

Diagnosis

Alejandro Keymer

11/11/2019

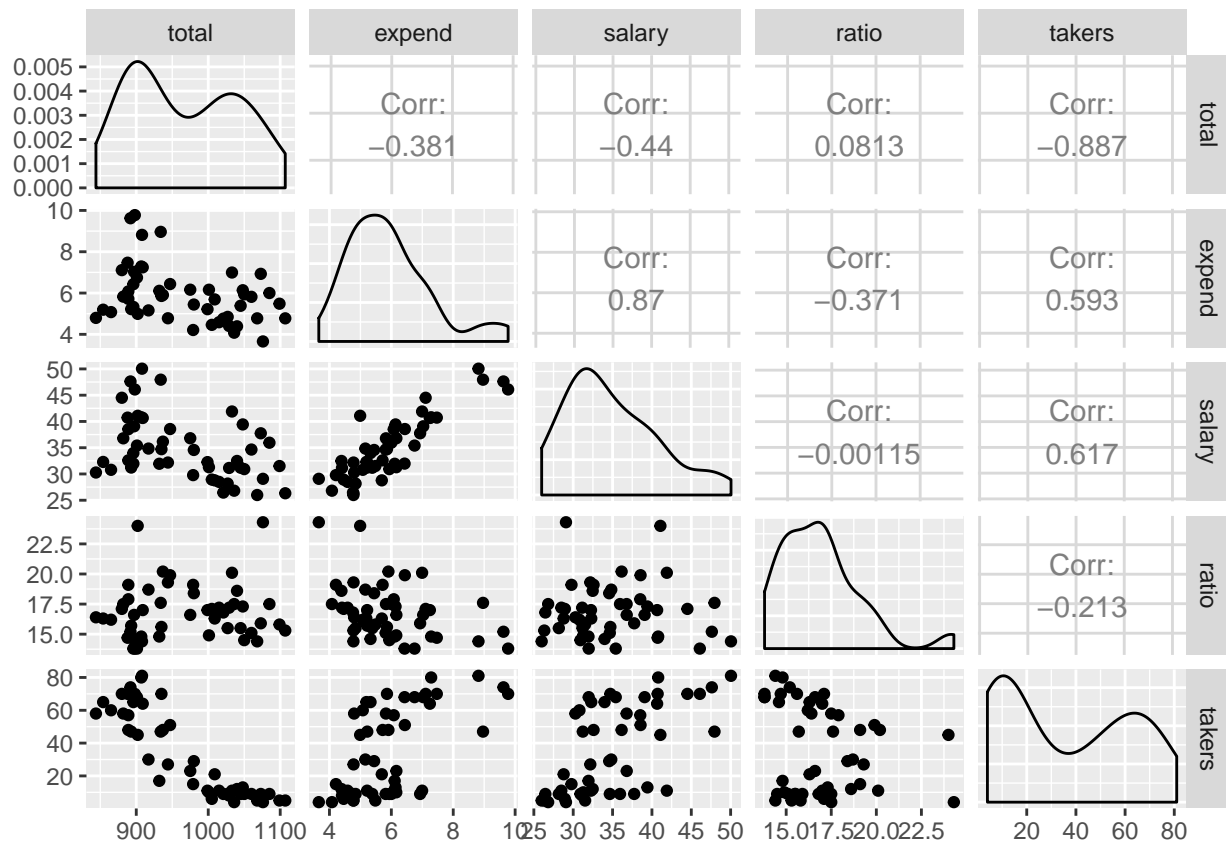
Ejercicios del libro de Faraway

1. (Ejercicio 1 cap. 6 pág. 97)

Using the `sat` dataset, fit a model with the total SAT score as the response and `expend`, `salary`, `ratio` and `takers` as predictors. Perform regression diagnostics on this model to answer the following questions. Display any plots that are relevant. Do not provide any plots about which you have nothing to say. Suggest possible improvements or corrections to the model where appropriate.

```
n <- dim(sat)[1]
p <- dim(sat)[2]
model <-
  lm(total ~ expend + salary + ratio + takers, data = sat)

sat %>%
  select(total, expend, salary, ratio, takers) %>%
  ggpairs(progress = F)
```



Lo primero que hago es crear una matriz de dispersión para tener una idea de como se distribuyen las variables y la relación 2x2 que tienen entre si. Utilizo la librería `GGally` que tiene la función `ggpairs`.

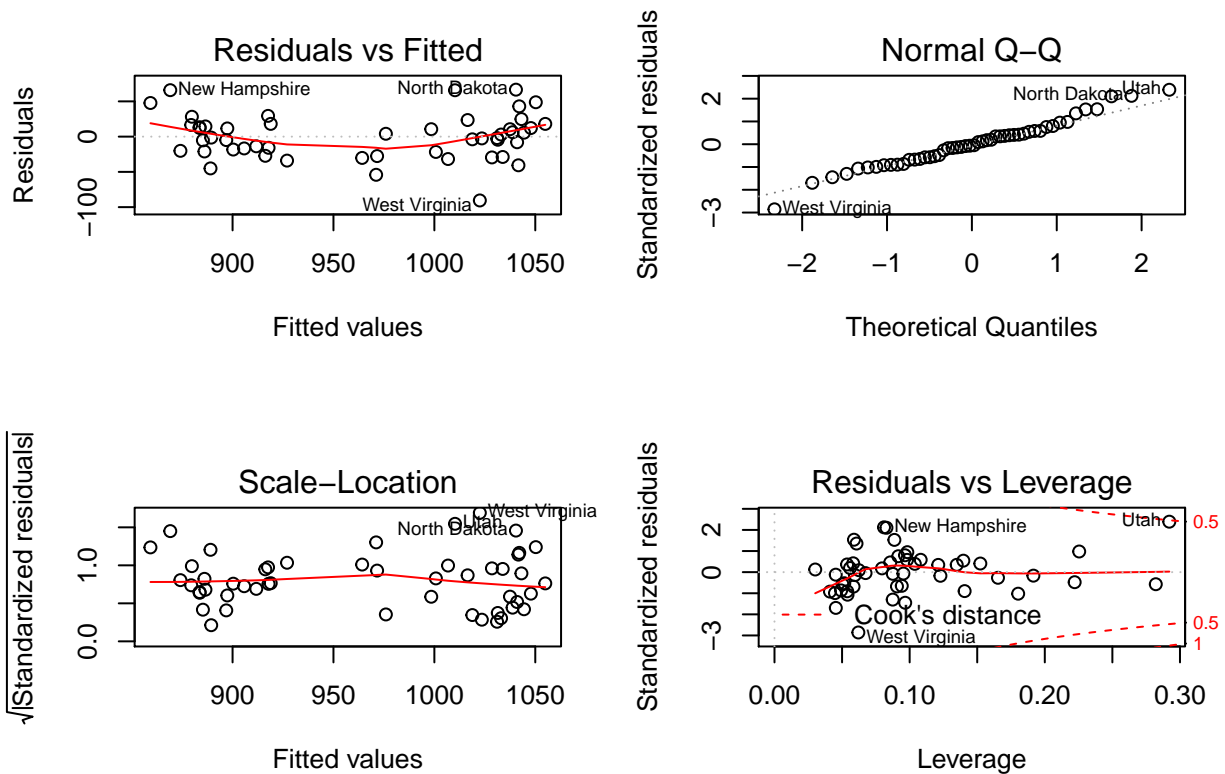


Figure 1: Gráficos Diagnósticos

(a) Check the constant variance assumption for the errors.

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

Para evaluar la homocedasticidad del error utilizo los gráficos de:

- residuales v/s valor ajustado
- residuales estandarizados v/s valor ajustado.

En este caso se observa cierta irregularidad en el gráfico $\hat{y} \times \hat{e}$ que podría traducir no-linearidad, aunque dentro de todo no se observa un patrón definido, ni tan anormal que haga pensar que el error no sigue una homocedasticidad.

(b) Check the normality assumption.

```
shapiro.test(resid(model))

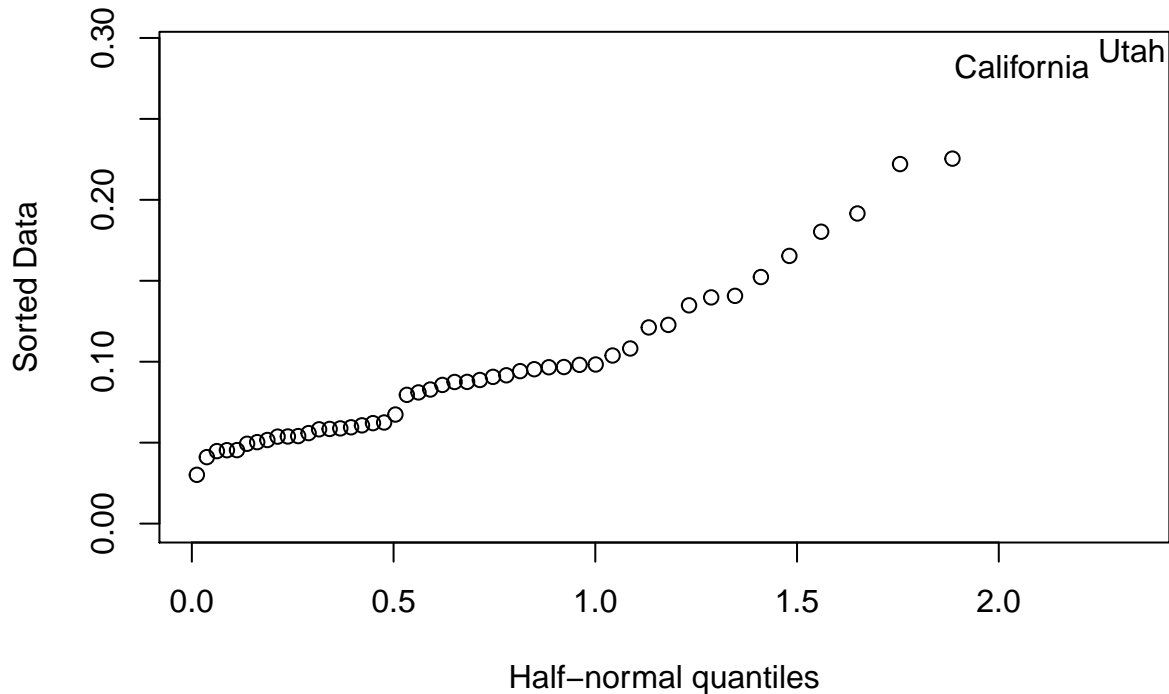
##
## Shapiro-Wilk normality test
##
## data:  resid(model)
## W = 0.97691, p-value = 0.4304
```

Para evaluar la normalidad del error, utilizo el gráfico de $Q-Q$ de los residuales. En este caso el modelo se acerca bastante a la línea recta, con algunos valores *raros* que se alejan, que corresponden a los con valores mas altos de residuales, y que habría que examinar mejor.

Por otra parte podemos utilizar el test de Shapiro-Wilk cuya H_0 es que el error no difiere de la distribución normal. En este caso no hay evidencia para rechazar la H_0

(c) Check for large leverage points.

```
halfnorm(hatvalues(model), labs = rownames(sat))
```



```
augment(model) %>%
  filter(.hat > 2*(p/n))
```

```
## # A tibble: 2 x 13
##   .rownames total expend salary ratio takers .fitted .se.fit .resid .hat
##   <chr>      <int> <dbl> <dbl> <dbl> <int> <dbl> <dbl> <dbl> <dbl>
## 1 Californ~  902  4.99  41.1  24      45   918.   17.4 -15.8 0.282
## 2 Utah      1076  3.66  29.1  24.3     4  1010.   17.7  65.8 0.292
## # ... with 3 more variables: .sigma <dbl>, .cooksad <dbl>, .std.resid <dbl>
```

Para evaluar los valores extremos en el modelo, utilizamos dos gráficos.

- La gráfica de los valores ajustados versus los residuales estandarizados ($\hat{y} \times \sqrt{|\hat{\epsilon}|}$)
- Una gráfica de *media normal (halfnormal)* versus los residuales estandarizados. De esta manera se pueden identificar los valores mas extremos en X .

En este caso podemos ver que los valores para California y Utah cumplen con las características para valores extremos.

(d) Check for outliers.

```
(outliers <-
  augment(model) %>%
  mutate(.t.resid = rstudent(model)) %>%
  arrange(-abs(.t.resid)) %>%
  head(5))
```

```
## # A tibble: 5 x 14
##   .rownames total expend salary ratio takers .fitted .se.fit .resid .hat
##   <chr>      <int> <dbl> <dbl> <dbl> <int> <dbl> <dbl> <dbl> <dbl>
## 1 West Vir~  932   6.11  31.9  14.8    17  1023.   8.15 -90.5 0.0621
## 2 Utah      1076   3.66  29.1  24.3     4  1010.  17.7   65.8 0.292
## 3 North Da~ 1107   4.78  26.3  15.3     5  1040.   9.31  66.6 0.0810
## 4 New Hamp~  935   5.86  34.7  15.6    70   869.   9.41  65.9 0.0828
## 5 Nevada     917   5.16  34.8  18.7    30   971.   6.97 -54.1 0.0454
## # ... with 4 more variables: .sigma <dbl>, .cooksdi <dbl>,
## #   .std.resid <dbl>, .t.resid <dbl>
```

```
abs(outliers$.t.resid) > abs(qt(.05/(n * 2), n - p))
```

```
## [1] FALSE FALSE FALSE FALSE FALSE
```

Para valorar posibles *outliers* podemos utilizar la estrategia planteada en el libro de Faraway, y calcular los residuales *studentizados*. En este sentido los residuales cuyo valor superen un limite determinado por la corrección de Bonferroni, se podrían considerar posibles *outliers*.

En este caso el valor que mas puede ser un outlier es West Virginia, aunque el valor de *.t.resid* de west virgina no es mayor al valor limite de p corregido.

```
model_1 <-
  sat %>%
  rownames_to_column() %>%
  filter(rownames != "West Virginia") %>%
  lm(total ~ expend + salary + ratio + takers, data = .)

tidy(model)
```

```
## # A tibble: 5 x 5
##   term          estimate std.error statistic p.value
##   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)  1046.      52.9      19.8  7.86e-24
## 2 expend         4.46     10.5      0.423 6.74e- 1
## 3 salary         1.64      2.39      0.686 4.96e- 1
## 4 ratio        -3.62      3.22     -1.13 2.66e- 1
## 5 takers        -2.90      0.231    -12.6 2.61e-16
```

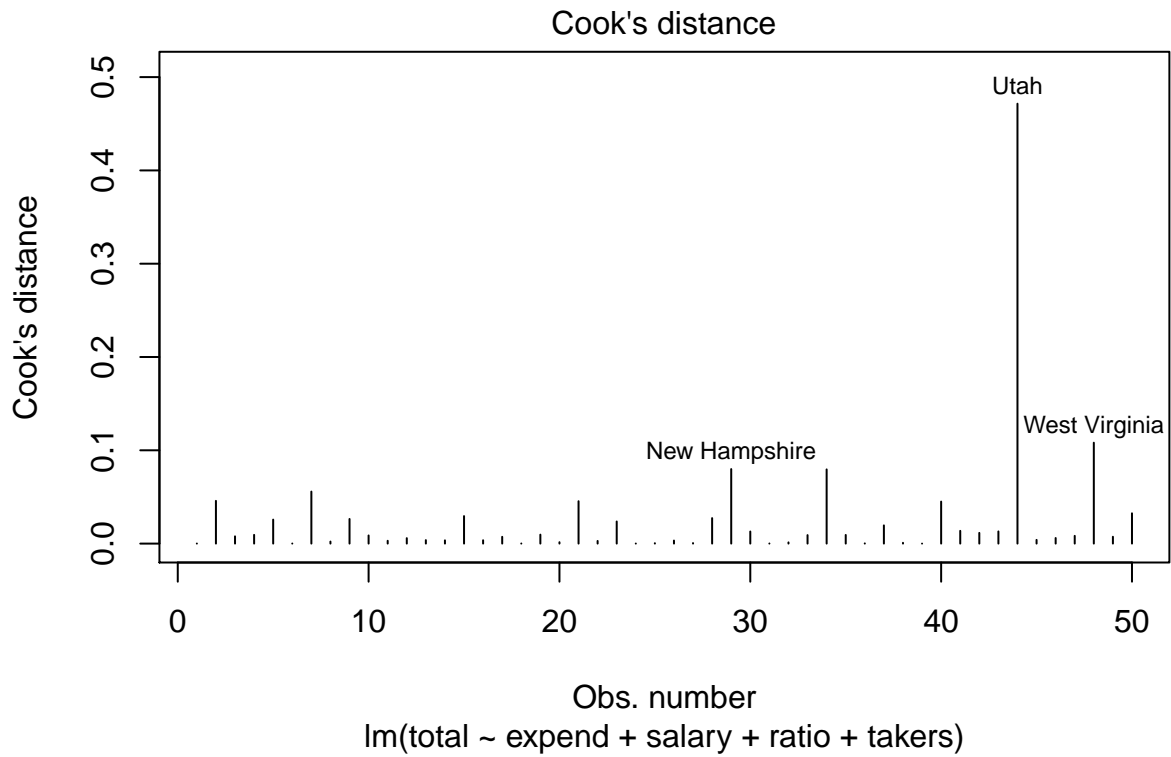
```
tidy(model_1)
```

```
## # A tibble: 5 x 5
##   term          estimate std.error statistic p.value
##   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)  1058.      48.5     21.8  3.26e-25
## 2 expend         7.36     9.69      0.759 4.52e- 1
## 3 salary         1.09      2.19      0.496 6.22e- 1
## 4 ratio        -3.94      2.94     -1.34 1.88e- 1
## 5 takers        -2.97      0.213    -14.0 8.68e-18
```

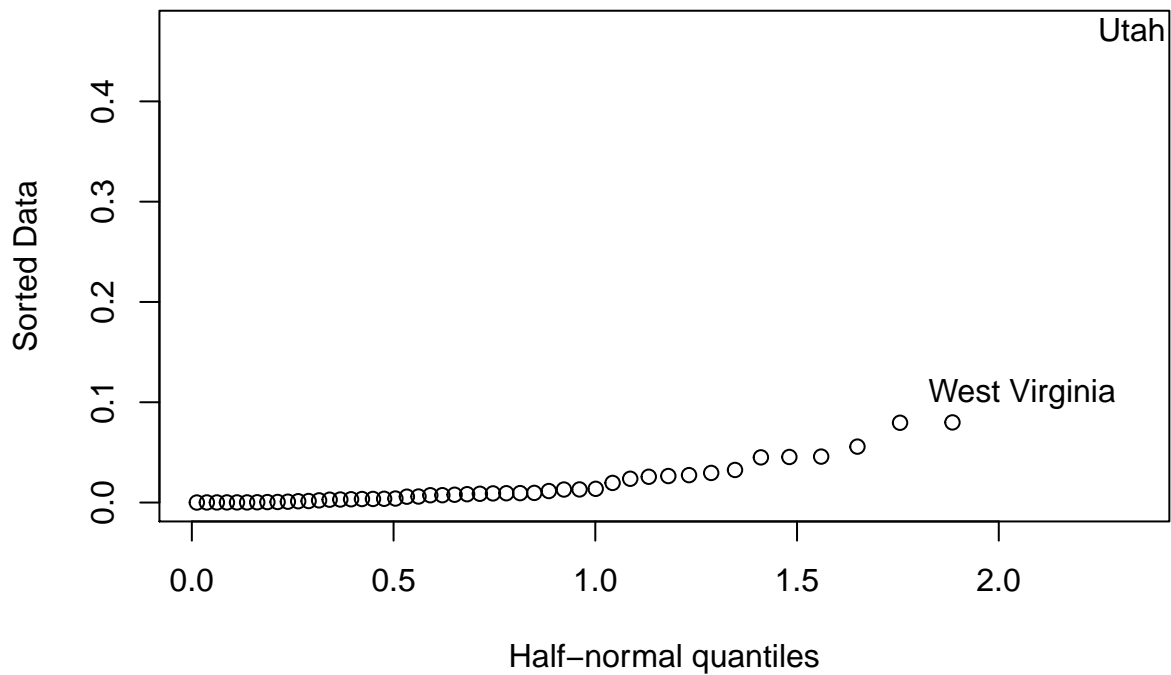
Como West Virginia es un valor extremo (esta lejos de la linea del 0 en el plot de $\hat{y} \times \sqrt{|\hat{\epsilon}|}$)

(e) Check for influential points.

```
plot(model, 4)
```



```
halfnorm(cooks.distance(model), labs = rownames(sat))
```



```
model_2 <-  
  sat %>%  
  rownames_to_column() %>%  
  filter(rowname != "Utah") %>%  
  lm(total ~ expend + salary + ratio + takers, data = .)
```

```
tidy(model)
```

```
## # A tibble: 5 x 5
##   term      estimate std.error statistic  p.value
##   <chr>      <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept) 1046.      52.9      19.8  7.86e-24
## 2 expend       4.46     10.5       0.423  6.74e- 1
## 3 salary       1.64      2.39       0.686  4.96e- 1
## 4 ratio      -3.62      3.22      -1.13  2.66e- 1
## 5 takers      -2.90      0.231    -12.6  2.61e-16
```

```
tidy(model_1)
```

