Final Assignment

Greta Riva

2024-03-08

1. DESCRIPTION OF THE DATASET AND DATA STRUCTURE

The data set I have chosen contains data from the department of Breast Diseases at the University of Coimbra, collected between 2015 and 2019. (It was selected by the website kaggle.com)

It consist of 116 observations among (116 patients) of 10 different variables.

There is one cathegorical variable with two possible outcomes: 1, that represent 52 healthy patients, or 2, that represents 64 cancer patients. The other 9 variables are continuous and defined as it follows:

- Age (years): Age of the individuals.
- BMI (kg/m2): Body Mass Index, a measure of body fat based on weight and height.
- Glucose (mg/dL): Blood glucose levels, an important metabolic indicator.
- Insulin (nU/mL): Insulin levels, a hormone related to glucose regulation.
- HOMA: Homeostatic Model Assessment, a method for assessing insulin resistance and beta-cell function.
- Leptin (ng/mL): Leptin levels, a hormone involved in regulating appetite and energy balance.
- Adiponectin (g/mL): Adiponectin levels, a protein associated with metabolic regulation.
- Resistin (ng/mL): Resistin levels, a protein implicated in insulin resistance.
- MCP-1 (pg/dL): Monocyte Chemoattractant Protein-1, a cytokine involved in inflammation.
- Classification: 1: Healthy controls 2: Patients with breast cancer.

2. GOALS

The goal of my study is to fit a linear regression model in order to understand the relationship between the response variable BMI and the predictors. It will help us understand which biomarker has a significant impact on the BMI level.

3. EXPLORATORY ANALYSIS

Let's represent a scatterplot matrix in order to understand the relations between quantitative variables. (Figure 1) The most relevant relationship through the variables is the one between the levels of Insulin and HOMA: they increase together. This isn't surprising because Homeostatic Model Assessment (HOMA) is a method for assessing Insulin resistance. Their correlation will be adressed later.

```r
plot(data[, -10])
mtext("Figure 1: Scatterplot matrix of the quantitative variables", side = 1,
      line = 4, cex = 0.8)
```

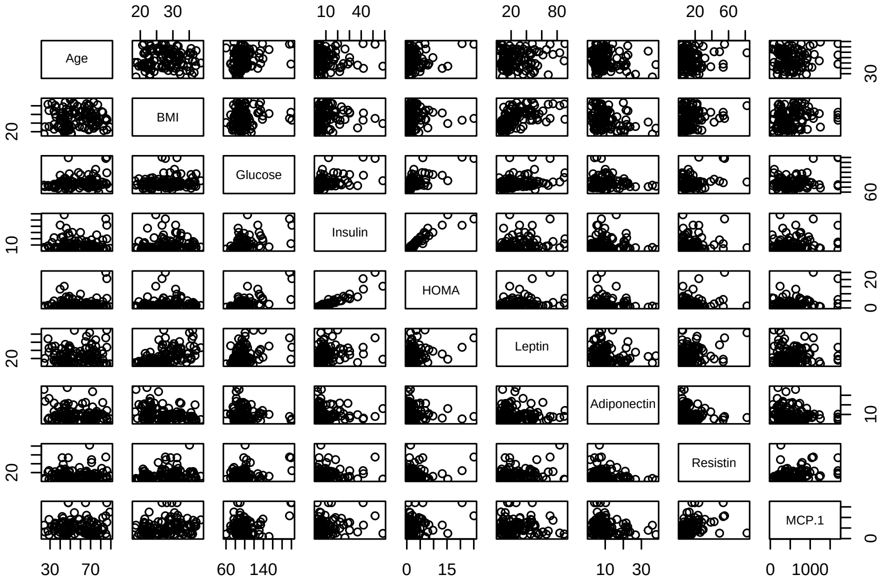


Figure 1: Scatterplot matrix of the quantitative variables

We now represent the cathegorical variable 'Classification' as a factor with 2 different levels : level 1 is assigned to healty patients while level 2 refers to cancer patients.

```
data$Classification= as.factor(data$Classification)
summary(data$Classification)
```

```
##  1  2
## 52 64
```

The boxplot below (Figure 2) is useful to visualize the correlation between BMI of patients and the Presence of breast cancer (Classification), which is a categorical variable.

The Target does not affect BMI of patients: that is easy to see because the median values of the 2 groups are very similar.

```
boxplot(BMI~Classification, data=data,
        col=c("turquoise", "violet"),
        main="Correlation between
        BMI and target",
        xlab= "Presence of Breast Cancer",
        ylab= "BMI of Patients")

mtext("Figure 2: Boxplot to represent the categorical variable", side = 1,
      line = 4, cex = 0.8)
```

