro caso el valor es relativamente alto: 381.4.
- El p-valor: El modelo es estadísticamente significativo al ser $p - value = 2.2 \cdot e^{-16} < 0.05$.

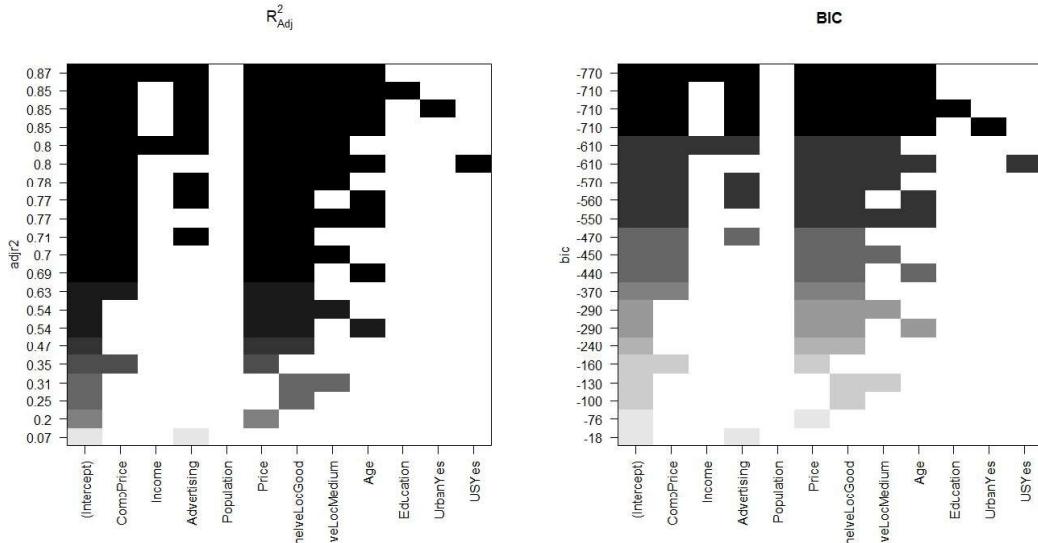
2.4 Metodos basados en criterios:

Veamos los mejores modelos usando como criterios R^2 ajustado y BIC :

La función regsubsets realiza una búsqueda exhaustiva de los mejores subconjuntos de variables X para explicar Y . El objeto `modelos_mejoresCompleto` contiene una tabla de posibilidades desde 1 hasta 7 covariables:

	(Intercept)	CompPrice	Income	Advertising	Population	Price	ShelveLocGood	ShelveLocMedium	Age	Education	UrbanYes	USYes
1	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE
1	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
1	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
2	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE
2	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
2	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
3	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE
3	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE
3	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE
4	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE
4	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE
4	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE
4	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE
5	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE
5	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE
5	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE
6	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE
6	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE
6	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE
6	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
7	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
7	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
7	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

De entre toda esta batería de modelos posibles, veamos unos gráficos de como son R^2 ajustado y AIC :



El mejor modelo será aquel con mayor R^2 *ajustado* y menores BIC y AIC :

A la izquierda vemos que el modelo con mayor R^2 *ajustado* es $Sales \sim ShelveLoc + Price + CompPrice + Advertising + Age + Income$ que ya obtuvimos con el procedimiento mecánico anterior, además también sabemos que tienen el AIC bajo. En el segundo plot, las variables dependientes que conforman el modelo mencionado tienen un valor muy bajo, lo que es buen indicio.

De hecho, guiándonos por el BIC , podemos obtener analíticamente el mejor modelo del objeto `modelos_mejoresCompleto`:

1. Hacemos una lista con todos los BIC de `modelos_mejoresCompleto`:

```
[1] -103.36220 -76.26634 -18.17982 -235.90366 -159.18848 -134.66784 -373.71021 -288.89418 -288.74490 -468.52957 -449.96052 -441.40171 -566.50696
[14] -559.56850 -549.98260 -711.31065 -608.13362 -607.03462 -774.30360 -707.37999 -706.95421
```

2. Tomamos el modelo con menor BIC : Me devuelve 19.
3. Ahora vemos cuál es ese modelo 19:

(Intercept)	TRUE	CompPrice	TRUE	Income	Advertising	TRUE	Population	FALSE	Price	ShelveLocGood	ShelveLocMedium	TRUE	Age	TRUE
	TRUE	UrbanYes	FALSE	TRUE	USYes	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
	Education													

El modelo resultante es, de nuevo, $Sales \sim ShelveLoc + Price + CompPrice + Advertising + Age + Income$, con $BIC = -774.30360$

Procediendo de forma análoga, se obtiene que el mejor modelo es el anterior:

(Intercept)	CompPrice	Income	Advertising	Population	Price	ShelveLocGood	ShelveLocMedium	Age
TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE
Education	UrbanYes	USYes	FALSE					

Con R^2 *ajustado* = 0.86969647, que ya obtuvimos en las tablas.

En resumen,

- El $R^2_{ajustado}$ da una explicabilidad de 86.97% como era de esperar.
- El AIC : Tiene un valor de 23.32. Un valor bajo es buen indicio.
- El BIC : Tiene un valor de -774.30360 . Un valor bajo es buen indicio.

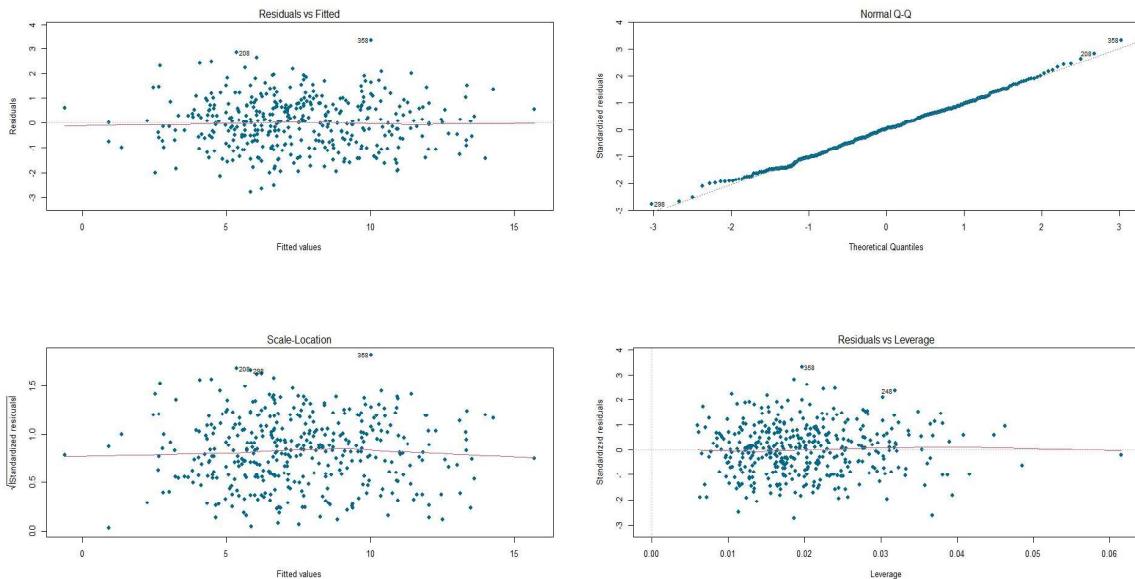
- El C_p *Mallows* es una variante del *AIC*, devolvería 6.3, que es ligeramente superior al número de parámetros, pero dado que se encuentra cerca de dicho número, consideramos que no es un valor tan desastre como los C_p *Mallows* que saldrán más adelante cuando quitemos Price o CompPrice ([modelo 2.3](#) y [modelo 2.4](#)).

Si en un principio eligiésemos este modelo como el definitivo:

En los siguientes gráficos se tiene:

- *Residuals vs Fitted*: Los puntos se distribuyen de forma aleatoria alrededor de la línea horizontal que marca un residuo nulo, lo que es buen indicio de control sobre la linealidad (y la homocedasticidad que confirmaremos más adelante con *Scale-Location*). Analíticamente, podemos hacer las pruebas siguientes:
 - *Test de Braunsch-Pagan*: Tenemos un p-valor $0.6914 > 0.05$, luego aceptamos la hipótesis nula: Hay homocedasticidad.
 - *Test de varianza no constante*: El p-valor es 0.67408 mayor que 0.05 . Luego aceptaríamos homocedasticidad según este test también.

La gráfica *Scale-Location*: Se observa que no hay una tendencia clara, lo cual es buena señal pues sería indicio de que las varianzas no varían a medida que aumentamos los valores ajustados \hat{Y} . Tenemos por tanto linealidad y homocedasticidad.



En el gráfico *Residuals vs Leverage*, tenemos identificados outliers influyentes, es decir, valores extremos que pueden influir en los resultados finales, a saber: 208 y 358.

- *Normal Q-Q*: Los puntos se encuentran bastante bien ubicados sobre la recta, habrá que tratar algunos outliers que aparecen en los diagramas anteriores: 208, 298, 357, 358 y 248 serán candidatos.

No obstante, utilizaremos dos test para analizar la normalidad de los residuos:

- *Test de Shapiro-Wilk*: Obtenemos un p-valor de $0.7439 > 0.05$, luego aceptamos la normalidad de los residuos.

- Test de Kolmogorov-Smirnov: Obtenemos un p-valor de $0.9349 > 0.05$, luego aceptamos la normalidad de los residuos holgadamente.

- Independencia de residuos, ¿hay autocorrelación?:

El valor de DW es cercano a 2 y el p-valor es superior a 0.05, luego aceptamos la hipótesis alternativa: no hay autocorrelación.

```
Durbin-Watson test
data: modelo_reducido
DW = 1.9882, p-value = 0.4523
alternative hypothesis: true autocorrelation is greater than 0
```

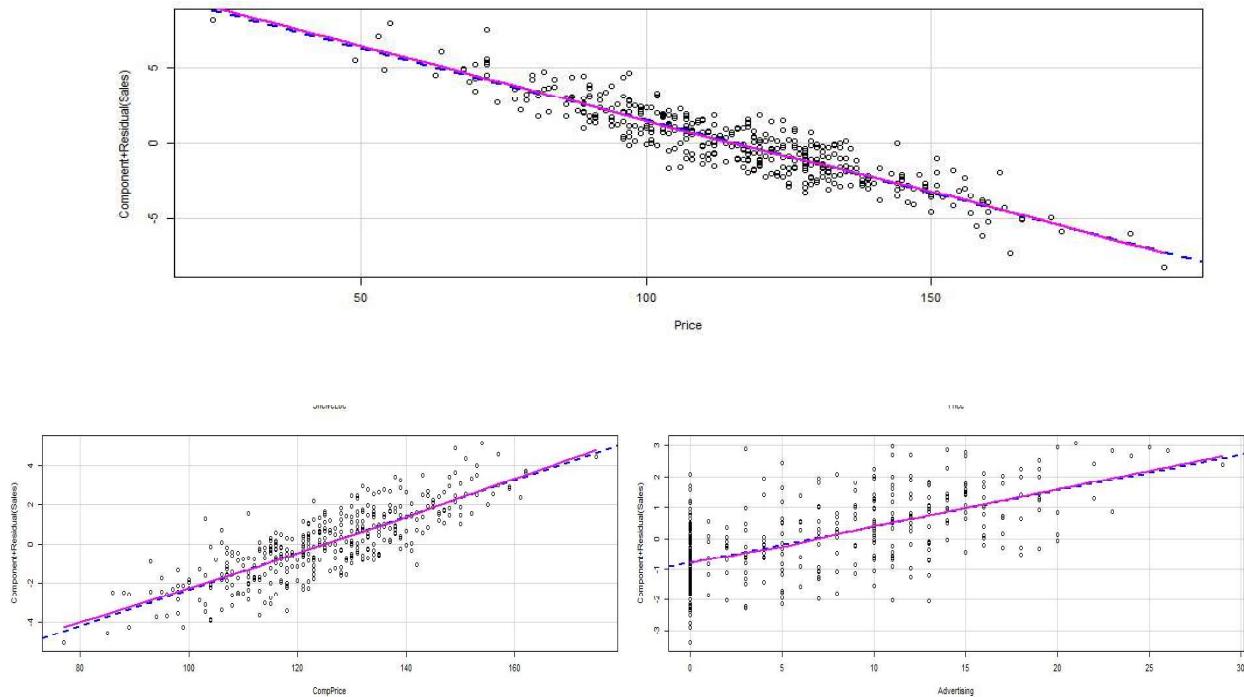
- Multicolinealidad: Utilizamos el VIF como medidor de la colinealidad, lo más óptimo sería $0 < VIF < 1$. Se observa que CompPrice y Price superan los umbrales por lo que debe de existir quizás un poco de colinealidad:

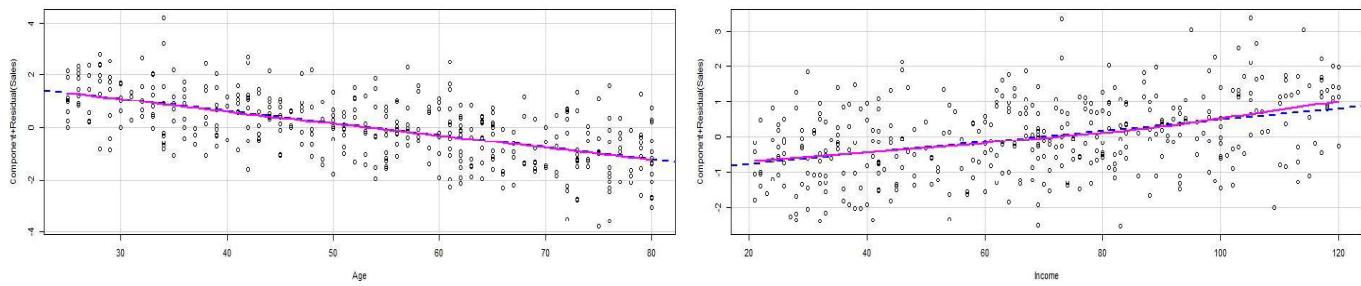
	VIF	df	$VIF^{(1/(2*df))}$
ShelveLoc	1.015139	2	1.003763
Price	1.534425	1	1.238719
CompPrice	1.534883	1	1.238904
Advertising	1.012935	1	1.006447
Age	1.016830	1	1.008380
Income	1.015448	1	1.007694

- Linealidad: Anteriormente ya vimos una buena tendencia lineal general del modelo gráficamente. Analíticamente, para la linealidad, lo importante es que la asociación de variables en nuestro modelo sea lineal, luego la media de los residuos debería ser cero o casi cero, si hacemos `mean(modelo_reducido_residuos)` nos devuelve $7.729413e^{-17}$.

Podemos también analizar la linealidad y capacidad explicativa para cada variable dependiente, para ver cuáles afectarían más a la linealidad con el comando `crPlots`:

La **línea discontinua azul** representa la verdadera recta de regresión, la **recta rosa** representa lo que sigue nuestro modelo:





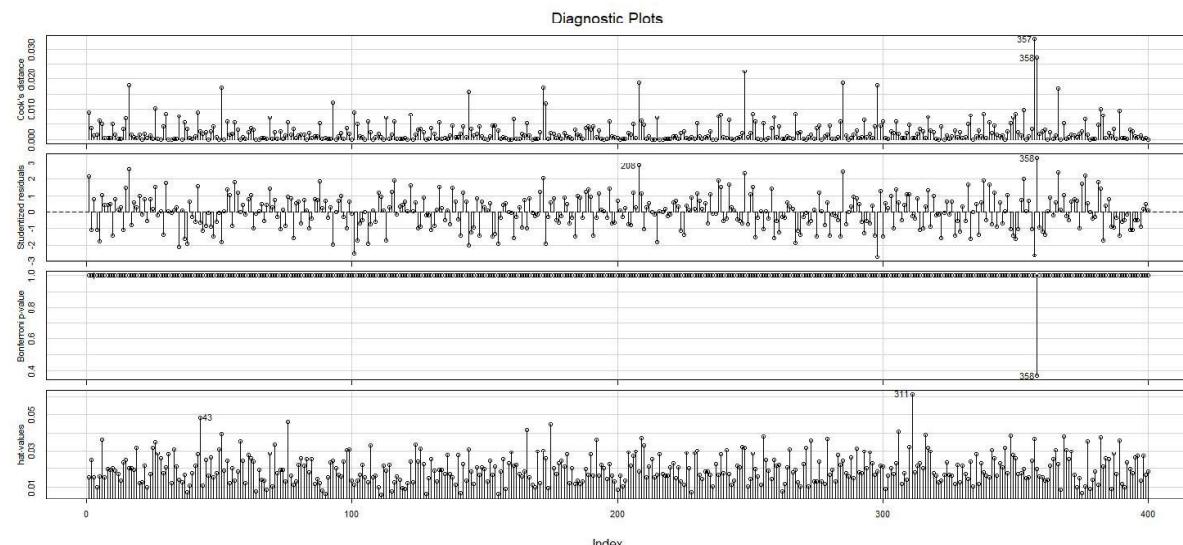
Como se aprecia en los gráficos, todas siguen una buena tendencia lineal (además de asemejarse mucho a la verdadera recta de regresión), exceptuando quizás el último, para la variable Income, que pierde la linealidad en el extremo derecho y podría explicar “peor” para puntuaciones más grandes. No obstante, los tests al llevar a cabo la asociación general de variables explicativas en nuestro modelo de regresión denotan buen comportamiento lineal y explicativo.