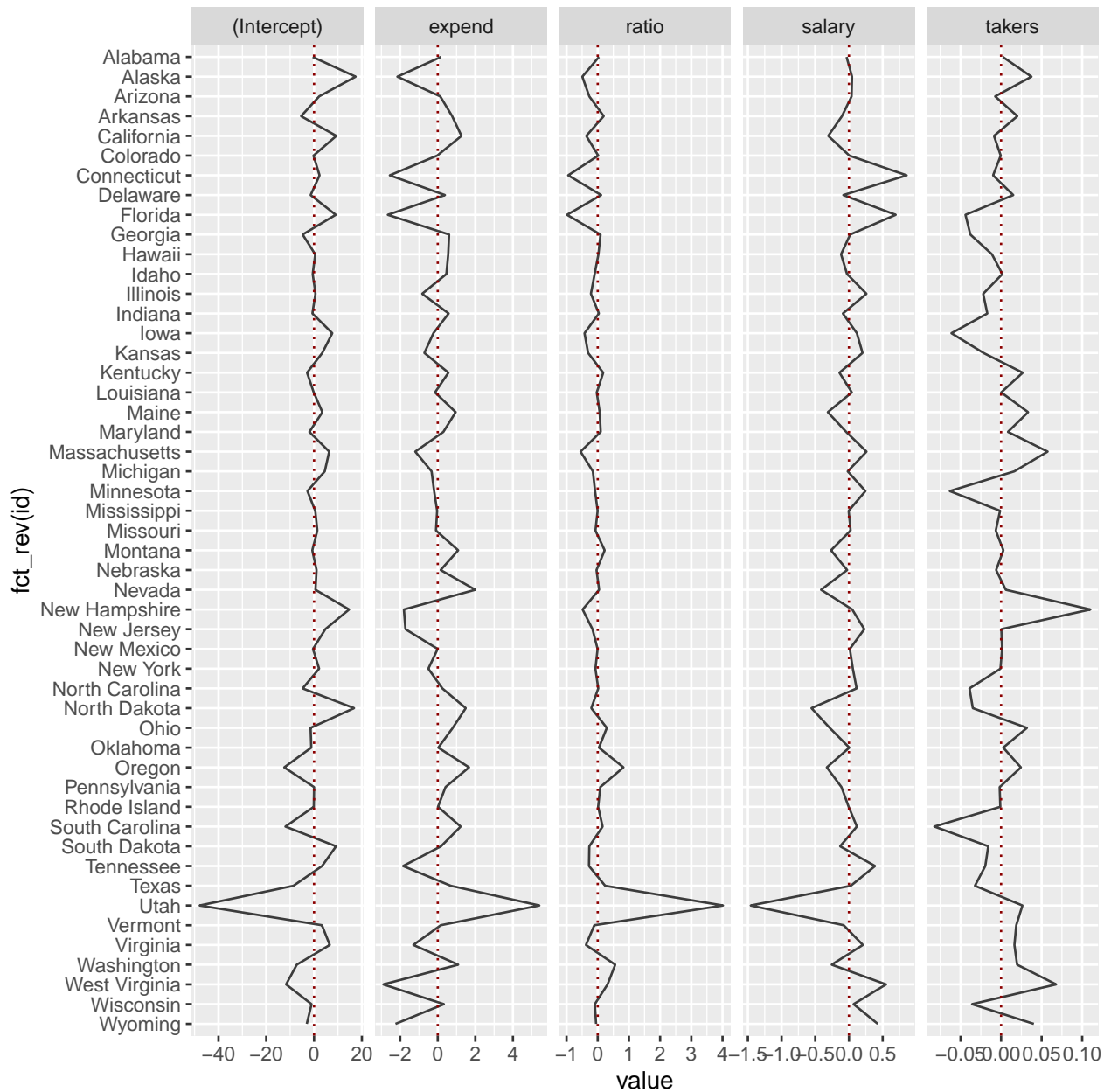
```
## # A tibble: 5 x 5
##   term      estimate std.error statistic  p.value
##   <chr>      <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept) 1058.      48.5      21.8  3.26e-25
## 2 expend       7.36     9.69       0.759  4.52e- 1
## 3 salary       1.09      2.19       0.496  6.22e- 1
## 4 ratio      -3.94      2.94      -1.34  1.88e- 1
## 5 takers      -2.97      0.213    -14.0  8.68e-18
```

```
tidy(model_2)
```

```
## # A tibble: 5 x 5
##   term      estimate std.error statistic  p.value
##   <chr>      <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept) 1094.      53.4      20.5  4.04e-24
## 2 expend     -0.943     10.2     -0.0925  9.27e- 1
## 3 salary       3.10      2.33       1.33  1.90e- 1
## 4 ratio      -7.64      3.43      -2.23  3.10e- 2
## 5 takers      -2.93      0.219    -13.4  3.95e-17
```

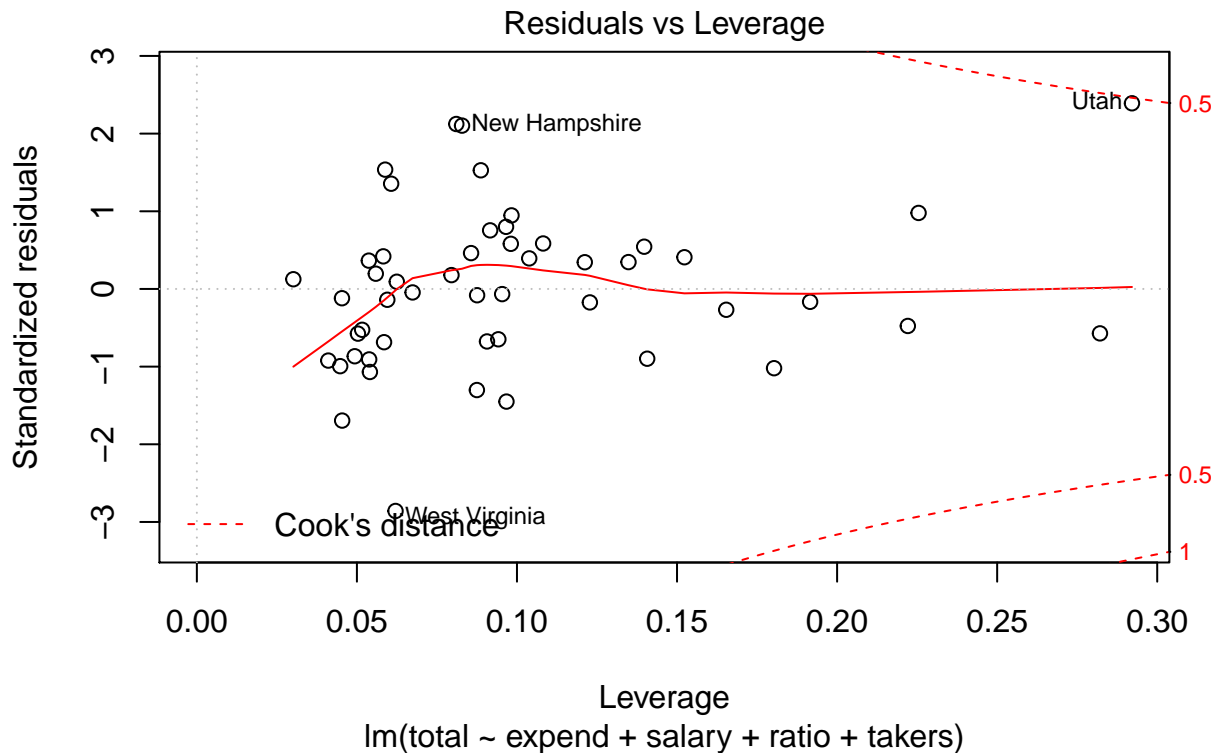
Para evaluar los puntos mas influyentes podemos utilizar la *distancia de Cook* que refleja los cambios en el modelo al no incluir una observación determinada. En este caso destacan Utah y West Virginia como observaciones influyentes. Además podemos volver a mirar el gráfico de *Residuales v/s leverage*, que también refleja puntos con un grado alto de *palanca* y un residual alto.

```
influence(model)$coefficients %>%
  as_tibble(rownames = "id") %>%
  gather(key, value, -id) %>%
  ggplot(aes(fct_rev(id), value, group = key)) +
  geom_line(alpha = .75)+
  coord_flip() +
  geom_hline(yintercept = 0, linetype = 3, color = "darkred") +
  facet_grid(~ key, scales = "free") +
  scale_colour_viridis_d(end=.85)
```



La función **influence** permite obtener los coeficientes al restar una observación x_i (“leave one out”). De esta manera puedo construir un gráfico que revela los cambios en los diferentes coeficientes para cada una de las observaciones. En este gráfico se ve claramente como **Utah** genera los mayores cambios en todos los coeficientes.

```
plot(model, 5)
```



```
sat[44,]
```

```
##      expend ratio salary takers verbal math total
## Utah  3.656  24.3 29.082      4    513  563  1076
```

El modelo multivariable es explicativo, con un valor de R^2 relativamente alto. En el modelo el gasto por estudiante **expend** y el sueldo **salary** se relacionan de manera positiva con el resultado del test **total**. La **ratio** de profesores y el porcentaje de estudiantes que hacen el examen de manera negativa.

Utah es un caso *raro* en la medida que tiene un score muy alta en relación a un gasto **expend** y un sueldo **salary** inferior al promedio por una parte, y una **ratio**. El porcentaje de alumnos elegibles **takers** llama la atención, ya que parece extremadamente bajo en relación a los otros parámetros.

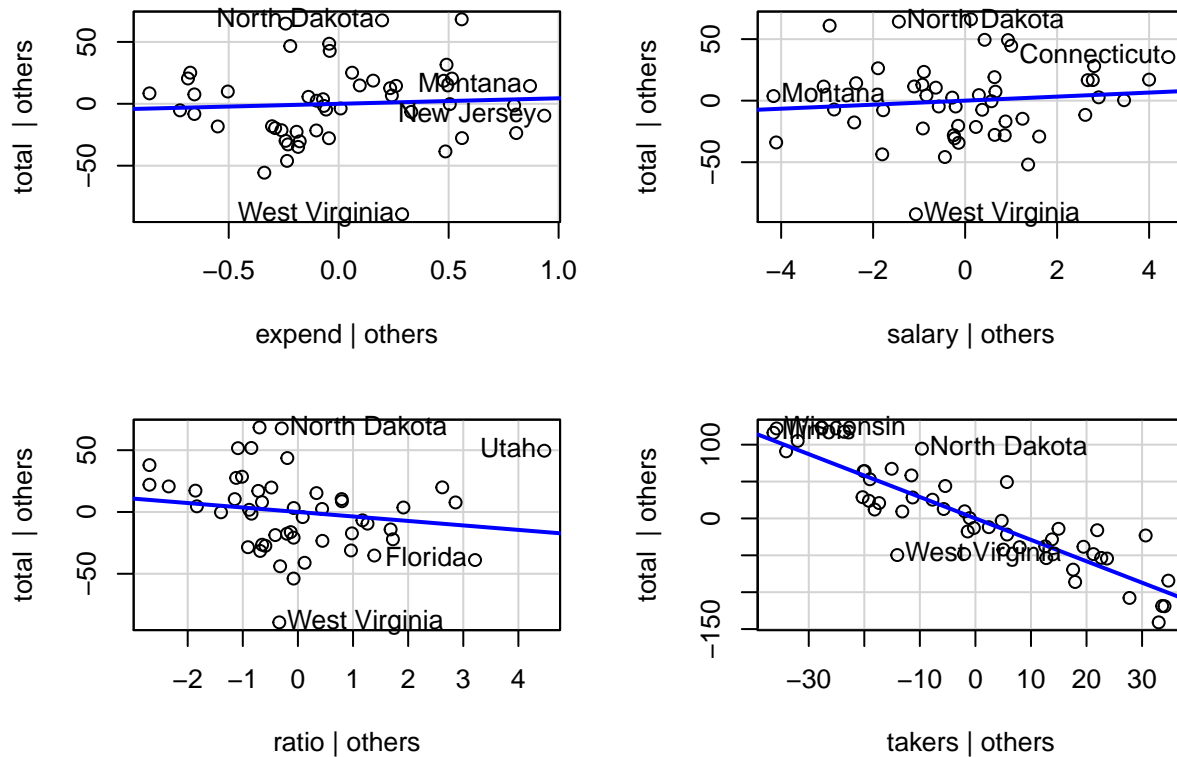
(f) Check the structure of the relationship between the predictors and the response.

En primer lugar me remito al gráfico de dispersión de (a), que permite reflejar la distribución y la relación 1 a 1 de las variables.

```
# added variable
car::avPlots(model)
```

```
## Registered S3 methods overwritten by 'car':
##   method                      from
##   influence.merMod             lme4
##   cooks.distance.influence.merMod lme4
##   dfbeta.influence.merMod      lme4
##   dfbetas.influence.merMod    lme4
```

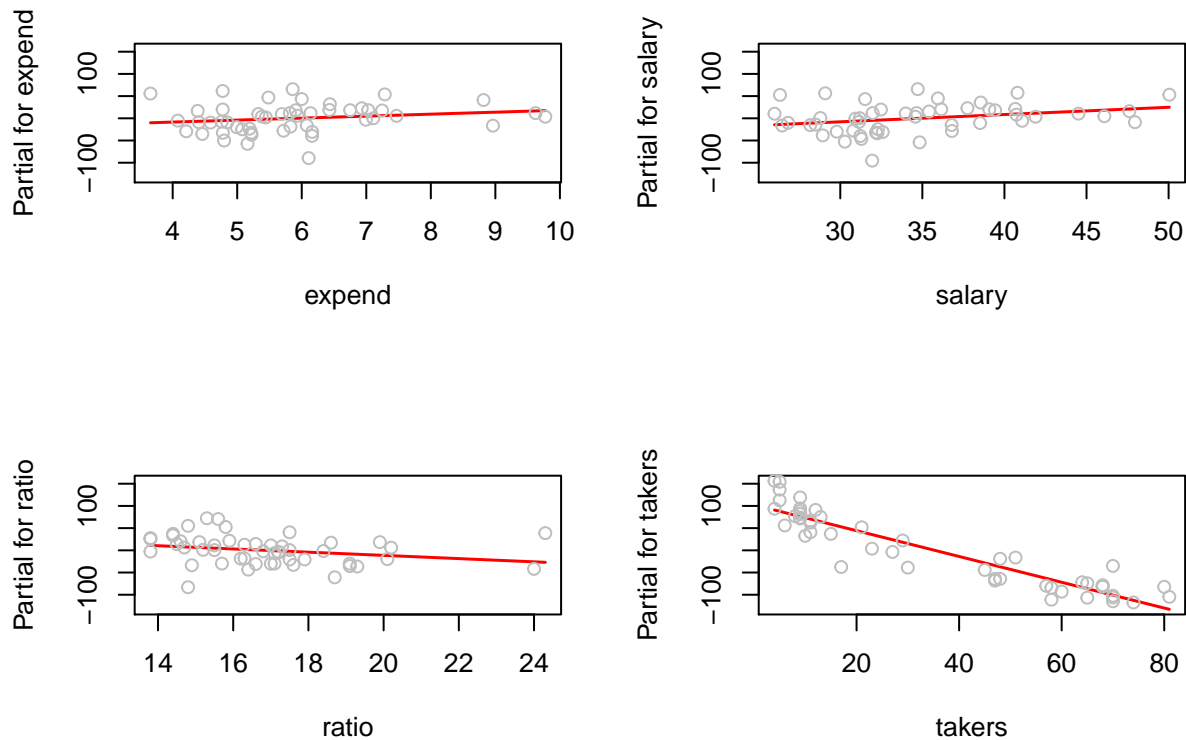

Added-Variable Plots



Para evaluar la estructura del modelo son de utilidad los gráficos de regresión parcial. La regresión parcial permite *aislar* la influencia de una variable independiente con la variable dependiente, restando la influencia de las otras variables independientes. Además permiten valorar la presencia de *outliers* y puntos influyentes.

En este caso podemos ver la clara relación inversa de la variable **takers** con el **total** y la relación mas débil de las otras variables. Podemos observar además el caso *raro* de Utah

```
par(mfrow = c(2,2))
termplot(model, partial.resid = T)
```



Los gráficos de residuales parciales, permiten conocer la respuesta aislando el *efecto* del resto de variables independientes. También permiten poder observar la presencia de diferencias estructurales, de no-linealidad

2. (Ejercicio 2 cap. 6 pág. 97)

Using the `teengamb` dataset, fit a model with `gamble` as the response and the other variables as predictors. Answer the questions posed in the previous question.

```
n <- dim(teengamb)[1]
p <- dim(teengamb)[2]

teengamb_cl <-
  teengamb %>%
  rownames_to_column(".rownames") %>%
  mutate(.rownames = factor(.rownames),
         sex = factor(sex, levels = c(0,1), labels = c('male', 'female')))

model <-
  lm(gamble ~ sex + status + income + verbal, data = teengamb_cl)

teengamb_cl %>%
  select(gamble, sex, status, income, verbal) %>%
  ggpairs(progress = F, aes(color = sex), lower = list(combo = wrap("facethist", bins = 10)))
```



En este caso gráfico con utilizando `sex` como un factor.

(a) Check the constant variance assumption for the errors.

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

En este caso el gráfico podría definir cierta heterocedasticidad en la medida que a mayor valor de \hat{y} parece haber mayor dispersión del error.

```
var.test(resid(model)[teengamb_cl$sex == "female"], resid(model)[teengamb_cl$sex == "male"])

##
## F test to compare two variances
##
## data: resid(model)[teengamb_cl$sex == "female"] and resid(model)[teengamb_cl$sex == "male"]
## F = 0.31604, num df = 18, denom df = 27, p-value = 0.01374
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
## 0.1379388 0.7808184
## sample estimates:
## ratio of variances
## 0.3160421
```

En este caso al menos en cuanto a las poblaciones según sexo, hay una diferencia significativa en las varianzas

(b) Check the normality assumption.

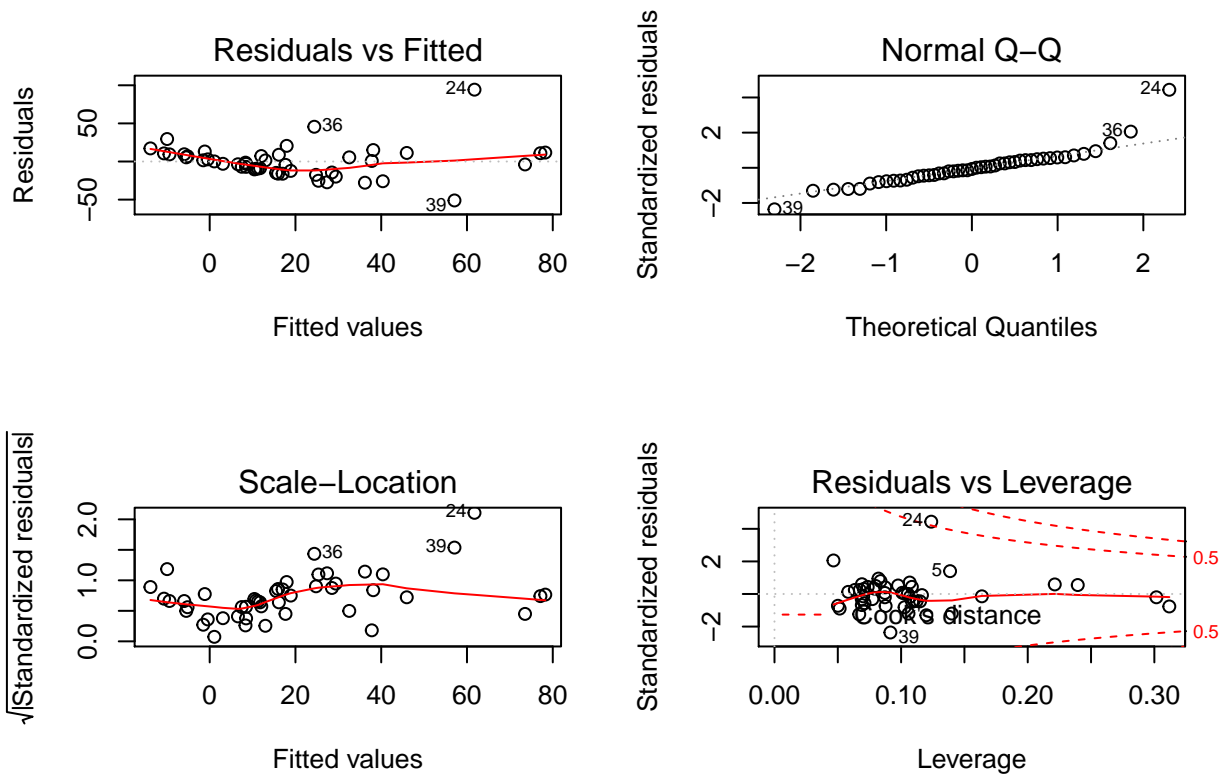


Figure 2: Gráficos Diagnósticos

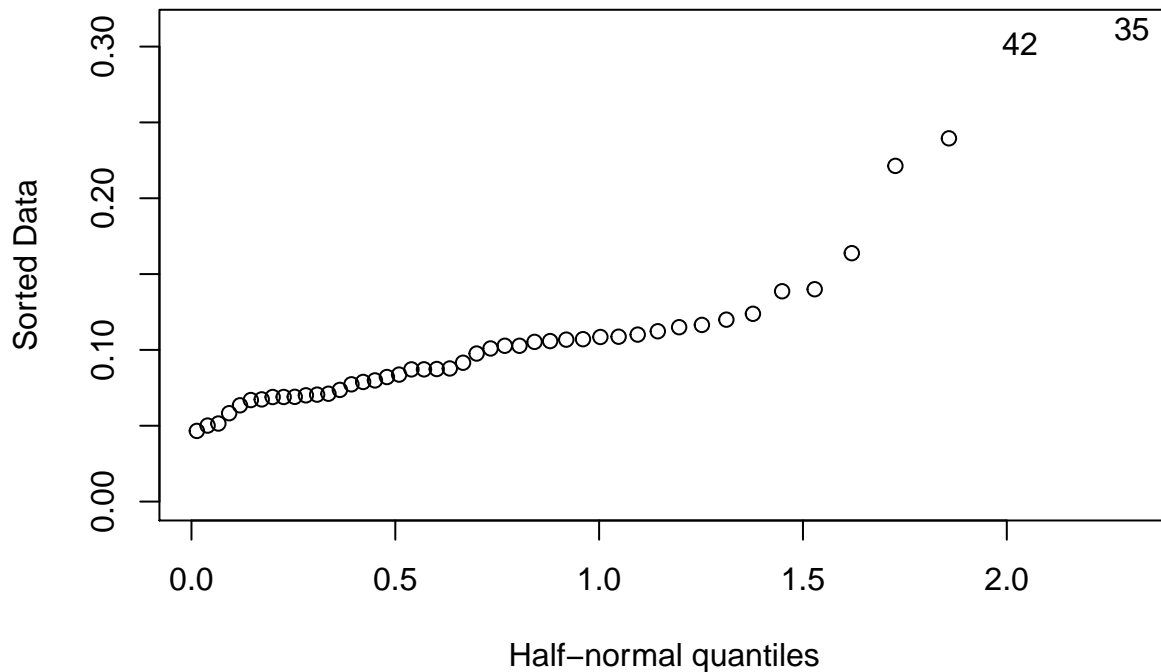
```
shapiro.test(resid(model))
```

```
##
## Shapiro-Wilk normality test
##
## data: resid(model)
## W = 0.86839, p-value = 8.16e-05
```

En este caso podemos observar que el gráfico $Q-Q$ difiere de la normalidad con un patrón de *cola larga*. Por otra parte la preba de Shapiro Wilk refleja que se rechaza la H_0 de normalidad de la distribución del error.