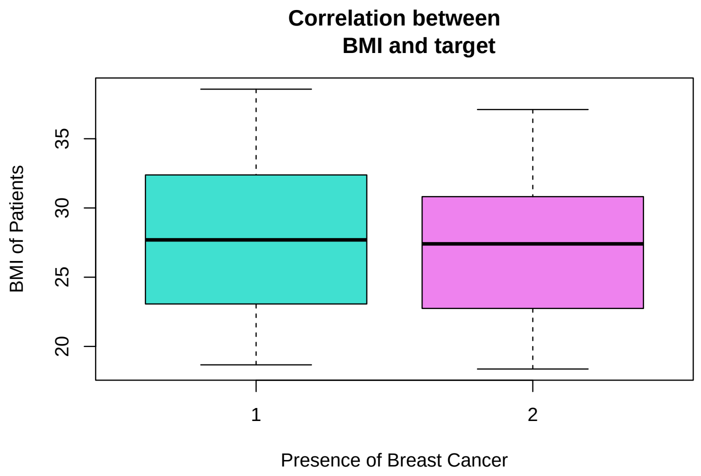


Figure 2: Boxplot to represent the categorical variable

4. LINEAR REGRESSION MODEL AND BEST SUBSET SELECTION

I fit a linear regression model using all my predictors.

```
ols = lm(BMI ~ Age + Insulin  + Glucose + HOMA + Leptin + Adiponectin + MCP.1
         + Resistin + Classification, data = data)
summ = summary(ols)
summ
```

```
##
## Call:
## lm(formula = BMI ~ Age + Insulin + Glucose + HOMA + Leptin +
##     Adiponectin + MCP.1 + Resistin + Classification, data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -8.459 -2.208 -0.436  2.335 10.541
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     20.452974   3.067784   6.667 1.21e-09 ***
## Age             -0.023218   0.023208  -1.000  0.31939
## Insulin          0.348563   0.121973   2.858  0.00514 **
## Glucose          0.054235   0.028116   1.929  0.05641 .
## HOMA            -1.241157   0.402095  -3.087  0.00258 **
## Leptin           0.148240   0.020243   7.323 4.94e-11 ***
## Adiponectin     -0.163953   0.054344  -3.017  0.00320 **
## MCP.1            0.003531   0.001132   3.121  0.00232 **
## Resistin        -0.007111   0.032997  -0.215  0.82979
## Classification2 -1.900641   0.801478  -2.371  0.01953 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.704 on 106 degrees of freedom
## Multiple R-squared:  0.4981, Adjusted R-squared:  0.4555
## F-statistic: 11.69 on 9 and 106 DF,  p-value: 1.362e-12
```

Exploring the way in which predictors vary depending on Classification (cathegorical variable), I notice that the ranges of values assumed by Age in the two cathegories have some differences. This could be an indicator to assume an interaction between Age and Classification. In order to that, I fit another linear regression model including this interaction.

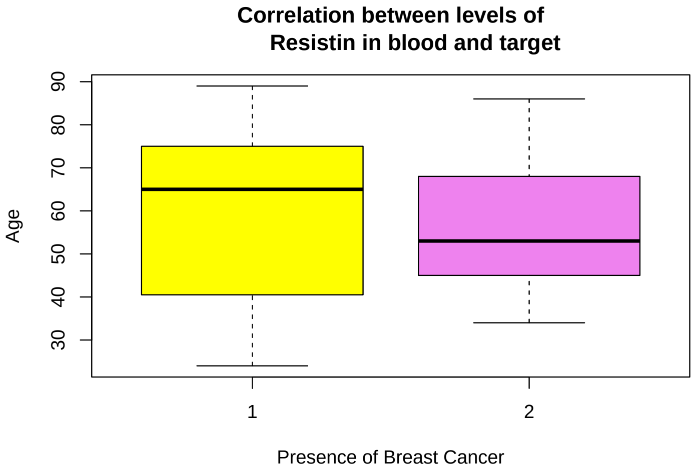


Figure 3: Boxplot: distribution of Age in the categories

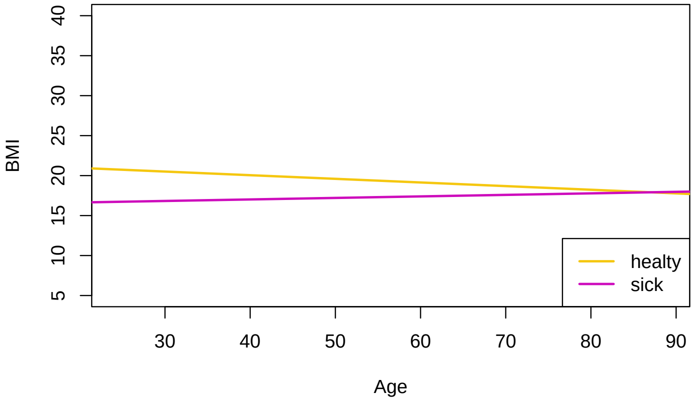


Figure 4: Fitted regression lines with interactions between Age and

This suggests that increases in Age are associated with slight decrease in BMI among healthy patients. On the other hand, as the variable Age increases it's possible to see a slight increse in BMI of cancer patients. However the interaction term is not significant, i.e the p-value of Age:Classification2 does not indicate evidence against the null hypothesis H0 : Beta10 = 0. This idea is enforced comparing the two R-squared of each model, that are very similar.

