Project I: Regression analysis SF2930

Asmi Souhail , Foufa Mastafa

February 19th, 2019

1 The chosen data

For this project we have chosen the data set Bodyfatmen which represents Scenario I. In the following section, we will present our methodology for constructing a linear model that fits well the data that we have chosen.

2 Introduction and project goals

We start by distinguishing the target variable which corresponds to the feature 'density' from the other columns that are gathered in the matrix X

We scaled the data so that we avoid problems of regressors's magnitude. We brought, for each feature, variance to one and mean to zero for this purpose.

Then, we divided our data into two sets: a training set (that represent 80% of the data) and a test set so that we can compare the methods using some metrics on the test set which allows us to avoid overfitting.

Our methodology to construct a linear model that fits well the data is presented in these steps:

1. We fitted the largest possible model to the data (Full model)
2. Residual analysis.
3. The possibility of a transformation
4. Diagnostic for leverage and influence points.
5. Checking multicolinearity : determining all possible subsets and comparing them using Cp and R^2_{adj} .
6. PCR method on the data to see the best subset of regressor in terms of R^2_{adj} and MSEP.

7. Ridge regression and Lasso on the data.

8. Results and conclusion.

3 Analysis and comments

Through this analysis we aim to fit a linear model to our data that verifies all the assumptions that we made on the error. We think at first that our responses verify:

$$Y = X\beta + \epsilon \tag{1}$$

such that ϵ follows a normal distribution $\mathcal{N}(0, \sigma^2)$. we will check the normality assumption and the constant variance through the residuals analysis since the residuals can be considered as observations of the error.

3.1 An overview of the Data

The response variables represent the density of 198 men (training set) and the features are presented in Figure 2 after scaling the data matrix.

Figure 1 presents a histogram of the scaled responses. we see that the distribution of the responses has a normal shape which is good sign because under the normality assumption of the error, the responses must also follow a normal distribution with mean $X\beta$ and constant variance σ^2

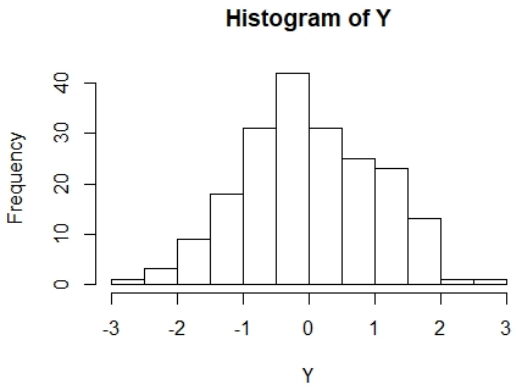

Figure 1: Histogram of the responses

Figure 2 presents a short summary of the scaled Data. We see that we have 13 features and 198 observations. The most important thing in this analysis is the prior knowledge

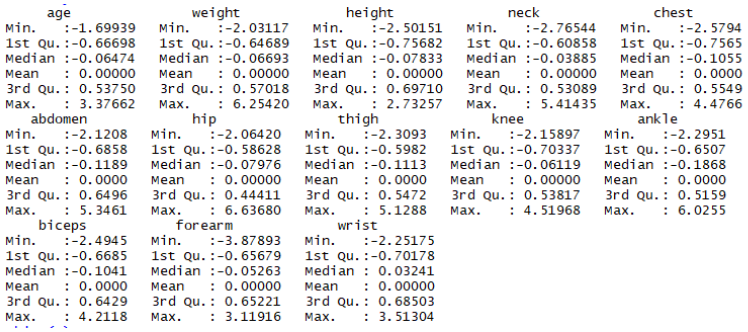

Figure 2: Summary of the scaled data

of the dependencies between all the variables. Since we are interested in finding a linear relationship between the Y (density) and the other features it's necessary to check the correlation matrix using Pearson dependence. We focus on the first row (correlation of the response Y with the other regressors)of the correlation matrix presented in Figure3. We see a high negative correlation between density and abdomen so we can expect rejecting the t statistic concerning $\beta_{abdomen}$ with a small p value. This means that the feature abdomen is relevant in our linear model. We see also a remarkable correlation between density and the regressors weight, chest, hip ,thigh and knee. In addition to that, an important aspect that we have to focus on in our choice of regressors is the correlation between the features. This aspect is further analyzed in section 3.6.

3.2 Fitting the full model

Our first step in this analysis consists in fitting the largest possible model to the data. Figure 4 show a summary of the full model fit. We can see that the significance of the fit is R^2_{adj}=0.7366 which is a good value. The null hypothesis H_0 of the F-statistic were rejected with a p-value=0 which proves that the linearity of the full model is significant. Now let's focus on the contribution of each regressor to the response. Since the data is scaled, the estimated $\hat{\beta}$'s present the contribution to the response after a unit variation of the regressor. We can see that abdomen has the highest estimate $\hat{\beta}_{abdomen}$=-1.234 and that the H_0 of the corresponding t test were rejected with probability 0. Focus on the beta estimates can be valuable here since data is scaled. For example, some features present near-zero coefficient estimates meaning that their contribution to the linear model is very likely to be low (to strengthen our assessment, we can also check the t-test for a particular feature). It is the case of chest, height and knee. Indeed when computing the linear regression model in python on all the data set, we have (cf. Fig.5.):

- $\hat{\beta}_{height}$ = -6.694e-05
- $\hat{\beta}_{chest}$ = 9.823e-05

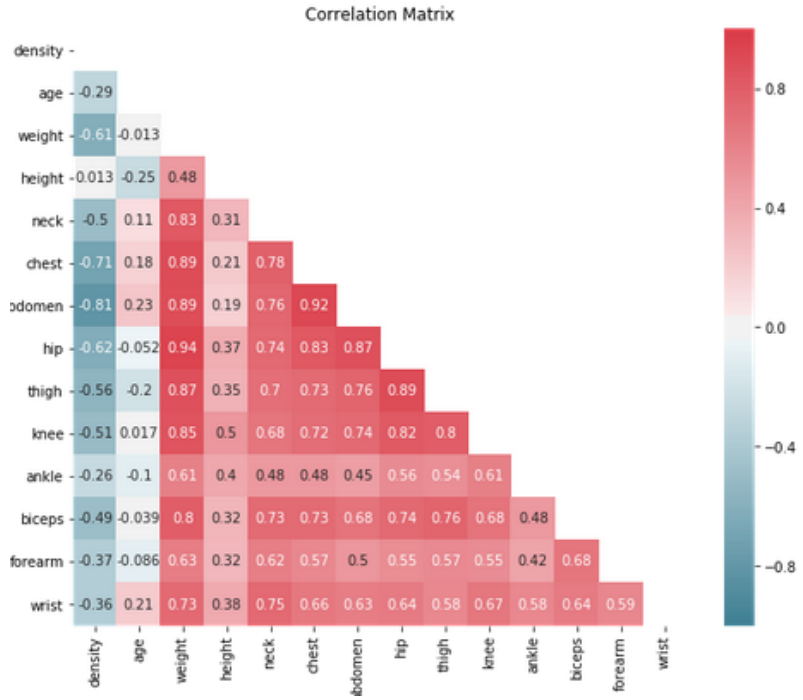

Figure 3: Correlation matrix

- $\hat{\beta}_{knee}$ = -8.217e-05

We can see also that some values has good linearity significance in this model since there t tests where rejected with small probabilities for example wrist ,age and neck . There is also some features as height , biceps and Knee that have low estimates (e-02) and there t test were rejected with high probabilities. As an example : $Pr_{knee}(> \|t\|) = 0.857$. Now we have to check the test error on the testing set. our testing set consist on the last 50 observation of the main data . After predicting the responses of the testing set the model has a test error equal to MSE_full=0.299 .