(c) Check for large leverage points.

```
halfnorm(hatvalues(model), labs = teengamb_cl$.rownames)
```



```
augment(model) %>%
  mutate(.rownames = teengamb_cl$.rownames) %>%
  filter(.hat > 2*(p/n))
```

```
## # A tibble: 4 x 13
##   gamble sex   status income verbal .fitted .se.fit .resid .hat .sigma
##   <dbl> <fct> <int> <dbl> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1    88  male     18    12      2   77.1   11.1  10.9  0.240  22.9
## 2    90  male     38    15      7   78.3   10.7  11.7  0.221  22.9
## 3   14.1 male     28   1.5      1   28.5   12.7 -14.4  0.312  22.8
## 4   69.7 male     61    15      9   73.5   12.5 -3.84  0.302  23.0
## # ... with 3 more variables: .cooksd <dbl>, .std.resid <dbl>,
## #   .rownames <fct>
```

En este caso podemos ver que los valores para 42 y 35 cumplen con las características para valores extremos. Habría que revisar también los 31 y 33.

(d) Check for outliers.

```
augment(model) %>%
  mutate(.rownames = teengamb_cl$.rownames) %>%
  mutate(.t.resid = rstudent(model)) %>%
  filter(abs(.t.resid) > abs(qt(.05/(n * 2), n - p)))
```

```
## # A tibble: 1 x 14
##   gamble sex   status income verbal .fitted .se.fit .resid .hat .sigma
##   <dbl> <fct> <int> <dbl> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1   156 male     27    10      4   61.7   7.98  94.3  0.124  16.7
## # ... with 4 more variables: .cooksd <dbl>, .std.resid <dbl>,
## #   .rownames <fct>, .t.resid <dbl>
```

En este caso el valor 24 parece ser un *outlier*

```
model_1 <-
  teengamb_cl %>%
  filter(.rownames != "24") %>%
  lm(gamble ~ sex + status + income + verbal, data = .)
```

```
tidy(model)
```

```
## # A tibble: 5 x 5
##   term      estimate std.error statistic  p.value
##   <chr>      <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept) 22.6      17.2      1.31  0.197
## 2 sexfemale  -22.1       8.21     -2.69  0.0101
## 3 status      0.0522    0.281     0.186  0.853
## 4 income       4.96     1.03      4.84  0.0000179
## 5 verbal      -2.96     2.17     -1.36  0.180
```

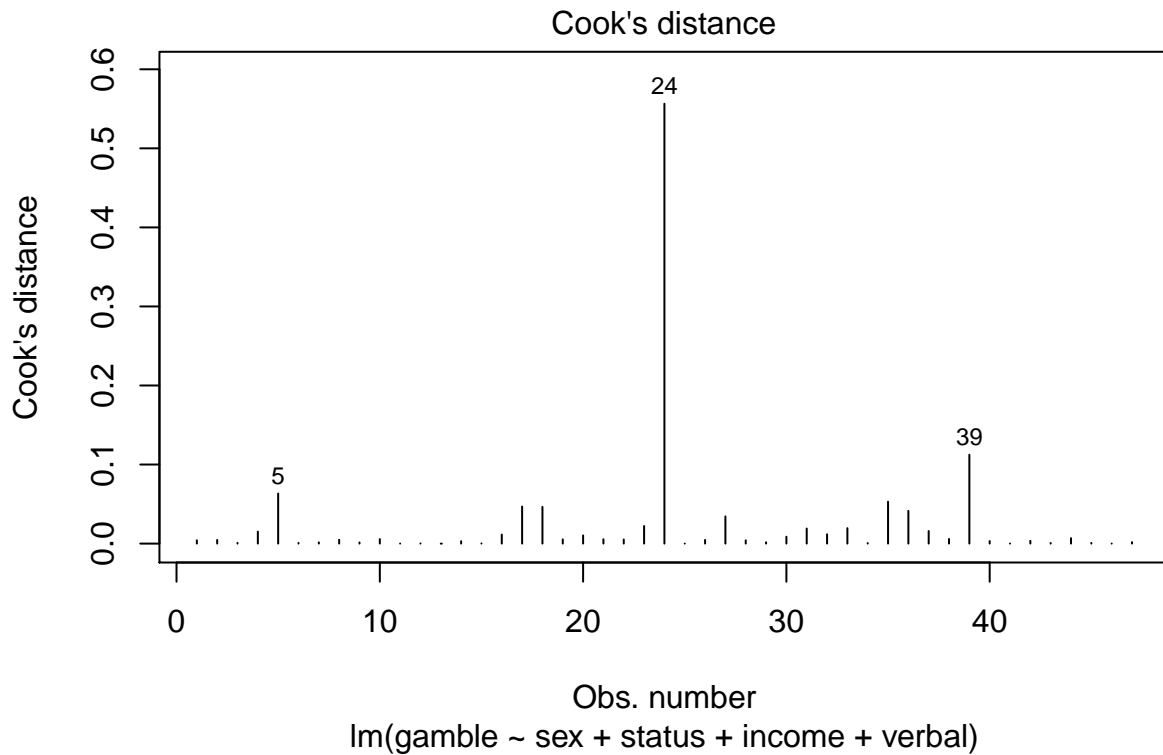
```
tidy(model_1)
```

```
## # A tibble: 5 x 5
##   term      estimate std.error statistic  p.value
##   <chr>      <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)  7.63     12.9     0.590  0.558
## 2 sexfemale  -16.3      6.13    -2.66  0.0112
## 3 status      0.174    0.208     0.835  0.409
## 4 income       4.33    0.764     5.67  0.00000126
## 5 verbal      -1.80     1.61    -1.12  0.271
```

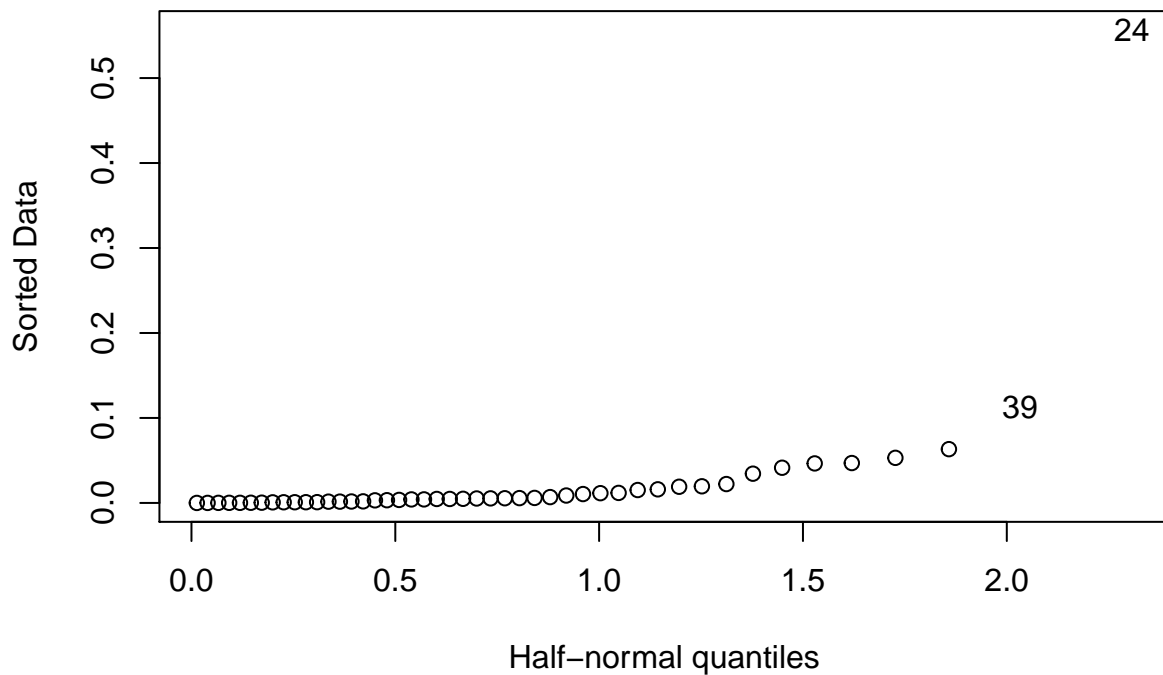
La exclusión de la observación 24 genera varios cambios en todos los coeficientes del modelo. Además de ser un candidato a *outlier* es un valor muy influyente

(e) Check for influential points.

```
plot(model, 4)
```



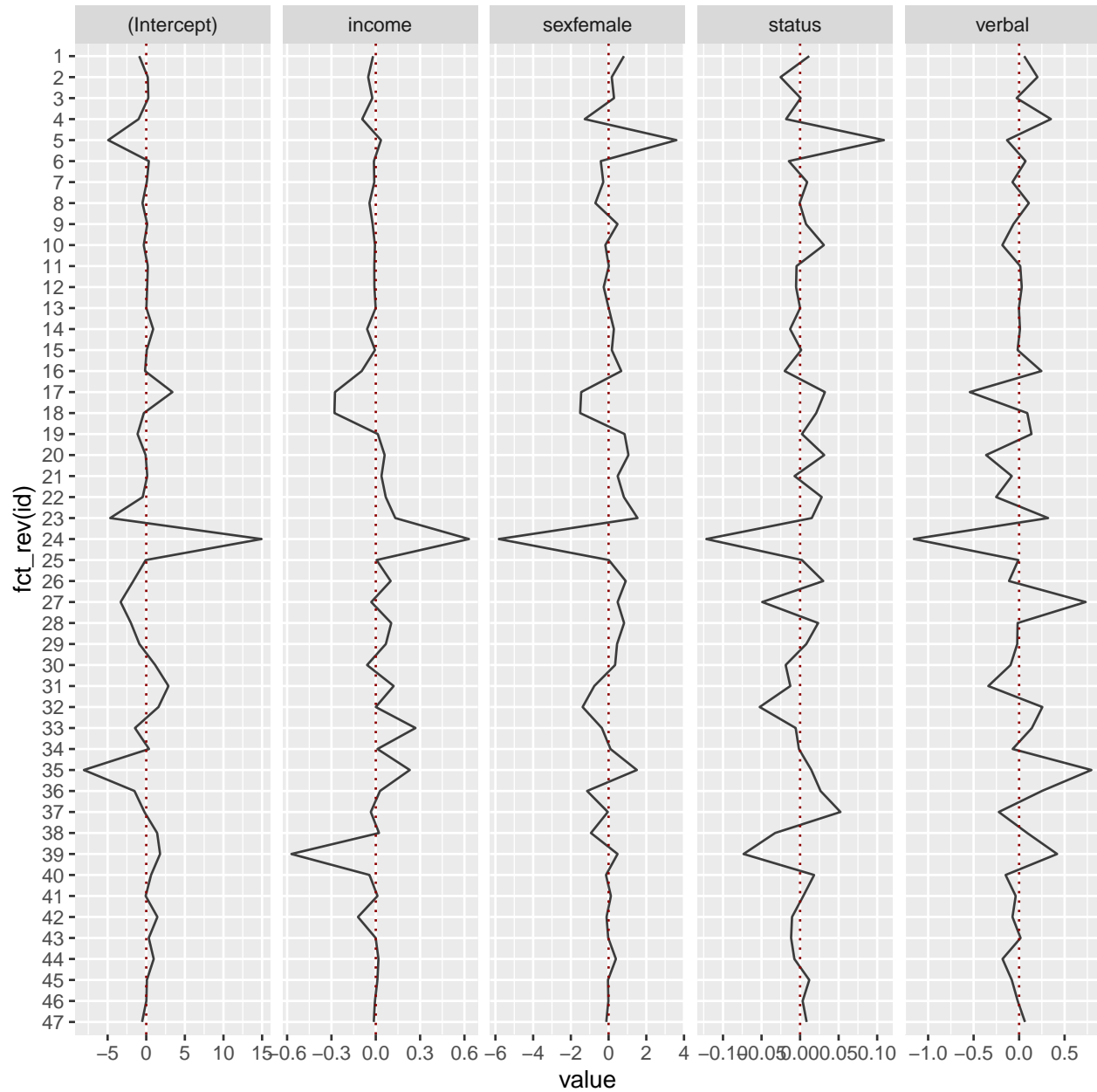
```
halfnorm(cooks.distance(model), labs = teengamb_cl$.rownames)
```



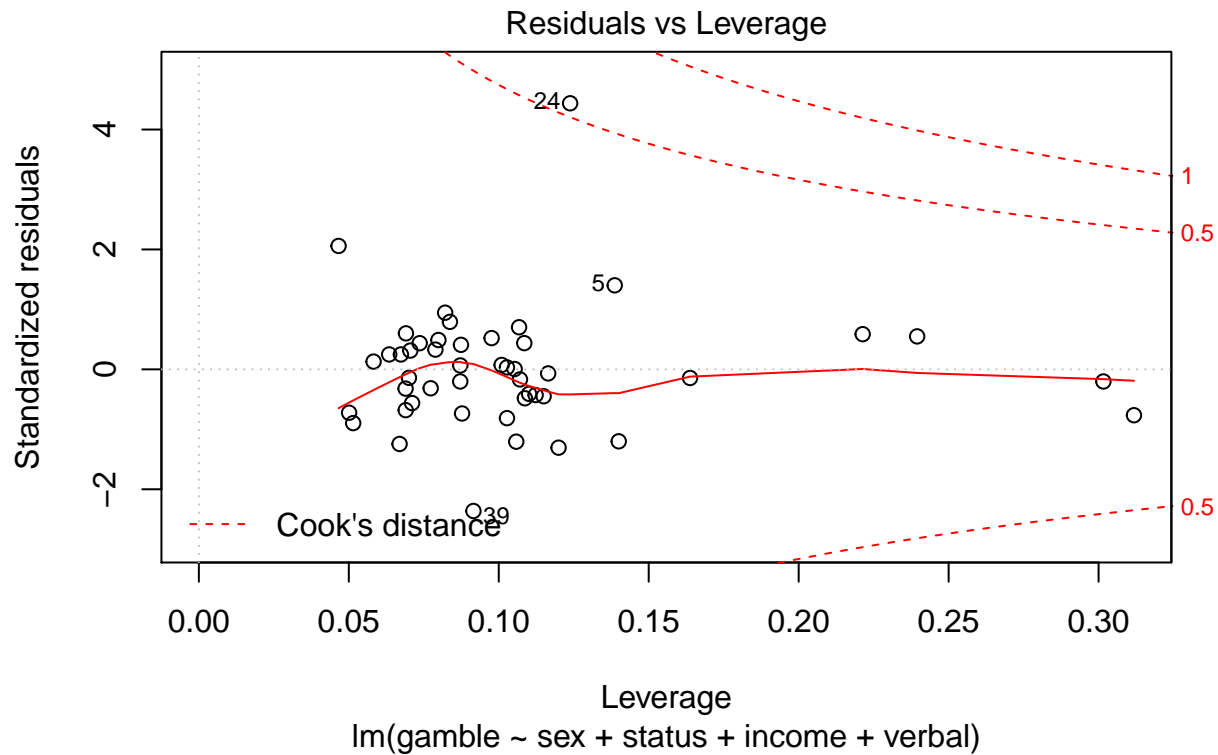
El gráfico de distancias de Cook confirma que el valor 24 se aleja mucho del comportamiento de las otras variables, que es un valor influyente y probablemente un outlier.

```
influence(model)$coefficients %>%
  as_tibble(rownames = "id") %>%
  mutate(id = factor(id, levels = c(1:n))) %>%
  gather(key, value, -id) %>%
```

```
ggplot(aes(fct_rev(id), value, group = key)) +
  geom_line(alpha = .75)+
  coord_flip() +
  geom_hline(yintercept = 0, linetype = 3, color = "darkred") +
  facet_grid(~ key, scales = "free")
```



```
plot(model, 5)
```

```
tidy(model)
```

```
## # A tibble: 5 x 5
##   term      estimate std.error statistic  p.value
##   <chr>      <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)  22.6      17.2      1.31  0.197
## 2 sexfemale  -22.1       8.21     -2.69  0.0101
## 3 status      0.0522    0.281     0.186  0.853
## 4 income       4.96     1.03      4.84  0.0000179
## 5 verbal      -2.96     2.17     -1.36  0.180
```

```
teengamb_cl[24,]
```

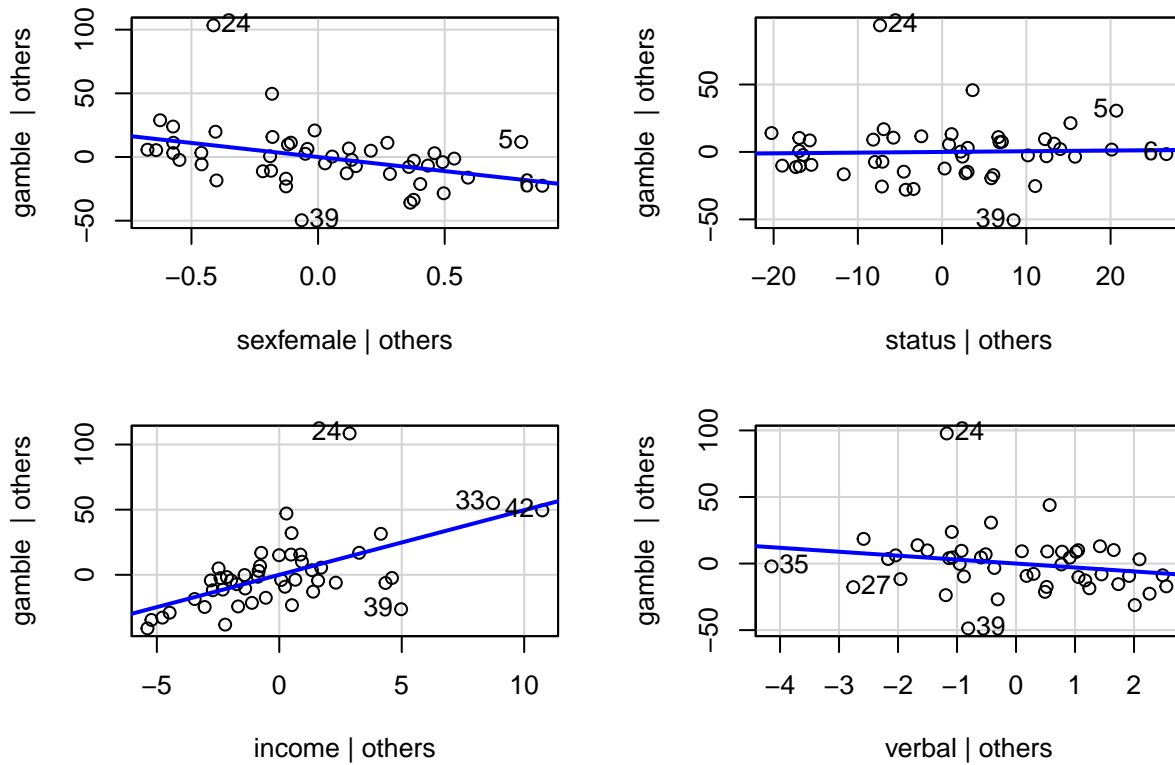
```
##   .rownames sex status income verbal gamble
## 24      24 male   27    10     4    156
```

(f) Check the structure of the relationship between the predictors and the response.

En primer lugar me remito al gráfico de dispersión de (a), que permite reflejar la distribución y la relación 1 a 1 de las variables.

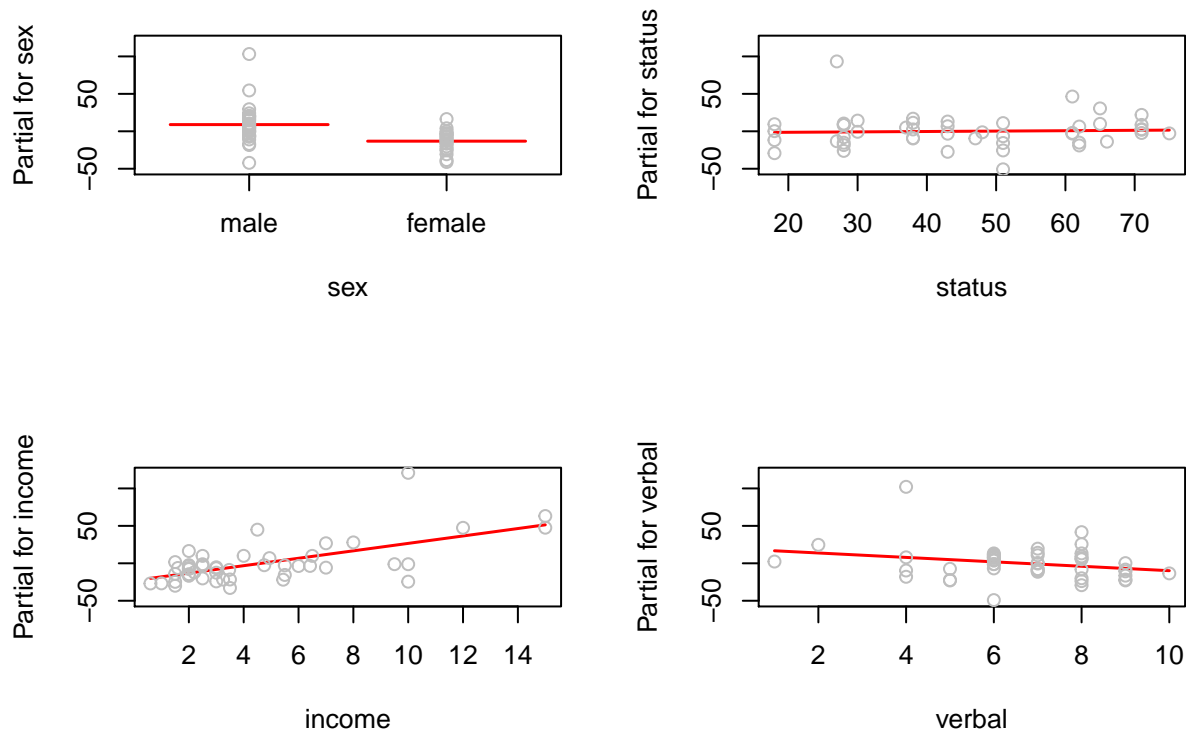
```
car::avPlots(model)
```

Added-Variable Plots



En este caso el modelo refleja la relación directa del sexo masculino y del `income` con `gamble`, y la relación inversa suave con el score `verbal`. Es posible apreciar claramente que la observación 24 es un caso *raro* que se sale de las líneas de predicción para todos los coeficientes.

```
par(mfrow = c(2,2))
termplot(model, partial.resid = T)
```



En este caso los parciales de residuales reflejan lo que se ve en el primer gráfico de dispersión y es que hay variables que no tienen una distribución normal, como `income` y `verbal`, que podrían hacer pensar en hacer que transformaciones de las variables mejorarían la estructura del modelo.