We observe now what are the best sets of variables for each model size.

```
library(leaps)
ols1 = regsubsets(BMI  ~ ., data= data)
summ1 = summary(ols1)
summ1
```

```
## Subset selection object
## Call: regsubsets.formula(BMI ~ ., data = data)
## 9 Variables  (and intercept)
##                Forced in Forced out
## Age                FALSE      FALSE
## Glucose            FALSE      FALSE
## Insulin            FALSE      FALSE
## HOMA               FALSE      FALSE
## Leptin             FALSE      FALSE
## Adiponectin        FALSE      FALSE
## Resistin           FALSE      FALSE
## MCP.1               FALSE      FALSE
## Classification2     FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: exhaustive
##          Age Glucose Insulin HOMA Leptin Adiponectin Resistin MCP.1
## 1  ( 1 ) " " " "     " "     " "  "*"    " "         " "      " "
## 2  ( 1 ) " " " "     " "     " "  "*"    "*"         " "      " "
## 3  ( 1 ) " " " "     " "     " "  "*"    "*"         " "      "*"
## 4  ( 1 ) " " " "     " "     " "  "*"    "*"         " "      "*"
## 5  ( 1 ) " " " "     "*"     "*"  "*"    "*"         " "      "*"
## 6  ( 1 ) " " " "     "*"     "*"  "*"    "*"         " "      "*"
## 7  ( 1 ) " " "*"     "*"     "*"  "*"    "*"         " "      "*"
## 8  ( 1 ) "*" "*"     "*"     "*"  "*"    "*"         " "      "*"
##          Classification2
## 1  ( 1 ) " "
## 2  ( 1 ) " "
## 3  ( 1 ) " "
## 4  ( 1 ) "*"
## 5  ( 1 ) " "
## 6  ( 1 ) "*"
## 7  ( 1 ) "*"
## 8  ( 1 ) "*"
```

5. BEST MODEL SELECTION

We now determine which is the optimal number of regressors that must be used to fit the model. In order to take this decision we can use four different criteria: adjusted Rˆ2, BIC, AIC and Cp Mallow's.

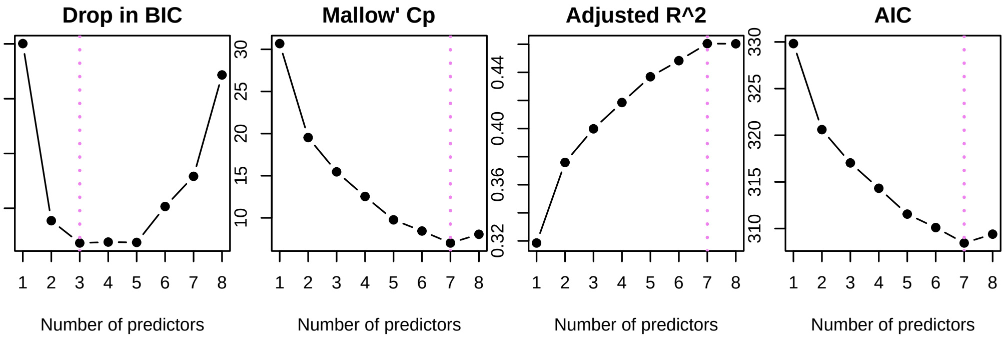


Figure 5: BIC, CP, R−squared and AIC

The coefficient estimates associated with the best model (with 7 predictors) are:

```
round(coef(ols1,7),7)
```

```
##     (Intercept)         Glucose         Insulin            HOMA          Leptin
##      19.0578486       0.0517881       0.3705407      -1.3050130       0.1468790
##     Adiponectin           MCP.1 Classification2
##      -0.1504048       0.0035773      -1.8554516
```

Through cross validation we perform best subset selection within each of the k training sets. We create a vector that allocates each observation to one of k = 116 folds (number of rows), and we create a matrix in which we will store the results.

Cross Validation method is performed to estimate the goodness of predictions made by a model. However, since this is not the goal of our study, the previous four criteria are preferred.

```
p = 8
k = nrow(data)
set.seed (1)
folds = sample (1:k,nrow(data),replace =FALSE)
cv.errors = matrix (NA ,k, p, dimnames =list(NULL , paste (1:p) ))
for(j in 1:k){
best.fit =regsubsets (BMI ~ ., data= data[folds!=j,])
for(i in 1:p) {
    mat = model.matrix(as.formula(best.fit$call[[2]]), data[folds==j,])
    coefi = coef(best.fit ,id = i)
    xvars = names(coefi )
    pred = mat[,xvars ]%*% coefi
    cv.errors[j,i] <- mean( (data$BMI[folds==j] - pred)^2)
  }
}
cv.mean = colMeans(cv.errors)
cv.mean
```

```
##        1        2        3        4        5        6        7        8
## 17.58653 16.17246 16.00220 15.52030 15.42991 14.49234 14.26306 14.41253
```

Also according to the Cross Validation criterion the best model is the one with 7 predictors.