3.3 Empirical distributions of the regression variables via boostraped samples

Another way to check the normality assumption of the errors is to verify the distributions of the $\hat{\beta}$'s. Since the least square estimates are linear combination of the responses y they must follow a normal distribution. the problem is how to generate a sample of $\hat{\beta}$. the solution to this kind of problem is resampling from $\mathcal{S}$=$\{(x_1, y_1), ..., (x_n, y_n)\}$ via Boostrap. Each histogram in Figure 5 show the empirical distribution of each least square estimate and we can see the normal shape of these estimates which confirms one more time our

	coef	std err	t	P>|t|	[0.025	0.975]
ones	1.0554	0.001	1699.337	0.000	1.054	1.057
age	-0.0017	0.001	-1.785	0.075	-0.004	0.000
weight	0.0070	0.004	1.689	0.093	-0.001	0.015
height	-6.694e-05	0.001	-0.064	0.949	-0.002	0.002
neck	0.0026	0.001	1.995	0.047	3.24e-05	0.005
chest	9.823e-05	0.002	0.050	0.961	-0.004	0.004
abdomen	-0.0237	0.002	-10.618	0.000	-0.028	-0.019
hip	0.0037	0.002	1.579	0.116	-0.001	0.008
thigh	-0.0033	0.002	-1.901	0.058	-0.007	0.000
knee	-8.217e-05	0.001	-0.061	0.952	-0.003	0.003
ankle	-0.0008	0.001	-0.871	0.385	-0.002	0.001
biceps	-0.0013	0.001	-1.084	0.279	-0.004	0.001
forearm	-0.0021	0.001	-2.298	0.022	-0.004	-0.000
wrist	0.0034	0.001	2.976	0.003	0.001	0.006

Omnibus:	3.742	Durbin-Watson:	1.725
Prob(Omnibus):	0.154	Jarque-Bera (JB):	2.571
Skew:	0.051	Prob(JB):	0.277
Kurtosis:	2.512	Cond. No.	21.0

Figure 4: Summary after fitting all the regressors on ALL the data – PYTHON (Y is not standardized here)

assumptions about the error.

3.4 Analysis of the residuals

3.4.1 Normality assumption

In this section we will verify the assumptions that we made about the errors and we check if there is a need of transformation on the data . The first step of the residuals analysis will be verifying that the residuals follows a normal distribution. To do that we will compare the empirical quantiles of the residuals to the theoretical quantiles of the normal distribution.

```
Call:
lm(formula = Y ~ ., data = X)

Residuals:
    Min      1Q  Median      3Q     Max
-1.1187 -0.3653  0.0217  0.3650  1.0682

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -6.317e-16  3.647e-02   0.000   1.0000
age         -1.326e-01  5.129e-02  -2.586   0.0105 *
weight       4.477e-01  2.476e-01   1.808   0.0722 .
height      -2.710e-02  6.250e-02  -0.434   0.6651
neck         1.829e-01  7.590e-02   2.410   0.0169 *
chest       -1.210e-01  1.117e-01  -1.083   0.2801
abdomen     -1.234e+00  1.279e-01  -9.643   <2e-16 ***
hip          2.928e-01  1.438e-01   2.037   0.0431 *
thigh       -2.793e-01  1.108e-01  -2.522   0.0125 *
knee         1.454e-02  8.149e-02   0.178   0.8586
ankle       -7.265e-02  5.038e-02  -1.442   0.1510
biceps      -4.613e-02  7.008e-02  -0.658   0.5112
forearm     -1.354e-01  5.445e-02  -2.487   0.0138 *
wrist        2.051e-01  6.770e-02   3.030   0.0028 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5132 on 184 degrees of freedom
Multiple R-squared:  0.754,     Adjusted R-squared:  0.7366
F-statistic: 43.38 on 13 and 184 DF,  p-value: < 2.2e-16
```

Figure 5: summary after fitting all the regressors on TRAINING data (Y is also standardized here)

Figure 6 shows the QQ plot of the residuals quantiles vs the normal quantiles and we see a linear relationship between these two which confirms the normality assumption, we can see that we don't have problem on the left and right tail of the empirical distribution which indicates a perfect match.

3.4.2 Constant variance assumption

Step 2 will consist on verifying the constant variance of the residual's normal distribution. In other words we want to verify that the variance does not depend on the x_i's. To do so, we plot the graph of the residuals versus the fitted values. Figure 8 presents the residuals in three forms (studentized and R-student residual) vs the fitted values.

The graphs indicates that the residuals can be contained in a horizontal band, then there are no obvious model defects (the variance of the residuals does not depend on the observations). So we can conclude that we have constant variance.

After checking the two main assumption, we can be sure that we don't need any trans-

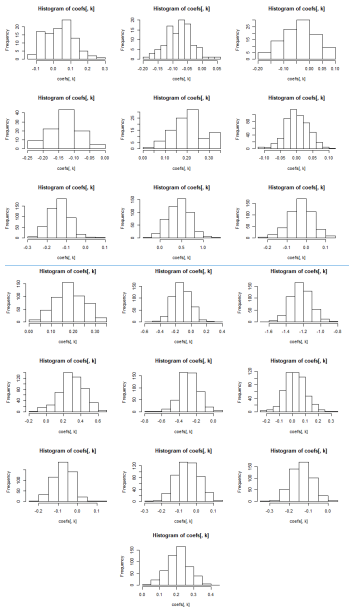

Figure 6: The histogram of the regression parameters after using bootstrap by Data resampling

formation on the responses or the data.

3.4.3 Partial regression plot

This part is not really made to check any assumption. Partial regression is a way to verify that a regressor is really linearly dependant of the responses.

The partial regression plot for regressor variable x_i is obtained by plotting the y residuals $e_i(y\|Others)$ (where others means all the regressors except x_i) against the x_i residuals $e_i(x_i\|Others)$. If the regressor x_i enters the model linearly, then the partial regression plot should show a linear relationship, that is, the partial residuals will fall along a straight line with a nonzero slope. The slope of this line will be the regression coefficient of x_i in the multiple linear regression model .If the partial regression shows a curvilinear graph , a transformation must be made on x_i.