3. (Ejercicio 3 cap. 6 pág. 97)

For the `prostate` data, fit a model with `lpsa` as the response and the other variables as predictors. Answer the questions posed in the first question.

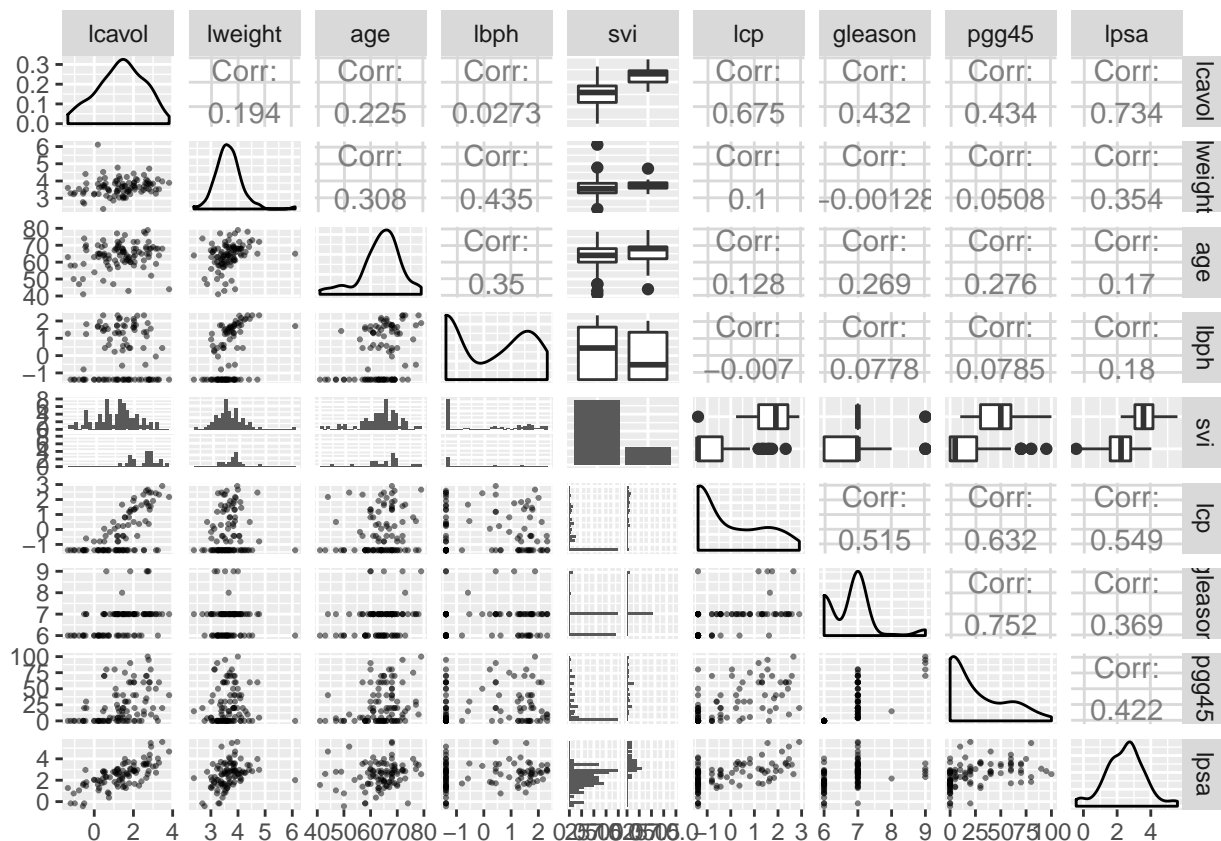
```
n <- dim(prostate)[1]
p <- dim(prostate)[2]

prostate_cl <-
  prostate %>%
  mutate(svi = as.logical(svi))

model <-
  lm(lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason + pgg45, data = prostate_cl)

ggpairs(prostate_cl, progress = F, lower = list(continuous = wrap("points", alpha = 0.5, size = .4)))

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Lo primero que hago es crear una matriz de dispersión para tener una idea de como se distribuyen las variables y la relación 2x2 que tienen entre si. Utilizo la librería `GGally` que tiene la función `ggpairs`.

(a) Check the constant variance assumption for the errors.

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

Para evaluar la homocedasticidad del error utilizo los gráficos de:

- residuales v/s valor ajustado
- residuales estandarizados v/s valor ajustado.

En este caso no se observa un patrón definido por lo que se asume homocedasticidad.

(b) Check the normality assumption.

```
shapiro.test(resid(model))
```

```
##
## Shapiro-Wilk normality test
##
## data: resid(model)
## W = 0.99113, p-value = 0.7721
```

La gráfica $Q-Q$ parece bastante pareja y aproximada a la linea. El test de Shapiro Wilk confirma la normalidad del error

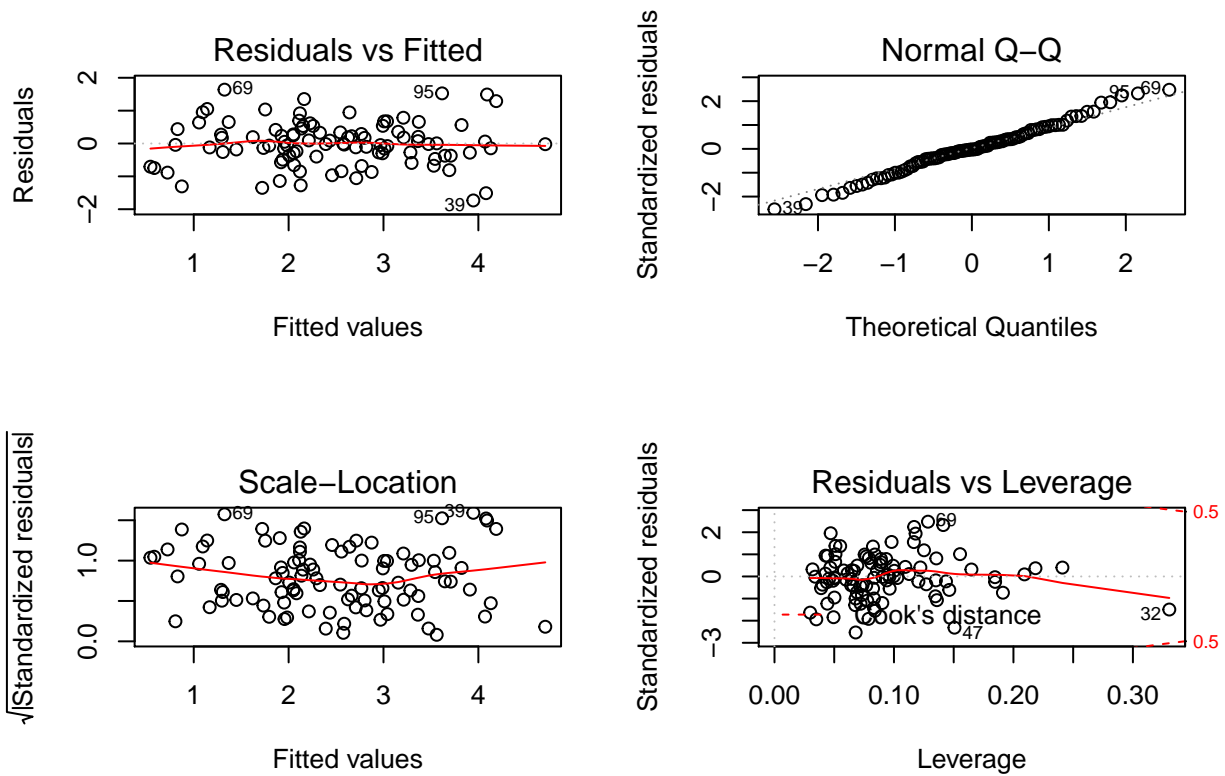
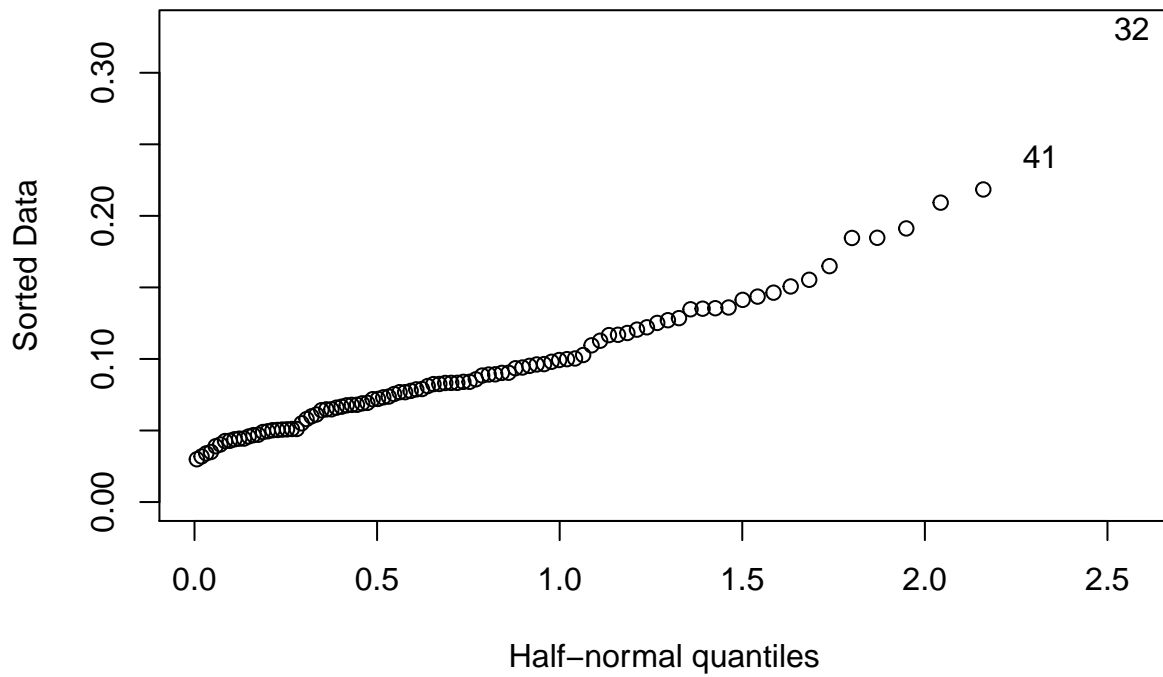


Figure 3: Gráficos Diagnósticos

(c) Check for large leverage points.

```
halfnorm(hatvalues(model), labs = rownames(prostate))
```



```
augment(model) %>%
  mutate(.rownames = rownames(prostate)) %>%
  filter(.hat > 2*(p/n))
```

```
## # A tibble: 5 x 17
##   lpsa lcavol lweight age lbph svi lcp gleason pgg45 .fitted
##   <dbl> <dbl> <dbl> <int> <dbl> <lgl> <dbl> <int> <int> <dbl>
## 1  2.01  0.182   6.11   65  1.70 FALSE -1.39     6     0   2.88
## 2  2.16  1.42   3.66   73 -0.580 FALSE  1.66     8    15   1.93
## 3  2.30  0.621   3.14   60 -1.39 FALSE -1.39     9    80   2.05
## 4  3.08  1.84   3.24   60  0.438 TRUE  1.18     9    90   3.54
## 5  4.13  2.53   3.68   61  1.35 TRUE  -1.39     7    15   4.07
## # ... with 7 more variables: .se.fit <dbl>, .resid <dbl>, .hat <dbl>,
## #   .sigma <dbl>, .cooksdi <dbl>, .std.resid <dbl>, .rownames <chr>
```

Para evaluar los valores extremos en el modelo, utilizamos dos gráficos.

- La gráfica de los valores ajustados versus los residuales estandarizados ($\hat{y} \times \sqrt{|\hat{\epsilon}|}$)
- Una gráfica de *media normal* (*halfnormal*) versus los residuales estandarizados. De esta manera se pueden identificar los valores mas extremos en X .

En este caso podemos ver que los valores para el caso 32, 37 y el 41 cumplen con las características para valores extremos.

(d) Check for outliers.

```
augment(model) %>%
  mutate(.t.resid = rstudent(model)) %>%
  filter(abs(.t.resid) > abs(qt(.05/(n * 2), n - p)))
```

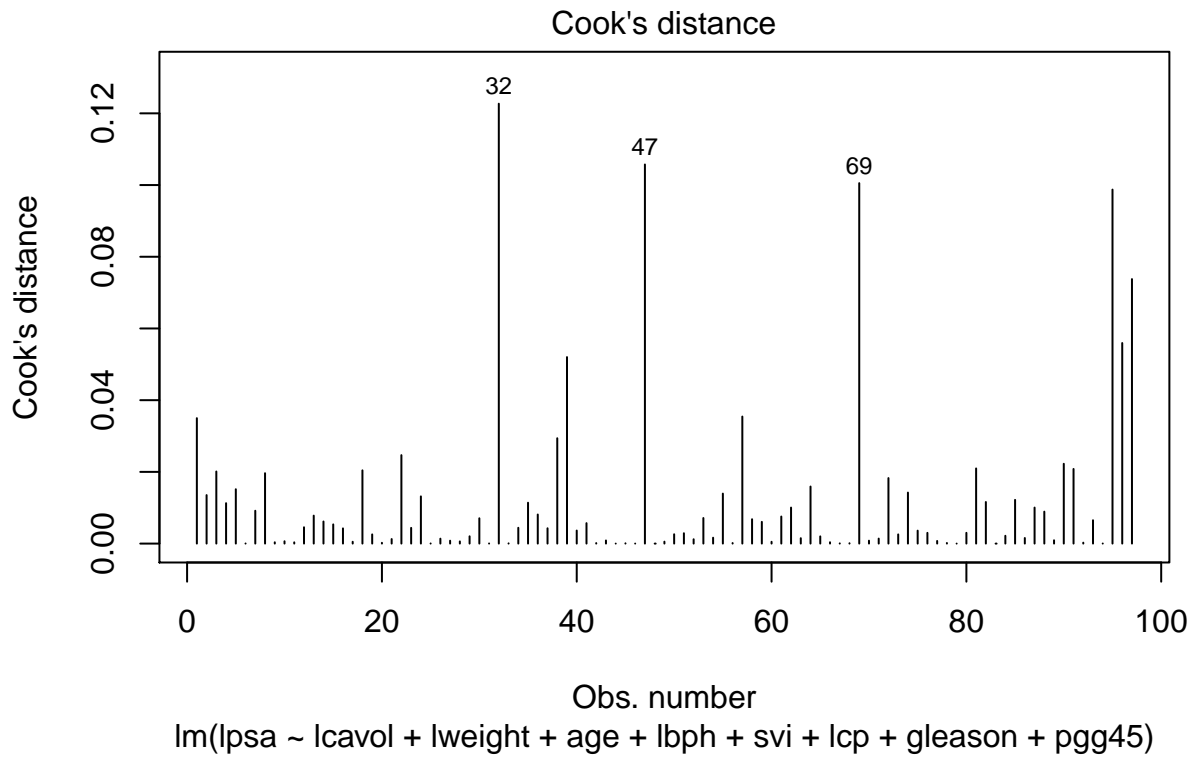
```
## # A tibble: 0 x 17
## # ... with 17 variables: lpsa <dbl>, lcavol <dbl>, lweight <dbl>,
## #   age <int>, lbph <dbl>, svi <lgl>, lcp <dbl>, gleason <int>,
## #   pgg45 <int>, .fitted <dbl>, .se.fit <dbl>, .resid <dbl>, .hat <dbl>,
## #   .sigma <dbl>, .cooksdi <dbl>, .std.resid <dbl>, .t.resid <dbl>
```

Para valorar posibles *outliers* podemos utilizar la estrategia planteada en el libro de Faraway, y calcular los residuales *studentizados*. En este sentido los residuales cuyo valor superen un limite determinado por la corrección de Bonferroni, se podrían considerar posibles *outliers*.

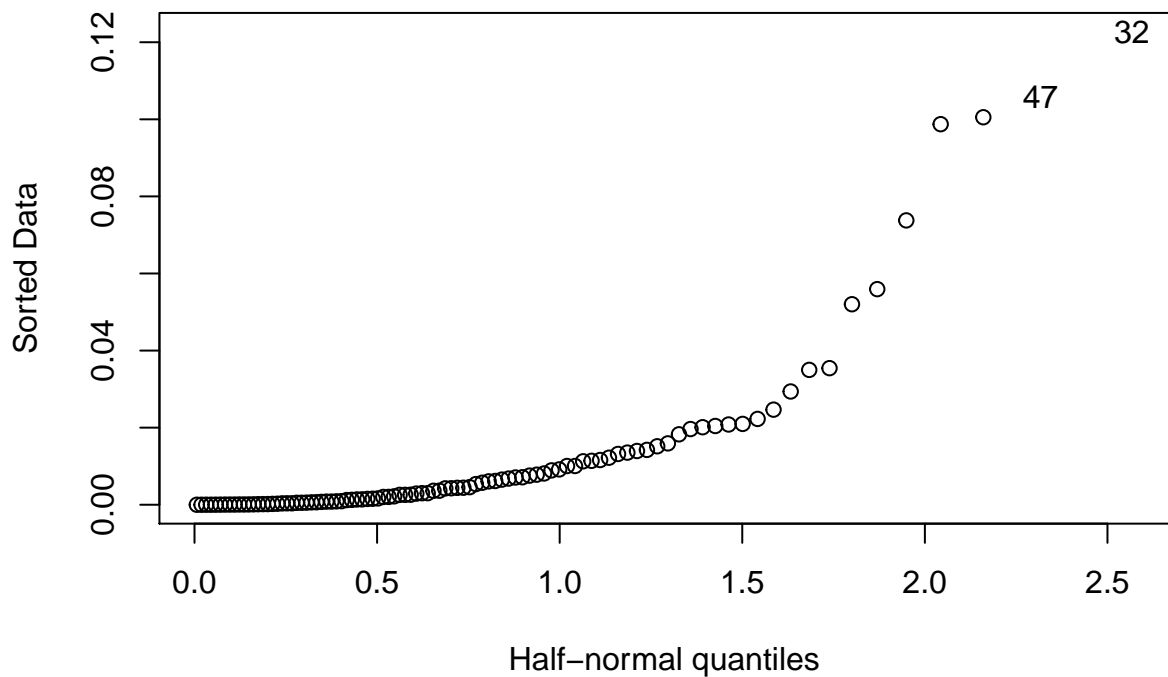
En este caso no hay valores que cumplan con lo propuesto

(e) Check for influential points.