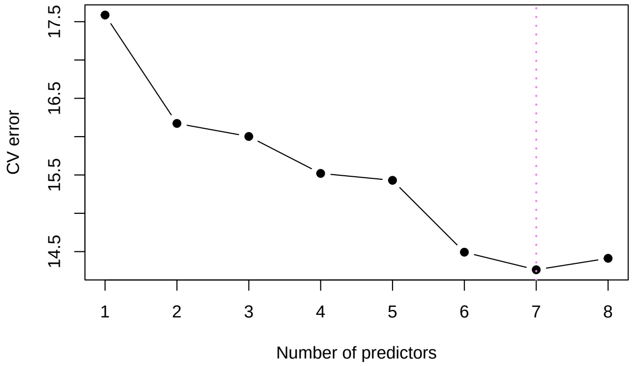


Figure 6: Cross−Validation error for the best model of each side

Based on our analysis, the optimal model comprises seven predictors. The fitted model equation is as follows: y = 19.06 + 0.05 x1 + 0.37 x2 -1.31 x3 + 0.15 x4 - 0.15 x5 + 0.004 x6 - 1.86 x7

6. COLLINEARITY ISSUES AND IMPROVEMENT OF THE MODEL

At first, in order to identify collinearity issues, it is useful to analyse the correlation matrix. that takes in account bivariate correlations.

```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
cormatrix = cor(model.matrix(best)[,c(-1)])
corrplot(cormatrix, method = "ellipse")

mtext("Figure 7: Correlation matrix.",side = 1, line = 4.3, cex =0.8)
```

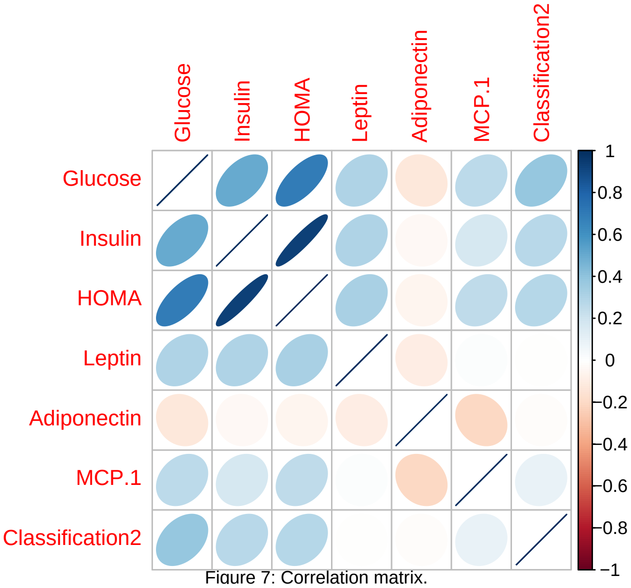


Figure 7: Correlation matrix.

The presence of collinearity can pose problems in the regression context, since it can be difficult to separate out the individual effects of collinear variables on the response.

```
cor(model.matrix(best)[, c(-1,-5,-6,-7,-8)])
```

```
##           Glucose   Insulin      HOMA
## Glucose 1.0000000 0.5046531 0.6962118
## Insulin 0.5046531 1.0000000 0.9321978
## HOMA    0.6962118 0.9321978 1.0000000
```

After plotting the full correlation matrix, considering all predictors of the best subset, i observe that: Correlation between Insulin and HOMA is very high : 0.93. That is due to the fact that HOMA (Homeostatic Model Assessment) is a method for assessing Insulin resistance and beta-cell function, reason why it is difficult to separate the effects of this predictor from the ones of Insulin. Correlation betweem Glucose and HOMA is substantial : 0.69. This isn't surprising because insulin is a hormone related to glucose regulation.

It is also possible for collinearity to exist between three or more variables even if no pair of variables has a particularly high correlation. We call this situation multicollinearity. A better way to assess multi-collinearity is to compute the variance inflation factor (VIF). As a rule of thumb, a VIF>10 indicates a problematic amount of collinearity.

As we suspected, we can affirm that there's a strong collinearity in our data (VIF of Insulin and HOMA assume values respectively 12 and 17), but no issues of multicollinearity.

```
library(car)
```

```
## Loading required package: carData
```

```
vif(best)
```

```
##        Glucose        Insulin           HOMA         Leptin    Adiponectin
##       3.330884      12.054500      17.333747       1.185139       1.061191
##          MCP.1 Classification
##       1.165096       1.273997
```

A possible way to solve this problem is to drop one of the problematic variables (we drop Insulin, because it has higher p-value and the R-squared of the model is lower than the one in the model that includes HOMA) or creating another predictor that combine them togheter.