From Fig.10, we see first that the highest slope is the slope of abdomen which confirms one more time the strong linear relation between the density and abdomen. Another remark that has to be made is that the slope of knee, height and chest is practically null which proves that we have to think about removing these regressors when we deal with the choice of regressors.(check section on Multicolinearity for further analysis). We don't see any particular curvilinear shape so we don't need any transformation on the regressors.

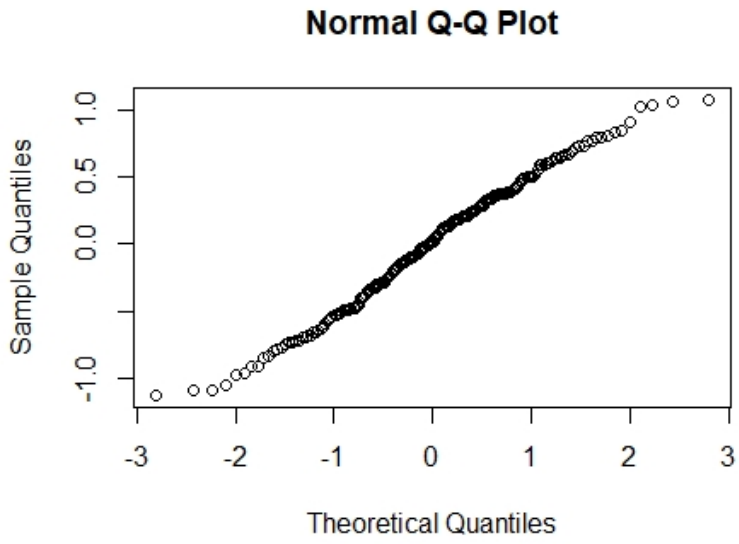

Figure 7: Q-Q plot of the residuals vs normal

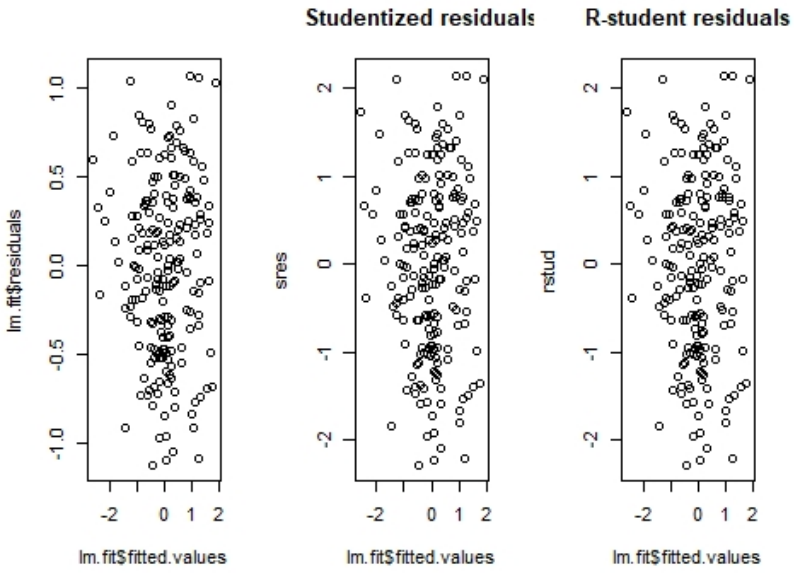

Figure 8: Residuals vs fitted

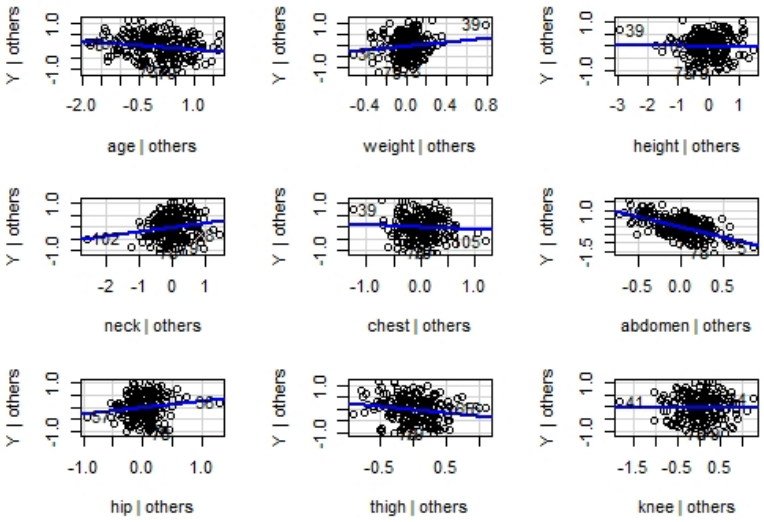

Figure 9: Residuals Y—others vs X_i—others

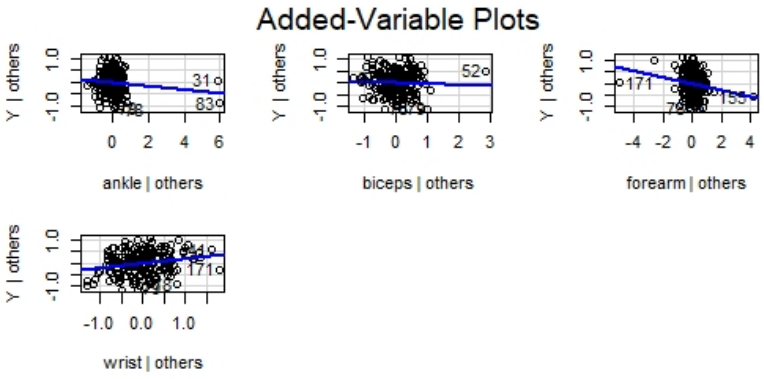

Figure 10: Residuals Y—others vs X_i—others

3.5 Detecting influential points and outliers

In this section we aim to reprocess the data, in other words we will try to detect the points that have bad influence on the data by using several measures.
First measure that we will consider in our analysis is the Hat matrix. We know that $h_i i$ is a standardized measure of the distance of the i'th observation from the center (or centroid) of the x space. If the diagonal value exceeds the leverage cutoff=0.1414 we have a leverage point. The second measure is the Cook's measure, we can interpret it as the squared Euclidean distance that the vector of fitted values moves when the i'th observation is deleted. Each point of the data having a cook distance exceeding the Cook's cutoff =0.9528 will be considered as an influencial point.

```
> leverage.cutoff
[1] 0.1414141
> cooks.cutoff
[1] 0.9562812
> studres.cutoff
[1] 1.972941
>
```

Figure 11: Different distance cutoffs

We see from Fig.13 that the maximal Cook's distance in the training set is 0.25 corresponding to x_{39} , this value is way far from the cutoff . So we can conclude that we don't have influence points in our data . On the other hand , we have some leverage points as x_{171},x_{31} and x_{39}.