```
plot(model, 4)
```



```
halfnorm(cooks.distance(model), labs = rownames(prostate))
```



```
model_1 <-
  prostate_cl %>%
  rownames_to_column() %>%
  filter(rowname != "32") %>%
  lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi + lcp +
    gleason + pgg45, data = .)
```

```
model_2 <-
  prostate_cl %>%
  rownames_to_column() %>%
  filter(rowname != "47") %>%
  lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi + lcp +
    gleason + pgg45, data = .)
```

```
tidy(model)
```

```
## # A tibble: 9 x 5
##   term          estimate std.error statistic    p.value
##   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)  0.669      1.30      0.516  0.607
## 2 lcavol       0.587     0.0879    6.68  0.00000000211
## 3 lweight      0.454     0.170     2.67  0.00896
## 4 age         -0.0196    0.0112   -1.76  0.0823
## 5 lbph         0.107     0.0584    1.83  0.0704
## 6 sviTRUE      0.766     0.244     3.14  0.00233
## 7 lcp         -0.105     0.0910   -1.16  0.250
## 8 gleason      0.0451     0.157     0.287  0.775
## 9 pgg45        0.00453  0.00442    1.02  0.309
```

```
tidy(model_1)
```

```
## # A tibble: 9 x 5
##   term          estimate std.error statistic    p.value
##   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)  0.172      1.33      0.129  0.897
## 2 lcavol       0.565     0.0885    6.39  0.00000000793
## 3 lweight      0.622     0.202     3.08  0.00279
## 4 age         -0.0213    0.0111   -1.91  0.0596
## 5 lbph         0.0956     0.0585    1.63  0.106
## 6 sviTRUE      0.760     0.243     3.13  0.00235
## 7 lcp         -0.106     0.0904   -1.17  0.244
## 8 gleason      0.0507     0.156     0.324  0.747
## 9 pgg45        0.00447  0.00439    1.02  0.312
```

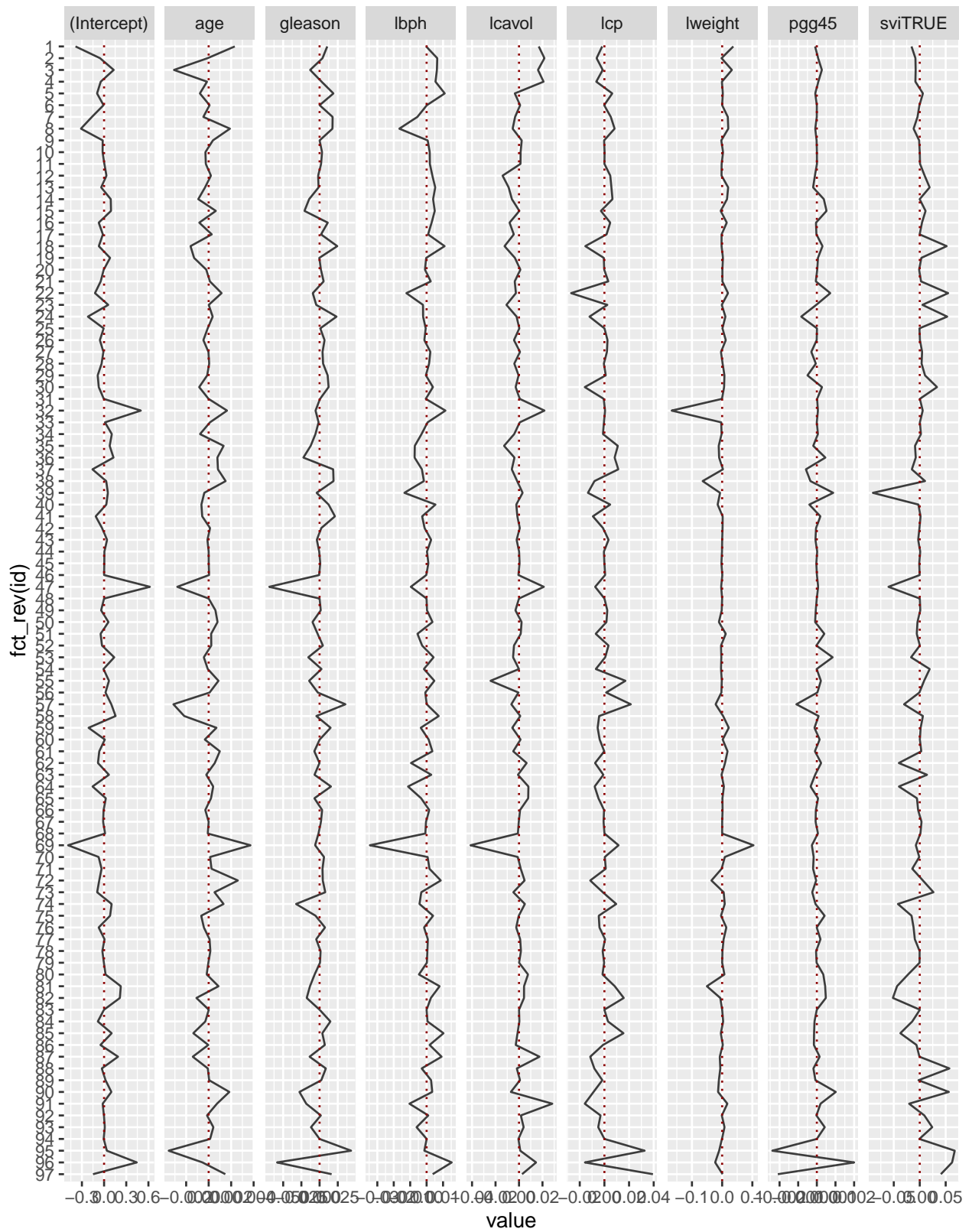
```
tidy(model_2)
```

```
## # A tibble: 9 x 5
##   term          estimate std.error statistic    p.value
##   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)  0.0488     1.29     0.0378  0.970
## 2 lcavol       0.566     0.0861    6.57  0.00000000357
## 3 lweight      0.458     0.166     2.76  0.00701
## 4 age         -0.0168    0.0110   -1.54  0.128
## 5 lbph         0.117     0.0571    2.04  0.0441
## 6 sviTRUE      0.826     0.239     3.45  0.000863
## 7 lcp         -0.0980    0.0888   -1.10  0.273
## 8 gleason      0.114     0.156     0.731  0.467
## 9 pgg45        0.00446  0.00431    1.03  0.304
```

Los casos mas influyentes son el 32 y el 47. Si sacamos del modelo la observación 32, el coeficiente de

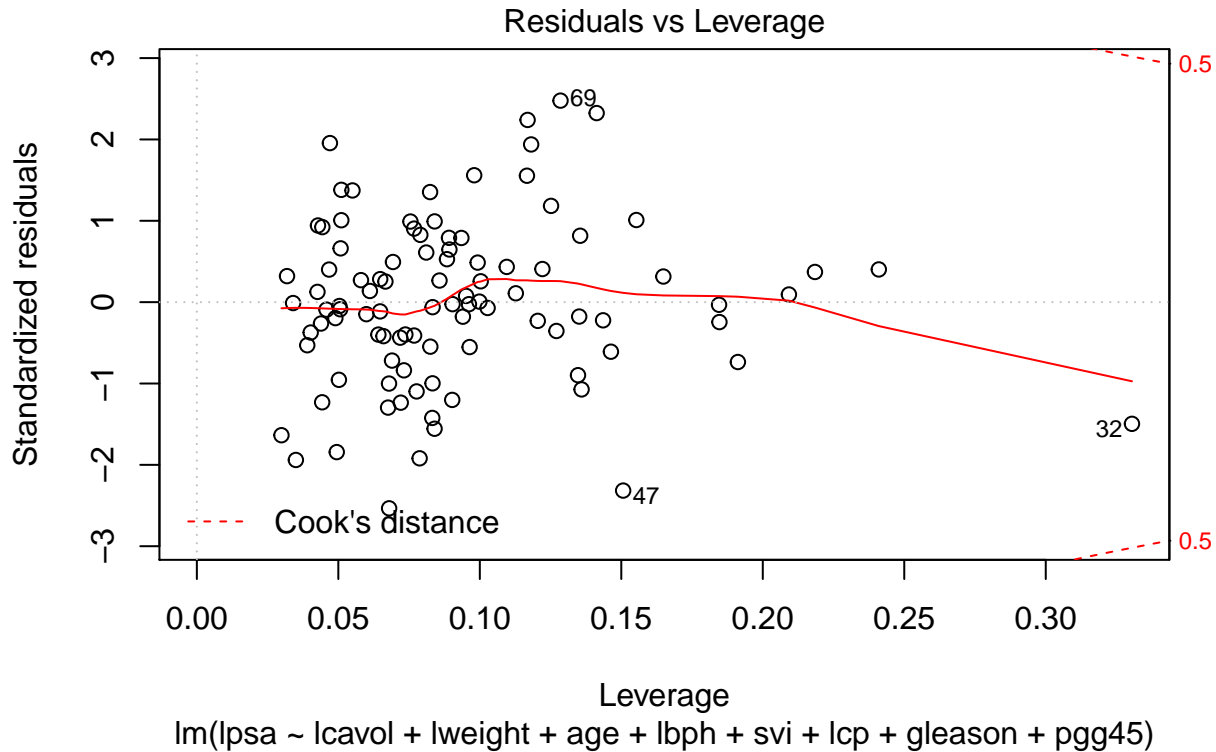
lweight cambia en .15. Al quitar la observación 47, el modelo se preserva mejor.

```
influence(model)$coefficients %>%  
  as_tibble(rownames = "id") %>%  
  mutate(id = factor(id, levels = c(1:n))) %>%  
  gather(key, value, -id) %>%  
  ggplot(aes(fct_rev(id), value, group = key)) +  
  geom_line(alpha = .75)+  
  coord_flip() +  
  geom_hline(yintercept = 0, linetype = 3, color = "darkred") +  
  facet_grid(~ key, scales = "free")
```



En este gráfico se ve como 32 genera un cambio importante en **lweight**. Llama la atención además la observación 69 que genera cambios en varios coeficientes y que tiene una distancia de cook importante.

```
plot(model, 5)
```



```
prostate_cl[c(69,32, 47),]
```

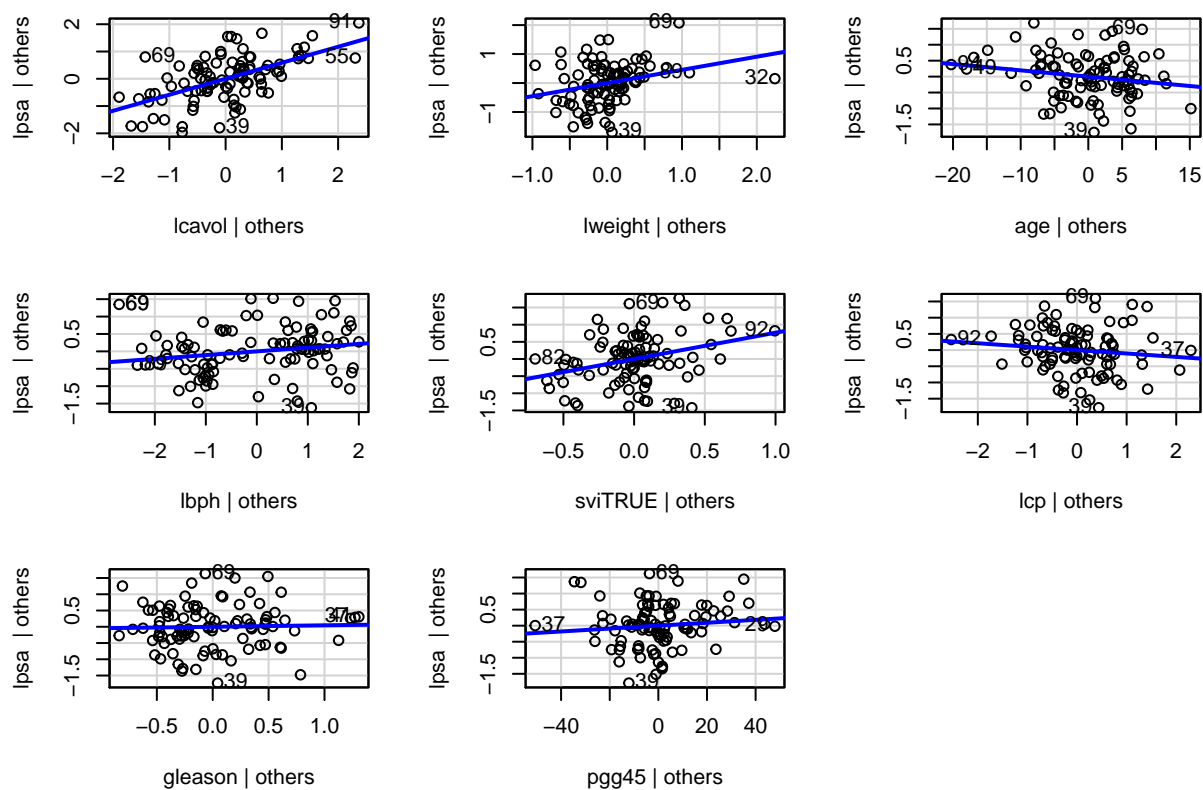
##	lcavol	lweight	age	lbph	svi	lcp	gleason	pgg45	lpsa
## 69	-0.4462871	4.4085	69	-1.386294	FALSE	-1.38629	6	0	2.96269
## 32	0.1823216	6.1076	65	1.704748	FALSE	-1.38629	6	0	2.00821
## 47	2.7278528	3.9954	79	1.879465	TRUE	2.65676	9	100	2.56879

Los casos 69, 32 y 47 son casos influyente. El caso 32 ademas tiene mucha *palanca*. Pero no tengo mucha evidencia como para decir que son outliers.

(f) Check the structure of the relationship between the predictors and the response.

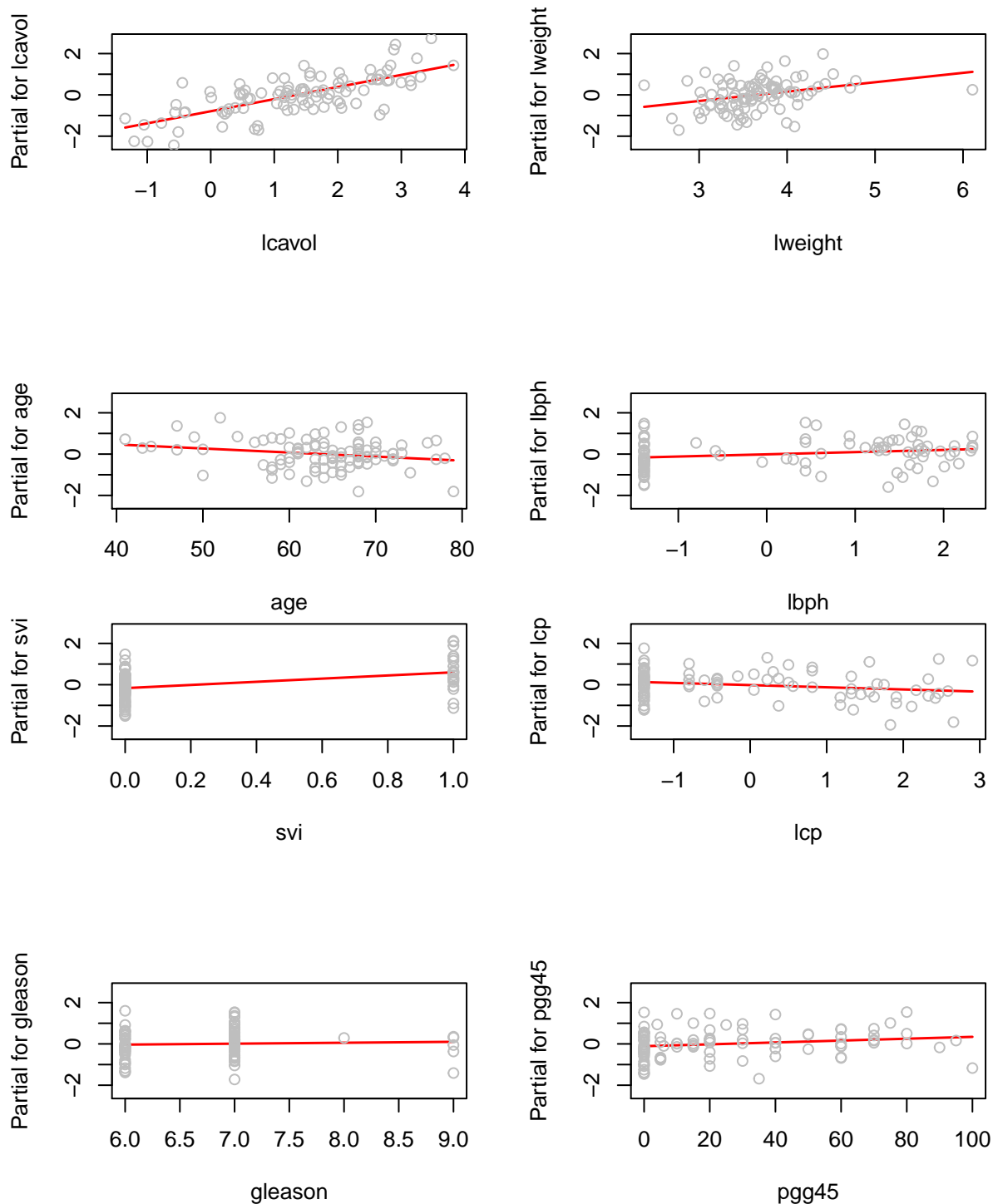
```
car::avPlots(model)
```

Added-Variable Plots



La regresión parcial confirma que 32 y mas aún el 69 son casos *raros* que se comportan de manera diferente a la mayoría en cuanto a las líneas de regresión parcial.

```
par(mfrow = c(2,2))
termplot(model, partial.resid = T)
```



Aquí destaca el caso 32 con un `lweight` muy alejado del resto, y la estructura de `lbph` (y en menor grado, de `lcp`, `pgg45` y `gleason`) que no es para nada homogénea, lo que ya se veía en el gráfico de dispersión.

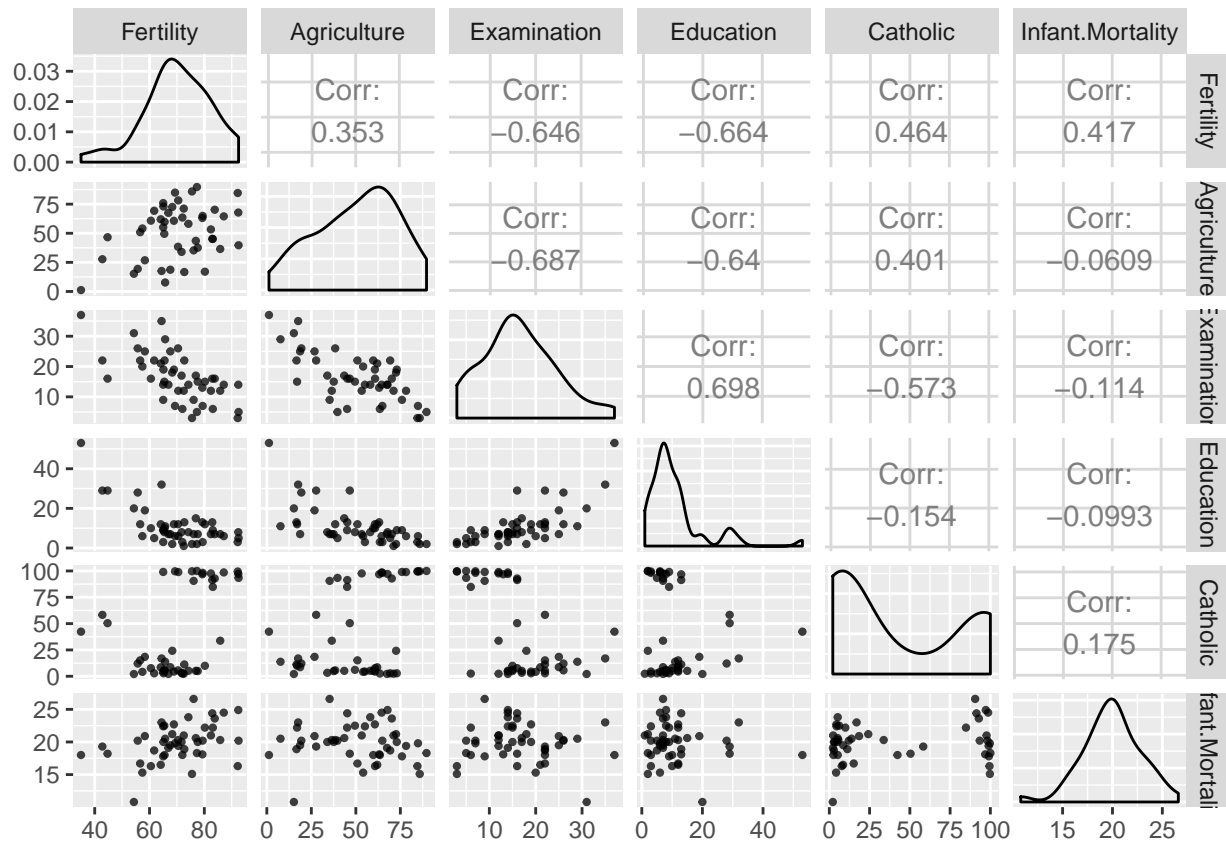
4. (Ejercicio 4 cap. 6 pág. 97)

For the `swiss` data, fit a model with `Fertility` as the response and the other variables as predictors. Answer the questions posed in the first question.

```
n <- dim(swiss)[1]
p <- dim(swiss)[2]

model <-
  lm(Fertility ~ Agriculture + Examination + Education + Catholic + Infant.Mortality, data = swiss)

swiss %>%
  ggpairs(progress = F, lower = list(continuous = wrap("points", alpha = 0.75, size = .8)))
```



Lo primero que hago es crear una matriz de dispersión para tener una idea de como se distribuyen las variables y la relación 2x2 que tienen entre si. Utilizo la librería **GGally** que tiene la función **ggpairs**.

(a) Check the constant variance assumption for the errors.

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

No se observa un patrón definido, por lo que se acepta una homocedasticidad del error.

(b) Check the normality assumption.

```
shapiro.test(resid(model))
```

```
##
## Shapiro-Wilk normality test
```

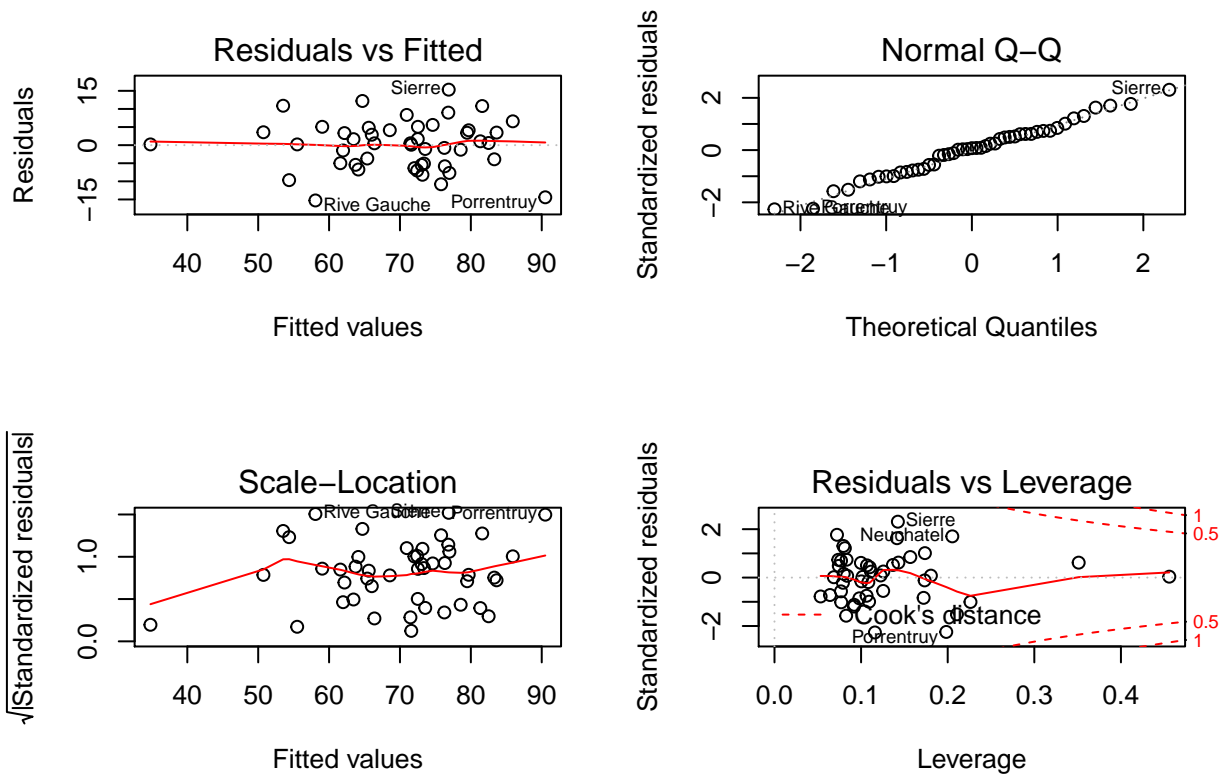


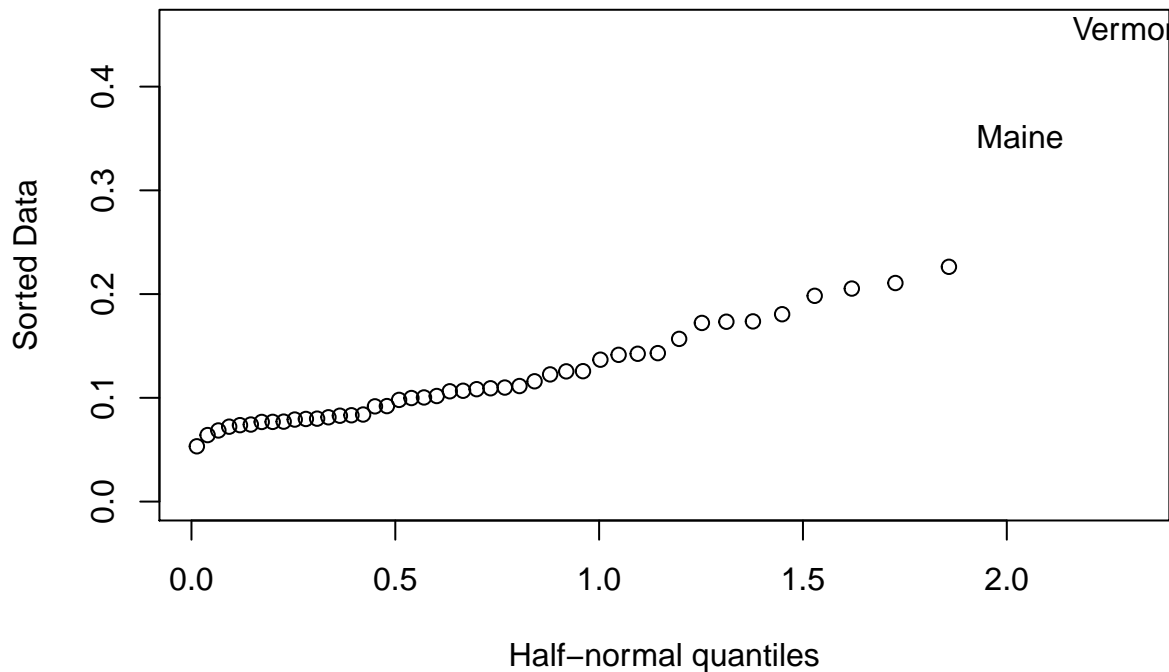
Figure 4: Gráficos Diagnósticos

```
##
## data: resid(model)
## W = 0.98892, p-value = 0.9318
```

El gráfico *Q-Q se adapta bien a la linea. La prueba no permite rechazar la H_0 de normalidad del error.

(c) Check for large leverage points.