After the two trials, we can conclude that the first method in order to remove collinearity issue is more convenient for our dataset: we see a slighter drop in R-squared value.

```
best1 = lm(BMI ~ Glucose + HOMA + Leptin + Adiponectin + MCP.1 + Classification, data = data )
summary(best1)
```

```
##
## Call:
## lm(formula = BMI ~ Glucose + HOMA + Leptin + Adiponectin + MCP.1 +
##     Classification, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6730 -2.4794 -0.6159  2.5869 11.1359
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     24.4923983  2.1744090  11.264  < 2e-16 ***
## Glucose          0.0009811  0.0235157   0.042  0.96680
## HOMA            -0.1514420  0.1403146  -1.079  0.28284
## Leptin           0.1519879  0.0202116   7.520 1.65e-11 ***
## Adiponectin     -0.1561696  0.0537676  -2.905  0.00445 **
## MCP.1            0.0030772  0.0011028   2.790  0.00622 **
## Classification2 -1.2661242  0.7835282  -1.616  0.10900
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.833 on 109 degrees of freedom
## Multiple R-squared:  0.4475, Adjusted R-squared:  0.4171
## F-statistic: 14.71 on 6 and 109 DF,  p-value: 2.974e-12
```

```
summbest = summary(best1)
```

However, comparing R squared, F statistics and Residual standard error of the two models (first model is the best one, while the second is the one without collinearity issues) I prefer the best model, since my objective is to obtain precise predictions for the response BMI.

7. DIAGNOSTICS

Outliers In order to discover if there are any outliers in the model, we compute standardized residuals and plot them against fitted values. A point is considered an outlier if its studentized residuals are lower than -3 or higher than 3.

```
rsta = rstandard(best)
range(rsta)
```

```
## [1] -2.430917  2.937989
```

The maximum standardized residual, corresponding to patient 109, is less than 3. We conclude that patient 109 is not an outlier.

High Leverage points We now check for leverages points : the i-th data is an High Leverage Point if it is extreme in the regressors space and have unusual value of the predictors.

In order to the rule of thumb, a leverage point is considered high if is higher than 2 * (p+1)/n. Our threshold is equal to $2 * (7 + 1)/116 = 0.138$ We compute the leverage starting from the Hat matrix:

```
infl = influence(best)
hat = infl$hat
```

```
sum(hat)
```

```
## [1] 8
```

```
hat[which(hat>=(2*8/nrow(data)))]
```

```
##        14        17        72        75        79        86        87        88
## 0.1590392 0.1677641 0.3168345 0.1681942 0.3836548 0.1508952 0.1445500 0.5757920
##        89       110       115       116
## 0.3937724 0.1481789 0.1407101 0.1430079
```

High leverage points are extreme points in the regression space, it means that they are not “bad”, it is useful to identify them because this may have been wrongly reported or in general this is a red flag for further problems.

Influential points We now search for influential points: elements that, if removed from the model, would cause a large change in the fit. In general, Influential points could be outliers or high-leverage points (or both).

```
cook = cooks.distance(best)
cook[which.max(cook)]
```

```
##        116
## 0.09182972
```

In order to the rule of thumb for cook’s distance data whose Di (cook) is larger than 1 should be declared to be influential. In practice, Di $>$ 0.5 is used as an empirical threshold. The largest Di of the model is substantially less than one Di = 0.09, it means that this is not considered an influential point and the deletion of the case will not change the behaviour of the model too much.

```
par(pty="s",mfrow=c(1,3))

plot(fitted(best), rsta, xlab="Fitted Values", ylab="Standardized Residuals",
     main = "Outliers" , pch=19, cex=0.8,ylim=c(-3,3))
abline(h=0, col='violet', lwd = 2)

plot(hat, ylab="Leverages", main="High leverage points", pch=16)
abline(h= 2*8/116, lty=2, col="violet", lwd=2)
high_lev = hat > 0.14
x_hl = which(high_lev)
y_hl = (hat)[high_lev]
text(x_hl, y_hl, labels = x_hl, cex = 1, pos = 2)

mtext("Figure 8: Outliers, High Leverage points, Influential points.",
      side = 1, line = 5, cex =0.8)

cook = cooks.distance(best)
plot( cook, pch = 16, ylim = c(0,1), ylab = "Cook's distance",
      main =  "Influential points")
abline(h = 1, col = 'violet', lty = 2, lwd = 2)
abline(h = 0.5, col = 'turquoise', lty = 2, lwd = 2)
```

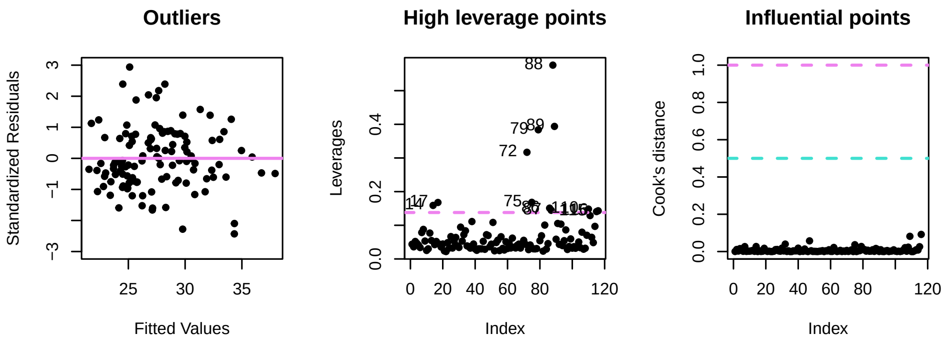


Figure 8: Outliers, High Leverage points, Influential points.

From the graphs emerge that there are 12 points that have a leverage higher than 0.138. However, none of that points appears to be an Influential point, reason why we keep them in the model.