2.5 Analizando las observaciones extremas:

En la normal-QQ y el residuals-fitted values ya teníamos signos de outliers. Tenemos observaciones influyentes respecto de la variable respuesta y respecto las covariables (ver código), resumimos aquí los outliers: 208, 298, 357, 358, 248, 43 y 311. Utilizando el valor crítico de Bonferroni averiguamos que la observación 358 es influyente, algo que podemos comprobar con el comando outlierTest, que nos devuelve:

```
No Studentized residuals with Bonferroni p < 0.05
Largest |rstudent|:
      rstudent unadjusted p-value Bonferroni p
358   3.34075      0.00091592    0.36637
```

Si hacemos un plot de diagnóstico:



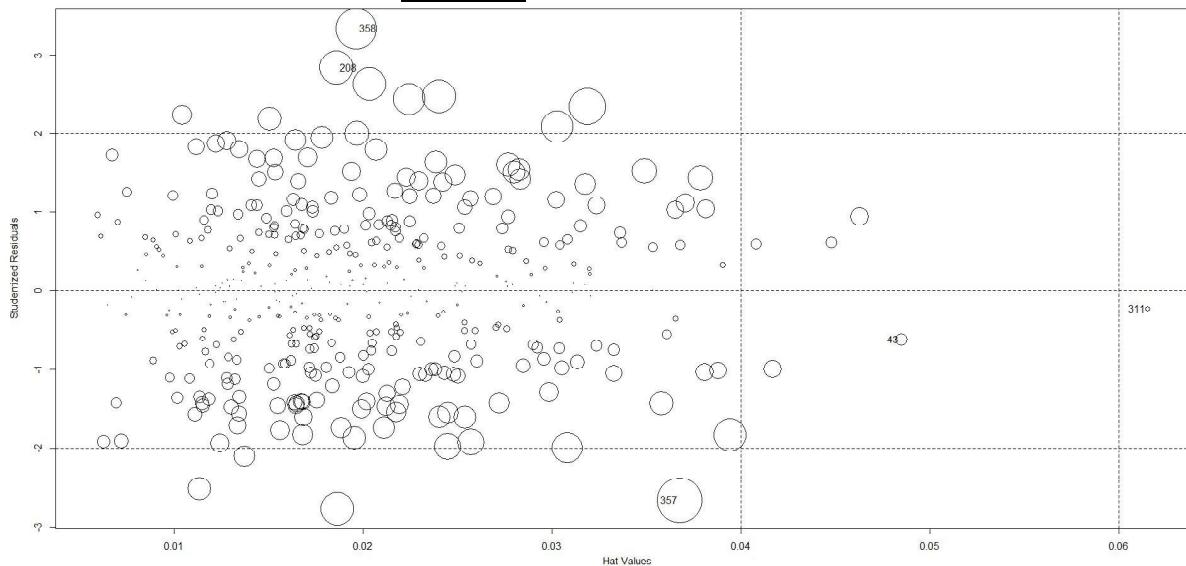
La distancia de Cook asegura que las observaciones 357 y 358 son influyentes y no deberían extraerse del modelo (en el modelo 2.1 veremos lo que ocurre al extraer estas dos observaciones)

y los cambios que se producirían). El plot de Bonferroni nos muestra que la observación 358 es influyente, tal y como habíamos estudiado analíticamente.

Las observaciones 43 y 311 tienen un valor hat alto, pero que tengan un gran leverage no significa que sean influyentes, luego habría que estudiar el caso en que eliminamos las observaciones 43 y 311 del modelo para saber si son influyentes.

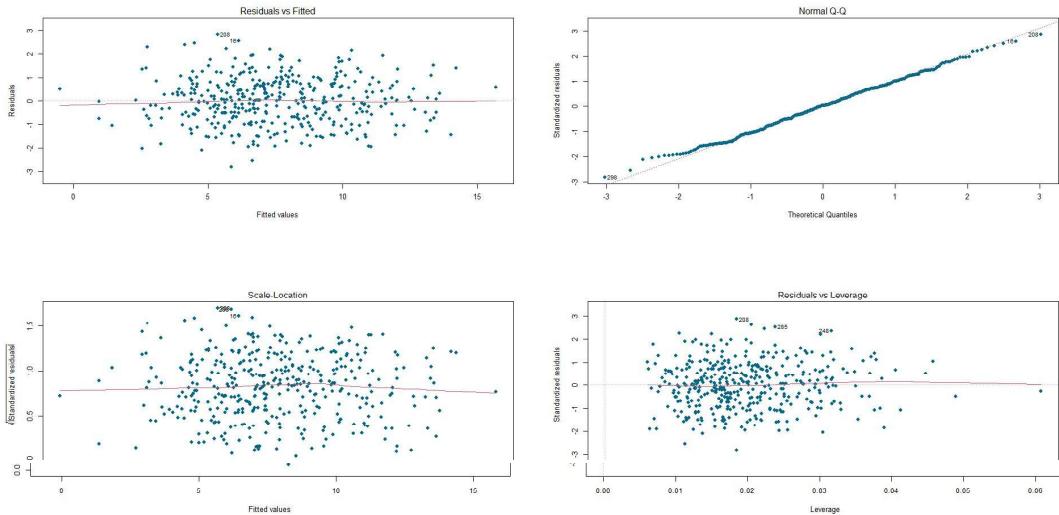
En el siguiente plot vemos los *residuos studentizados vs hat-values*. Todas aquellas observaciones con un residuo studentizado y un hat-value muy extremos, serán influyentes en el modelo: 357, 358 y 208 son ejemplos de observaciones influyentes.

En cambio, la observación 311 parece más un outlier sin mucha influencia, junto con la observación 43. En la discusión del modelo 2.3 eliminaremos estas dos observaciones.



3.MODELO 2.1 ESTUDIADO:

Modelo 2.1: Quitamos las observaciones influyentes 357 y 358. ¡OJO! Es solo para medir la capacidad que tienen estas variables de alterar el modelo, pero NO deben eliminarse las observaciones significativas. Lo incluimos en la tabla final para facilitar la comparativa, pero no seleccionaremos este modelo en ningún momento.



En comparación con el **Modelo 2**, se alteran bastante algunos supuestos:

En los anteriores gráficos se tiene:

- **Residuals vs Fitted**: La linealidad (y homocedasticidad, aunque se confirma mejor con la gráfica **Scale-Location**) parece que se altera un poco. Lo que confirman las pruebas siguientes (en comparación con el **modelo 2**):
 - **Test de Braunsch-Pagan**: Tenemos un p-valor 0.4135 mayor que 0.05, luego aceptamos la hipótesis nula: los residuos tienen varianza constante.
 - **Test de varianza no constante**: El p-valor es 0.50195, que es mayor a 0.05. Luego aceptaríamos homocedasticidad también.

En resumen, se pierde “fuerza” en la homocedasticidad:

	Modelo_2	Modelo_2.1
<u>Test de Braunsch-Pagan</u>	0.6914	0.4135
<u>Test de varianza no constante</u>	0.67408	0.50195

- **Normal Q-Q**: Los puntos se encuentran bastante bien ubicados sobre la recta, veamos analíticamente cómo de bien se acoplan sobre la recta en comparación con el **modelo 2**.

Utilizaremos dos test para analizar la normalidad de los residuos:

- **Test de Shapiro-Wilk**: Obtenemos un p-valor de $0.6029 > 0.05$, luego aceptamos la normalidad de los residuos.
- **Test de Kolmogorov-Smirnov**: Obtenemos un p-valor de $0.897 > 0.05$, luego aceptamos la normalidad de los residuos holgadamente.

En resumen, se pierde ”fuerza” en la normalidad:

	Modelo_2	Modelo_2.1
<u>Test de Shapiro-Wilk</u>	0.7439	0.6029
<u>Test de Kolmogorov-Smirnov</u>	0.9349	0.897

En comparación con el modelo 2, sí que mejora el C_p de Mallows y la R_{aj}^2 . Si hacemos un summary del modelo, se obtiene:

```

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 5.190661  0.498875 10.405 <2e-16 ***
ShelveLocGood 4.865046  0.149822 32.472 <2e-16 ***
ShelveLocMedium 1.933402  0.122888 15.733 <2e-16 ***
Price        -0.094081  0.002631 -35.755 <2e-16 ***
CompPrice     0.092946  0.004039 23.013 <2e-16 ***
Advertising   0.115181  0.007581 15.193 <2e-16 ***
Age          -0.044726  0.003127 -14.301 <2e-16 ***
Income        0.016175  0.001806  8.957 <2e-16 ***
---
Signif. codes:  0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9982 on 390 degrees of freedom
Multiple R-squared:  0.8759,    Adjusted R-squared:  0.8737 
F-statistic: 393.4 on 7 and 390 DF,  p-value: < 2.2e-16

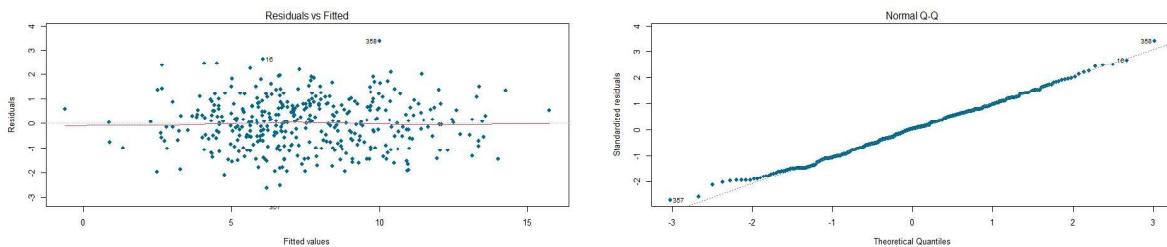
```

En resumen,

- El $R_{ajustado}^2$ da una explicabilidad de 87.37%.
- El AIC: Tiene un valor de 1137.96. Un valor muchísimo más elevado que en el anterior modelo.
- El BIC: Tiene un valor de 1173.838. Un valor muy elevado también.
- El C_p Mallows devolvería 5.55, que es inferior al número de parámetros. Lo que sugiere un modelo bien ajustado.

4. MODELO 2.2 ESTUDIADO:

Modelo 2.2: Quitamos los outliers 208 y 298. Analicemos como siempre los primeros plots siguientes:



En comparación con el Modelo 2, se alteran un poco algunos supuestos:

En los siguientes gráficos se tiene:

- Residuals vs Fitted: La linealidad y homocedasticidad mejoran. Veamos cómo cambia la homocedasticidad con las pruebas siguientes (en comparación con el modelo 2):
 - Test de Braunch-Pagan: Tenemos un p-valor 0.9349 mayor que 0.05, luego aceptamos holgadamente la hipótesis nula: los residuos tienen varianza constante.
 - Test de varianza no constante: El p-valor es 0.96548, que es mayor a 0.05. Luego aceptaríamos sobradamente la homocedasticidad también.

En resumen, se gana “fuerza” en la homocedasticidad:

	Modelo_2	Modelo_2.2
<u>Test de Braunch-Pagan</u>	0.6914	0.9349
<u>Test de varianza no constante</u>	0.67408	0.96548

- Normal Q-Q: Los puntos se encuentran bastante bien ubicados sobre la recta, de hecho, mejora significativamente en la cola izquierda y derecha con respecto al modelo 2. Veamos analíticamente cómo de bien se acoplan sobre la recta en comparación con el modelo 2.

Utilizaremos dos test para analizar la normalidad de los residuos:

- Test de Shapiro-Wilk: Obtenemos un p-valor de $0.6029 > 0.05$, luego aceptamos la normalidad de los residuos.
- Test de Kolmogorov-Smirnov: Obtenemos un p-valor de $0.897 > 0.05$, luego aceptamos la normalidad de los residuos holgadamente.

En resumen, se gana ”fuerza” en la normalidad:

	Modelo_2	Modelo_2.2
<u>Test de Shapiro-Wilk</u>	0.7439	0.6305
<u>Test de Kolmogorov-Smirnov</u>	0.9349	0.9862

En comparación con el modelo 2, sí que mejora el C_p de Mallows y la R_{aj}^2 . Si hacemos un summary del modelo, se obtiene:

```

coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 5.425401  0.496612 10.925  <2e-16 ***
ShelfLocGood 4.832372  0.150576 32.093  <2e-16 ***
ShelfLocMedium 1.949837  0.124112 15.710  <2e-16 ***
Price        -0.095342  0.002625 -36.327  <2e-16 ***
CompPrice     0.092845  0.004054 22.904  <2e-16 ***
Advertising   0.118075  0.007611 15.515  <2e-16 ***
Age          -0.045733  0.003131 -14.608  <2e-16 ***
Income        0.015559  0.001809  8.599  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.002 on 390 degrees of freedom
Multiple R-squared:  0.8762,    Adjusted R-squared:  0.874
F-statistic: 394.3 on 7 and 390 DF,  p-value: < 2.2e-16

```

En resumen,

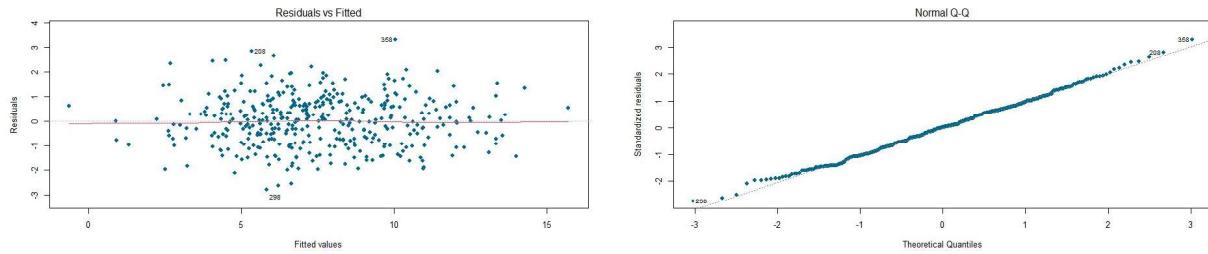
- El $R^2_{ajustado}$ da una explicabilidad de 87.4%.
- El AIC: Tiene un valor de 1140.931. Un valor muchísimo más elevado que en el modelo 2.
- El BIC: Tiene un valor de 1176.809. Un valor muy elevado también.
- El C_p Mallows se mantiene en 5.55, que es inferior al número de parámetros. Lo que sugiere un modelo bien ajustado.

Aunque este modelo mejore en un 1% la explicabilidad, nos interesa también que tenga una buena capacidad predictiva y los valores tan elevados de AIC y BIC no ayudan.

Insistimos en que solo queríamos ver la influencia de eliminar esas observaciones, por eso no nos explayamos más en el análisis de este modelo (autocorrelación, linealidad, validación, predicción, etc.)

5.MODELO 2.3 ESTUDIADO

Modelo 2.3: Quitamos los outliers 43 y 211. Analicemos como siempre los primeros gráficos siguientes:



En comparación con el Modelo 2, se tiene:

- Residuals vs Fitted: La linealidad y homocedasticidad parece que se mantienen. Veamos cómo cambia la homocedasticidad con las pruebas siguientes (en comparación con el modelo 2):
 - Test de Braunch-Pagan: Tenemos un p-valor 0.6918 mayor que 0.05, luego aceptamos holgadamente la hipótesis nula: los residuos tienen varianza constante.
 - Test de varianza no constante: El p-valor es 0.6638, que es mayor a 0.05. Luego aceptaríamos sobradamente la homocedasticidad también.

En resumen, no se notan muchos cambios en la homocedasticidad:

	Modelo_2	Modelo_2.3
<u>Test de Braunch-Pagan</u>	0.6914	0.6918
<u>Test de varianza no constante</u>	0.67408	0.6638

- Normal Q-Q: Los puntos se encuentran bastante bien ubicados sobre la recta, de hecho, no se notan grandes cambios en la normalidad con respecto al modelo 2. Veámoslo analíticamente en comparación con el modelo 2.

Utilizaremos dos test para analizar la normalidad de los residuos:

- Test de Shapiro-Wilk: Obtenemos un p-valor de $0.7243 > 0.05$, luego aceptamos la normalidad de los residuos.
- Test de Kolmogorov-Smirnov: Obtenemos un p-valor de $0.9209 > 0.05$, luego aceptamos la normalidad de los residuos también holgadamente.