```
halfnorm(hatvalues(model), labs = rownames(sat))
```



```
augment(model) %>%
  filter(.hat > 2*(p/n))
```

```
## # A tibble: 2 x 14
##   .rownames Fertility Agriculture Examination Education Catholic
##   <chr>      <dbl>      <dbl>      <int>      <int>      <dbl>
## 1 La Vallee  54.3        15.2        31        20        2.15
## 2 V. De Ge~  35          1.2         37        53        42.3
## # ... with 8 more variables: Infant.Mortality <dbl>, .fitted <dbl>,
## #   .se.fit <dbl>, .resid <dbl>, .hat <dbl>, .sigma <dbl>, .cooksdi <dbl>,
## #   .std.resid <dbl>
```

Para evaluar los valores extremos en el modelo, utilizamos dos gráficos.

- La gráfica de los valores ajustados versus los residuales estandarizados ($\hat{y} \times \sqrt{\hat{\epsilon}}$)
- Una gráfica de *media normal* (*halfnormal*) versus los residuales estandarizados. De esta manera se pueden identificar los valores mas extremos en X .

En este caso podemos ver que los valores para **Vermont** y **Maine** cumplen con las características para valores extremos. Llamen la atención también los puntos de **La Valle** y **V. De Geneve** que presentan valores extremos en una de las variables.

(d) Check for outliers.

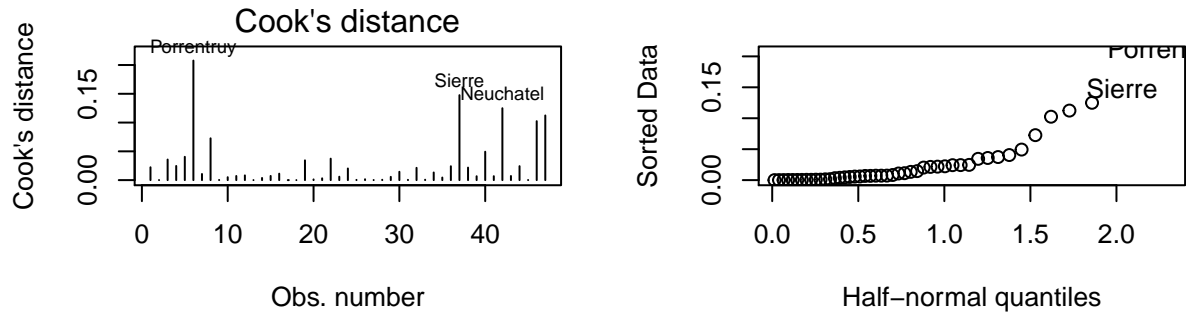
```
augment(model) %>%
  mutate(.t.resid = rstudent(model)) %>%
  filter(abs(.t.resid) > abs(qt(.05/(n * 2), n - p)))
```

```
## # A tibble: 0 x 15
## # ... with 15 variables: .rownames <chr>, Fertility <dbl>,
## #   Agriculture <dbl>, Examination <int>, Education <int>, Catholic <dbl>,
## #   Infant.Mortality <dbl>, .fitted <dbl>, .se.fit <dbl>, .resid <dbl>,
## #   .hat <dbl>, .sigma <dbl>, .cooksdi <dbl>, .std.resid <dbl>,
## #   .t.resid <dbl>
```

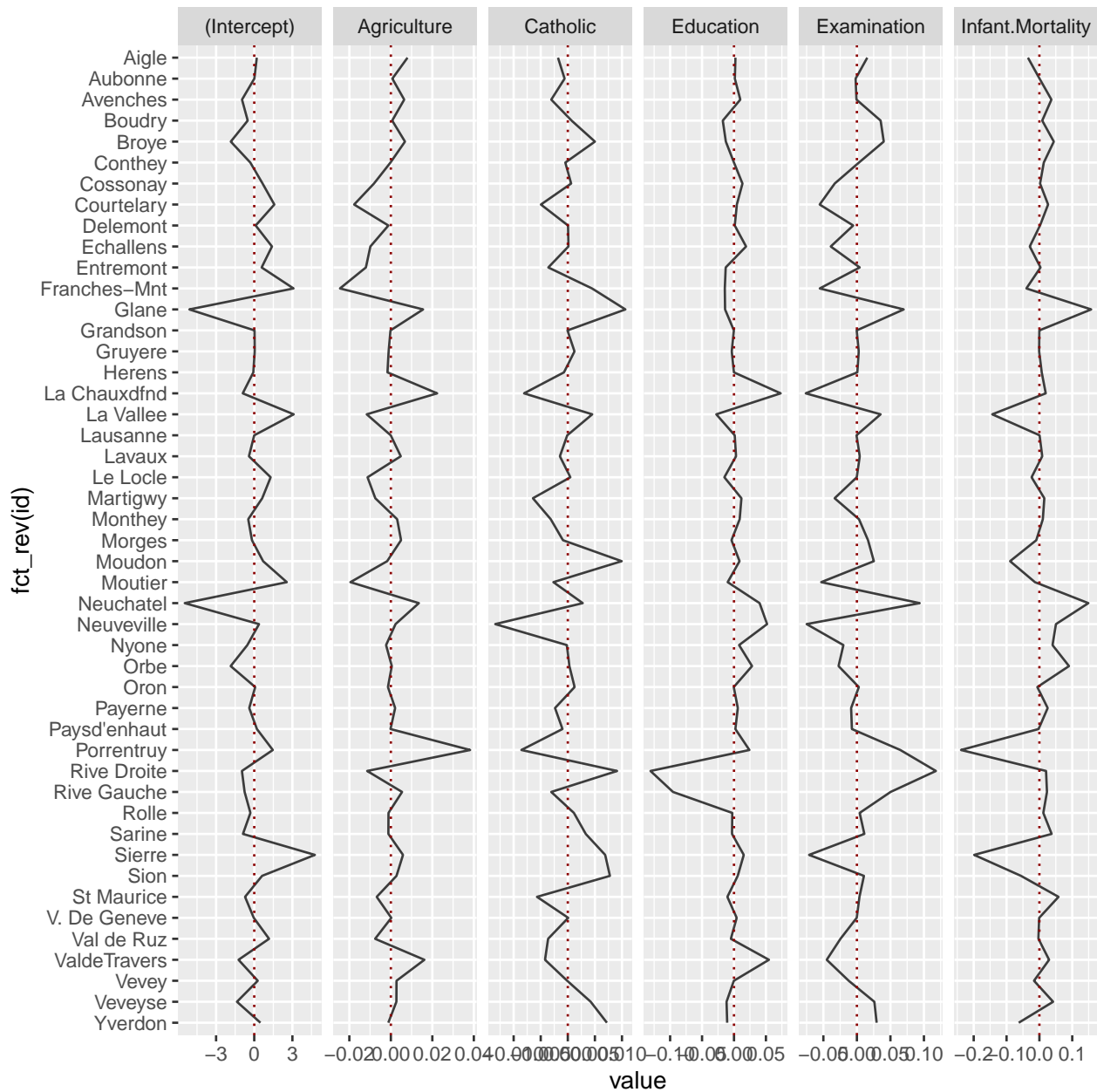

No hay *outliers* por este criterio. Tampoco se observan en el gráfico de las distancias de Cook en los gráficos diagnósticos

(e) Check for influential points.

```
par(mfrow = c(2,2))
plot(model, 4)
halfnorm(cooks.distance(model), labs = rownames(swiss))
```



```
influence(model)$coefficients %>%
  as_tibble(rownames = "id") %>%
  gather(key, value, -id) %>%
  ggplot(aes(fct_rev(id), value, group = key)) +
  geom_line(alpha = .75) +
  coord_flip() +
  geom_hline(yintercept = 0, linetype = 3, color = "darkred") +
  facet_grid(~ key, scales = "free") +
  scale_colour_viridis_d(end=.85)
```



```
colMeans(swiss)
```

```
##      Fertility      Agriculture      Examination      Education
##      70.14255      50.65957      16.48936      10.97872
##      Catholic Infant.Mortality
##      41.14383      19.94255
```

```
swiss %>%
  rownames_to_column() %>%
  filter(rowname %in% c("Porrentruy", "Sierre"))
```

```
##      rowname Fertility Agriculture Examination Education Catholic
## 1 Porrentruy      76.1      35.3          9          7      90.57
## 2 Sierre        92.2      84.6          3          3      99.46
##      Infant.Mortality
```

```
## 1      26.6
## 2      16.3
```

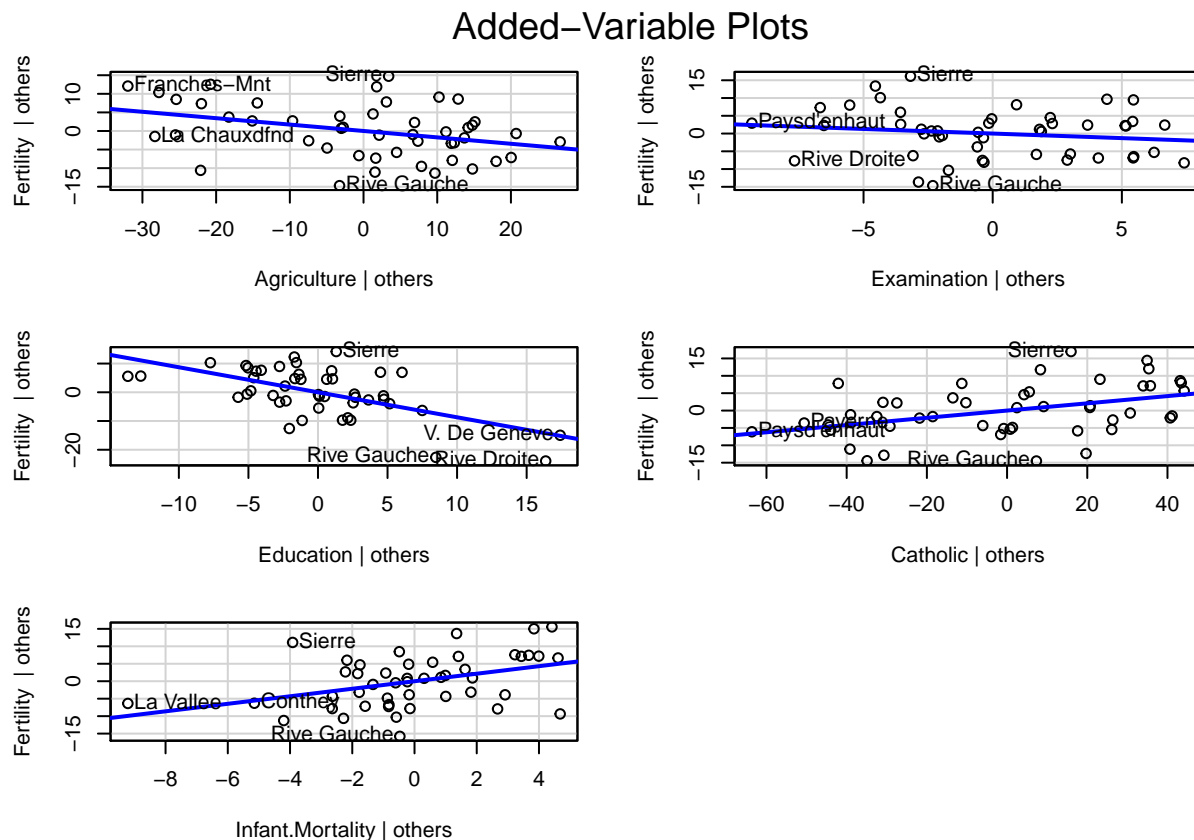
Hay varias observaciones que resultan influyentes. Por el criterio de distancias de Cook, destaca **Porrentruy** que en segundo gráfico podemos ver, modifica **agriculture** y **Infant.Mortality**. **Sierre** modifica **Infant.Mortality** hacia el otro sentido, tiene una mortalidad baja, y **Examination** baja.

Hay dos puntos con alta palanca pero baja influencia, **La Valle** y **V. De Geneve**.

(f) Check the structure of the relationship between the predictors and the response.

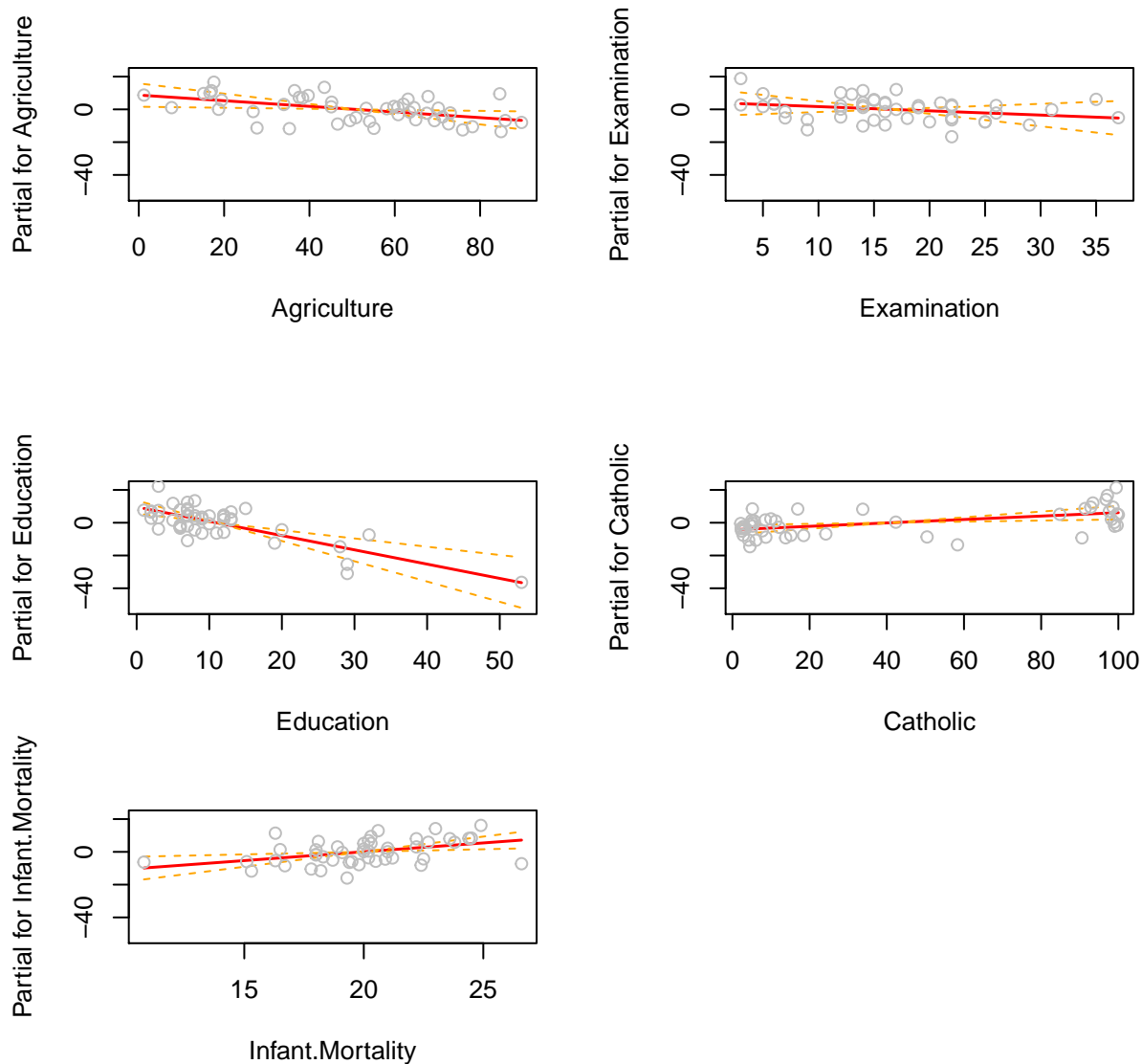
En primer lugar me remito al gráfico de dispersión de (a), que permite reflejar la distribución y la relación 1 a 1 de las variables.

```
car::avPlots(model)
```



La estructura parece correcta. Podemos ver como **Sierre** se aleja de la tendencia de los modelos.

```
par(mfrow = c(2,2))
termplot(model, partial.resid = T, se = T)
```



Se pueden observar los valores extremos pero con poca influencia en `Education` y en `Infant.Mortality`. Destaca la estructura bimodal de `Catholic`. Se podría realizar una transformación de la variable a una categorial o incluso binaria.

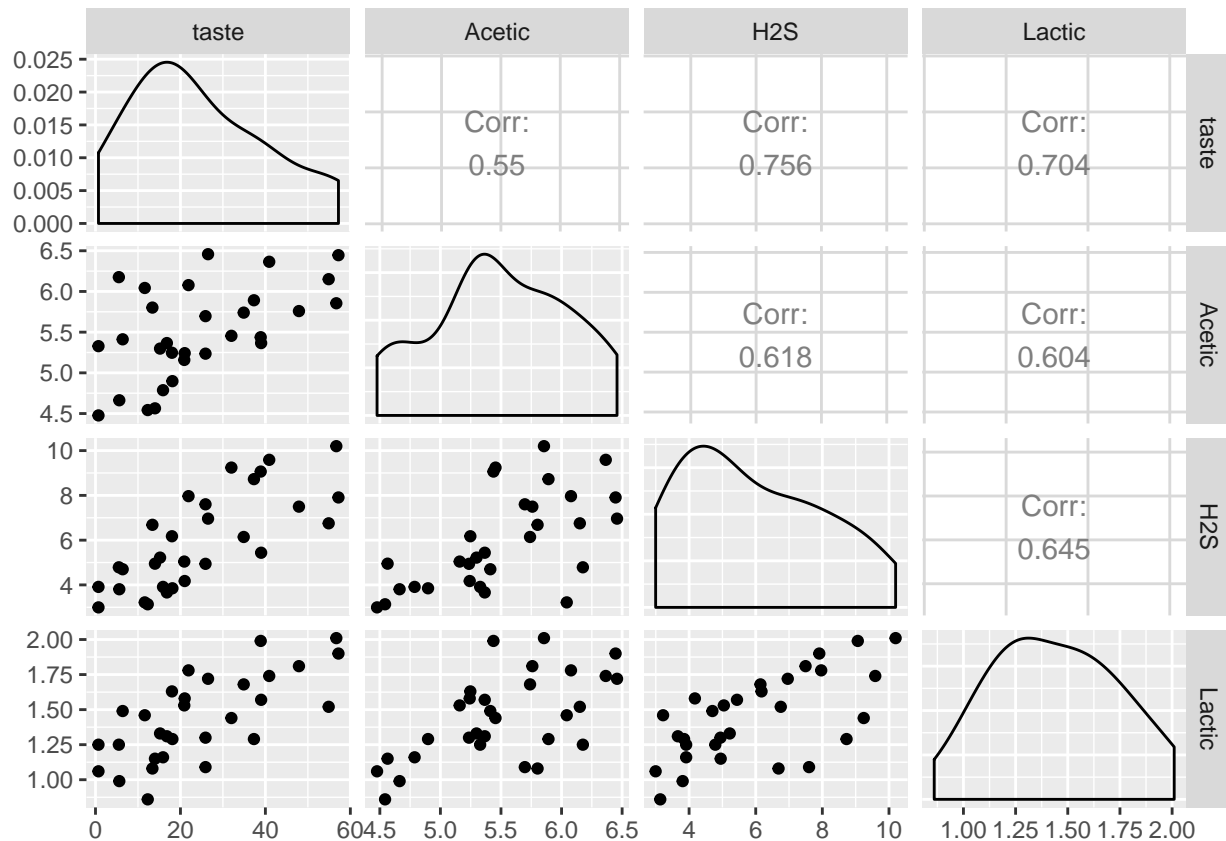
5. (Ejercicio 5 cap. 6 pág. 97)

Using the `cheddar` data, fit a model with `taste` as the response and the other three variables as predictors. Answer the questions posed in the first question.