8. MODEL IMPROVEMENT

Indeed, after the drop of the point with maximum cook distance, we see a slight improvement in our model : R-squared and F statistics increase while the p-value decreases. In addition to that, p-value of Glucose coefficient decrease, which means that the predictor becomes more significant.

```
bestcook = lm(BMI ~ Glucose + Insulin + HOMA + Leptin + Adiponectin + MCP.1 +
                Classification, data = data, subset=(cook < max(cook)))
summary(bestcook)
```

```
##
## Call:
## lm(formula = BMI ~ Glucose + Insulin + HOMA + Leptin + Adiponectin +
##     MCP.1 + Classification, data = data, subset = (cook < max(cook)))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.1707 -2.4238 -0.6132  2.3595 10.1390
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    18.306616   2.699945   6.780 6.82e-10 ***
## Glucose         0.057593   0.027549   2.091 0.038933 *
## Insulin         0.375485   0.116704   3.217 0.001712 **
## HOMA           -1.336317   0.387054  -3.453 0.000797 ***
## Leptin          0.158886   0.020009   7.941 2.14e-12 ***
## Adiponectin    -0.142380   0.051070  -2.788 0.006279 **
## MCP.1           0.003278   0.001065   3.078 0.002645 **
## Classification2 -1.766676   0.765722  -2.307 0.022968 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.628 on 107 degrees of freedom
## Multiple R-squared:  0.5139, Adjusted R-squared:  0.4821
## F-statistic: 16.16 on 7 and 107 DF,  p-value: 2.336e-14
```

9. INTERPRETATION OF THE PARAMETERS OF THE BEST MODEL

```
best = lm(BMI ~ Glucose + Insulin + HOMA + Leptin + Adiponectin + MCP.1 +
            Classification, data = data )
summary(best)
```

```
##
## Call:
## lm(formula = BMI ~ Glucose + Insulin + HOMA + Leptin + Adiponectin +
##     MCP.1 + Classification, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6352 -2.2885 -0.4291  2.3866 10.4456
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     19.057849   2.720463   7.005 2.21e-10 ***
## Glucose          0.051788   0.027860   1.859  0.06577 .
## Insulin          0.370541   0.118581   3.125  0.00229 **
## HOMA            -1.305013   0.393076  -3.320  0.00123 **
## Leptin           0.146879   0.019514   7.527 1.65e-11 ***
## Adiponectin     -0.150405   0.051761  -2.906  0.00445 **
## MCP.1            0.003577   0.001073   3.334  0.00117 **
## Classification2 -1.855452   0.777044  -2.388  0.01868 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.687 on 108 degrees of freedom
## Multiple R-squared:  0.4933, Adjusted R-squared:  0.4605
## F-statistic: 15.02 on 7 and 108 DF,  p-value: 1.423e-13
```

Considering the best model, we now give an interpretation of parameters. The intercept is 19.05 kg/m^2. It corresponds to the expected Body Mass Index of a person with no age and no levels of Glucose, Insulin, HOMA, Leptin, Adiponectin and Resistin in the blood. In order to that, it is not interpretable because no such person could exist.

An increase of one unit in Glucose is associated with an estimated increase of 0.0518 kg/m^2 in BMI, holding other predictors costant.

All the others quantitative variables can be interpreted in the same way.

The coefficient of Classification2 represents the estimated average difference in BMI between healthy patients and cancer patients. In this case, a value of -1.855 indicates that patients classified as cancer patients have an estimated average BMI lower by 1.855 kg/m^2 compared to those that are healthy.

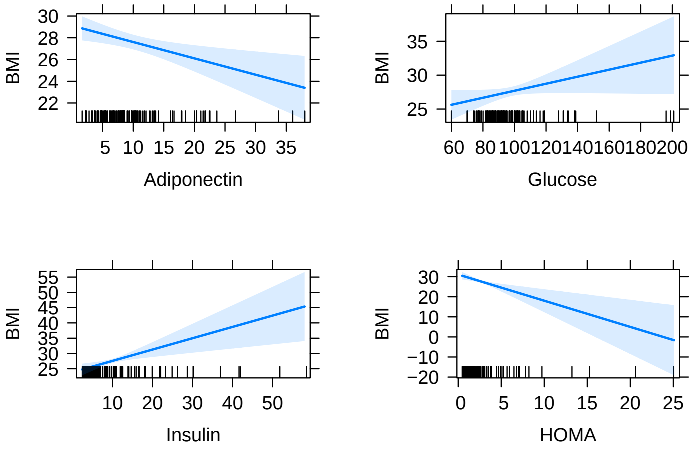
Confidence Intervals

We represent the confidence intervals obtained with the function predictorEffect(). The table below contains 95% confidence intervals cof estimated coefficients.

```
confint(best)
```

```
##                        2.5 %       97.5 %
## (Intercept)     13.665418022 24.450279190
## Glucose         -0.003435955  0.107012182
## Insulin          0.135491960  0.605589481
## HOMA            -2.084156944 -0.525869107
## Leptin           0.108199738  0.185558229
## Adiponectin     -0.253004251 -0.047805316
## MCP.1            0.001450499  0.005704142
## Classification2 -3.395686639 -0.315216483
```

```r
library(gridExtra)
grid_top = grid.arrange(plot(e1.ols, main = ""), plot(e2.ols, main = ""),
                        plot(e3.ols, main = ""), plot(e4.ols, main = ""),
                        ncol = 2)
```



```r
grid_bottom = grid.arrange(plot(e5.ols, main = ""), plot(e6.ols, main = ""),
                           plot(e7.ols, main = ""), ncol = 2)
```