Now we will consider other measures that provides us with information about the importance of each point of the data toward the estimated least square variables $DFBETAS_{j,i}$ and the fitted values $DFFITS_i$ .These two measures computes respectively the variation in $\hat{\beta}_j$ and $\hat{y}$ after removing the observation x_i. Another measure that we will consider is the $COVRATIO_i$. If the $COVRATIO_i > 1$ than the ith observation improves the precision of estimation, while if $COVRATIO_i < 1$, inclusion of the ith point degrades precision. The table in Fig.13 presents all the measures that we have presented already of the critical points in the data. The conclusion that we can make from this table is that if we base our analysis on the COVRATIO we see that practically all the point improves the precision of the estimation. In addition , all the DFBETAS and the DFFITS are approximately null which proves that there is not an observation that has major effect on the estimated least squares or the fitted values.

Conclusion: the result of our diagnostic of influencial point is that there is no need to remove any point from our data set.

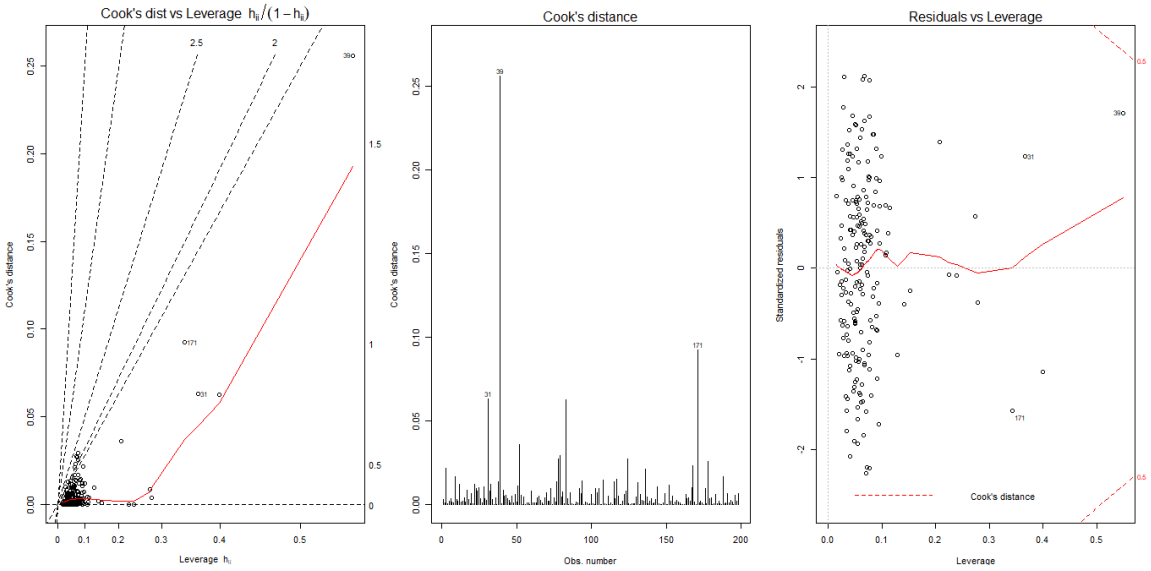

Figure 12: Remote and outliers analysis

```
Potentially influential observations of
        lm(formula = Y ~ ., data = X) :

    dfb.1_ dfb.age dfb.wght dfb.hght dfb.neck dfb.chst dfb.abdm dfb.hip dfb.thgh dfb.knee dfb.ankl dfb.bcps dfb.frrm dfb.wrst dffit   cov.r   cook.d
5   -0.03   0.08   -0.01    0.02     0.06     0.06    -0.09     0.05    0.02    -0.07     0.00    -0.02     0.01    -0.02    -0.16  1.24_*  0.00
31   0.11  -0.09   -0.26    0.22     0.23     0.10     0.14     0.08   -0.06    -0.15     0.89     0.14    -0.11    -0.17     0.94_*  1.52_*  0.06
36  -0.03   0.00    0.11   -0.03    -0.02    -0.10    -0.01    -0.16    0.06     0.01    -0.01     0.01    -0.07     0.06    -0.24  1.48_*  0.00
39   0.18   0.15    0.98   -0.91     0.19    -0.69    -0.38     0.13   -0.43    -0.12    -0.03     0.03    -0.72    -0.26     1.90_*  1.91_*  0.26
41   0.05  -0.01   -0.03   -0.02    -0.07     0.05     0.01     0.14    0.00    -0.19     0.02    -0.08     0.05     0.14     0.35  1.45_*  0.01
57  -0.02  -0.01   -0.04    0.00     0.00    -0.03    -0.01     0.07   -0.02     0.02     0.01     0.03     0.02     0.02    -0.11  1.27_*  0.00
83  -0.11  -0.22    0.02    0.09     0.00     0.03    -0.02     0.00    0.06     0.13    -0.87    -0.13     0.00     0.22    -0.94_*  1.63_*  0.06
102 -0.01   0.00   -0.01    0.02     0.03     0.01     0.00     0.00    0.01    -0.01     0.00     0.00    -0.01    -0.01    -0.04  1.39_*  0.00
155 -0.01   0.00    0.00    0.00     0.00     0.01     0.00    -0.01    0.01     0.00     0.00     0.01    -0.04     0.01    -0.05  1.42_*  0.00
171 -0.14   0.26    0.12   -0.06    -0.11    -0.27     0.09    -0.02   -0.08    -0.10     0.18    -0.16     1.02_*  -0.46    -1.14_*  1.36_*  0.09
    hat
5    0.14
31   0.37_*
36   0.28_*
39   0.55_*
41   0.27_*
57   0.15
83   0.40_*
102  0.23_*
155  0.24_*
171  0.34_*
```

Figure 13: Different measures for checking influence points

3.6 Multicolinearity Analysis

In this section, we first apply different methods to diagnose multicolinearity in our design matrix. Then we try to only focus on the relevant features which leads us to a problem of data reduction. Several methods are hence used to perform relevant data reduction.

3.6.1 Multicolinearity Diagnostic

Method 1: In figure 3, the correlation matrix indicates via the Pearson correlation coefficients the pairwise linear relationship among our features (excluding density target values).

The inspection of the correlation matrix indicates indeed that there are several near-linear dependencies in the bodyfat data.

From this matrix, we can directly state the following, where Covn designs the normalised covariance defining Pearson correlation coefficients:

- abdomen and chest : Covn(adbdomen, chest) = 0.92 (check figure 13)
- hip and weight: Covn(hip, weight) = 0.94 (check figure 13)
- chest and weight: Covn(chest, weight) = 0.89 (check figure 13)
- abdomen and weight: Covn(abdomen, weight) = 0.89 (check figure 13)

We can plot the pairwise relationships of those features to better see how relevant is the correlation matrix. For that, we fit simple linear models to each feature given the other. To be more precise, we fit chest with weight, abdomen with weight and hip with abdomen.

Those 4 features (chest, weight, abdomen and hip) are hence linearly linked two by two. And an interesting strategy could be to reduce our dimensions by using only one of those features instead of all of them as in the original dataset. It can be interesting in a real life scenario with many observations and low computational power for example.