```
n <- dim(cheddar)[1]
p <- dim(cheddar)[2]

model <-
  lm(taste ~ Acetic + H2S + Lactic, data = cheddar)

cheddar %>%
  ggpairs(progress = F)
```



(a) Check the constant variance assumption for the errors.

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

No se ve un patrón en el gráfico de valores ajustados v/s residuales que haga pensar en problemas de varianza del error.

(b) Check the normality assumption.

```
shapiro.test(resid(model))

##
##  Shapiro-Wilk normality test
##
## data:  resid(model)
## W = 0.98021, p-value = 0.8312
```

El gráfico $Q-Q$ sigue una distribución muy cercana a la recta y la prueba de S-W no permite desechar la H_0 por lo que se puede asumir la normalidad de los residuales.

(c) Check for large leverage points.

```
halfnorm(hatvalues(model), labs = rownames(cheddar))
```

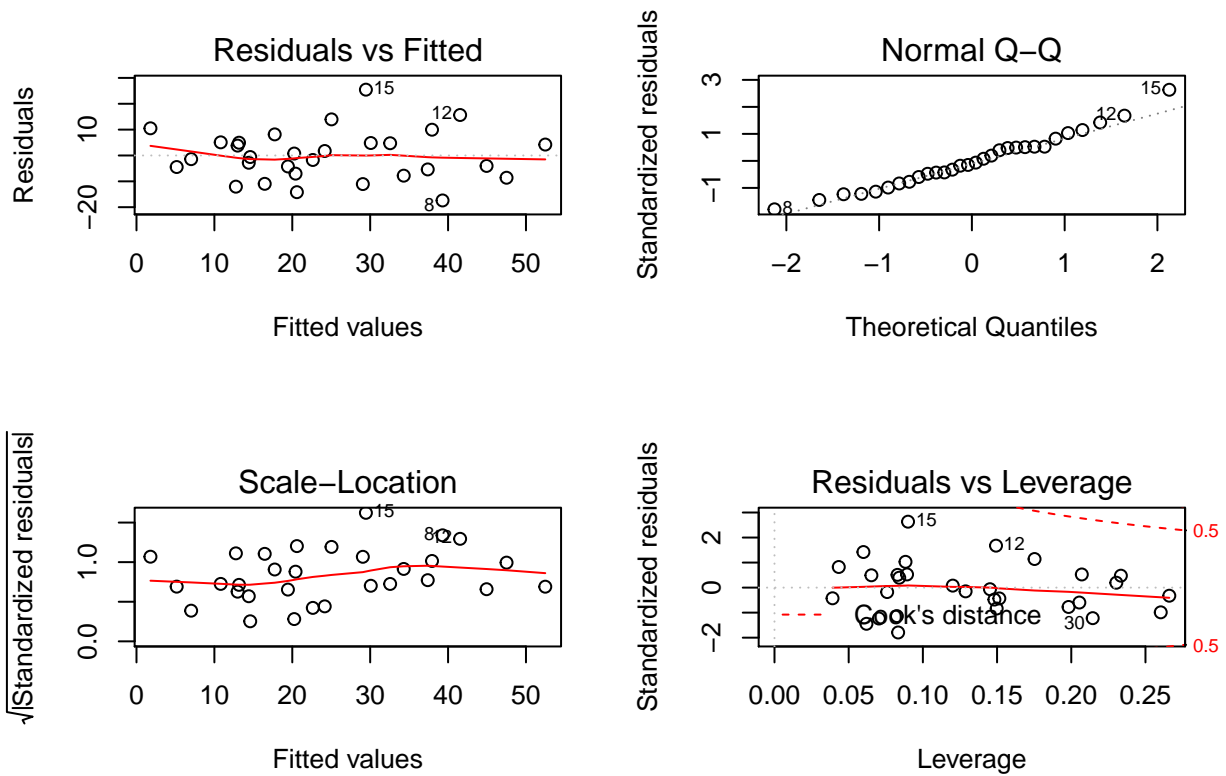
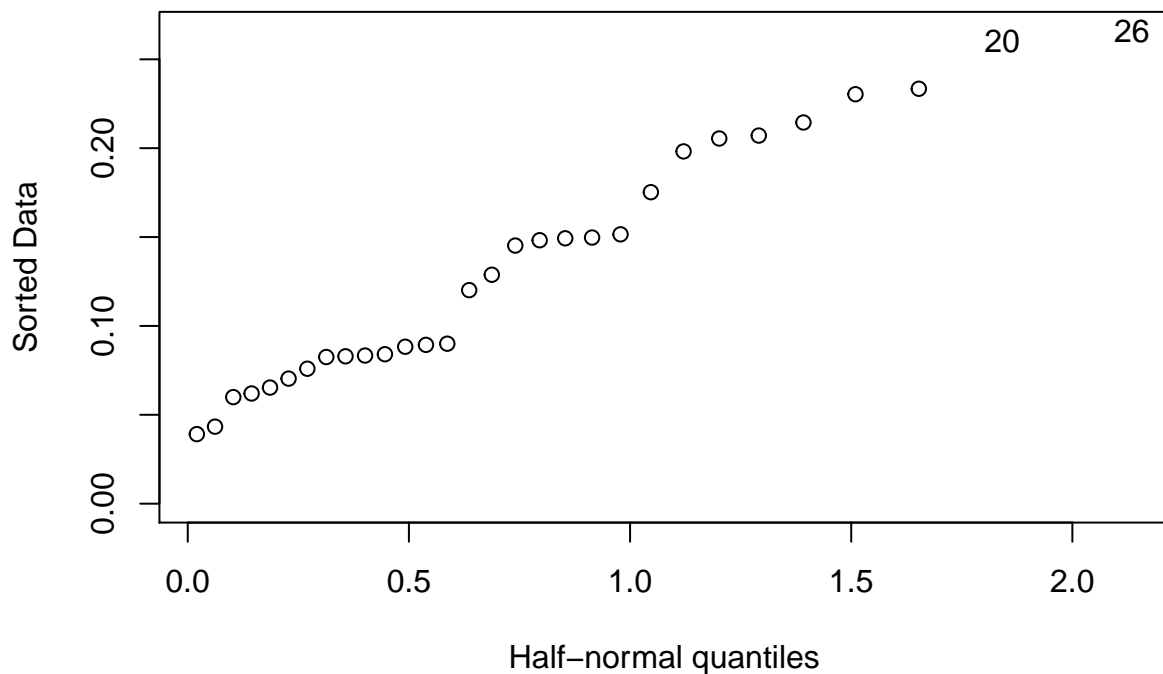


Figure 5: Gráficos Diagnósticos



```
augment(model) %>%
  rowid_to_column() %>%
  filter(.hat > 2*(p/n))
```

```
## # A tibble: 0 x 12
```

```
## # ... with 12 variables: rowid <int>, taste <dbl>, Acetic <dbl>,
## #   H2S <dbl>, Lactic <dbl>, .fitted <dbl>, .se.fit <dbl>, .resid <dbl>,
## #   .hat <dbl>, .sigma <dbl>, .cooksd <dbl>, .std.resid <dbl>
```

Los casos 20 y 26 son los casos mas extremos.

(d) Check for outliers.

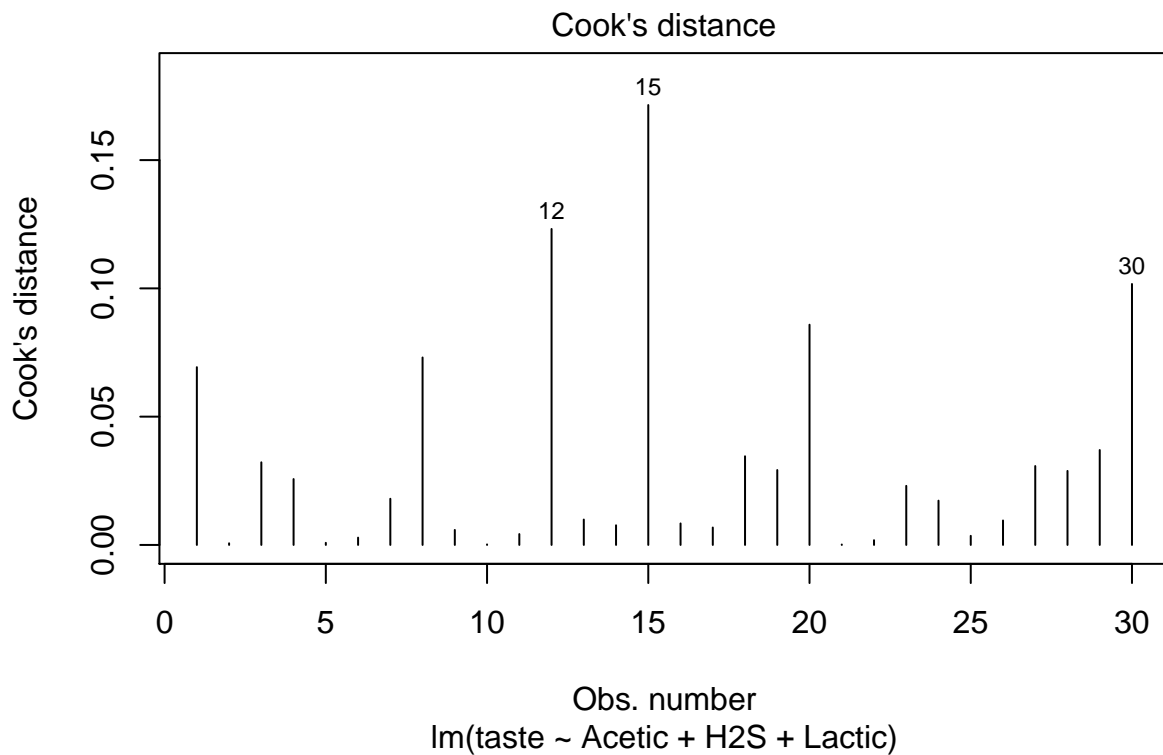
```
augment(model) %>%
  mutate(.t.resid = rstudent(model)) %>%
  filter(abs(.t.resid) > abs(qt(.05/(n * 2), n - p)))
```

```
## # A tibble: 0 x 12
## # ... with 12 variables: taste <dbl>, Acetic <dbl>, H2S <dbl>,
## #   Lactic <dbl>, .fitted <dbl>, .se.fit <dbl>, .resid <dbl>, .hat <dbl>,
## #   .sigma <dbl>, .cooksd <dbl>, .std.resid <dbl>, .t.resid <dbl>
```

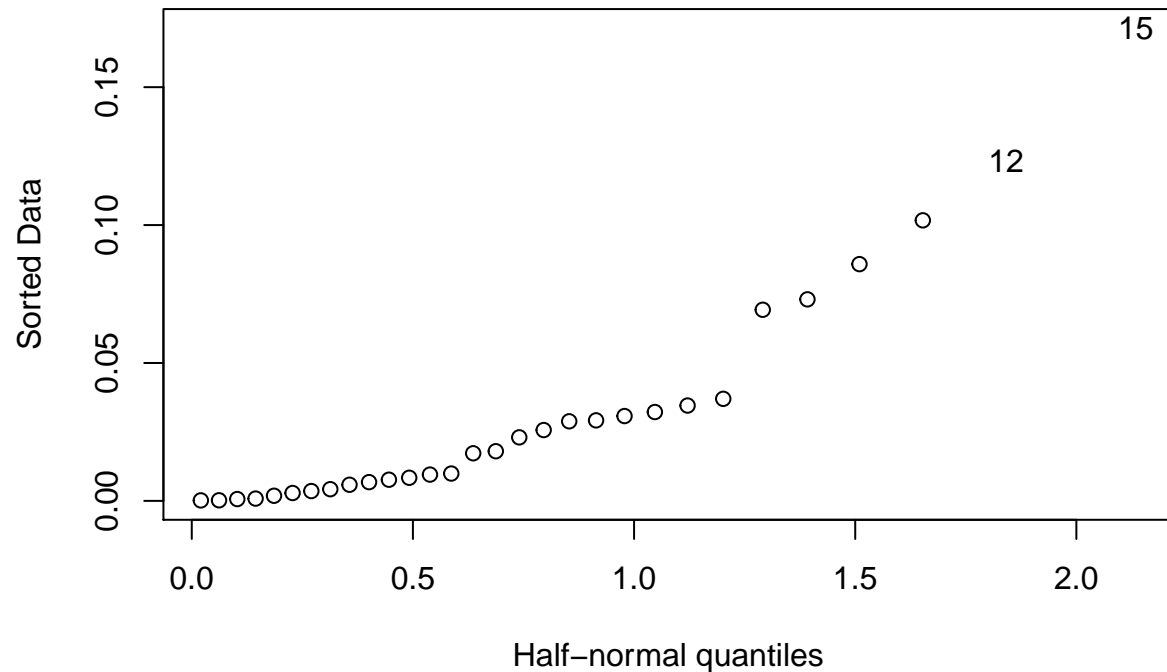
No hay outliers según este criterio.

(e) Check for influential points.

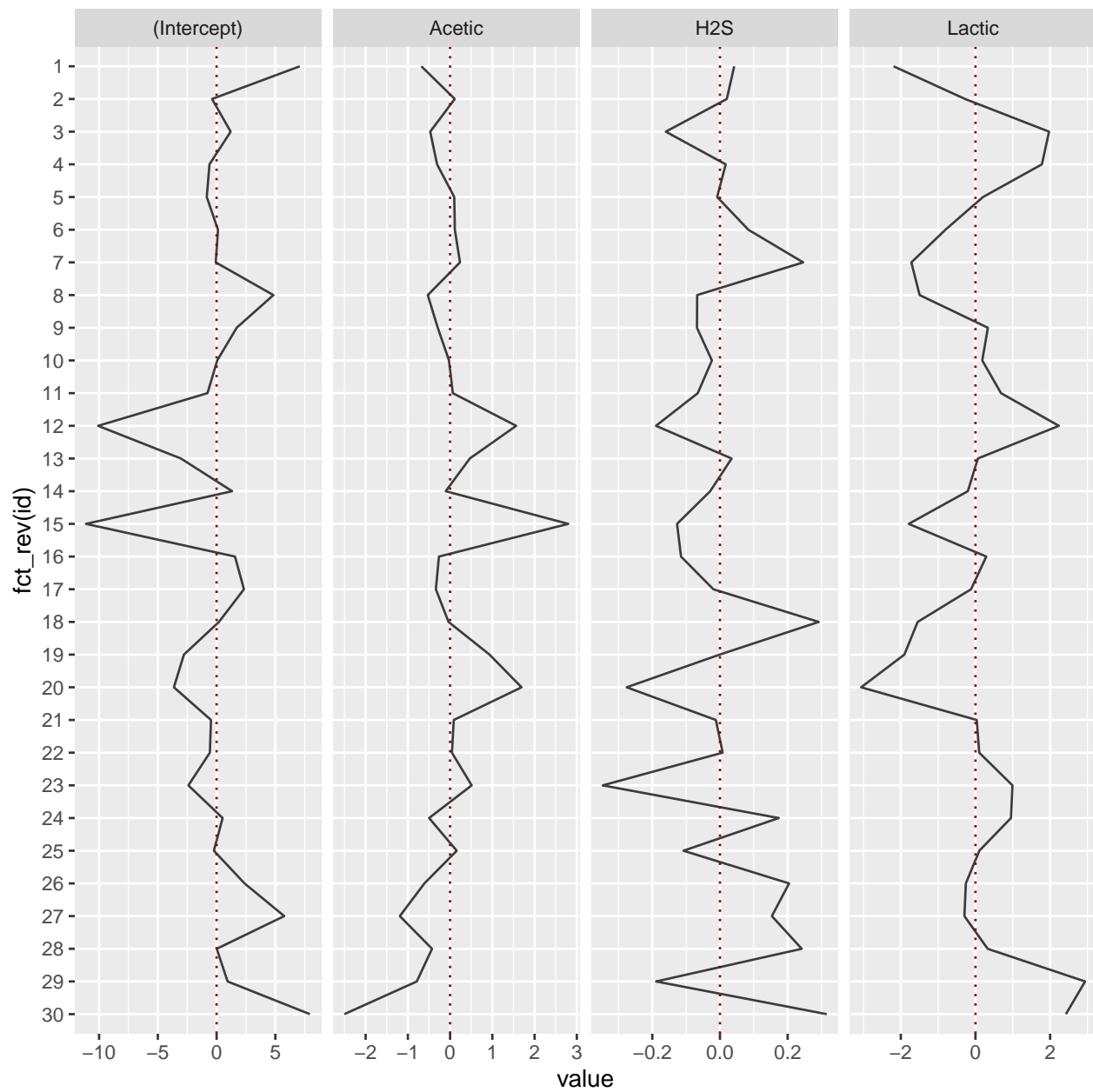
```
plot(model, 4)
```



```
halfnorm(cooks.distance(model), labs = rownames(cheddar))
```



```
influence(model)$coefficients %>%
  as_tibble(rownames = "id") %>%
  mutate(id = factor(id, levels = c(1:n))) %>%
  gather(key, value, -id) %>%
  ggplot(aes(fct_rev(id), value, group = key)) +
  geom_line(alpha = .75) +
  coord_flip() +
  geom_hline(yintercept = 0, linetype = 3, color = "darkred") +
  facet_grid(~ key, scales = "free")
```

Los casos 12 y 15 son casos influyentes.

```
model_1 <-
  cheddar %>%
  rownames_to_column() %>%
  filter(rowname != "15") %>%
  lm(formula = taste ~ Acetic + H2S + Lactic, data = .)
```

```
tidy(model)
```

```
## # A tibble: 4 x 5
##   term          estimate std.error statistic p.value
##   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
```

```
## 1 (Intercept) -28.9      19.7      -1.46  0.155
## 2 Acetic       0.328      4.46       0.0735 0.942
## 3 H2S          3.91       1.25       3.13   0.00425
## 4 Lactic       19.7       8.63       2.28   0.0311
```

```
tidy(model_1)
```

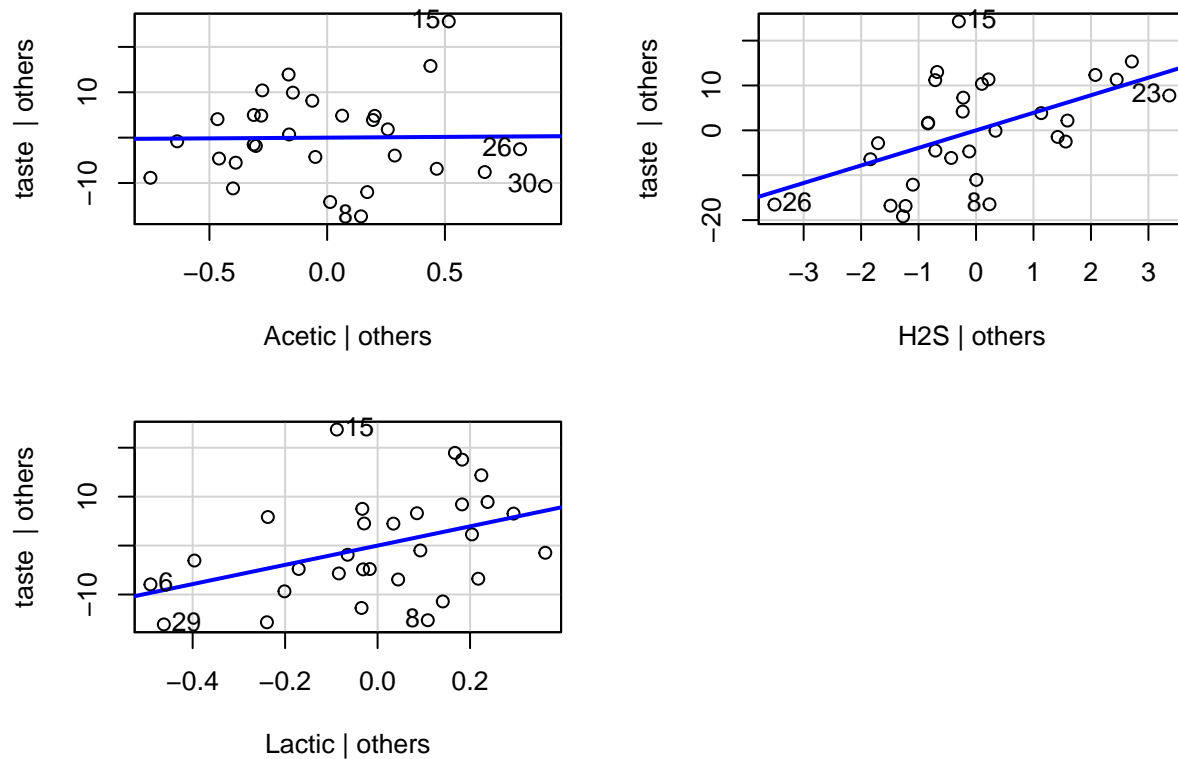
```
## # A tibble: 4 x 5
##   term      estimate std.error statistic p.value
##   <chr>      <dbl>    <dbl>    <dbl>   <dbl>
## 1 (Intercept) -17.8     17.6     -1.01  0.323
## 2 Acetic      -2.47     4.00     -0.617 0.543
## 3 H2S          4.04     1.09      3.70  0.00106
## 4 Lactic      21.5     7.56      2.84  0.00886
```

El caso 15 modifica el coeficiente de Acetic en 2 , y Lactic en casi 2.

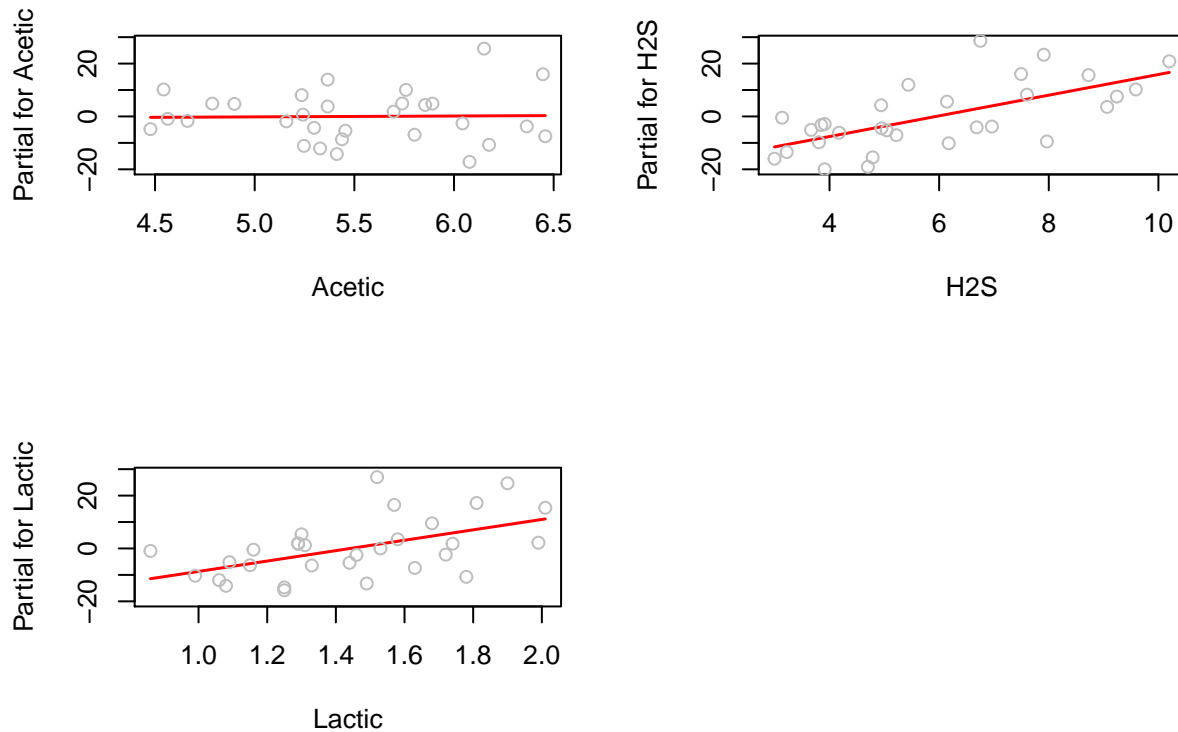
(f) Check the structure of the relationship between the predictors and the response.