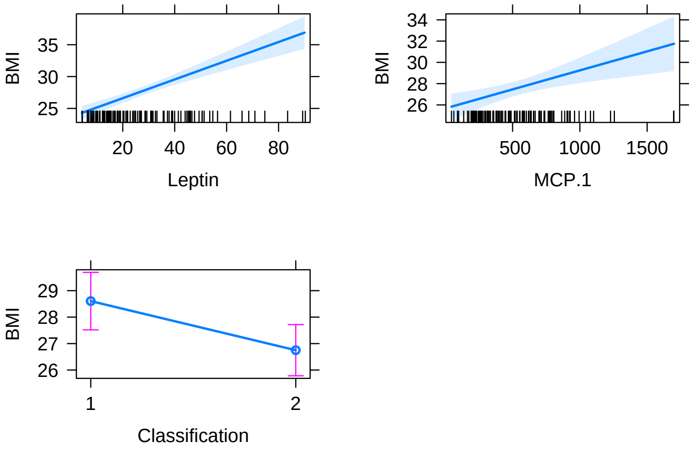
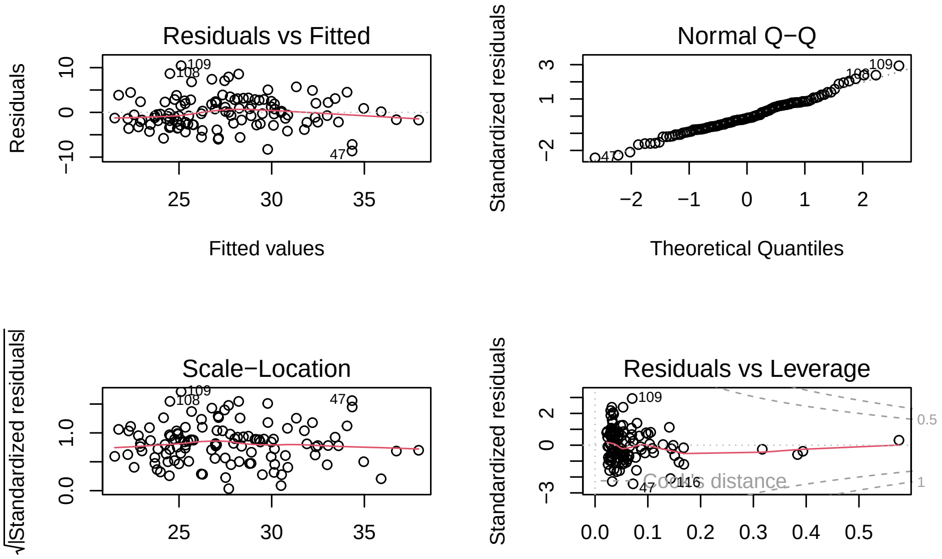


Figure 9: Confidence Intervals for the Coefficients

```r
par(mfrow=c(2,2))
plot(best)
mtext("Figura 10: Graphical interpretations of the Residuals.", side = 1,
      line = 4.2 , adj = 4, cex =0.8)
```



ıura 10: Graphical interpretations of the Residuals.

- Analyzing the Residual vs Fitted, we see a scatterplot that is overall symmetric to 0 (marked by a smooth line and without any significant inclinations) and a good distribution of points in vertical direction: it means that there aren't issues of non-linearity and non-costant variance.
- Scale-location plot show us that there aren't square roots of standardized residuals $> |1.7|$, which confirm the fact that there aren't any outliers.
- Q-Q plot is useful to check for normality of residuals: the model follows a normal distribution, since we do not see signifant tails of points that do not follow the straight line.
- Residuals vs Leverage confirm that there aren't ingìfluential points due to the low cook's distances.

10. TEST OF EACH Bj TO BE 0

We test the following hypotesis for every Beta coefficient of the fitted model: H0 : Betaj = 0 , while the other Betas are arbitrary H1 : Betaj != 0, while the other Betas are arbitrary.

```
pvaluecoef = summbest$coefficients[,4]
pvaluecoef
```

```
##     (Intercept)         Glucose            HOMA          Leptin     Adiponectin
##    5.521318e-20    9.667990e-01    2.828352e-01    1.646715e-11    4.454018e-03
##           MCP.1 Classification2
##    6.216503e-03    1.090009e-01
```

We refuse the null hypotesis in every case in which the p-value is lower than 0.05