Method 2: We focus on the eigen values of the matrix $X^T X$. The more correlated are the columns of such matrix the 'less' it is invertible. And hence we can expect the estimated coefficients of our regression model to inflate as those coefficients depend directly with the inverse of such matrix. The idea behind eigen values analysis is just that if we have very small eigen values then our matrix is not of full rank and hence cannot be inverted. As a consequence, we have unstable estimated coefficients. We start by computing the condition number $\kappa = \frac{\lambda_{max}}{\lambda_{min}}$ of the matrix $X^T X$. Recall that:

- if k $\geq$ 1000 then strong multicolinearity
- if 100 $\leq$ k $\leq$ 1000 then moderate multicolinearity
- if k $\leq$ 100 quasi no multicolinearity

In our case we have κ=464 that indicates moderate multicolinearity.

This should catch our attention as we have seen some serious pairwise linearities in our design matrix. To dive more into this method of eigen values, we plot the ratio between the highest eigen values and the other eigen values. Also, we plot the different eigen values and priori we should have some low values.

We note that analysis of eigenvalues is not such a great technique to catch multicolinearity in our data. Indeed the eigen values are not so low here (even though data is normalized).

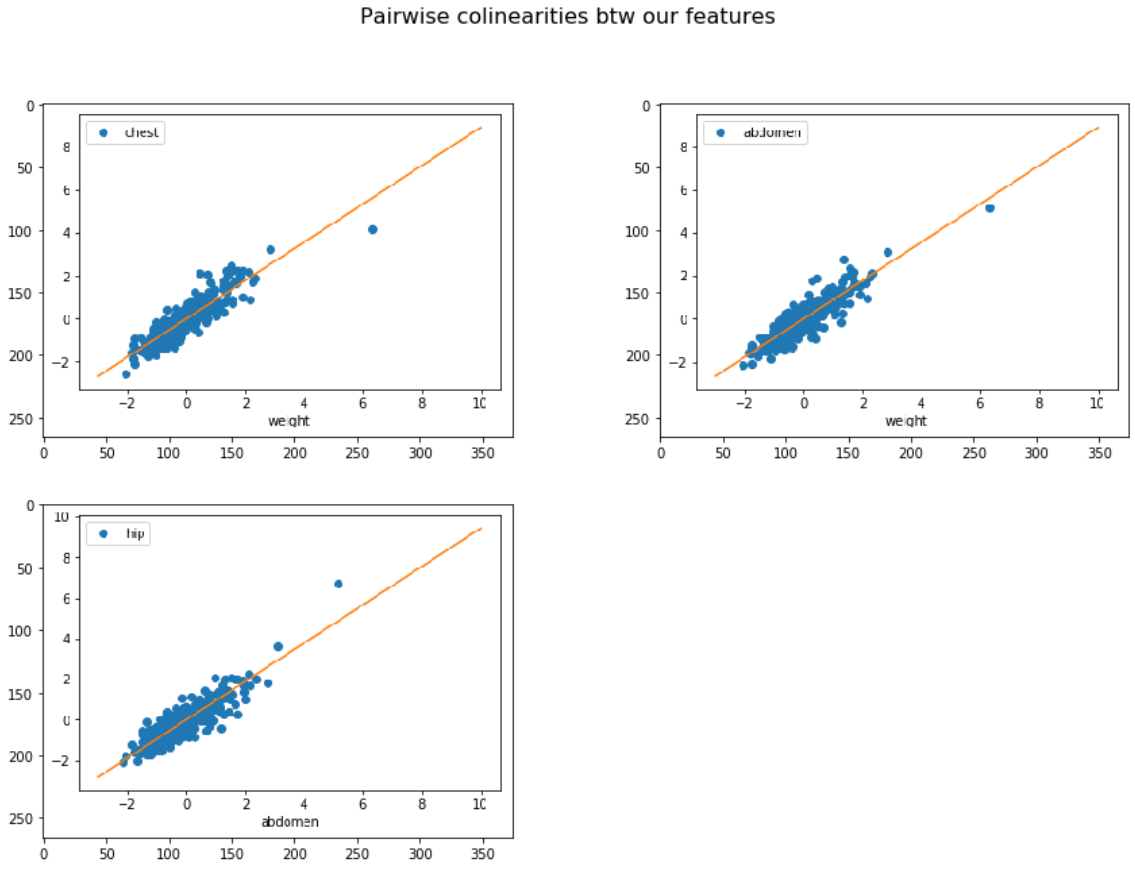

Figure 14: Pairwise plots between certain features

3.6.2 Multicolinearity solutions: focus on four features with Python

In this section, we decide to focus on four features of interest that caught attention after analysis of the covariance matrix as well as the plots of the fitted simple models.

One idea that comes to mind when having four correlated features is to only keep the most relevant one. Relevancy here is examined via Covn(feature, target) ie by analysing the Pearson coefficient between our features and the target feature density.
Step 1: we check the relevancy of each feature. We have:

- Cov(abdomen, y) = 0.81
- Cov(chest, y) = 0.71
- Cov(hip, y) = 0.62
- Cov(weight, y) = 0.61