En resumen, no se notan muchos cambios en la normalidad:

	Modelo_2	Modelo_2.3
<u>Test de Shapiro-Wilk</u>	0.7439	0.7243
<u>Test de Kolmogorov-Smirnov</u>	0.9349	0.9209

En comparación con el modelo 2, sí que mejora el C_p de Mallows y la R^2_{aj} . Si hacemos un summary del modelo, se obtiene:

```

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 5.517150  0.512677 10.761 < 2e-16 ***
ShelveLocGood 4.840492  0.153199 31.596 < 2e-16 ***
ShelveLocMedium 1.959171  0.126235 15.520 < 2e-16 ***
Price        -0.095550  0.002698 -35.421 < 2e-16 ***
CompPrice     0.092437  0.004139 22.332 < 2e-16 ***
Advertising   0.115760  0.007750 14.937 < 2e-16 ***
Age           -0.046185  0.003185 -14.499 < 2e-16 ***
Income         0.015807  0.001844   8.572 2.4e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.021 on 390 degrees of freedom
Multiple R-squared:  0.8714,    Adjusted R-squared:  0.8691 
F-statistic: 377.5 on 7 and 390 DF,  p-value: < 2.2e-16

```

En resumen,

- El $R^2_{ajustado}$ da una explicabilidad de 86.9%.
- El AIC: Tiene un valor de 1156.251. Un valor también muchísimo más elevado que en el modelo 2.
- El BIC: Tiene un valor de 1192.13. Un valor muy elevado también.
- El C_p Mallows se mantiene en 5.56, que es inferior al número de parámetros. Lo que sugiere un modelo bien ajustado.

Este modelo no mejora mucho el modelo 2, nos interesa también que tenga una buena capacidad predictiva y los valores tan elevados de AIC y BIC no ayudan. Lo único que mejora el C_p Mallows.

- Independencia de residuos, ¿hay autocorrelación?:
El valor de DW es cercano a 2 y el p-valor es superior a 0.05, luego aceptamos que no hay autocorrelación.

```

Durbin-watson test

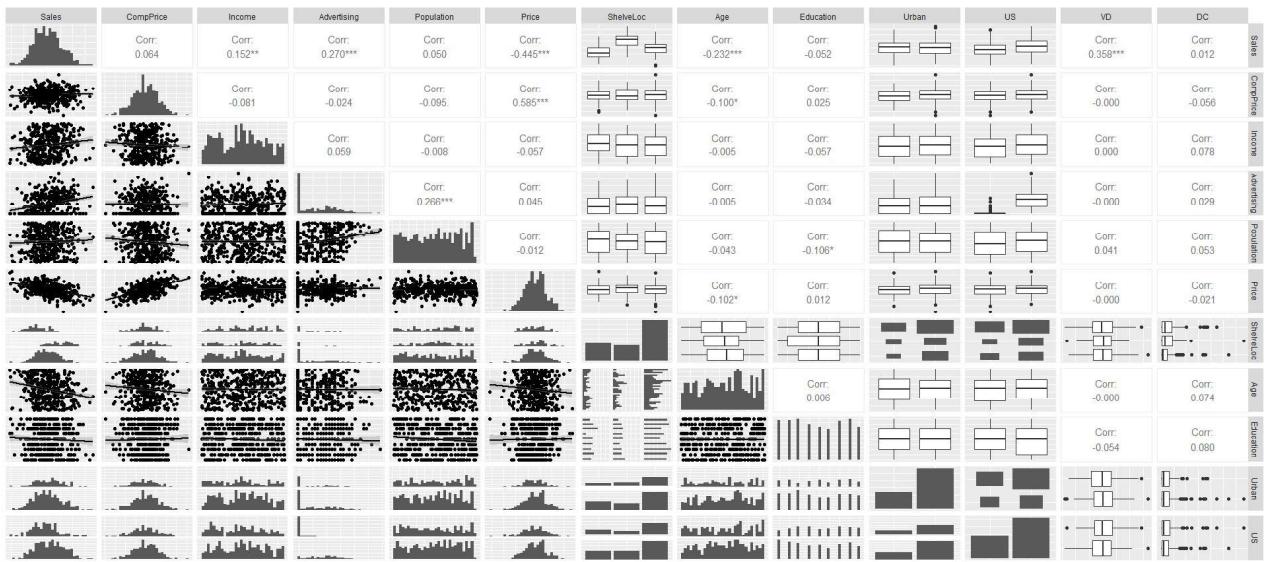
data: Carseats_obs_out_reducido_2
DW = 1.9976, p-value = 0.4902
alternative hypothesis: true autocorrelation is greater than 0

```

Con respecto al modelo 2, sí que mejora un poco el p-valor en el test.

- Linealidad: La media de los residuos debería ser cero o casi cero, si hacemos `mean(Carseats_obs_out_reducido_residuos_2)` nos devuelve $5.834744e^{-17}$. Luego es un muy buen valor, aunque menor al obtenido en el modelo 2.

Retomando el modelo 2, si hacemos un `ggpairs(modelo_completo)`, obtenemos lo siguiente:



Es un plot muy completo, en la **diagonal** podemos ver los histogramas para cada covariable y así tener una idea de la distribución de cada una de ellas antes de analizarlas globalmente, esto podría sugerirnos de antemano si hay sospechas de la necesidad de llevar a cabo una transformación BoxCox.

En la **parte inferior izquierda**, se muestran los gráficos de dispersión que sirven para tener una idea de qué variables dependientes pueden estar linealmente más asociadas con la variable respuesta Sales. Se observa que Population y Education podrían ser las que menos estén linealmente relacionadas con Sales.

En cualquier caso, lo que nos interesa de aquí es la correlación entre las covariables, que se ubica en la **parte superior derecha**. En general se observa una correlación más bien baja, donde la correlación “más alta” se da entre Price y CompPrice, con un valor de 0.585, que se considera una correlación moderada.

Veamos qué ocurre al eliminar Price o CompPrice del modelo 2:

6. MODELO 2.4 ESTUDIADO

Modelo 2.4: Quitamos la variable cuantitativa CompPrice:

Al hacer un `summary()`, se obtiene:

```

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 13.400594  0.545299 24.575 < 2e-16 ***
ShelveLocGood 4.875602  0.230253 21.175 < 2e-16 ***
ShelveLocMedium 2.004566  0.189280 10.590 < 2e-16 ***
Price        -0.060559  0.003285 -18.436 < 2e-16 ***
Advertising   0.105665  0.011642  9.076 < 2e-16 ***
Age          -0.049826  0.004790 -10.401 < 2e-16 ***
Income         0.013550  0.002771  4.891 1.47e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.539 on 393 degrees of freedom
Multiple R-squared:  0.7074, Adjusted R-squared:  0.7029 
F-statistic: 158.3 on 6 and 393 DF,  p-value: < 2.2e-16

```

- El R^2 : Tenemos una explicabilidad del 70.29%, que es un valor óptimo.
- El error estándar residual (RSE): En la columna Std. Error se aprecia que son todos los errores son relativamente pequeños y que el valor de Residual standard error del modelo completo es 1.539, unos 0.5 puntos más superiores a los modelos que tenemos hasta ahora.
- El F-statistic: En nuestro caso el valor decae a 158.3.
- El p-valor: El modelo es estadísticamente significativo al ser $p - value = 2.2 \cdot e^{-1} < 0.05$.

En comparación con el Modelo 2, se alteran, de nuevo, algunos supuestos:

- En cuanto a la homocedasticidad:
 - Test de Braunsch-Pagan: Tenemos un p-valor 0.9625 mayor que 0.05, luego aceptamos sobradamente la hipótesis nula: los residuos tienen varianza constante.
 - Test de varianza no constante: El p-valor es 0.8889, que es mayor a 0.05. Luego aceptaríamos homocedasticidad también.

En resumen, se gana mucha fuerza en la homocedasticidad:

	Modelo_2	Modelo_2.4
<u>Test de Braunsch-Pagan</u>	0.6914	0.9625
<u>Test de varianza no constante</u>	0.67408	0.8889

- Normal Q-Q: Veamos analíticamente cómo de bien se acoplan sobre la recta en comparación con el modelo 2.

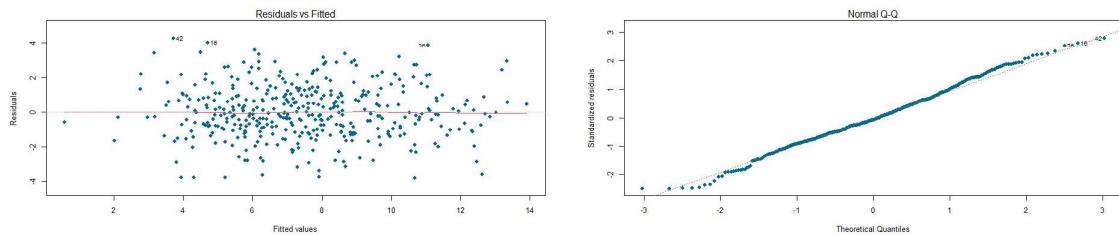
Utilizaremos dos test para analizar la normalidad de los residuos:

- Test de Shapiro-Wilk: Obtenemos un p-valor de $0.1717 > 0.05$, luego aceptamos la normalidad de los residuos.
- Test de Kolmogorov-Smirnov: Obtenemos un p-valor de $0.897 > 0.05$, luego no aceptamos la normalidad de los residuos.

En resumen, se pierde mucha fuerza en la normalidad:

	Modelo_2	Modelo_2.4
<u>Test de Shapiro-Wilk</u>	0.7439	0.1717
<u>Test de Kolmogorov-Smirnov</u>	0.9349	0.897

Veamos un dos plots para tener una visualización gráfica de lo anterior:



En el primer gráfico, los puntos se distribuyen de forma aleatoria alrededor de la línea horizontal que marca un residuo nulo. Además, la línea roja se mantiene casi en línea casi horizontal, luego al aumentar los valores ajustados, no aumenta el valor de los residuos. Esto es muy buen indicio para la linealidad. En cuanto a la homocedasticidad, en la gráfica Scale-Location nos ha dado muy buenos resultados, más visibles en los test.

En el segundo plot, se pierde mucha normalidad en comparación con el modelo 2. Se produce en el lado derecho un abombamiento bastante notable, aunque tienda a ajustarse a la línea punteada.

- Independencia de residuos, ¿hay autocorrelación?:
El valor de DW es cercano a 2 y el p-valor es superior a 0.05, luego aceptamos la hipótesis alternativa: no hay autocorrelación.

```
Durbin-Watson test
data: modelo_reducido_final_pruebas_1
DW = 1.9296, p-value = 0.4789
alternative hypothesis: true autocorrelation is not 0
```

- Multicolinealidad: Utilizamos el VIF como medidor de la colinealidad, lo más óptimo sería $0 < VIF < 1$. Se observa que superan por muy poco los umbrales por lo que debe existir una colinealidad muy escasa:

	GVIF	DF	$GVIF^{(1/(2*DF))}$
shelveLoc	1.014783	2	1.003676
Price	1.018505	1	1.009210
Advertising	1.009404	1	1.004691
Age	1.014095	1	1.007023
Income	1.012469	1	1.006215

En comparación con el modelo 2, se tiene:

- El $R^2_{ajustado}$ da una explicabilidad de 70.29%.
- El AIC: Tiene un valor de 1489.163. Un valor muy elevado.

- El BIC: Tiene un valor de 1521.094. Un valor muy elevado también.
- El C_p Mallows devolvería 37133044, que es muy superior al número de parámetros. Sugiere un modelo mal ajustado.

7. MODELO 2.5 ESTUDIADO:

Modelo 2.5: Quitamos la variable cuantitativa Price. Pero al quitar dicha variable resulta que CompPrice no es significativa en el modelo al ser su p-valor $0.297 > 0.05$, luego también hay que eliminarla.

Al hacer un summary() de este modelo, se obtiene:

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 5.854636  0.491421 11.914 < 2e-16 ***
ShelveLocGood 4.675320  0.313686 14.904 < 2e-16 ***
ShelveLocMedium 1.919267  0.258076  7.437 6.50e-13 ***
Advertising    0.096022  0.015862  6.053 3.30e-09 ***
Age           -0.040744  0.006499 -6.270 9.52e-10 ***
Income         0.016500  0.003773  4.374 1.57e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.099 on 394 degrees of freedom
Multiple R-squared:  0.4543, Adjusted R-squared:  0.4474 
F-statistic: 65.6 on 5 and 394 DF,  p-value: < 2.2e-16
```

- R^2 : Tenemos una explicabilidad del 44.74%, que no es un valor óptimo.
- El error estándar residual (RSE): En la columna Std. Error se aprecia que son todos los errores son relativamente pequeños y que el valor de Residual standard error del modelo completo es el más alto hasta ahora, con un valor de 2.099.
- F -statistic: En nuestro caso el valor decae a 65.6. También el valor más bajo hasta ahora. Nos interesa que el F -statistic sea grande.
- p -valor: El modelo es estadísticamente significativo al ser $p - value = 2.2 \cdot e^{-16} < 0.05$.

En comparación con el Modelo 2, se alteran, de nuevo, algunos supuestos:

En los siguientes gráficos se tiene:

- En cuanto a la homocedasticidad:
 - Test de Braunch-Pagan: Tenemos un p-valor 0.2165 mayor que 0.05, luego aceptamos la hipótesis nula: los residuos tienen varianza constante.
 - Test de varianza no constante: El p-valor es 0.56871, que es mayor a 0.05. Luego aceptaríamos homocedasticidad también.

En resumen, se pierde mucha fuerza en la homocedasticidad:

Modelo_2	Modelo_2.5
<u>Test de Braunch-Pagan</u>	0.6914
	0.2165

<u>Test de varianza no constante</u>	0.67408	0.56871
--------------------------------------	---------	---------

- Normal Q-Q: Veamos analíticamente cómo de bien se acoplan sobre la recta en comparación con el modelo 2.

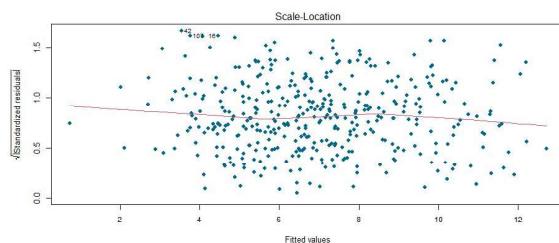
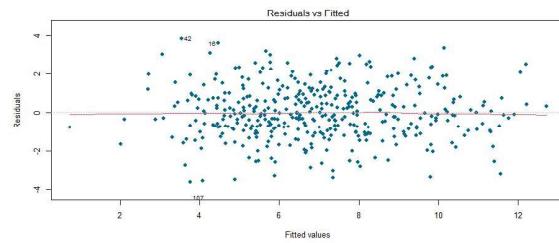
Utilizaremos dos test para analizar la normalidad de los residuos:

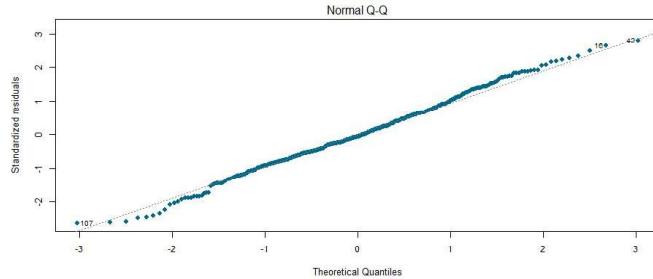
- Test de Shapiro-Wilk: Obtenemos un p-valor de $0.6702 > 0.05$, luego aceptamos la normalidad de los residuos.
- Test de Kolmogorov-Smirnov: Obtenemos un p-valor de $2.01e^{-12} > 0.05$, luego no aceptamos la normalidad de los residuos.

En resumen, se pierde mucha fuerza en la normalidad:

	Modelo_2	Modelo_2.5
<u>Test de Shapiro-Wilk</u>	0.7439	0.6702
<u>Test de Kolmogorov-Smirnov</u>	0.9349	$2.01e^{-12}$

Veamos un dos plots para tener una visualización gráfica de lo anterior:





En el primer plot, los puntos se distribuyen de forma aleatoria alrededor de la línea horizontal que marca un residuo nulo, esta línea roja se mantiene prácticamente casi horizontal, lo que es muy buena señal para la linealidad. En el segundo plot se observa un poco de varianza, pero igualmente los puntos se encuentran bien distribuidos en la parte superior e inferior de la gráfica, luego visualmente tenemos homocedasticidad en el modelo, aunque la homocedasticidad será de menor calidad que en el modelo 2, algo que ya demostraron también los test.

En el tercer plot, se pierde mucha normalidad en comparación con el modelo 2. Se produce también en la cola derecha un abombamiento bastante notable, aunque tiende a ajustarse a la línea punteada. Pero los test no afirman con suficiente contundencia la normalidad. Por lo que habrá que llevar a cabo una transformación. No obstante, creemos que sí que hay normalidad visualmente.

	R^2_{aj}	F-statistic	p-valor	Res. Standard Error	AIC	BIC	C_p Mallows
Modelo_1	1	$9.47e + 06$	$2.2e^{-16}$	0.004899	-3103.058	-3039.195	13
Modelo_2	0.8696	381.4	$2.2e^{-16}$	1.019	23.32	-774.30360	$6.3 > p = 6$
Modelo_2.1	0.8737	393.4	$2.2e^{-16}$	0.9982	1137.96	1173.838	$5.55 < 6$
Modelo_2.2	0.874	394.3	$2.2e^{-16}$	1.002	1140.931	1176.809	$5.55 < 6$
Modelo_2.3	0.8691	377.5	$2.2e^{-16}$	1.0021	1156.251	1192.13	$5.56 < 6$
Modelo_2.4	0.7029	158.3	$2.2e^{-16}$	1.539	1489.163	1521.094	37133044
Modelo_2.5	0.4474	65.6	$2.2e^{-16}$	2.099			69248967
Modelo_3							
Modelo_4							

Si lleváramos a cabo dicha transformación, no sería suficiente para afirmar la normalidad de los residuos por el test de Kolmogórov-Smirnov, cuyo p-valor mejora a 0.009942, pero sigue siendo menor que 0.05. Se ha intentado eliminar algunos outliers para conseguir la normalidad por el test de Kolmogórov-Smirnov, pero sin éxito.

8. TABLA RESUMEN DE LOS MODELOS ESTUDIADOS:

No obstante, el **modelo 2** es muy explicativo, con un número de explicativas no excesivo y con *AIC* y *BIC* más tolerables, que serán útiles para la capacidad predictiva del modelo. No solo queremos explicabilidad, sino también una buena capacidad de predicción. El objetivo es llegar a un compromiso entre explicabilidad y capacidad predictiva.

9. VALIDACION DEL MODELO DEFINITIVO:

MODELO DEFINITIVO. **MODELO 2:**

Nuestro modelo es Carseats, con variable respuesta Sales y con 13 variables explicativas. Seleccionamos el **modelo 2**, con 6 variables explicativas:

- 5 variables cuantitativas: Price, CompPrice, Advertising, Age e Income.
- 1 variable explicativa: ShelveLoc.

Sabemos que este modelo es estadísticamente significativo porque todas sus variables son significativas al tener su p-valor menor que 0.05, al igual que el del modelo. De entre las posibilidades de incluir predictores cualitativos, nos quedamos solo con ShelveLoc. El tratamiento de dicho predictor cualitativo se discute en el código, el cual aumenta la explicabilidad del modelo y es influyente sobre la variable respuesta Sales, aumentando su intercepto. Líneas del código 1099 hasta 1435.

Ahora ya con el modelo definitivo, procedemos a su **validación**:

9.1 VALIDACION DEL MODELO CON LAS VARIABLES SELECCIONADAS EN DICHO MODELO.

El primer paso que vamos a realizar para proceder a su validación, es calcular el error cuadrático medio de predicción (*MSPE*) del modelo seleccionado:

9.1.1.CALCULO DE MSPE:

Ajustamos el modelo en el conjunto de entrenamiento y el modelo ajustado lo utilizamos para predecir las respuestas de las observaciones en el conjunto de validación. Dividimos de forma aleatoria los datos en conjunto de entrenamiento (train) y test. En nuestro caso, el 70% de los datos van al conjunto de entrenamiento y el 30% al conjunto de test.

Utilizamos la función regsubsets() con Sales la variable de salida y ShelveLoc, Price, CompPrice, Advertising, Age e Income las variables explicativas. Observamos, que ejecutando el código descrito en la imagen, nos aparece la lista de errores del conjunto de validación de cada modelo de los obtenidos antes con regsubsets(). El menor error de validación es 1,115, que es un error bastante aceptable.

The screenshot shows the RStudio interface with several tabs open at the top: TRABAJO MEST.R, PCualitativos.R, RLM1.R, Entrada_1_definitiva.R, trabajo mestR, Entrada_2_definitiva.R. The main area displays R code for selecting models and calculating validation errors. A red box highlights the last few lines of the console output:

```

> val.errors
[1] 5.998933 4.270787 2.849702 2.646947 2.220623
[6] 1.333992 1.115834

```

Para saber qué modelo tiene menor error cuadrático medio y por tanto una mejor aproximación al conjunto de validación, calculamos los coeficientes del modelo con menor error de validación:

```

> coef(model.exh, which.min(val.errors))
   (Intercept) ShelfLocGood ShelfLocMedium Price CompPrice Advertising
   5.40197712    4.90279302     2.07681522 -0.09334074  0.09250221    0.10222908
      Age           Income
      -0.05009512    0.01689838
  
```

Observamos, que el modelo con menor error de validación, es el modelo que hemos seleccionado previamente como mejor modelo de regresión de nuestro conjunto de datos, luego, de momento, se confirma nuestra hipótesis.

9.1.2 LOOCV:

Ahora, vamos a utilizar la validación cruzada (*LOOCV*), para ver si también, por este procedimiento el mejor modelo es el que hemos seleccionado.

The screenshot shows the RStudio interface with the following details:

- Code Editor:** Displays R code for a reduced model (`modelo_reducido`) and a cross-validation loop. The loop iterates over 5 folds, fits a regression model on 6 folds, and calculates the mean squared error for each fold. The last line of the code is a comment about using cross-validation to select the best model.
- Console:** Shows the execution of the R code. The output includes the mean cross-validation errors for each fold (1 through 7) and the overall mean cross-validation error (6.023830).

```

1531 modelo_reducido <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age + Income, data = cseats)
1532 summary(modelo_reducido)
1533 
1534 n <- nrow(cseats)
1535 k <- n #número de grupos igual a n
1536 set.seed(5)
1537 folds <- sample(x=1:k, size =nrow(cseats), replace = FALSE)
1538 cv.errors <- matrix(NA, k, 7, dimnames = list(NULL,paste(1:7)))
1539 for (j in 1:k){
1540   best.fit <- regsubsets(Sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income, data=cseats[folds != j,])#cojemos datos del conjunto de entrenamiento
1541   for (i in 1:7){
1542     pred <- predict.regsubsets(best.fit, newdata=cseats[folds==j,], id=i)#datos de test
1543     cv.errors[j,i] <- mean((cseats$Sales[folds == j]-pred)^2)
1544   }
1545 }
1546 mean.cv.errors <- apply(cv.errors, 2, mean)#calcula la media de los betas_i
1547 mean.cv.errors
1548 coef(regfit.best, which.min(mean.cv.errors))
1549 
1550 #x3) Usando validación cruzada con 5 grupos elegir el mejor modelo de regresión de
1551 
1547:15 (Top Level) 

```

```

> for (j in 1:k){
+   best.fit <- regsubsets(Sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income, data=cseats[folds != j,])#cojemos datos del conjunto de entrenamiento
+   for (i in 1:7){
+     pred <- predict.regsubsets(best.fit, newdata=cseats[folds==j,], id=i)#datos de test
+     cv.errors[j,i] <- mean((cseats$Sales[folds == j]-pred)^2)
+   }
+ }
> mean.cv.errors <- apply(cv.errors, 2, mean)#calcula la media de los betas_i
> mean.cv.errors
 1      2      3      4      5      6      7 
6.023830 4.278404 3.000016 2.344444 1.820027 1.253935 1.060475 

```

Observamos, que el menor error de validación cruzada es 1.0604, que al igual que antes es un error bastante aceptable, indicativo de que estamos en un modelo bien ajustado.

Los coeficientes del modelo de este error de validación son:

```

> coef(regfit.best, which.min(mean.cv.errors))
  (Intercept) ShelveLocGood ShelveLocMedium Price CompPrice Advertising
  5.4019712    4.90279302   2.07681522 -0.09334074   0.09250221   0.10222908
  Age          Income
 -0.05009512   0.01689838
> 

```

De nuevo, el modelo seleccionado con menor error de validación cruzada, es el modelo 2, nuestro modelo seleccionado como mejor modelo de regresión de nuestro conjunto de datos.

9.1.3 K.-Fold Cross Validation:

Por último, utilizando el método de K-fold Cross Validation:

En nuestro caso vamos a utilizar $K = 10$, es decir, nuestros datos se dividen en 10 subconjuntos, de los cuales, uno se utiliza como dato de prueba, y el resto como datos de entrenamientos:

```

## VALIDACION CRUZADA K-TODAS
1553 modelo_reducido <- lm(sales ~ ShelveLoc + Price + CompPrice + Advertising + Age + Income, data = Carseats)
1554 summary(modelo_reducido)
1555 n <- nrow(Carseats)
1556 k <- 10 #número de grupos igual a n
1557 set.seed(5)
1558 folds <- sample(x=1:k, size =nrow(Carseats), replace = FALSE)
1559 cv.errors <- matrix(NA, k, 7, dimnames = list(NULL,paste(1:7)))
1560 for (j in 1:k){
1561   best.fit <- regsubsets(sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income, data=Carseats)
1562   for (i in 1:7){
1563     pred <- predict.regsubsets(best.fit, newdata=Carseats[folds==j,], id=i)
1564     cv.errors[j,i] <- mean((Carseats$sales[folds == j]-pred)^2)
1565   }
1566 }
1567 mean.cv.errors <- apply(cv.errors, 2, mean)
1568 mean.cv.errors
1569 coef(regfit.best, which.min(mean.cv.errors))
1570 install.packages("car")
1571 library(car)
1572
1573

```

Console output:

```

> for (j in 1:k){
+   best.fit <- regsubsets(sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income, data=Carseats)
+   folds !={{j}}]
+   for (i in 1:7){
+     pred <- predict.regsubsets(best.fit, newdata=Carseats[folds=={{j}}], id=i)
+     cv.errors[{{j}},i] <- mean((Carseats$sales[folds == {{j}}]-pred)^2)
+   }
+ }
> mean.cv.errors <- apply(cv.errors, 2, mean)
> mean.cv.errors
 1    2    3    4    5    6    7
10.345575 7.236907 7.385628 6.823970 4.408228 1.931200 1.766940
>

```

Observamos que el menor error de validación es 1.766. Para ver los coeficientes del modelo al que pertenece este error:

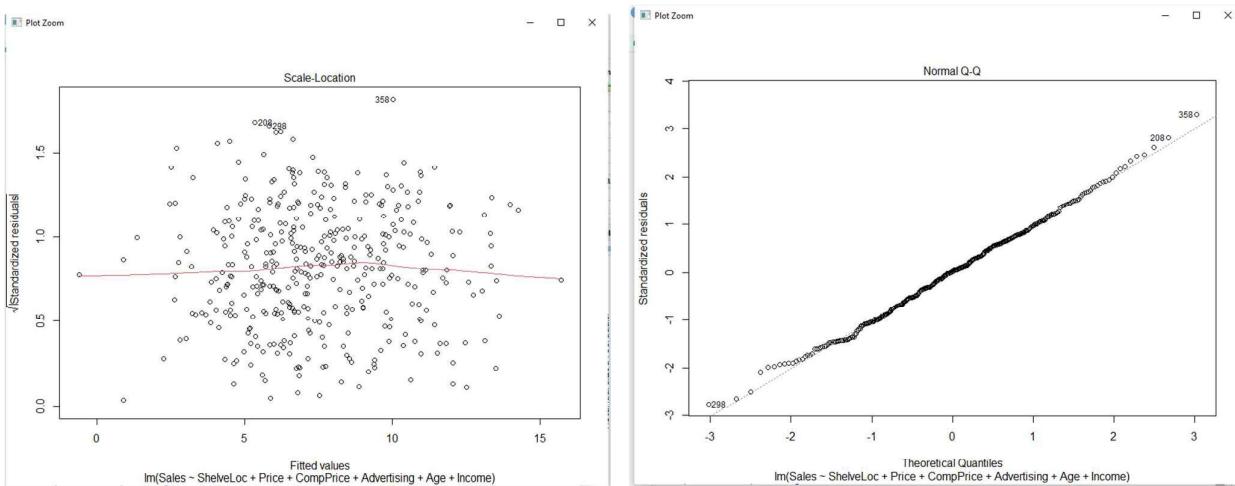
```

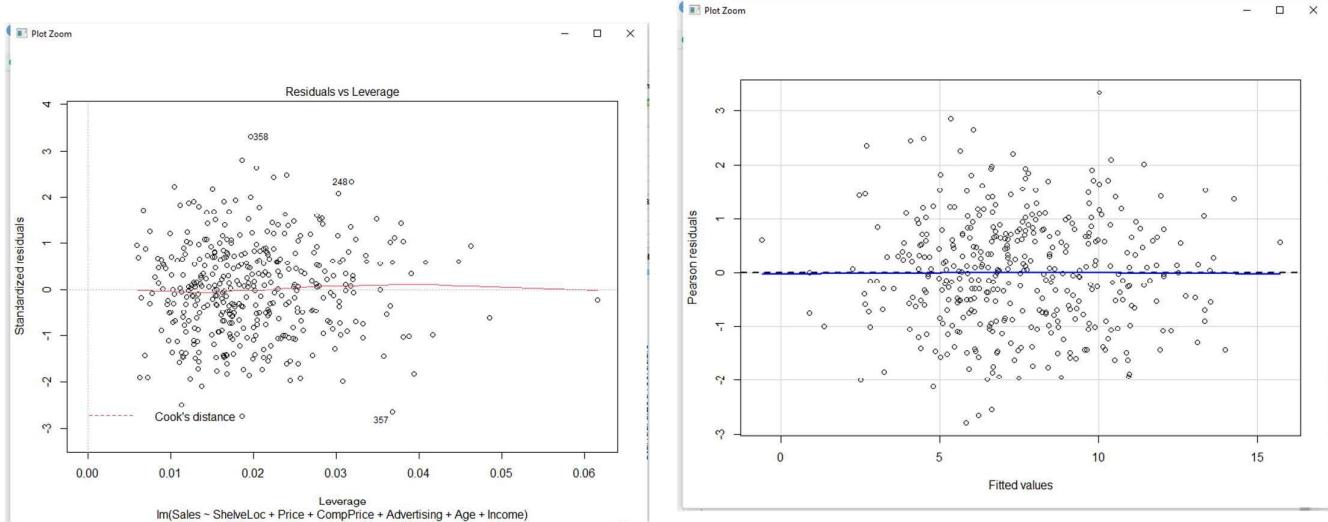
> coef(regfit.best, which.min(mean.cv.errors))
(Intercept) ShelveLocGood ShelveLocMedium      Price      CompPrice      Advertising
5.40197712     4.90279302     2.07681522    -0.09334074     0.09250221     0.10222908
Age           Income
-0.05009512      0.01689838
>

```

De nuevo, el modelo seleccionado por este procedimiento, es ¡el modelo 2!. Por tanto, este modelo, además de ser un buen modelo de regresión para nuestro conjunto de datos, es el modelo que tiene menor error, lo que nos indica, que el modelo está muy bien ajustado.

Para asegurarnos, lo comprobamos gráficamente:





Como observamos en las imágenes, el **modelo 2**, está bien ajustado, ya que los residuos se distribuyen aleatoriamente alrededor de la recta $y = 0$, y el QQplot, de la distribución normal, queda también muy bien ajustado.

Anteriormente, hemos basado la predicción en un modelo cuyas variables explicativas coincidían con nuestro modelo seleccionado en el estudio de la regresión, a continuación, lo vamos a realizar teniendo en cuenta todas las variables explicativas, y a confirmar, que efectivamente el **modelo 2**, es el que menor error tiene, y por tanto mejor aproximación al conjunto de validación.

9.2 VALIDACION DEL MODELO DE REGRESION CON TODAS LAS VARIABLES EXPLICATIVAS.

9.2.1 CALCULO MSPE:

En este caso, utilizando la función regsubsets() análogamente a como hemos hecho antes, nos salen 8 modelos posibles, con las variables que tenemos (que son todas, excepto Urban y US, que las excluimos al principio)

Ahora, análogamente a como hemos hecho anteriormente, calculamos los errores de validación de cada modelo:

```
+ }
> val.errors
[1] 5.989083 4.270787 2.849702 2.646947 2.220623 1.233992 1.115834 1.112543
```

Observamos, que el modelo 7 y 8 son los modelos con más bajo error de validación, con una diferencia de tan solo 0.0003. El modelo 8 incluye la variable no significativa Population, la cual excluimos en el estudio previo a la validación, ya que su inclusión, a pesar de provocar una pequeña ganancia de explicabilidad, añadiría ruido innecesario, por lo tanto, ignoramos la opción de incluir Population en nuestro modelo, y nos quedamos con que el mejor modelo teniendo en cuenta el menor error de validación es el modelo 7 del regsubsets(), que coincide con el modelo 2 (nuestro modelo elegido como mejor modelo de regresión).

9.2.2 METODO VALIDACION CRUZADA

Ahora, pasamos a estudiar el error de validación cruzada:

En un proceso análogo al que hemos hecho antes, nos quedan los siguientes errores de validación cruzada: El menor error es 1.0604, que coincide con el menor error hecho anteriormente para el modelo con variables explicativas solo las del modelo previamente seleccionado (modelo2)

```
> mean.cv.errors
   1      2      3      4      5      6      7      8
6.023830 4.278404 3.000016 2.344444 1.820027 1.253935 1.060475 1.085778
```

De nuevo, nos queda que el modelo con menor error de validación cruzada es el modelo 7, que coincide con nuestro modelo 2 y cuyos coeficientes son:

```
> coef( regfit.best, which.min(mean.cv.errors))
  (Intercept) CompPrice Income Advertising Price ShelveLocGood
  5.40197712  0.09250221 0.01689838  0.10222908 -0.09334074 4.90279302
ShelveLocMedium Age
  2.07681522 -0.05009512
```

9.2.3 K-Fold Cross Validation:

Finalmente, utilizando el K-fold Cross Validation, la lista de errores de validacion obtenida es:

```
> mean.cv.errors
   1      2      3      4      5      6      7      8
10.345575 7.236907 7.385628 6.823970 4.408228 1.931200 1.766940 1.800800
```

El menor error es de 1.76, del modelo 7, que coincide con nuestro modelo 2, y cuyos coeficientes son los mismos que antes.

Por tanto, con estas dos comprobaciones de validación, queda bastante evidente que el mejor modelo de regresión para nuestro conjunto de datos es el modelo 2, ya que además de realizar un buen estudio y predicción de los mismos, es el que mejor aproximación tiene al conjunto de validación.

10. PREDICCIONES:

Si hacemos una predicción puntual, tomando los siguientes valores para las variables explicativas: ShelveLoc, Price=180, CompPrice=150, Advertising=10, Age=60, Income=120.

Obtenemos dos tablas: nuevos_datos y predicción_datos para las 400 observaciones con esos valores fijos. Si tomamos las dos primeras observaciones de nuevos_datos, tenemos:

	ShelveLoc	Price	CompPrice	Advertising	Age	Income
1	Bad	180	150	10	60	120
2	Good	180	150	10	60	120

Ahora bien, si vamos a la tabla predicción_datos y tomamos las dos primeras observaciones, tenemos:

Name	Type	Value
• predicción_datos	double [400]	2.49 7.32 4.44 4.44 2.49 2.49 ...
1	double [1]	2.489114
2	double [1]	7.324789

Esto quiere decir que para la observación 1 y 2, con esos valores de las variables explicativas, tendríamos unas 2 y 7 ventas aproximadamente, respectivamente.

Intervalo de predicción para la media: Veamos cuantas ventas vamos a tener en todas aquellas variables que tengan Price=180, CompPrice=150, Advertising=10, Age=60, Income=120:

	fit	lwr	upr
1	2.489114	2.085583	2.892645
2	7.324789	6.920101	7.729477

La columna fit nos dice que, puntualmente, en promedio vamos a vender 2.489 y 7.32 para las observaciones 1 y 2, respectivamente.

Las otras dos columnas nos dan la cota superior e inferior de nuestro intervalo con un nivel de confianza al 95%. Así, en todas aquellas observaciones que ponga esos datos, voy a obtener *en promedio* para la observación 1 unas ventas de entre 2.08 y casi 3, mientras que para la observación 2 tendremos entre casi 7 y 7.7 ventas.

Intervalo de predicción: Queremos el IC para una nueva observación y no para la respuesta media. Para este intervalo utilizamos en el comando predict el intervalo “prediction”. Así, para las dos primeras observaciones (de las 400 que hay), se tiene:

	fit	lwr	upr
1	2.489114	0.4446463	4.533582
2	7.324789	5.2800924	9.369485

Como antes:

La columna fit nos da una predicción puntual (la que obtuvimos antes) significaba que, para los valores mencionados de las variables explicativas, tendremos para las observaciones 1 y 2 entonces predecimos que las ventas van a ser de 2 y 7, respectivamente.

Las otras dos columnas nos dan la cota superior e inferior de nuestro intervalo con un nivel de confianza al 95%, es decir, para la observación 1, las ventas (Sales) van a estar entre 0.44 y 4.5, y para la observación 2 estarán entre 5.28 y 9.369 ventas, a un nivel de confianza del 95%.

ANEXO:

#Paquetes necesarios para la práctica:

```
install.packages("ISLR")
library(ISLR)
install.packages("GGally") #Para ggpairs.
library(GGally)
install.packages("leaps")
library(leaps)
install.packages("lmtest")
library(lmtest)
install.packages("car")
library("car")
install.packages("corrplot")
library(corrplot)
install.packages("olsrr")
library("olsrr")

head(carseats)
attach(Carseats);
names(Carseats);
```

```
#MODELOS:
```

```
#TIPO 1: Con variables predictoras cuantitativas y cualitativas:
```

```
#Modelo 1: Completo:
```

```
#Construimos el modelo utilizando todos los predictores y la variable respuesta Sales:
```

```
RLM.Completo_1 = lm(Sales~, data=Carseats)  
summary(RLM.Completo_1)  
anova(RLM.Completo_1)
```

```
#Comparamos gráficos para cada variable dependiente y correlaciones (esto servirá para la  
discusión del
```

```
#modelo 2, para los modelos 2.3 y 2.4):
```

```
ggpairs(Carseats, lower = list(continuous = "smooth"),  
diag = list(continuous = "barDiag"), axisLabels = "none")
```

```
#VARIANZA CONSTANTE:
```

```
bptest(RLM.Completo_1) #Prueba de Braunch-Pagan.
```

```
ncvTest(RLM.Completo_1) #Non-constant Variance Score Test.
```

```
#Colinealidad:
```

```
vif(RLM.Completo_1)
```

```
#INDEPENDENCIA RESIDUOS: ¿Autocorrelación?
```

```
dwtest(RLM.Completo_1)
```

#EXTRAYENDO OUTLIERS Y NORMALIDAD GRÁFICAMENTE: residuos vs valores ajustados (fitted values), residuos estandarizados frente a leverage:

```
par(mfrow=c(2, 2))
plot(RLM.Completo_1, col='deepskyblue4', pch=19)
```

```
BC_2 <- boxCox(RLM.Completo_1, family="yjPower", plotit = TRUE)
```

```
lambda <- BC_2$x[which.max(BC_2$y)]
```

```
lambda
```

```
#####
[1] 0.989899 #El valor exacto de lambda.
```

```
#####
potencia <- yjPower(Sales, lambda)
```

```
RLM.Completo_1_transf <- lm(potencia ~ ., data = Carseats)
```

```
RLM.Completo_1_transf
```

```
summary(RLM.Completo_1_transf)
```

```
anova(RLM.Completo_1_transf)
```

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

```
plot(RLM.Completo_1_transf, col='deepskyblue4', pch=19)
```

#Modelo 2:

#MÉTODO FORWARD: Este método inicia con el modelo vacío y
#busca el mejor modelo con una, luego dos, luego tres
#variables y así sucesivamente.

```
RLM.Vacio <- lm(Sales~1, data=Carseats)  
summary(RLM.Vacio)
```

```
RLM.Forward <- step(RLM.Vacio, scope=  
  list(lower=RLM.Vacio, upper=RLM.Completo),  
  direction = "forward")  
summary(RLM.Forward)
```

#Modelo con todas las variables explicativas:

```
RLM.Completo = lm(Sales~., data=Carseats)  
summary(RLM.Completo)
```

#MÉTODO BACKWARD: Este inicia con el modelo completo.

```
RLM.Backward <- step(RLM.Completo, scope=  
  list(lower=RLM.Vacio, upper=RLM.Completo),  
  direction = "backward")  
summary(RLM.Backward)
```

#MÉTODO STEPWISE: Este inicia con el modelo vacío. Este método nos quitaría #las variables no significativas (a diferencia de los otros dos métodos).

```
RLM.Stepwise <- step(RLM.Vacio, scope=
  list(lower=RLM.Vacio, upper=RLM.Completo),
  direction = "both")
summary(RLM.Stepwise)
anova(RLM.Stepwise)
```

#Veamos qué obtenemos con regsubsets para todas las variables explicativas:

```
modelos_mejoresCompleto <- regsubsets(Sales ~ ., data=Carseats, nbest=3, nvmax=7)
summary(modelos_mejoresCompleto)$which
```

#El Cp de Mallows compara la precisión y el sesgo del modelo completo con modelos que incluyen solo algunos predictores:

```
modelo_completo <- lm(Sales ~ ., data = Carseats)
modelo_reducido <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age +
Income, data = Carseats)
summary(modelo_reducido)
anova(modelo_reducido)
ols_mallows_cp(modelo_reducido, modelo_completo)

par(mfrow=c(1, 2))
```

```

plot(modelos_mejoresCompleto, scale="adjr2", main=expression(R[Adj]^2))
plot(modelos_mejoresCompleto, scale="bic", main="BIC")

#Mejores variables guiándonos por el BIC (ANALÍTICAMENTE):
library(leaps)
summary(modelos_mejoresCompleto)$bic #Lista de todos los BIC
#####
[1] -103.36220 -76.26634 -18.17982 -235.90366 -159.18848 -134.66784 -373.71021 -
288.89418 -288.74490 -468.52957 -449.96052 -441.40171 -566.50696
[14] -559.56850 -549.98260 -711.31065 -608.13362 -607.03462 -774.30360 -707.37999 -
706.95421
#####

which(summary(modelos_mejoresCompleto)$bic ==
min(summary(modelos_mejoresCompleto)$bic)) #Tomar el modelo con menor BIC.
#####
[1] 19
#####

summary(modelos_mejoresCompleto)$which[19, ] #¿Cuál es el modelo 19?
#####
(Intercept) CompPrice Income Advertising Population Price
ShelveLocGood ShelveLocMedium Age
TRUE TRUE TRUE TRUE FALSE TRUE
TRUE TRUE TRUE
Education UrbanYes USYYes
FALSE FALSE FALSE
#####

```

#Mejores variables guiándonos por el R^2 ajustado:

```
summary(modelos_mejoresCompleto)$adjr2  
which(summary(modelos_mejoresCompleto)$adjr2 ==  
max(summary(modelos_mejoresCompleto)$adjr2)) #Tomar el modelo con mayor adjr2.  
summary(modelos_mejoresCompleto)$which[19, ]
```

#Correlación de predictores:

```
#Correlación:  
install.packages("ISLR")  
library(ISLR)  
install.packages("GGally")  
library(GGally)  
ggpairs(modelo_reducido_final, lower = list(continuous = "smooth"),  
diag = list(continuous = "barDiag"), axisLabels = "none") #REVISAR CORRELACIÓN  
COMPRICE Y PRICE.
```

#Colinealidad o multicolinealidad entre predictores:

```
modelo_reducido_final <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age +  
Income, data=Carseats)
```

#Análisis de Inflación de Varianza (VIF): Se calcula un VIF por cada variable explicativa:

```
library(car)  
vif(modelo_reducido_final)
```

#Autocorrelación con Durbin-Watson: Veamos la
INDEPENDENCIA DE ERRORES:

```
library(lmtest)  
dwtest(modelo_reducido_final, alternative = "two.sided")
```

"""

Durbin-Watson test

data: modelo_reducido

DW = 1.9882, p-value = 0.4523 #DW está lejos de 0, siendo un valor MUY cercano a 2, y el p-valor es mayor que 0.05, luego hay independencia. No hay autocorrelación entre los errores.

alternative hypothesis: true autocorrelation is greater than 0

"""

#Comprobación Hipótesis:

#VARIANZA CONSTANTE:

```
library(lmtest)  
bptest(modelo_reducido) #Prueba de Braunch-Pagan
```

"""

H_0 : Los errores tienen varianza constante (homoceodasticidad)

H_1 : Los errores tienen NO varianza constante (NO hay homoceodasticidad)

studentized Breusch-Pagan test

data: modelo_reducido

BP = 4.7418, df = 7, p-value = 0.6914 #p-valor grande, luego aceptamos la hipótesis nula: Hay homocedasticidad.

Si el p-valor es menor que 0.05, entonces hay evidencias para decir que NO se cumple la homocedasticidad de los

e_{i}.

"""

ncvTest(modelo_reducido)

"""

Non-constant Variance Score Test

Variance formula: ~ fitted.values

Chisquare = 0.1768675, Df = 1, p = 0.67408 #El p-valor es mayor que 0.05, aceptamos homocedasticidad.

"""

#NORMALIDAD DE LOS RESIDUOS:

modelo_reducido_residuos <- modelo_reducido\$residuals

modelo_reducido_residuos

shapiro.test(modelo_reducido_residuos) #Ya vimos normalidad en algunos datos en ggpairs (foto del plot grande con muchas gráficas).

"""

Shapiro-Wilk normality test

```

data: modelo_reducido_residuos
W = 0.99724, p-value = 0.7439 #El p-valor es mayor que 0.05, aceptamos normalidad de los
residuos.

"""

ks.test(modelo_reducido_residuos, pnorm)

"""

One-sample Kolmogorov-Smirnov test

data: modelo_reducido_residuos
D = 0.026871, p-value = 0.9349 #El p-valor es holgadamente mayor que 0.05, aceptamos
normalidad de los residuos.

alternative hypothesis: two-sided

"""

qqnorm(modelo_reducido_residuos) #La normalidad de los residuos implicaría normalidad en
los Y.

qqline(modelo_reducido_residuos)

#Gráficos más completo:
par(mfrow=c(2, 2))
plot(modelo_reducido, col='deepskyblue4', pch=19)

crPlots(modelo_reducido)

#Linealidad:
mean(modelo_reducido_residuos)

"""

[1] 7.729413e-17

"""

#DETECCIÓN DE OUTLIERS:

```

#Veamos cuales de los outliers son INFLUYENTES:

```
tabla_influencias <- influence.measures(modelo_reducido) #Cook D_{i}, DFITS_{i},  
DFBETAS_{i}.
```

```
#La tabla anterior proporciona una tabla de 400 observaciones, vamos a escoger de todas ellas  
#las que son outliers y, además, cumplen los criterios para ser INFLUYENTES (Cook,  
#DFITS, DFBETAS).
```

#Observaciones extremas respecto a Y=Sales:

```
modelo_reducido <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age +  
Income, data = Carseats)
```

```
plot(predict(modelo_reducido), rstudent(modelo_reducido))
```

```
Carseats$VD <- rstudent(modelo_reducido)
```

```
which(Carseats$VD >= 2*6/400) #Sacamos aquellas observaciones con alto leverage (NO  
tienen por qué ser influyentes)
```

#2a forma: Cálculo con la teoría: El valor crítico de Bonferroni es $t_{\{1-\alpha/2n;n-p-1\}}=t_{\{1-0.05/2;400-6-1\}}=1.966019$

```
CriticBonferroni <- qt(1-0.05/2,393)#valor crítico de Bonferroni. ¡OJO! ¡¡¡Inestable para n  
grande!!!!(Apuntes Rosa).
```

```
CriticBonferroni
```

```
sum(abs(rstudent(modelo_reducido))>CriticBonferroni)
```

####

```
[1] 16 #I.e, 16 valores satisfacen la desigualdad.
```

####

```
which.max(abs(rstudent(modelo_reducido)))  
"""  
358 #¿Cuál de ellos es el residuo studentizado más grande en valor absoluto? El 358  
"""
```

#Comprobación del rstudent utilizando outlierTest:

```
install.packages("car")  
library("car")
```

```
outlierTest(modelo_reducido)
```

"""

No Studentized residuals with Bonferroni p < 0.05

Largest |rstudent|:

```
rstudent unadjusted p-value Bonferroni p  
358 3.34075 0.00091592 0.36637 #La observación 358.
```

"""

#Observaciones extremas respecto a las X:

```
#La F de tablas se calcula con qt: t_{0.05;n-p}=t_{0.5;6;400-6}=411.1304  
F_tabla <- qt(0.5,6,400-6)  
F_tabla  
obs_cook <- cooks.distance(modelo_reducido)
```

```
which(obs_cook>F_tabla)

obs_dffits <- dffits(modelo_reducido)
which(abs(obs_dffits) > 2*sqrt(6/400)) #Quiero los influentes solo, aquellos que
abs(dffits)>2*sqrt(p/n)
```

```
obs_dfbetas <- dfbetas(modelo_reducido)
which(abs(obs_dfbetas) > 2/sqrt(400))
```

```
influenceIndexPlot(modelo_reducido) #Valores hat, Bonferroni, Cook, Student estandarizado.
#Ver plot_diagnostico_reducido.png
influencePlot(modelo_reducido) #VISTA ANALÍTICA DE LAS INFLUENCIAS.
ALTERNATIVA A influenceIndexPlot.
```

```
#Modelo 2.1. Quitar observaciones 357 y 358:
```

```
obs.out <- c(357,358) #observaciones a quitar
Carseats_obs_out <- Carseats[-obs.out,1:8]
```

```
Carseats_obs_out_reducido <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age  
+ Income, data = Carseats_obs_out)
```

```
par(mfrow=c(2, 2))  
plot(Carseats_obs_out_reducido, col='deepskyblue4', pch=19)
```

bptest(Carseats_obs_out_reducido) #Prueba de Braunsch-Pagan. #p-valor grande, luego aceptamos la hipótesis nula: Hay homocedasticidad.

"""

studentized Breusch-Pagan test

```
data: Carseats_obs_out_reducido  
BP = 7.1492, df = 7, p-value = 0.4135
```

"""

ncvTest(Carseats_obs_out_reducido) #p-valor grande, luego aceptamos la hipótesis nula: Hay homocedasticidad.

"""

Non-constant Variance Score Test

Variance formula: ~ fitted.values

Chisquare = 0.4508098, Df = 1, p = 0.50195

"""