11.a TESTING A GROUP OF REGRESSORS

We test the hypothesis that both Glucose and Classification may be excluded from the model, i.e. Beta1 = Beta7 = 0, I chose Glucose and Classification because they're the less significant predictors in the model.

```
best0 = lm( BMI ~ Insulin + HOMA + Leptin + Adiponectin +
    MCP.1, data = data)
anova(best0, best)
```

```
## Analysis of Variance Table
##
## Model 1: BMI ~ Insulin + HOMA + Leptin + Adiponectin + MCP.1
## Model 2: BMI ~ Glucose + Insulin + HOMA + Leptin + Adiponectin + MCP.1 +
##     Classification
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1    110 1561.1
## 2    108 1468.5  2    92.626 3.4061 0.03677 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

P-value < 0.05 so we can reject the null Hypotesis and confirm that the two predictors are significant in our model.

11.b TESTING ALL THE REGRESSORS

This test helps us to understand if the predictors are useful in predicting the response. We write the null hypothesis as: Beta1 = Beta2 = Beta3 = Beta4 = Beta5 = Beta6 = Beta7 = 0

```
Fstat = (summbest$r.squared/7) / ((1-summbest$r.squared)/(n-7-1))
1 - pf(Fstat, 7, n-7-1)
```

```
## [1] 1.20608e-11
```

Since the p-value is so small, this null hypothesis is rejected and we can affirm that all the predictors considered are useful.

12. MEASURING THE GOODNESS OF FIT

```
summbest$sigma
```

```
## [1] 3.83281
```

```
summbest$r.squared
```

```
## [1] 0.4474999
```

```
summbest$adj.r.squared
```

```
## [1] 0.417087
```

```
summbest$fstatistic
```

```
##     value     numdf     dendf
##  14.71417   6.00000 109.00000
```

In order to analyze the goodness of the fit, is useful to considerate this values:

- The residual standard error (sigma) is a measure of the typical deviation of the observed values from the fitted values, indicating that the typical difference between observed and predicted BMI values is approximately 3.687 units.
- The multiple R-squared value (0.4933) represents the proportion of variance in the response variable (BMI) that is explained by the predictors in the model. This suggests that there are other factors not included in the model that influence BMI: they could be total calorie intake, macronutrient intake (protein, carbohydrates, fats), consumption of high-fat or high-sugar foods, physical activity habits, chronic stress and mental health. However, it is understandable that this informations are not included because the dataset was a collection of blood analysis controls in patients of breast unit disease.
- The adjusted R-squared value (0.4605) is a modified version of the R-squared value that adjusts for the number of predictors in the model. In this case, the adjusted R-squared is slightly lower than the multiple R-squared, indicating that some of the predictors may not be contributing significantly to the model.
- The F-statistic tests the overall significance of the regression model by comparing the variability explained by the model to the variability not explained. In this case, the F-statistic is 15.02 with

degrees of freedom (DF) of 7 and 108. The associated p-value (1.423e-13) is very small, indicating that the regression model is statistically significant overall.

13. PREDICTION OF THE RESPONSE VARIABLE WITH ASSOCIATED UNCERTAINTY

```r
newdata = data.frame(Glucose = 90, Insulin = 8.8, HOMA = 3, Leptin = 28.1,
          Adiponectin = 12.3, MCP.1 = 700.34, Classification = factor(1))
predict(best, newdata = newdata, interval = "prediction", level = 0.95)
```

```
##        fit      lwr      upr
## 1 27.84716 20.43037 35.26395
```

The predicted value of the response variable (BMI) for the new data is approximately 27.85. The lower bound of the prediction interval at a 95% confidence level is approximately 20.43 while the upper bound is approximately 35.26.

14. SIMULATION OF N DATA POINTS

I create a vector y containing 116 fitted values. To assess the accuracy of the model's predictions compared to the actual data points within the dataset, we can calculate the mean squared error (MSE). This high MSE value indicates that the model demonstrates a relatively far match to the dataset, as, on average, the difference between the observed and predicted data points is high.

```r
n = 116
set.seed(3)
beta = coefficients(best)
X = model.matrix(best)
y = X %*% beta + rnorm(116, 0, sigma(best))

MSE = mean((data$BMI - y)^2)
MSE
```

```
## [1] 21.71944
```

CONCLUSIONS

This study aimed to explore the correlation between BMI and various biomarkers. Through this analysis, several key findings have emerged. Firstly, our exploratory analysis revealed significant relationships between BMI and biomarkers such as Glucose, Insulin, HOMA, Leptin, Adiponectin, and MCP-1. The scatterplot matrix and boxplot visualization demonstrated the associations between these variables and BMI, providing insights into potential predictors of BMI variation. Secondly, using linear regression and best subset selection techniques, we identified the optimal model with seven predictors: Glucose, Insulin, HOMA, Leptin, Adiponectin, MCP-1, and Classification. Furthermore, diagnostic tests were conducted to assess model assumptions and limitations. While the model performed well overall, there were some violations of assumptions such as collinearity between certain predictors (HOMA and Insulin). However, these issues were mitigated through careful model selection and interpretation. The findings contribute to our understanding of the complex interplay between biological factors and BMI, with implications for predictive modeling and clinical practice.