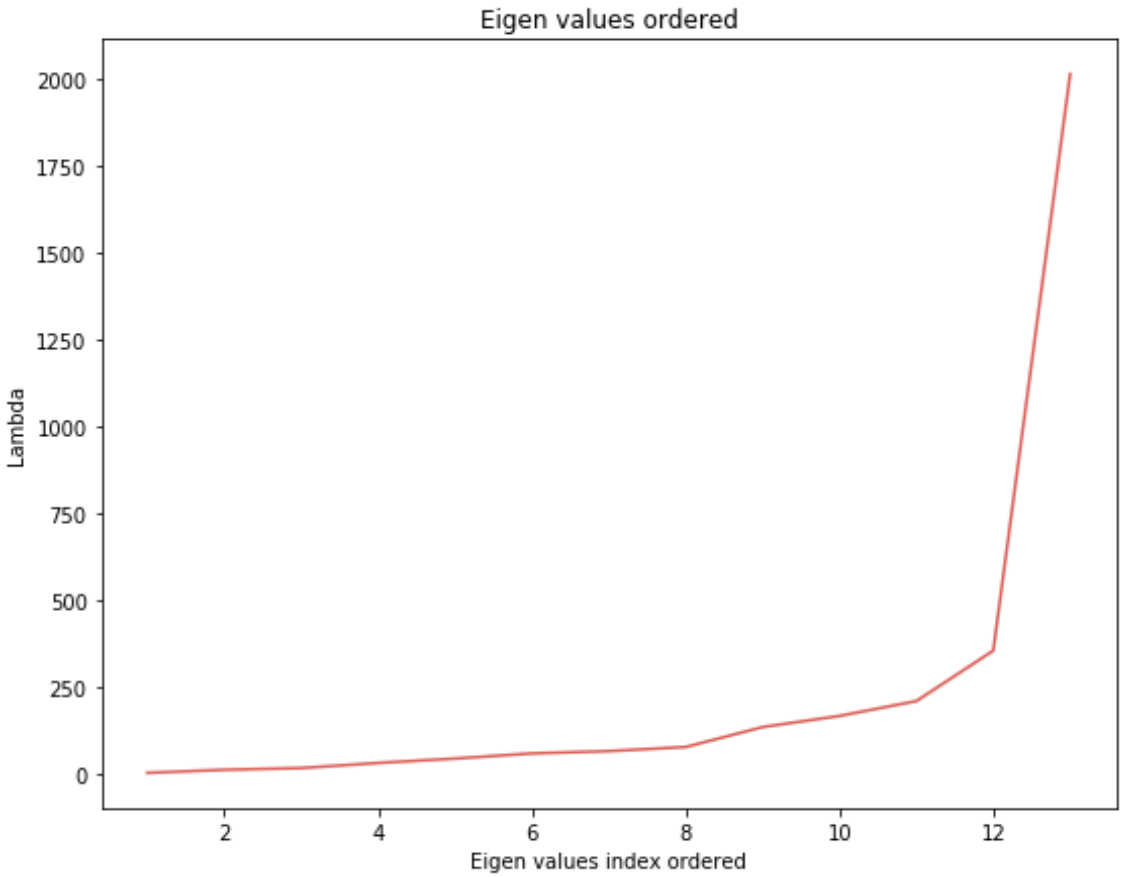

Figure 15: Eigen values of $X^T X$.

So, the most relevant feature among our four features of interest is a priori abdomen. We could decide thus to reject chest, hip and weight. However a more interesting approach is rather to fit feature by feature beginning with the most relevant one and checking certain metrics. We decide to focus on Radjusted**2 and F statistic. The results are summarized in figure 17. We start by only taking abdomen, hence removing chest, hip and weight features and then we add feature by feature starting from the most relevant one after abdomen.

Based on figure 14, a good trade-off may be to only keep two of those features ie keeping abdomen and chest.

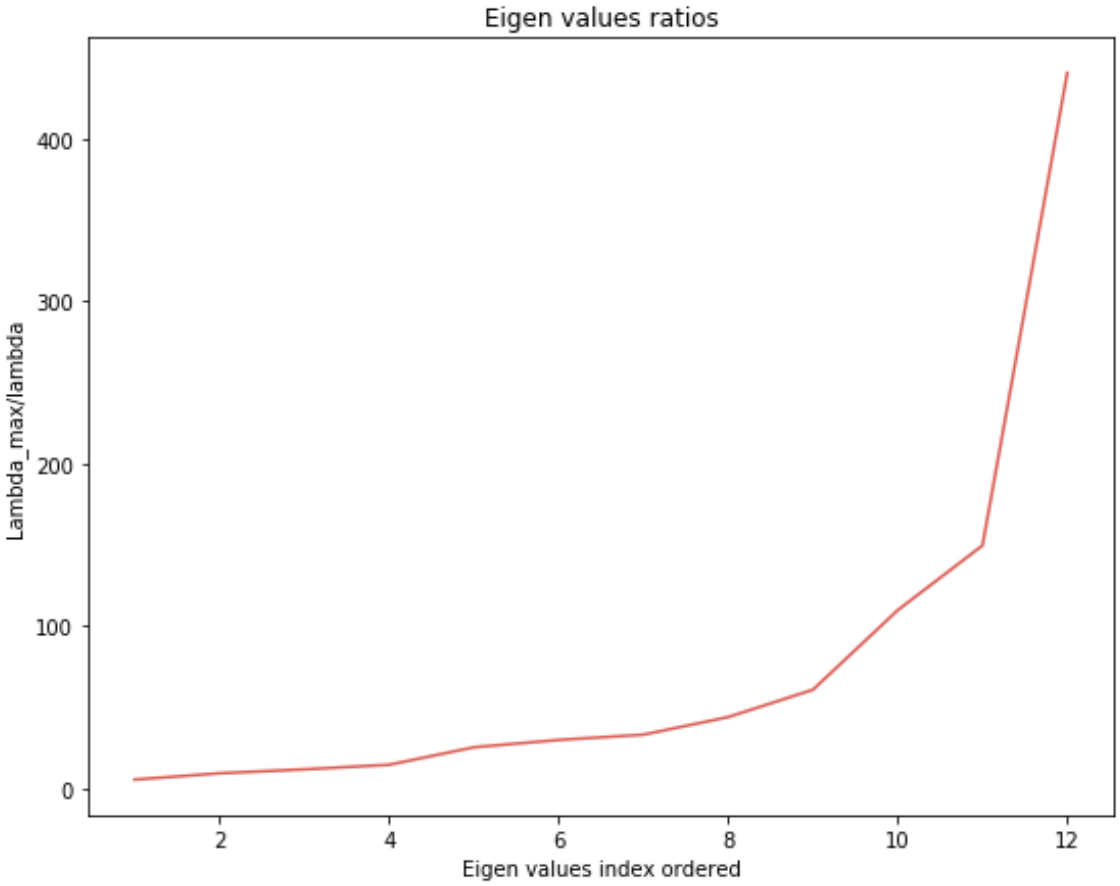

Figure 16: Ratios of eigenvalues $X^T X$.

3.6.3 Multicolinearity solution: method of all possible subsets with R

An alternative to deal with the problem of multicolinearity is to use the method of all possible subsets of features. By cross validation, we can evaluate the performance of our model with all subset of features starting from one feature to all possible features and for each subset we can have the best set of feature focusing on performance of the model. Figure 18 presents different criteria to compare all the possible subsets of this model. we will focus our analysis on three of them R_{adj},C_p and BIC.

Let's first recall that the $R^2_{adj,full}$=0.736 . We can see from graph 2 in Fig12 that the adjusted coefficient of determination corresponding to 8 regressors is practically the same as $R^2_{adj,full}$, we can explain this by the fact that the remaining features don't contribute to much to the model. the Bayesian information criteria shows us that the optimal number of regressors is 7 . So we will compare the accuracy of these two models on the test set.