```
dwtest(Carseats_obs_out_reducido)
```

"""

Durbin-Watson test

```
data: Carseats_obs_out_reducido
```

DW = 1.9341, p-value = 0.2547

alternative hypothesis: true autocorrelation is greater than 0

"""

```
par(mfrow=c(1, 2))
plot(Carseats_obs_out_reducido, scale="adjr2", main=expression(R[Adj]^2))
plot(Carseats_obs_out_reducido, scale="bic", main="BIC")

vif(Carseats_obs_out_reducido) #Los valores VIF_{j} son pequeños, luego no hay colinealidad.
Entre 0 y 1 bien, poco más de uno algo de colinealidad.
```

```
summary(Carseats_obs_out_reducido)
```

```
Carseats_obs_out_reducido_residuos <- Carseats_obs_out_reducido$residuals
Carseats_obs_out_reducido_residuos
```

```
shapiro.test(Carseats_obs_out_reducido_residuos)
```

####

Shapiro-Wilk normality test

```
data: Carseats_obs_out_reducido_residuos
```

```
W = 0.99673, p-value = 0.6029 #El p-valor es mayor que 0.05, aceptamos normalidad de los
residuos.
```

####

```
ks.test(Carseats_obs_out_reducido_residuos, pnorm)
```

####

One-sample Kolmogorov-Smirnov test

```

data: Carseats_obs_out_reducido_residuos

D = 0.022621, p-value = 0.897 #El p-valor es holgadamente mayor que 0.05, aceptamos
normalidad de los residuos.

alternative hypothesis: two-sided

"""

library("olsrr")

Carseats_obs_out_completo <- lm(Sales ~ ., data = Carseats_obs_out)

Carseats_obs_out_reducido <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age
+ Income, data = Carseats_obs_out)

ols_mallows_cp(Carseats_obs_out_reducido, Carseats_obs_out_completo)

```

"""

[1] 5.546158

"""

#Modelo 2.2. Quitar observaciones 298 y 208:

```

obs.out_1 <- c(298,208) #observaciones a quitar

Carseats_obs_out_1 <- Carseats[-obs.out_1,1:8]

Carseats_obs_out_reducido_1 <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising +
Age + Income, data = Carseats_obs_out_1)

```

```
par(mfrow=c(2, 2))
plot(Carseats_obs_out_reducido_1, col='deepskyblue4', pch=19)
```

bptest(Carseats_obs_out_reducido_1) #Prueba de Braunsch-Pagan. #p-valor grande, luego aceptamos la hipótesis nula: Hay homocedasticidad.

"""

studentized Breusch-Pagan test

data: Carseats_obs_out_reducido_1

BP = 2.3938, df = 7, p-value = 0.9349

"""

ncvTest(Carseats_obs_out_reducido_1) #p-valor grande, luego aceptamos la hipótesis nula: Hay homocedasticidad.

"""

Non-constant Variance Score Test

Variance formula: ~ fitted.values

Chisquare = 0.001873212, Df = 1, p = 0.96548

"""

```
dwtest(Carseats_obs_out_reducido_1)
```

"""

Durbin-Watson test

data: Carseats_obs_out_reducido_1

DW = 2.0224, p-value = 0.5885

alternative hypothesis: true autocorrelation is greater than 0

"""

```
par(mfrow=c(1, 2))
```

```
plot(Carseats_obs_out_reducido_1, scale="adjr2", main=expression(R[Adj]^2))
```

```
plot(Carseats_obs_out_reducido_1, scale="bic", main="BIC")
```

```
vif(Carseats_obs_out_reducido_1) #Los valores VIF_{j} son pequeños, luego no hay colinealidad. Entre 0 y 1 bien, poco más de uno algo de colinealidad.
```

```
summary(Carseats_obs_out_reducido_1)
```

```
Carseats_obs_out_reducido_residuos_1 <- Carseats_obs_out_reducido_1$residuals  
Carseats_obs_out_reducido_residuos_1
```

```
shapiro.test(Carseats_obs_out_reducido_residuos_1)
```

```
""""
```

```
Shapiro-Wilk normality test
```

```
data: Carseats_obs_out_reducido_residuos
```

```
W = 0.99647, p-value = 0.6305 #El p-valor es mayor que 0.05, aceptamos normalidad de los residuos.
```

```
""""
```

```
ks.test(Carseats_obs_out_reducido_residuos_1, pnorm)
```

```
""""
```

```
One-sample Kolmogorov-Smirnov test
```

```
data: Carseats_obs_out_reducido_residuos
```

```
D = 0.022742, p-value = 0.9862 #El p-valor es holgadamente mayor que 0.05, aceptamos normalidad de los residuos.
```

```
alternative hypothesis: two-sided
```

```
""""
```

```
library("olsrr")  
Carseats_obs_out_completo_1 <- lm(Sales ~ ., data = Carseats_obs_out_1)  
Carseats_obs_out_reducido_1 <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising +  
Age + Income, data = Carseats_obs_out_1)  
ols_mallows_cp(Carseats_obs_out_reducido_1, Carseats_obs_out_completo_1)
```

####

```
[1] 5.546158
```

####

```
AIC(Carseats_obs_out_reducido_1)
```

```
BIC(Carseats_obs_out_reducido_1)
```

#Modelo 2.3. Quitar observaciones 43 y 211:

```
obs.out_2 <- c(43,211) #observaciones a quitar  
Carseats_obs_out_2 <- Carseats[-obs.out_2,1:8]
```

```
Carseats_obs_out_reducido_2 <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising +  
Age + Income, data = Carseats_obs_out_2)
```

```
summary(Carseats_obs_out_reducido_2)
```

```
par(mfrow=c(2, 2))  
plot(Carseats_obs_out_reducido_2, col='deepskyblue4', pch=19)
```

```
bptest(Carseats_obs_out_reducido_2) #Prueba de Braunsch-Pagan. #p-valor grande, luego  
aceptamos la hipótesis nula: Hay homocedasticidad.
```

"""

```
studentized Breusch-Pagan test
```

```
data: Carseats_obs_out_reducido_2
```

```
BP = 4.7387, df = 7, p-value = 0.6918
```

"""

```
ncvTest(Carseats_obs_out_reducido_2) #p-valor grande, luego aceptamos la hipótesis nula: Hay  
homocedasticidad.
```

"""

```
Non-constant Variance Score Test
```

```
Variance formula: ~ fitted.values
```

```
Chisquare = 0.1889362, Df = 1, p = 0.6638
```

"""

```
dwtest(Carseats_obs_out_reducido_2)
```

"""

```
Durbin-Watson test
```

```
data: Carseats_obs_out_reducido_2
```

```
DW = 1.9976, p-value = 0.4902
```

```
alternative hypothesis: true autocorrelation is greater than 0
```

"""

```
par(mfrow=c(1, 2))
```

```
plot(Carseats_obs_out_reducido_2, scale="adjr2", main=expression(R[Adj]^2))
```

```
plot(Carseats_obs_out_reducido_2, scale="bic", main="BIC")
```

```
vif(Carseats_obs_out_reducido_2) #Los valores VIF_{j} son pequeños, luego no hay colinealidad. Entre 0 y 1 bien, poco más de uno algo de colinealidad.
```

```
summary(Carseats_obs_out_reducido_2)
```

```
Carseats_obs_out_reducido_residuos_2 <- Carseats_obs_out_reducido_2$residuals  
Carseats_obs_out_reducido_residuos_2
```

```
mean(Carseats_obs_out_reducido_residuos_2)
```

```
shapiro.test(Carseats_obs_out_reducido_residuos_2)
```

```
""""
```

```
Shapiro-Wilk normality test
```

```
data: Carseats_obs_out_reducido_residuos
```

```
W = 0.99716, p-value = 0.7243 #El p-valor es mayor que 0.05, aceptamos normalidad de los residuos.
```

```
""""
```

```
ks.test(Carseats_obs_out_reducido_residuos_2, pnorm)
```

```
""""
```

```
One-sample Kolmogorov-Smirnov test
```

```
data: Carseats_obs_out_reducido_residuos
```

```
D = 0.027662, p-value = 0.9209 #El p-valor es holgadamente mayor que 0.05, aceptamos normalidad de los residuos.
```

```
alternative hypothesis: two-sided
```

```
""""
```

```
library("olsrr")
Carseats_obs_out_completo_2 <- lm(Sales ~ ., data = Carseats_obs_out_2)
Carseats_obs_out_reducido_2 <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising +
Age + Income, data = Carseats_obs_out_2)
ols_mallows_cp(Carseats_obs_out_reducido_2, Carseats_obs_out_completo_2)
```

####

[1] 5.589415

####

```
AIC(Carseats_obs_out_reducido_2)
```

```
BIC(Carseats_obs_out_reducido_2)
```

```
ggpairs(Carseats, lower = list(continuous = "smooth"),
diag = list(continuous = "barDiag"), axisLabels = "none")
```

#Modelo 2.4: Quitamos CompPrice:

```
modelo_reducido_final_pruebas_1 <- lm(Sales ~ ShelveLoc + Price + Advertising + Age +  
Income, data=Carseats)  
summary(modelo_reducido_final_pruebas_1)
```

.....

Call:

```
lm(formula = Sales ~ ShelveLoc + Price + Advertising + Age +  
Income, data = Carseats)
```

Residuals:

Min	1Q	Median	3Q	Max
-3.7968	-0.9912	-0.1068	0.9670	4.2404

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	13.400594	0.545299	24.575	< 2e-16 ***
ShelveLocGood	4.875602	0.230253	21.175	< 2e-16 ***
ShelveLocMedium	2.004566	0.189280	10.590	< 2e-16 ***
Price	-0.060559	0.003285	-18.436	< 2e-16 ***
Advertising	0.105665	0.011642	9.076	< 2e-16 ***
Age	-0.049826	0.004790	-10.401	< 2e-16 ***
Income	0.013550	0.002771	4.891	1.47e-06 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.539 on 393 degrees of freedom

Multiple R-squared: 0.7074, Adjusted R-squared: 0.7029 #bajan la R^2 ajustada, la F-statistics.

F-statistic: 158.3 on 6 and 393 DF, p-value: < 2.2e-16

####

```
library("olsrr")
```

```
modelo_completo <- lm(Sales ~ ., data = Carseats)
```

```
ols_mallows_cp(modelo_reducido_final_pruebas_1, modelo_completo) #Número horrible:  
37347065. Muy mal ajustado.
```

```
AIC(modelo_reducido_final_pruebas_1)
```

```
BIC(modelo_reducido_final_pruebas_1)
```

```
library(lmtest)
```

```
dwtest(modelo_reducido_final_pruebas_1, alternative = "two.sided")
```

####

Durbin-Watson test

```
data: modelo_reducido_final_pruebas_1
```

```
DW = 1.9296, p-value = 0.4789 #NO autocorrelación :D
```

```
alternative hypothesis: true autocorrelation is not 0
```

####

#NORMALIDAD DE LOS RESIDUOS:

```
modelo_reducido_final_pruebas_residuos_1 <- modelo_reducido_final_pruebas_1$residuals  
modelo_reducido_final_pruebas_residuos_1
```

```
shapiro.test(modelo_reducido_final_pruebas_residuos_1) #El p-valor es mayor que 0.05,  
aceptamos normalidad de los residuos.
```

"""

Shapiro-Wilk normality test

```
data: modelo_reducido_final_pruebas_residuos_1
```

```
W = 0.99459, p-value = 0.1717      #El p-valor es mayor que 0.05, aceptamos normalidad de  
los residuos.
```

"""

```
ks.test(modelo_reducido_final_pruebas_residuos_1, pnorm)
```

"""

One-sample Kolmogorov-Smirnov test

```
data: modelo_reducido_final_pruebas_residuos_1
```

```
D = 0.10061, p-value = 0.0006087      #No se acepta la normalidad de los residuos.
```

```
alternative hypothesis: two-sided
```

"""

```
qqnorm(modelo_reducido_final_pruebas_residuos_1) #No se acepta la normalidad de los  
residuos.
```

```
qqline(modelo_reducido_final_pruebas_residuos_1)
```

```
#Homoceodasticidad:
```

```
library(lmtest)
```

```
bptest(modelo_reducido_final_pruebas_1) #Prueba de Braunch-Pagan
```

"""

studentized Breusch-Pagan test

data: modelo_reducido

BP = 1.4535, df = 6, p-value = 0.9625 #El p-valor es mayor que 0.05, aceptamos homocedasticidad.

"""

ncvTest(modelo_reducido_final_pruebas_1)

"""

Non-constant Variance Score Test

Variance formula: ~ fitted.values

Chisquare = 0.01949957, Df = 1, p = 0.88894 #El p-valor es mayor que 0.05, aceptamos homocedasticidad.

"""

library(car)

vif(modelo_reducido_final_pruebas_1)

"""

GVIF Df GVIF^(1/(2*Df))

ShelveLoc 1.014783 2 1.003676

Price 1.018505 1 1.009210

Advertising 1.009404 1 1.004691

Age 1.014095 1 1.007023

Income 1.012469 1 1.006215

"""

par(mfrow=c(2, 2))

plot(modelo_reducido_final_pruebas_1, col='deepskyblue4', pch=19)

influenceIndexPlot(modelo_reducido_final_pruebas_1)

```
#PRUEBAS:  
#Quitamos observaciones 42, 26  
obs.out <- c(26, 144, 51, 107) #observaciones a quitar  
Carseats_obs_out_prueba <- Carseats[-obs.out,1:8]  
  
Carseats_obs_out_prueba_1 <- lm(Sales ~ ShelveLoc + Price + Advertising + Age + Income,  
data = Carseats_obs_out_prueba)  
  
par(mfrow=c(2, 2))  
plot(Carseats_obs_out_prueba_1, col='deepskyblue4', pch=19)  
  
summary(Carseats_obs_out_prueba_1)
```

#OPCIÓN 2: Sin Price

```
modelo_reducido_final_pruebas_2 <- lm(Sales ~ ShelveLoc + Advertising + Age + Income + CompPrice, data=Carseats)
```

```
summary(modelo_reducido_final_pruebas_2)
```

```
anova(modelo_reducido_final_pruebas_2)
```

```
"""
```

Call:

```
lm(formula = Sales ~ ShelveLoc + CompPrice + Advertising + Age + Income, data = Carseats)
```

Residuals:

Min	1Q	Median	3Q	Max
-6.2818	-1.4829	-0.1108	1.4570	5.9593

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.898750	1.039363	4.713	3.39e-06 ***
ShelveLocGood	4.663241	0.313864	14.858	<2e-16 ***
ShelveLocMedium	1.911349	0.258159	7.404	8.13e-13 ***
CompPrice	0.007220	0.006917	1.044	0.297 #CompPrice se vuelve no significativo.
Advertising	0.096389	0.015864	6.076	2.92e-09 ***
Age	-0.040049	0.006532	-6.131	2.13e-09 ***
Income	0.016806	0.003784	4.442	1.16e-05 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				

Residual standard error: 2.099 on 393 degrees of freedom

Multiple R-squared: 0.4558, Adjusted R-squared: 0.4475 #Además baja todo. ES UNA MIERDA DE OPCIÓN.

F-statistic: 54.86 on 6 and 393 DF, p-value: < 2.2e-16

"""

#Quitamos CompPrice:

```
modelo_reducido_final_pruebas_2_final <- lm(Sales ~ ShelveLoc + Advertising + Age + Income, data=Carseats)
```

```
summary(modelo_reducido_final_pruebas_2_final)
```

"""

Call:

```
lm(formula = Sales ~ ShelveLoc + Advertising + Age + Income,  
  data = Carseats)
```

Residuals:

Min	1Q	Median	3Q	Max
-6.3079	-1.4777	-0.0678	1.4416	6.0422

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	5.854636	0.491421	11.914	< 2e-16 ***
ShelveLocGood	4.675320	0.313686	14.904	< 2e-16 ***
ShelveLocMedium	1.919267	0.258076	7.437	6.50e-13 ***
Advertising	0.096022	0.015862	6.053	3.30e-09 ***
Age	-0.040744	0.006499	-6.270	9.52e-10 ***
Income	0.016500	0.003773	4.374	1.57e-05 ***

Signif. codes:	0 ***	0.001 **	0.01 *	0.05 .
	0.1	'	'	1

Residual standard error: 2.099 on 394 degrees of freedom

Multiple R-squared: 0.4543, Adjusted R-squared: 0.4474

```
F-statistic: 65.6 on 5 and 394 DF, p-value: < 2.2e-16
```

```
"""
```

```
par(mfrow=c(1, 2))
```

```
plot(modelo_reducido_final_pruebas_2_final, scale="adjr2", main=expression(R[Adj]^2))
```

```
plot(modelo_reducido_final_pruebas_2_final, scale="bic", main="BIC")
```

```
library("olsrr")
```

```
modelo_completo <- lm(Sales ~ ., data = Carseats)
```

```
ols_mallows_cp(modelo_reducido_final_pruebas_2_final, modelo_completo) #Número  
horrible: 69455584. Muy mal ajustado.
```