```
car::avPlots(model)
```

Added-Variable Plots



```
par(mfrow = c(2,2))
termplot(model, partial.resid = T)
```



NO se ven mayores problemas estructurales en los gráficos.

6. (*) (Ejercicio 6 cap. 6 pág. 98)

Using thehappydata, fit a model withhappyas the response and the other four variables aspredictors. Answer the questions posed in the first question.

7. (*) (Ejercicio 7 cap. 6 pág. 98)

Using thetvdoctordata, fit a model withlifeas the response and the other two variables aspredictors. Answer the questions posed in the first question.

8. (*) (Ejercicio 8 cap. 6 pág. 98)

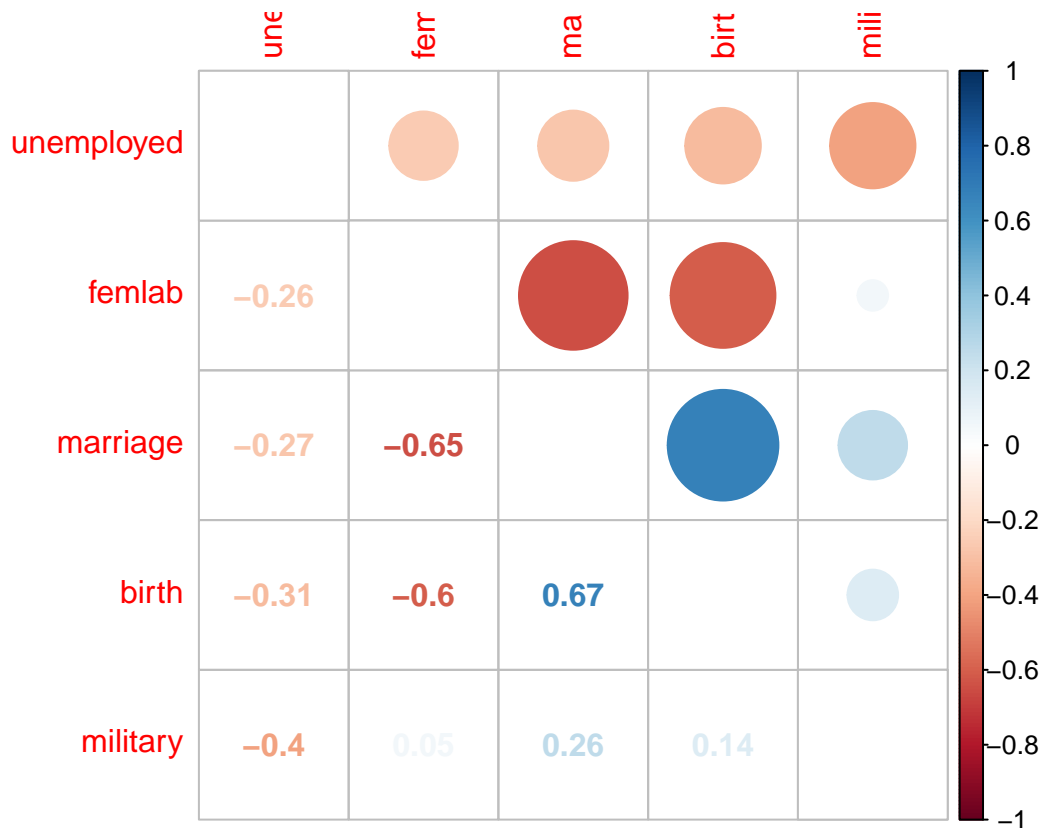
For thedivusadata, fit a model withdivorceas the response and the other variables, except yearas predictors. Check for serial correlation.

9. (Ejercicio 3 cap. 7 pág. 110)

Using the divusa data:

- Fit a regression model with `divorce` as the response and `unemployed`, `femlab`, `marriage`, `birth` and `military` as predictors. Compute the condition numbers and interpret their meanings.

```
# Miramos colinealidad
divusa %>%
select(-year, -divorce) %>%
cor() %>%
corrplot.mixed(., lower = "number", upper = "circle", tl.pos = "lt")
```



```
model <- lm(divorce ~ unemployed + femlab + marriage + birth + military, data = divusa)
```

Podemos observar que si hay varios predictores que presenta correlación entre si.

```
X <- model.matrix(model)[,-1]
eig <- eigen(t(X) %*% X)
```

```
eig$values
```

```
## [1] 1174600.548 21261.741 16133.842 6206.181 1856.894
```

```
(c_nums <- sqrt(eig$values[1] / eig$values))
```

```
## [1] 1.000000 7.432684 8.532498 13.757290 25.150782
```

Los *números de condición* son relativamente pequeños (< 30) por lo que a pesar de la correlación de parejas encontrada, no parece existir un problema grave de colinealidad.

- (b) For the same model, compute the VIFs. Is there evidence that collinearity causes some pre-dictors not to be significant? Explain.

```
(vifs <- vif(X))
```

```
## unemployed    femlab    marriage    birth    military
##    2.252888    3.613276    2.864864    2.585485    1.249596
```

```
sqrt(vifs)
```

```
## unemployed    femlab    marriage    birth    military
##    1.500962    1.900862    1.692591    1.607944    1.117853
```

Si bien hay algunas X_i donde la VIF se aleja de 1 que es la ortogonalidad, tampoco se aleja demasiado.

Donde más se aleja es en X_{femlab} donde se puede interpretar que el error estándar aumenta en 1.9008618

```
model_1 <-
  lm(formula = divorce+2*rnorm(dim(divusa)[1]) ~ unemployed + femlab + marriage + birth +
    military, data = divusa)
```

```
sumary(model)
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.487845   3.393779  0.7331  0.46594
## unemployed  -0.111252   0.055925 -1.9893  0.05052
## femlab       0.383649   0.030587 12.5430 < 2.2e-16
## marriage     0.118674   0.024414  4.8609 6.772e-06
## birth       -0.129959   0.015595 -8.3334 4.027e-12
## military    -0.026734   0.014247 -1.8764  0.06471
##
## n = 77, p = 6, Residual SE = 1.65042, R-Squared = 0.92
```

```
sumary(model_1)
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.188907   5.090770 -0.6264 0.5330552
## unemployed  -0.064336   0.083889 -0.7669 0.4456718
## femlab       0.451321   0.045881  9.8368 6.809e-15
## marriage     0.146915   0.036622  4.0117 0.0001474
## birth       -0.119572   0.023393 -5.1115 2.595e-06
## military    -0.023701   0.021371 -1.1090 0.2711662
##
## n = 77, p = 6, Residual SE = 2.47569, R-Squared = 0.85
```

A pesar de eso, si agregamos un poco de ruido el modelo se mantiene relativamente estable.

(c) Does the removal of insignificant predictors from the model reduce the collinearity? Investigate.

```
model_1 <-
  lm(formula = divorce+2*rnorm(dim(divusa)[1]) ~ unemployed + marriage + birth +
    military, data = divusa)
```

```
model_2 <-
  lm(formula = divorce+2*rnorm(dim(divusa)[1]) ~ unemployed + femlab +
    military, data = divusa)
```

```
sumary(model)
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.487845   3.393779  0.7331  0.46594
## unemployed  -0.111252   0.055925 -1.9893  0.05052
## femlab       0.383649   0.030587 12.5430 < 2.2e-16
## marriage     0.118674   0.024414  4.8609 6.772e-06
## birth       -0.129959   0.015595 -8.3334 4.027e-12
## military    -0.026734   0.014247 -1.8764  0.06471
##
## n = 77, p = 6, Residual SE = 1.65042, R-Squared = 0.92
```

```
sumary(model_1)
```

```
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept) 41.347882  2.755371 15.0063 < 2.2e-16
## unemployed  -0.611517  0.092624 -6.6021 5.935e-09
## marriage    -0.043837  0.043741 -1.0022  0.3196
## birth       -0.231181  0.029391 -7.8658 2.729e-11
## military    -0.014500  0.031060 -0.4668  0.6420
##
## n = 77, p = 5, Residual SE = 3.60519, R-Squared = 0.66
```

```
summary(model_2)
```

```
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.8058238  1.6140104 -2.3580  0.02106
## unemployed  -0.0075488  0.0797538 -0.0947  0.92485
## femlab       0.4480247  0.0316047 14.1759 < 2e-16
## military    -0.0301262  0.0263963 -1.1413  0.25747
##
## n = 77, p = 4, Residual SE = 3.12699, R-Squared = 0.75
```

```
vif(model_1)
```

```
## unemployed  marriage      birth  military
##   1.295131   1.927175   1.924532   1.244658
```

```
vif(model_2)
```

```
## unemployed      femlab  military
##   1.276363   1.074670   1.194897
```

En este caso reviso dos modelos alternativos. Uno sin la variable `femlab` cuyo VIF era alta en l modelo original, y un segundo modelo con la variable `femlab` pero sin `marriage` ni `birth` que tienen una correlación de pareja alta con `femlab` y entre si.

Los dos modelos alternativos pierden bastante poder explicativo lo que se ve reflejado en un R^2 mas pequeño, sobretodo el sin `femlab`. Por otra parte, es verdad que en el primer modelo sin `femlab` no baja la VIF (aumenta en las variables que tiene correlación, `birth` y `marriage`). En el segundo modelo, sin `birth` y `marriage`, la R^2 baja, pero la VIF también.

Como conclusión, creo que el modelo original es correcto.

10. (Ejercicio 4 cap. 7 pág. 110)

For the `longley` data, fit a model with `Employed` as the response and the other variables as predictors.

(a) Compute and comment on the condition numbers.

```
model <- lm(Employed ~ ., data = longley)

X <- model.matrix(model)[,-1]
eig <- eigen(t(X) %*% X)

eig$values

## [1] 6.665299e+07 2.090730e+05 1.053550e+05 1.803976e+04 2.455730e+01
## [6] 2.015117e+00

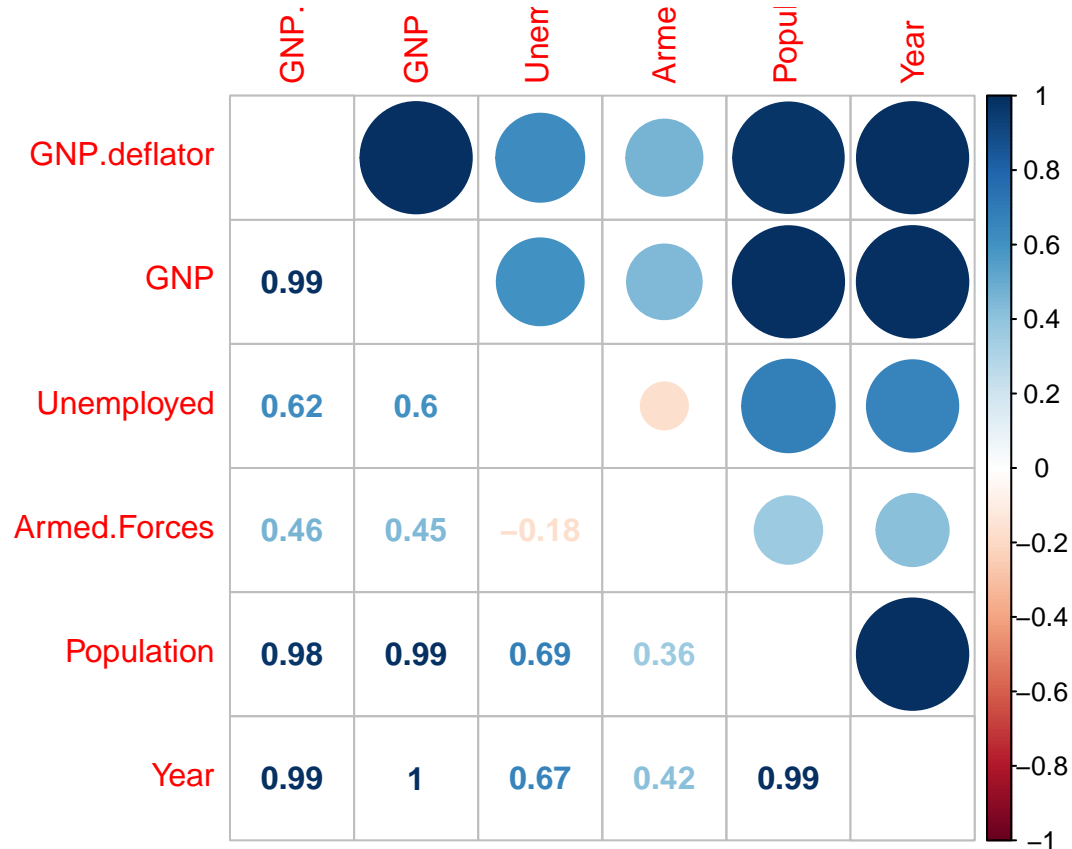
(c_nums <- sqrt(eig$values[1] / eig$values))

## [1] 1.00000 17.85504 25.15256 60.78472 1647.47771 5751.21560
```

Los *numeros de condición* son bastante altos al igual que los eigenvalues, lo que sugiere multicolinealidad

(b) Compute and comment on the correlations between the predictors.

```
corrplot.mixed(cor(X), lower = "number", upper = "circle", tl.pos = "lt")
```



En este caso las variables predictoras tienen mucha mayor correlación entre si.

(c) Compute the variance inflation factors.

```
(vifs <- vif(X))
```

```
## GNP.deflator      GNP      Unemployed Armed.Forces  Population
## 135.53244    1788.51348    33.61889      3.58893    399.15102
##      Year
## 758.98060
```

```
sqrt(vifs)
```

```
## GNP.deflator      GNP      Unemployed Armed.Forces  Population
## 11.641840    42.290820    5.798180      1.894447    19.978764
##      Year
## 27.549602
```

En este caso excepto por *Armed.Forces*, y algo menos *Unemployed*, la mayoría de las variables predictoras correlacionan entre si, y hacen que aumente el error estándar del modelo. Es probable que un modelo mas simple pueda funcionar mejor.

```
summary(model)
```

```
##
## Call:
## lm(formula = Employed ~ ., data = longley)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.41011 -0.15767 -0.02816  0.10155  0.45539
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.482e+03  8.904e+02  -3.911 0.003560 **
## GNP.deflator  1.506e-02  8.492e-02   0.177 0.863141
## GNP          -3.582e-02  3.349e-02  -1.070 0.312681
## Unemployed   -2.020e-02  4.884e-03  -4.136 0.002535 **
## Armed.Forces -1.033e-02  2.143e-03  -4.822 0.000944 ***
## Population   -5.110e-02  2.261e-01  -0.226 0.826212
## Year          1.829e+00  4.555e-01   4.016 0.003037 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3049 on 9 degrees of freedom
## Multiple R-squared:  0.9955, Adjusted R-squared:  0.9925
## F-statistic: 330.3 on 6 and 9 DF,  p-value: 4.984e-10
```

11. (Ejercicio 5 cap. 7 pág. 110)

For the prostate data, fit a model with `lpsa` as the response and the other variables as predictors.

```
model <-
  lm(lpsa ~ ., data = prostate)
```

(a) Compute and comment on the condition numbers.

```
X <- model.matrix(model)[,-1]
eig <- eigen(t(X) %*% X)
```

```
eig$values
```

```
## [1] 4.790826e+05 6.190704e+04 2.109042e+02 1.756329e+02 6.479853e+01
## [6] 4.452379e+01 2.023914e+01 8.093145e+00
```

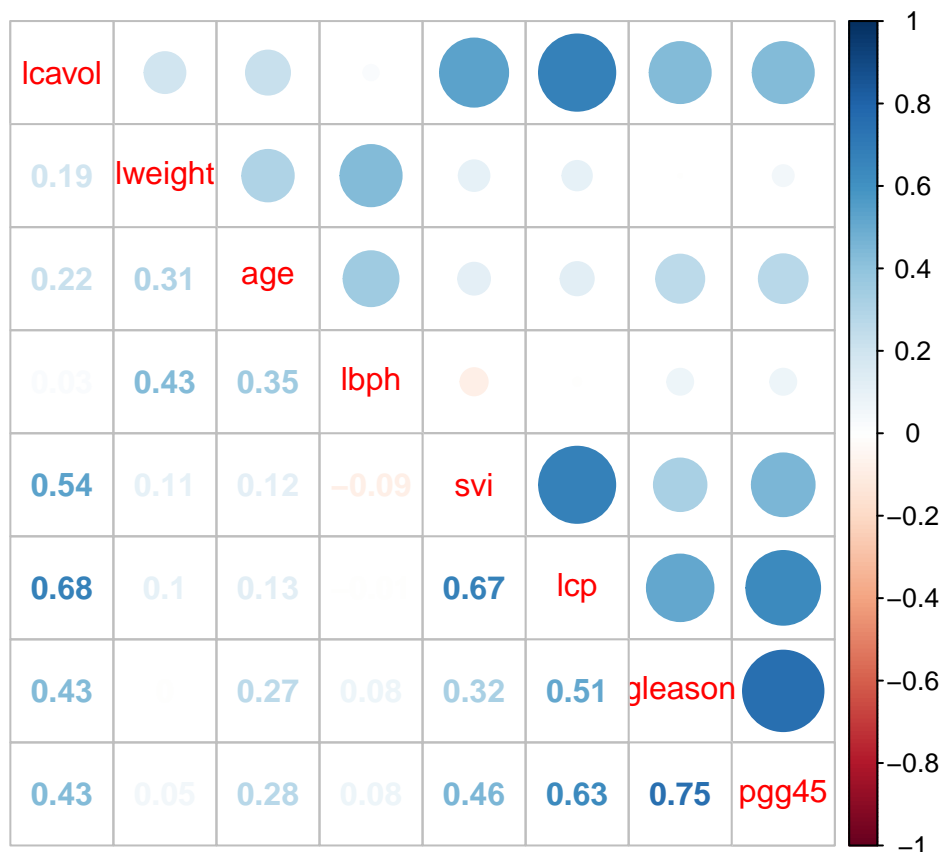
```
(c_nums <- sqrt(eig$values[1] / eig$values))
```

```
## [1] 1.00000 2.78186 47.66094 52.22787 85.98499 103.73114 153.85414
## [8] 243.30248
```

Algunos valores son altos y otros no, lo que sugiere que podría haber multicolinealidad en varias combinaciones lineales.

(b) Compute and comment on the correlations between the predictors.

```
corrplot.mixed(cor(X), lower = "number", upper = "circle", tl.pos = "d")
```

Se corrobora lo anterior en la medida de que hay alguno de los predictores que correlacionan bastante, como el caso de `pgg45` con `gleason` y `lcp`; `lcp` con `svi` y `lcavol`.

(c) Compute the variance inflation factors.

```
(vifs <- vif(X))

##   lcavol  lweight    age    lbph    svi    lcp  gleason   pgg45
## 2.054115 1.363704 1.323599 1.375534 1.956881 3.097954 2.473411 2.974361

sqrt(vifs)

##   lcavol  lweight    age    lbph    svi    lcp  gleason   pgg45
## 1.433218 1.167777 1.150478 1.172832 1.398886 1.760100 1.572708 1.724634
```

De todas formas, a pesar de la correlación, no parece haber un mayor problema al analizar la VIF. La mayoría de los predictores suman poco error

12. (*) (Ejercicio 8 cap. 7 pág. 111)

Use the fat data, fitting the model described in Section 4.2.

- Compute the condition numbers and variance inflation factors. Comment on the degree of collinearity observed in the data.
- Cases 39 and 42 are unusual. Refit the model without these two cases and recompute the collinearity diagnostics. Comment on the differences observed from the full data fit.
- Fit a model with `brozekas` as the response and `justage`, `weight` and `height` as predictors. Compute the collinearity diagnostics and compare to the full data fit.

- (d) Compute a 95% prediction interval for brozek for the median values of age , weight and height .
- (e) Compute a 95% prediction interval for brozek for $\text{age}=40$, $\text{weight}=200$ and $\text{height}=73$. How does the interval compare to the previous prediction?
- (f) Compute a 95% prediction interval for brozek for $\text{age}=40$, $\text{weight}=130$ and $\text{height}=73$. Are the values of predictors unusual? Comment on how the interval compares to the previous two answers.

Ejercicios del libro de Carmona 1.

(*) (Ejercicio 9.1 del Capítulo 9 página 172)

Realizar el análisis completo de los residuos del modelo de regresión parabólico propuesto en la sección 1.2 con los datos de tráfico.2.

(*) (Ejercicio 9.2 del Capítulo 9 página 172)

Realizar el análisis completo de los residuos de los modelos de regresión simple y parabólico propuestos en la sección 1.2 con los datos de tráfico, pero tomando como variable respuesta la velocidad (sin raíz cuadrada). Este análisis debe justificar la utilización de la raíz cuadrada de la velocidad como variable dependiente.