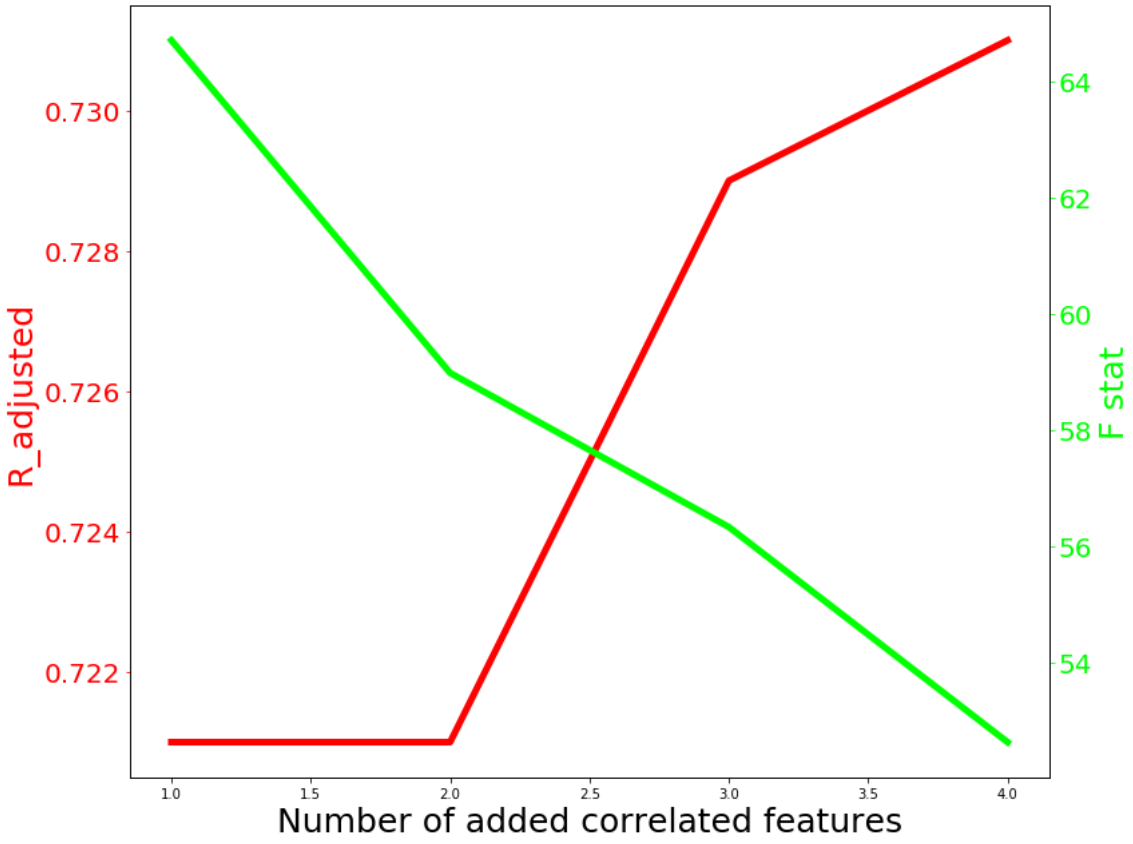

Figure 17: Performance of our linear regression with additional features $X^T X$.

Fig 18 presents the optimal features that we have to consider in our reduced model if we have a limited number of regressors. We can see that if we must choose one regressor we must choose abdomen. We tried also the forward method which gives us the same results as Fig14.

Fig 19 and Fig 20 presents the coefficients of the fit corresponding respectively to 7 regressors (which is the number proposed by BIC) and 8 regressors (which is the number proposed by R^2_{adj} and C_p. After testing the two models on the testing set we got that MSE_7=0.2927 and MSE_8=0.2830 . The two values are less than MSE_{full}=0.299 which proves that the reduced model is doing great on the test set.

Fig 17 presents the backward method which gives the same results as the forward method.

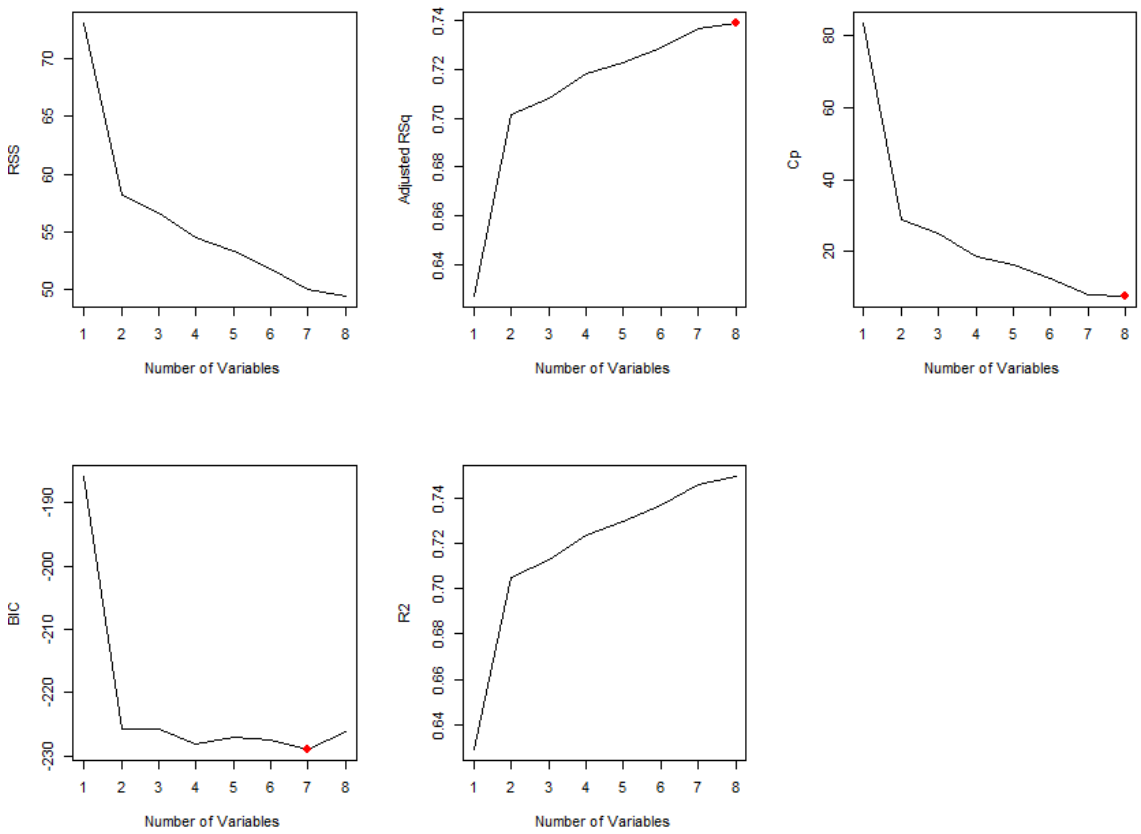

Figure 18: Different criteria for choosing the number of regressors

```
1 subsets of each size up to 8
Selection Algorithm: exhaustive
         age weight height neck chest abdomen hip thigh knee ankle biceps forearm wrist
1  ( 1 ) " " " "    " "    " "  " "   "*"     " " " "  " "  " "   " "    " "     " "
2  ( 1 ) " " "*"    " "    " "  " "   "*"     " " " "  " "  " "   " "    " "     " "
3  ( 1 ) " " "*"    " "    " "  " "   "*"     " " " "  " "  " "   " "    "*"     " "
4  ( 1 ) " " "*"    " "    " "  " "   "*"     " " " "  " "  " "   " "    "*"     "*"
5  ( 1 ) " " "*"    " "    "*"  " "   "*"     " " " "  " "  " "   " "    "*"     "*"
6  ( 1 ) "*" " "    " "    "*"  " "   "*"     "*" "*"  " "  " "   " "    " "     "*"
7  ( 1 ) "*" " "    " "    "*"  " "   "*"     "*" "*"  " "  " "   " "    "*"     "*"
8  ( 1 ) "*" "*"    " "    "*"  " "   "*"     "*" "*"  " "  " "   " "    "*"     "*"
```