```
library(lmtest)
```

```
dwtest(modelo_reducido_final_pruebas_2_final, alternative = "two.sided")
```

```
"""
```

Durbin-Watson test

```
data: modelo_reducido_final_pruebas_2_final
```

```
DW = 1.9023, p-value = 0.3255      #El p-valor es mayor que 0.05, aceptamos normalidad de  
los residuos.
```

```
alternative hypothesis: true autocorrelation is not 0
```

```
"""
```

#NORMALIDAD DE LOS RESIDUOS:

```
modelo_reducido_final_pruebas_2_final_residuos <-  
modelo_reducido_final_pruebas_2_final$residuals
```

```
modelo_reducido_final_pruebas_2_final_residuos
```

```
shapiro.test(modelo_reducido_final_pruebas_2_final_residuos) #El p-valor es mayor que 0.05,  
aceptamos normalidad de los residuos.
```

```
#Si DW está lejos de 0, siendo un valor MUY cercano a 2, y el p-valor es mayor que 0.05, luego  
hay independencia. No hay autocorrelación entre los errores.
```

```
""""
```

```
Shapiro-Wilk normality test
```

```
data: modelo_reducido_final_pruebas_2_final_residuos
```

```
W = 0.99698, p-value = 0.6702      #El p-valor es mayor que 0.05, aceptamos normalidad de  
los residuos.
```

```
""""
```

```
ks.test(modelo_reducido_final_pruebas_2_final_residuos, pnorm)
```

```
""""
```

```
One-sample Kolmogorov-Smirnov test
```

```
data: modelo_reducido_final_pruebas_2_final_residuos
```

```
D = 0.18583, p-value = 2.01e-12      #NO se acepta la normalidad de los residuos.
```

```
alternative hypothesis: two-sided
```

```
""""
```

```
qqnorm(modelo_reducido_final_pruebas_2_final_residuos) #No se acepta la normalidad de los  
residuos.
```

```
qqline(modelo_reducido_final_pruebas_2_final_residuos)
```

```
#Homoceodasticidad:
```

```
library(lmtest)
bptest(modelo_reducido_final_pruebas_2_final) #Prueba de Braunsch-Pagan
```

"""

studentized Breusch-Pagan test

```
data: modelo_reducido_final_pruebas_2_final
BP = 7.0565, df = 5, p-value = 0.2165    #El p-valor es mayor que 0.05, aceptamos
homocedasticidad.
```

"""

```
library(car)
```

```
ncvTest(modelo_reducido_final_pruebas_2_final)
```

"""

Non-constant Variance Score Test

Variance formula: ~ fitted.values

```
Chisquare = 0.3248429, Df = 1, p = 0.56871 #El p-valor es mayor que 0.05, aceptamos
homocedasticidad.
```

"""

```
library(car)
```

```
vif(modelo_reducido_final_pruebas_2_final)
```

"""

GVIF Df GVIF^(1/(2*Df))

```
ShelveLoc 1.012514 2     1.003114
```

```
Advertising 1.007367 1     1.003676
```

```
Age      1.003371 1     1.001684
```

```
Income   1.009094 1     1.004537
```

"""

```
kappa(modelo_reducido_final_pruebas_2_final) #INDICA FUERTE COLINEALIDAD:
379.0389
```

```
par(mfrow=c(2, 2))
plot(modelo_reducido_final_pruebas_2_final, col='deepskyblue4', pch=19)
```

#TRANSFORMACIÓN

```
#Me da error, me dice que la variable respuesta debe ser positiva, esto es porque hay un dato, el  
175, cuyo valor es y=0.00
```

```
#que es el causante del problema, pero no se si hay más. Para solucionarlo, vamos a añadir una  
constante a la variable
```

```
#respuesta Sales para poder aplicar boxcox, el problema es que al sumar una constante, no  
sabemos exactamente qué cantidad
```

```
#sumar sin afectar a las demás pruebas, lo que causa problemas futuros. Por tanto, recurrimos a  
las Transformaciones Yeo-Johnson:
```

```
BC <- boxCox(modelo_reducido_final_pruebas_2_final, family="yjPower", plotit = TRUE)
```

```
lambda <- BC$x[which.max(BC$y)]
lambda
#####
[1] 0.9494949 #El valor exacto de lambda.
#####
```

```
modelo_reducido_final_pruebas_2_transformado <- yjPower(Sales, lambda)
```

```
modelo_reducido_final_pruebas_2_2_transformado <-
lm(modelo_reducido_final_pruebas_2_2_transformado ~ ShelveLoc + Price + Advertising + Age
+ Income, data = Carseats)
```

```
modelo_reducido_final_pruebas_2_2_transformado
```

```
summary(modelo_reducido_final_pruebas_2_2_transformado)
```

"""

Call:

```
lm(formula = modelo_reducido_final_pruebas_2_2_transformado ~ ShelveLoc +
Price + Advertising + Age + Income, data = Carseats)
```

Residuals:

Min	1Q	Median	3Q	Max
-3.6111	-0.8757	-0.0962	0.8771	3.8408

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	12.260335	0.491160	24.962	< 2e-16 ***
ShelveLocGood	4.377784	0.207393	21.109	< 2e-16 ***
ShelveLocMedium	1.816476	0.170488	10.655	< 2e-16 ***
Price	-0.054473	0.002959	-18.411	< 2e-16 ***
Advertising	0.094977	0.010486	9.057	< 2e-16 ***
Age	-0.044827	0.004315	-10.390	< 2e-16 ***
Income	0.012219	0.002496	4.896	1.43e-06 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.386 on 393 degrees of freedom

Multiple R-squared: 0.7063, Adjusted R-squared: 0.7019

F-statistic: 157.5 on 6 and 393 DF, p-value: < 2.2e-16

"""

#Colinealidad o multicolinealidad entre predictores:

#Análisis de Inflación de Varianza (VIF): Se calcula un VIF por cada variable explicativa:

```
library(car)
```

```
vif(modelo_reducido_final_pruebas_2_2_transformado) #Los valores VIF_{j} son pequeños,  
luego no hay colinealidad. Entre 0 y 1 bien, poco más de uno algo de colinealidad.
```

#Autocorrelación con Durbin-Watson: Veamos la INDEPENDENCIA DE ERRORES:

```
library(lmtest)
```

```
dwtest(modelo_reducido_final_pruebas_2_2_transformado, alternative = "two.sided")
```

"""

Durbin-Watson test

```
data: modelo_reducido_final_pruebas_2_2_transformado
```

```
DW = 1.9294, p-value = 0.4779
```

```
alternative hypothesis: true autocorrelation is not 0
```

"""

```
influenceIndexPlot(modelo_reducido_final_pruebas_2_2_transformado) #Valores hat,  
Bonferroni, Cook, Student estandarizado.
```

```
#Ver plot_diagnstico_reducido.png
```

```
influencePlot(modelo_reducido_final_pruebas_2_2_transformado) #VISTA ANALÍTICA DE  
LAS INFLUENCIAS. ALTERNATIVA A influenceIndexPlot.
```

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

```
plot(modelo_reducido_final_pruebas_2_2_transformado, col='deepskyblue4', pch=19)
```

```
library(lmtest)
bptest(modelo_reducido) #Prueba de Braunch-Pagan. #p-valor grande, luego aceptamos la hipótesis nula: Hay homocedasticidad.
```

```
ncvTest(modelo_reducido) #p-valor grande, luego aceptamos la hipótesis nula: Hay homocedasticidad.
```

#NORMALIDAD DE LOS RESIDUOS:

```
modelo_reducido_final_pruebas_2_2_transformado_residuos <-
modelo_reducido_final_pruebas_2_2_transformado$residuals
modelo_reducido_final_pruebas_2_2_transformado_residuos
```

```
ks.test(modelo_reducido_final_pruebas_2_2_transformado_residuos, pnorm)
#####
One-sample Kolmogorov-Smirnov test
```

```
data: modelo_reducido_final_pruebas_2_2_transformado_residuos
D = 0.081426, p-value = 0.009942      #Mejora considerablemente el p-valor, pero sigue siendo insuficiente.
```

```
alternative hypothesis: two-sided
#####
#TRATAMIENTO DE LAS VARIABLES CUALITATIVAS DEL MODELO 2:
```

"""

La idea es hacer para cada una de las categorías de la variable cualitativa establecer una variable binaria.

Transformamos la variable categórica en variable dummy, y luego analizamos el efecto de esa variable categórica sobre Sales.

"""

```
modelo_complete <- lm(Sales ~ Price + CompPrice + Advertising +
Age + Income + Population + Education, data = Carseats)
summary(modelo_complete)
```

"""

Call:

```
lm(formula = Sales ~ Price + CompPrice + Advertising + Age +
Income + Population + Education, data = Carseats)
```

Residuals:

Min	1Q	Median	3Q	Max
-5.0598	-1.3515	-0.1739	1.1331	4.8304

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	7.7076934	1.1176260	6.896	2.15e-11 ***
Price	-0.0925226	0.0050521	-18.314	< 2e-16 ***
CompPrice	0.0939149	0.0078395	11.980	< 2e-16 ***
Advertising	0.1308637	0.0151219	8.654	< 2e-16 ***
Age	-0.0449743	0.0060083	-7.485	4.75e-13 ***
Income	0.0128717	0.0034757	3.703	0.000243 ***

```
Population -0.0001239 0.0006877 -0.180 0.857092 #Los signos negativos en la columna  
Estimate denotan que influyen (para mal)
```

```
Education -0.0399844 0.0371257 -1.077 0.282142 #esas variables sobre Sales, es decir,  
estaría asociado con una disminución
```

```
--- # en Sales.
```

```
Signif. codes: 0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 1.929 on 392 degrees of freedom
```

```
Multiple R-squared: 0.5417, Adjusted R-squared: 0.5335
```

```
F-statistic: 66.18 on 7 and 392 DF, p-value: < 2.2e-16
```

```
####
```

```
#ShelveLoc tiene 3 niveles, pero vamos a considerar solo ShelveLocGood y ShelveLocMedium  
que infiere R y que resumen muy bien ShelveLoc:
```

```
#Interpretamos esas variables porque son las significativas para el modelo, Urban y US  
#quedan descartadas. Dado que la variable cualitativa ShelveLoc tiene 3 niveles, se ne-  
#cesitarán 2 variables dummys para resumir la variable cualitativa.
```

```
#CASO 1: Todas las variables cuantitativas:
```

```
ventas <- 7.7076934 + (-0.0925226)*Price + 0.0939149*CompPrice + 0.1308637*Advertising  
+ (-0.0449743)*Age + 0.0128717*Income + (-0.0001239)*Population + (-  
0.0399844)*Education
```

```
#Quitamos Population y Education al no ser significativas para el modelo:
```

```
modelo_complete2 <- lm(Sales ~ Price + CompPrice + Advertising +  
Age + Income, data = Carseats)  
summary(modelo_complete2)
```

```
####
```

```
Call:
```

```
lm(formula = Sales ~ Price + CompPrice + Advertising + Age +
  Income, data = Carseats)
```

Residuals:

Min	1Q	Median	3Q	Max
-4.9071	-1.3081	-0.1892	1.1495	4.6980

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	7.109190	0.943940	7.531	3.46e-13 ***
Price	-0.092543	0.005044	-18.347	< 2e-16 ***
CompPrice	0.093904	0.007792	12.051	< 2e-16 ***
Advertising	0.130611	0.014572	8.963	< 2e-16 ***
Age	-0.044971	0.005994	-7.503	4.20e-13 ***
Income	0.013092	0.003465	3.779	0.000182 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.927 on 394 degrees of freedom

Multiple R-squared: 0.5403, Adjusted R-squared: 0.5345 #Mejora un poco el R^2, y además menos variables, luego mejorará AIC y BIC.

F-statistic: 92.62 on 5 and 394 DF, p-value: < 2.2e-16 #Además mejora el F-statistic haciéndolo más grande.

"""

```
ventas2 <- 7.7076934 + (-0.0925226)*Price + 0.0939149*CompPrice + 0.1308637*Advertising
+ (-0.0449743)*Age + 0.0128717*Income
```

#Incluimos ShelveLoc: MEJORA LA R^2 y la F-statistic:

```

modelo_complete3 <- lm(Sales ~ Price + CompPrice + Advertising +
Age + Income + Population + Education + ShelveLoc, data = Carseats)
summary(modelo_complete3)

#####

```

Call:

```

lm(formula = Sales ~ Price + CompPrice + Advertising + Age +
Income + Population + Education + ShelveLoc, data = Carseats)

```

Residuals:

Min	1Q	Median	3Q	Max
-2.8403	-0.6846	0.0151	0.6702	3.3481

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	5.6472014	0.5973864	9.453	< 2e-16 ***
Price	-0.0953854	0.0026726	-35.690	< 2e-16 ***
CompPrice	0.0929439	0.0041451	22.422	< 2e-16 ***
Advertising	0.1141447	0.0080123	14.246	< 2e-16 ***
Age	-0.0459891	0.0031817	-14.454	< 2e-16 ***
Income	0.0157321	0.0018426	8.538	3.09e-16 ***
Population	0.0002632	0.0003641	0.723	0.470
Education	-0.0197355	0.0196387	-1.005	0.316
ShelveLocGood	4.8350741	0.1527240	31.659	< 2e-16 ***
ShelveLocMedium	1.9553465	0.1255836	15.570	< 2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				

Residual standard error: 1.02 on 390 degrees of freedom

Multiple R-squared: 0.8725, Adjusted R-squared: 0.8696

F-statistic: 296.6 on 9 and 390 DF, p-value: < 2.2e-16

####

```

ventas3 <- 5.6472014 + (-0.0953854)*Price + 0.0929439*CompPrice + 0.1141447*Advertising
+ (-0.0459891)*Age + 0.0157321*Income + 4.8350741*ShelveLocGood +
1.9553465*ShelveLocMedium

```

#NOTA IMPORTANTE: Los p-valores utilizando summary PARA VARIABLES CUALITATIVAS pueden proporcionarnos conclusiones erróneas, es por

#ello que utilizaremos la función anova(nuestro_modelo) en lugar de summary() para comprobar la significancia de las

#variables cualitativas:

```

modelo_complete3 <- lm(Sales ~ Price + CompPrice + Advertising +
Age + Income + Population + ShelveLoc, data = Carseats)

anova(modelo_complete3)

```

####

Analysis of Variance Table

Response: Sales

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Price	1	630.03	630.03	605.7673 < 2.2e-16 ***	
CompPrice	1	508.69	508.69	489.1028 < 2.2e-16 ***	
Advertising	1	315.78	315.78	303.6165 < 2.2e-16 ***	
Age	1	211.86	211.86	203.7020 < 2.2e-16 ***	
Income	1	53.02	53.02	50.9742 4.609e-12 ***	
Population	1	0.02	0.02	0.0185 0.89201	
ShelveLoc	2	1052.94	526.47	506.1956 < 2.2e-16 ***	#Se ve que es significativa esa variable cualitativa.
Residuals	390	405.62	1.04		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1					

Como el p-valor es menor que 0.05, se concluye que hay evidencias para rechazar H_0 , es decir, ShelveLoc es significativa. Otra forma para confirmarlo:

```
modelo_reducido_1 <- lm(Sales ~ Price + CompPrice + Advertising +
Age + Income, data = Carseats)
```

```
modelo_complete4 <- lm(Sales ~ Price + CompPrice + Advertising +
Age + Income + ShelveLoc, data = Carseats)
```

```
require(MASS)
data()
anova(modelo_reducido_1, modelo_complete4) #MÁS RECOMENDABLE.
```

"""

#como Population y Education NO son estadísticamente significativas al ser su p-valor mayor que 0.05,

#los quitamos y generamos el nuevo modelo:

```
modelo_complete4 <- lm(Sales ~ Price + CompPrice + Advertising +
Age + Income + ShelveLoc, data = Carseats)
summary(modelo_complete4)
```

"""

Call:

```
lm(formula = Sales ~ Price + CompPrice + Advertising + Age +
Income + ShelveLoc, data = Carseats)
```

Residuals:

Min	1Q	Median	3Q	Max
-2.7728	-0.6954	0.0282	0.6732	3.3292

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept)	5.475226	0.505005	10.84	<2e-16 ***
Price	-0.095319	0.002670	-35.70	<2e-16 ***
CompPrice	0.092571	0.004123	22.45	<2e-16 ***
Advertising	0.115903	0.007724	15.01	<2e-16 ***
Age	-0.046128	0.003177	-14.52	<2e-16 ***
Income	0.015785	0.001838	8.59	<2e-16 ***
ShelveLocGood	4.835675	0.152499	31.71	<2e-16 *** #Las dummies son significativas, luego hay que interpretarlas.
ShelveLocMedium	1.951993	0.125375	15.57	<2e-16 *** #Las dummies son significativas, luego hay que interpretarlas.

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.019 on 392 degrees of freedom

Multiple R-squared: 0.872, Adjusted R-squared: 0.8697 #Mejora mucho más el R^2
ajustado.

F-statistic: 381.4 on 7 and 392 DF, p-value: < 2.2e-16 #Mejora mucho más el F-statistic.

"""

#CASO 2: Recta de regresión tomando ShelveLoc:

```
ventas_general <- 5.475226 + (-0.095319 )*Price + 0.092571*CompPrice +
0.115903*Advertising
+ (-0.046128)*Age + 0.015785*Income + 4.835675*ShelveLocGood +
1.951993*ShelveLocMedium
```

#Siguiendo la teoría, tenemos una variable cuantitativa: ShelveLoc y dos dummies: D1 =
ShelveLocGood y D2= ShelveLocGoodMedium:

```
#Sería Y = \beta_0 + \beta_1 x_1 + \beta_2 D_1 + \beta_3 x_1 D_1 +
\beta_4 D_2 + \beta_5 x_1 D_2 + \varepsilon
```

#Modelo de los que toman ShelveLocGood. Luego ShelveLocGood = 1 y
ShelveLocGoodMedium=0:

```
ventas_general_1 <- 5.475226 + (-0.095319 )*Price + 0.092571*CompPrice +  
0.115903*Advertising  
+ (-0.046128)*Age + 0.015785*Income + 4.835675*1 + 1.951993*0
```

#Así, el intercepto queda como: $5.475226 + 4.835675 = 10.3109$, luego la ecuación final es:

```
ventas_general_1_final <- 10.3109 + (-0.095319 )*Price + 0.092571*CompPrice +  
0.115903*Advertising  
+ (-0.046128)*Age + 0.015785*Income
```

#Modelo de los que toman ShelveLocGoodMedium. Luego ShelveLocGood = 0 y
ShelveLocGoodMedium=1:

```
ventas_general_2 <- 5.475226 + (-0.095319 )*Price + 0.092571*CompPrice +  
0.115903*Advertising  
+ (-0.046128)*Age + 0.015785*Income + 4.835675*0 + 1.951993*1
```

#Así, el intercepto queda como: $5.475226 + 1.951993 = 7.427219$, luego la ecuación final es:

```
ventas_general_2_final <- 7.427219 + (-0.095319 )*Price + 0.092571*CompPrice +  
0.115903*Advertising  
+ (-0.046128)*Age + 0.015785*Income
```

#Modelo sin ShelveLoc. Luego ShelveLocGood = 0 y ShelveLocGoodMedium=0:

```
ventas_general_3 <- 5.475226 + (-0.095319 )*Price + 0.092571*CompPrice +  
0.115903*Advertising  
+ (-0.046128)*Age + 0.015785*Income + 4.835675*0 + 1.951993*0
```

#Así, el intercepto queda como: $5.475226 + 1.951993 = 7.427219$, luego la ecuación final es:

```
ventas_general_3 <- 5.475226 + (-0.095319 )*Price + 0.092571*CompPrice +  
0.115903*Advertising  
+ (-0.046128)*Age + 0.015785*Income
```

"""

Se observa que solo cambia el intercepto y además en forma creciente,
luego tomando esas dos dummies, ¡¡estamos vendiendo más que sin tomarlas!!

En concreto, tomando ShelveLocGood vemos un incremento hasta 10 y con
ShelveLocMedium vemos otro incremento, aunque menor, de hasta 7 ventas más.

"""

#PREGUNTA 1: ¿Son iguales las rectas?

```
modelo_reducido_1 <- lm(Sales ~ Price + CompPrice + Advertising +  
Age + Income, data = Carseats)
```

```
modelo_complete4 <- lm(Sales ~ Price + CompPrice + Advertising +  
Age + Income + ShelveLoc, data = Carseats)
```

```
require(MASS)  
data("Carseats")  
anova(modelo_reducido_1, modelo_complete4)
```

"""

Analysis of Variance Table

Model 1: Sales ~ Price + CompPrice + Advertising + Age + Income

Model 2: Sales ~ Price + CompPrice + Advertising + Age + Income + ShelveLoc

Res.Df RSS Df Sum of Sq F Pr(>F)

1 394 1462.90

2 392 407.39 2 1055.5 507.82 < 2.2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Como el p-valor es aproximadamente 0, se concluye que hay evidencias para rechazar H_0 , es decir, las rectas NO son iguales.

.....

.....

Ahora bien, quizá el estudio del intercepto no sea suficiente, pues si se diera el caso de que las rectas tienen misma pendiente, significaría que

ShelveLocGood y ShelveLocMedium es igual de válido para las demás variables explicativas. Por

poner un ejemplo más sencillo, si tuviéramos

un modelo donde hay alumnos y alumnos_tercera_matrícula y una variable cualitativa seminario_Algébra y variable explicativa

que tome o no ese curso. Queremos ver como se relacionan las variables anteriores con las calificaciones

en la asignatura Ecuaciones Algebraicas:

Si tuviéramos en nuestro modelo pendientes iguales (supongamos que con mejoría de calificación), eso significaría que

el efecto positivo de los alumnos que toman el curso (la pendiente) sería igual para todo tipo de alumnos,

pero esto quizá no sea tan realista pues alumnos_tercera_matrícula puede que tengan más experiencia en la asignatura que

los que toman el curso. Por tanto, quizá sea más realista un modelo con rectas secantes, esperando así que el curso

tenga más efecto positivo sobre alumnos (primera matrícula que toman el curso) y menos sobre alumnos_tercera_matricula.

.....

#PREGUNTA 2: ¿Son las pendientes iguales?

anova(modelo_complete4)

.....

Analysis of Variance Table

Response: Sales

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Price	1	630.03	630.03	606.234 < 2.2e-16 ***	
CompPrice	1	508.69	508.69	489.480 < 2.2e-16 ***	
Advertising	1	315.78	315.78	303.851 < 2.2e-16 ***	
Age	1	211.86	211.86	203.859 < 2.2e-16 ***	
Income	1	53.02	53.02	51.014 4.494e-12 ***	
ShelveLoc	2	1055.51	527.76	507.822 < 2.2e-16 ***	#El p-valor es menor que 0.05, luego las dos rectas no tienen la misma pendiente.
Residuals	392	407.39	1.04		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

.....

#PREGUNTA 3: ¿Son los interteptos iguales? Se respondió en CASO 2.

#Otra forma sería utilizando el ANOVA y viendo el p-valor.

```
####
```

#VALIDACION DEL MODELO 2:

#x) Crear y validar modelos de predicción para Sales usando los datos de Carseats.

```
install.packages("ISLR")
library(ISLR)
install.packages("leaps")
library(leaps)
install.packages("PASWR")
library(PASWR)
```

```
modelo_reducido <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age +
Income, data = Carseats)

summary(modelo_reducido)

set.seed(5) #semilla

train <- sample (c(TRUE, FALSE), size=nrow(Carseats[1:6]),
replace=TRUE, prob=c(0.70,0.30)) #conjunto de entenamiento

prop.table(table(train))#calcula los percentiles en train

test <- (!train)
```

```

prop.table(table(test)) #calcula los percentiles en test

#Usando la aproximación al conjunto de validación elegir el mejor modelo de
#regresión obtenido con la función regsubsets() cuando Sales es la variable
#respuesta y ShelveLoc, Price, Compprice, Advertising, Age e Income las variables
explicativas. Usar
#la semilla set.seed=5 y dividir los datos disponibles en conjunto de entrenamiento
#y tst donde aproximadamente el 70% de los datos disponibles se usan
#para el entrenamiento y el resto se usa para el conjunto de test.

model.exh <- regsubsets(Sales ~ShelveLoc + Price + CompPrice + Advertising + Age + Income
, data = Carseats[train, 1:8] , method= "exhaustive")
summary(model.exh) #todos los modelos posibles para los `predictores
#vamos a calcular el error del conjunto de validación para el mejor modelo de entre los
#obtenidos antes
predict.regsubsets <- function(object, newdata, id,...){
  form <-as.formula(object$call[[2]])
  mat <- model.matrix(form,newdata)
  coefi <- coef(object,id=id)
  xvars <- names(coefi)
  mat[,xvars] %*% coefi
}
val.errors <- rep(NA,7)
Y <- Carseats[test,]$Sales
for (i in 1:7){
  Yhat <- predict.regsubsets (model.exh, newdata=Carseats[test,], id=i)
  val.errors[i] <- mean((Y-Yhat)^2)
}
val.errors
coef(model.exh, which.min(val.errors))

#otra forma

```

```

regfit.best <- regsubsets(Sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income,
  data=Carseats[train, 1:8])

coef( regfit.best, which.min(val.errors))

#Usando validación cruzada dejando uno fuera elegir el mejor modelo de regresión
#ón de entre los que obtenemos con la función regsubsets cuando cuando
#SALES es la variable respuesta y ShelveLoc, Price, CompPrice, Advertising, Age e Income
son las variables
#explicativas.

##LOOCV

modelo_reducido <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age +
Income, data = Carseats)

summary(modelo_reducido)

n <- nrow(Carseats)

k <- n #número de grupos igual a n

set.seed(5)

folds <- sample(x=1:k, size =nrow(Carseats), replace = FALSE)

cv.errors <- matrix(NA, k, 7, dimnames = list(NULL,paste(1:7)))

for (j in 1:k){

  best.fit <- regsubsets(Sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income,
  data=Carseats[folds !=j,])#cojemos datos del conjunto de entrenamiento

  for (i in 1:7){

    pred <- predict.regsubsets(best.fit, newdata=Carseats[folds==j], id=i)#datos de test

    cv.errors[j,i] <- mean((Carseats$Sales[folds == j]-pred)^2)

  }

}

mean.cv.errors <- apply(cv.errors, 2, mean)#calcula la media de los betas_i

mean.cv.errors

#x3)Usando validación cruzada con 5 grupos elegir el mejor modelo de regresión de
#entre los que obtenemos con la función regsubsets cuando cuando Sales es la

```

```

#variable respuesta y ShelveLoc, Price, CompPrice, Advertising, Age, e Income las variables
explicativas.

## VALIDACIÓN CRUZADA k-folds

modelo_reducido <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age +
Income, data = Carseats)

summary(modelo_reducido)

n <- nrow(Carseats)

k <- 10 #número de grupos igual a n

set.seed(5)

folds <- sample(x=1:k, size =nrow(Carseats), replace = FALSE)

cv.errors <- matrix(NA, k, 7, dimnames = list(NULL,paste(1:7)))

for (j in 1:k){

  best.fit <- regsubsets(Sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income,
data=Carseats[folds !=j,])

  for (i in 1:7){

    pred <- predict.regsubsets(best.fit, newdata=Carseats[folds==j], id=i)

    cv.errors[j,i] <- mean((Carseats$Sales[folds == j]-pred)^2)

  }

}

mean.cv.errors <- apply(cv.errors, 2, mean)

mean.cv.errors

coef(regfit.best, which.min(mean.cv.errors))

install.packages("car")

library(car)

#comprobación grafica

model.cv <- lm(Sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income,
data=Carseats)

summary(model.cv)

plot(lm(Sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income, data=Carseats))

plot(lm(Sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income, data=Carseats),
which=c(1,2))

residualPlot(model.cv)

influenceIndexPlot(model.cv)

```

```
####
```

```
###
```

```
#VALIDACION SIMPLE: CON TODAS LAS VARIABLES EXPLICATIVAS:
```

```
install.packages("ISLR")
library(ISLR)
install.packages("leaps")
library(leaps)
install.packages("PASWR")
library(PASWR)

modelo_reducido <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age +
Income, data = Carseats)

summary(modelo_reducido)
set.seed(5) #semilla
train <- sample (c(TRUE, FALSE), size=nrow(Carseats[1:6]),
replace=TRUE, prob=c(0.70,0.30)) #conjunto de entenamiento
prop.table(table(train))#calcula los percentiles en train
test <- (!train)
prop.table(table(test)) #calcula los percentiles en test

#Usando la aproximación al conjunto de validación elegir el mejor modelo de
#regresión obtenido con la función regsubsets() cuando Sales es la variable
#respuesta y ShelveLoc, Price, Compprice, Advertising, Age e Income las variables
explicativas. Usar
#la semilla set.seed=5 y dividir los datos disponibles en conjunto de entrenamiento
#y tst donde aproximadamente el 70% de los datos disponibles se usan
#para el entrenamiento y el resto se usa para el conjunto de test.
```

```

model.exh <- regsubsets(Sales ~. , data = Carseats[train, 1:8] , method= "exhaustive")
summary(model.exh) #todos los modelos posibles para los `predictores
#vamos a calcular el error del conjunto de validación para el mejor modelo de entre los
#obtenidos antes
predict.regsubsets <- function(object, newdata, id,...){
  form <-as.formula(object$call[[2]])
  mat <- model.matrix(form,newdata)
  coefi <- coef(object,id=id)
  xvars <- names(coefi)
  mat[,xvars] %*% coefi
}
val.errors <- rep(NA,8)
Y <- Carseats[test,]$Sales
for (i in 1:8){
  Yhat <- predict.regsubsets (model.exh, newdata=Carseats[test,], id=i)
  val.errors[i] <- mean((Y-Yhat)^2)
}
val.errors
coef(model.exh, which.min(val.errors))

#otra forma
regfit.best <-regsubsets(Sales~., data=Carseats[train, 1:8])
coef( regfit.best, which.min(val.errors))

#Usando validación cruzada dejando uno fuera elegir el mejor modelo de regresión
#ón de entre los que obtenemos con la función regsubsets cuando cuando
#SALES es la variable respuesta y ShelveLoc, Price, CompPrice, Advertising, Age e Income
son las variables
#explicativas.
##LOOCV

```

```

modelo_reducido <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age +
Income, data = Carseats)

summary(modelo_reducido)

n <- nrow(Carseats)

k <- n #número de grupos igual a n

set.seed(5)

folds <- sample(x=1:k, size =nrow(Carseats), replace = FALSE)

cv.errors <- matrix(NA, k, 8, dimnames = list(NULL,paste(1:8)))

for (j in 1:k){

  best.fit <- regsubsets(Sales~., data=Carseats[folds !=j,]) #cogemos datos del conjunto de
entrenamiento

  for (i in 1:8){

    pred <- predict.regsubsets(best.fit, newdata=Carseats[folds==j,], id=i)#datos de test

    cv.errors[j,i] <- mean((Carseats$Sales[folds == j]-pred)^2)

  }

}

mean.cv.errors <- apply(cv.errors, 2, mean)#calcula la media de los betas_i

mean.cv.errors

coef( regfit.best, which.min(mean.cv.errors))

#x3)Usando validación cruzada con 5 grupos elegir el mejor modelo de regresión de
#entre los que obtenemos con la función regsubsets cuando cuando Sales es la
#variable respuesta y ShelveLoc, Price, CompPrice, Advertising, Age, e Income las variables
explicativas.

## VALIDACIÓN CRUZADA k-folds

modelo_reducido <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age +
Income, data = Carseats)

summary(modelo_reducido)

n <- nrow(Carseats)

k <- 10 #número de grupos igual a n

set.seed(5)

folds <- sample(x=1:k, size =nrow(Carseats), replace = FALSE)

cv.errors <- matrix(NA, k, 8, dimnames = list(NULL,paste(1:8)))

for (j in 1:k){

```

```

best.fit <- regsubsets(Sales~., data=Carseats[folds !=j,])
for (i in 1:8){

  pred <- predict.regsubsets(best.fit, newdata=Carseats[folds==j], id=i)
  cv.errors[j,i] <- mean((Carseats$Sales[folds == j]-pred)^2)
}

mean.cv.errors <- apply(cv.errors, 2, mean)
mean.cv.errors
coef(regfit.best, which.min(mean.cv.errors))
install.packages("car")
library(car)

#comprobación grafica

model.cv <- lm(Sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income,
data=Carseats)

summary(model.cv)

plot(lm(Sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income, data=Carseats))
plot(lm(Sales~ShelveLoc + Price + CompPrice + Advertising + Age + Income, data=Carseats),
which=c(1,2))

residualPlot(model.cv)
influenceIndexPlot(model.cv)

#####

```

#Estimación para la respuesta media para nuevos valores:

```

modelo_reducido <- lm(Sales ~ ShelveLoc + Price + CompPrice + Advertising + Age +
Income, data = Carseats)

summary(modelo_reducido)

```

```

nuevos_datos <- data.frame(ShelveLoc, Price=180, CompPrice=150, Advertising=10, Age=60,
Income=120)

prediccion_datos <- predict(object=modelo_reducido, newdata=nuevos_datos)

```

```
View(prediccion_datos)
```

```
#Calculamos el IC al 95% para E[Y|ShelveLoc + Price + CompPrice + Advertising + Age + Income]
```

```
#para el nuevo conjunto de datos siguiente:
```

```
#Intervalo para la respuesta media:
```

```
nuevos_datos <- data.frame(ShelveLoc, Price=180, CompPrice=150, Advertising=10, Age=60, Income=120)
```

```
prediccion_media <- predict(object=modelo_reducido, newdata=nuevos_datos, interval="confidence", level=0.95)
```

```
#Intervalo de predicción:
```

```
nuevos_datos <- data.frame(ShelveLoc, Price=180, CompPrice=150, Advertising=10, Age=60, Income=120)
```

```
prediccion_datos <- predict(object=modelo_reducido, newdata=nuevos_datos, interval="prediction", level=0.95)
```

```
prediccion_datos
```

```
nuevos_datos_2 <- cbind(nuevos_datos, prediccion_datos)
```