Figure 19: The Optimal regressors

3.6.4 Multicolinearity solution: PCR

The principal component regression approach combats multicollinearity by using less than the full set of principal components in the model. To obtain the principal component

```
  (Intercept)           age          neck       abdomen           hip         thigh       forearm         wrist
-1.538700e-15 -1.509750e-01  2.302034e-01 -1.194171e+00  4.516745e-01 -2.694263e-01 -1.310495e-01  2.148820e-01
```

Figure 20: BIC regressors

```
  (Intercept)           age        weight          neck       abdomen           hip         thigh       forearm         wrist
-1.538700e-15 -1.311842e-01  2.533962e-01  1.951677e-01 -1.256956e+00  3.314700e-01 -2.732947e-01 -1.479106e-01  1.793220e-01
```

Figure 21: R_{adj} regressors

```
Selection Algorithm: backward
         age weight height neck chest abdomen hip thigh knee ankle biceps forearm wrist
1  ( 1 ) " " " "    " "    " "  " "   "*"     " " " "  " "  " "   " "    " "     " "
2  ( 1 ) " " " "    " "    " "  " "   "*"     "*" " "  " "  " "   " "    " "     " "
3  ( 1 ) " " " "    " "    "*"  " "   "*"     "*" " "  " "  " "   " "    " "     " "
4  ( 1 ) " " " "    " "    "*"  " "   "*"     "*" "*"  " "  " "   " "    " "     " "
5  ( 1 ) " " " "    " "    "*"  " "   "*"     "*" "*"  " "  " "   " "    " "     "*"
6  ( 1 ) "*" " "    " "    "*"  " "   "*"     "*" "*"  " "  " "   " "    " "     "*"
7  ( 1 ) "*" " "    " "    "*"  " "   "*"     "*" "*"  " "  " "   " "    "*"     "*"
8  ( 1 ) "*" "*"    " "    "*"  " "   "*"     "*" "*"  " "  " "   " "    "*"     "*"
> coef(regfit.backward, 7) # BIC suggests 3 variables
  (Intercept)           age          neck       abdomen           hip         thigh       forearm         wrist
-1.538700e-15 -1.509750e-01  2.302034e-01 -1.194171e+00  4.516745e-01 -2.694263e-01 -1.310495e-01  2.148820e-01
> coef(regfit.backward, 8) # Cp suggests 6 variables
  (Intercept)           age        weight          neck       abdomen           hip         thigh       forearm         wrist
-1.538700e-15 -1.311842e-01  2.533962e-01  1.951677e-01 -1.256956e+00  3.314700e-01 -2.732947e-01 -1.479106e-01  1.793220e-01
>
```

Figure 22: Regressor selection using backward method

estimator, assume that the regressors are arranged in order of decreasing eigenvalues $\lambda_1 \leq \lambda_2 \leq ..... \leq \lambda_p$, PCR eliminates the s regressors that have the lowest eigen values where s is a hyperparameter to be tuned. In our case the least eignenvalue is equal to 4.

```
Data:   X dimension: 198 13
        Y dimension: 198 1
Fit method: svdpc
Number of components considered: 13
TRAINING: % variance explained
   1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps  9 comps  10 comps
X    63.41    73.44    80.05    85.56    89.99    92.49    94.67    96.41    97.88     98.90
Y    35.20    51.23    55.27    57.96    58.13    62.64    66.96    67.52    67.91     67.95
   11 comps  12 comps  13 comps
X     99.42     99.86     100.0
Y     68.72     75.01      75.4
```

Figure 23: Summary PCR

We see from figure 23 and 24 that the number of that we can take is between 7 and 10 using two measures MSEP and R^2

3.6.5 Ridge regression

The ridge regression is a way to deal with multicolinearity , since it shrinks the regression estimates and produces a biased $\hat{B}$ with less variance than the least square estimates. the biasing parameter of the ridge regression is 0.0868 and the mean square error is MSE_{Ridge}=0.2857

So we had an intuitive idea is to to do a ridge regression on the reduced model with 7 regressors and we had a better result on the testing set.

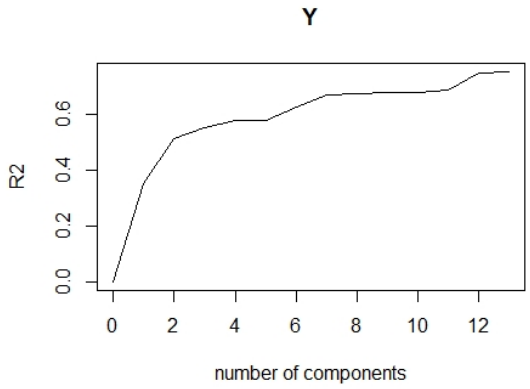

Figure 24: R^2 vs the number of regressors

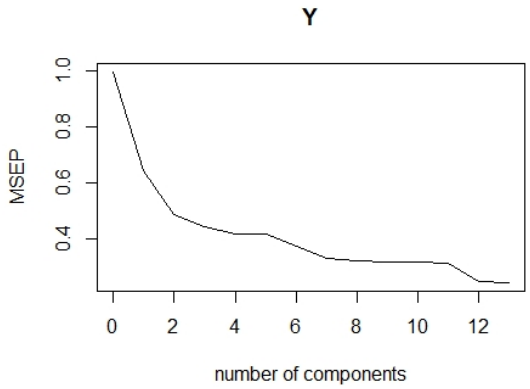

Figure 25: MSEP vs the number of regressors

We can see that the test mean square error has decreased MSE_{red}=0.2693 , which is until now the best result that we had.

4 Results

Overall, we dealt with many aspects of a data science problem. We started by checking the standard assumptions on the normality of our errors which led to the conclusion that it was okay to do such an assumption without any need of a transformation of our target feature. Also, we checked if the data was clean by several metrics and we came to the conclusion that there was no outliers. From this point a focus has been around the model

```
> cv.ridge$lambda.min
[1] 0.08684103
> ridge.pred <- predict(cv.ridge, s = "lambda.min", newx = X2)
> mse.cv.ridge <- mean( (ridge.pred - Ytest) ^ 2)
> mse.cv.ridge
[1] 0.2857578
> coef(cv.ridge)
14 x 1 sparse Matrix of class "dgCMatrix"
                             1
(Intercept) -1.201175e-15
age         -1.894778e-01
weight      -3.643245e-02
height       1.039315e-01
neck         1.018644e-01
chest       -1.897704e-01
abdomen     -4.538838e-01
hip         -2.195155e-02
thigh       -1.247483e-01
knee        -4.661651e-02
ankle       -1.242710e-03
biceps      -1.895077e-02
forearm     -6.409854e-02
wrist        1.538402e-01
```

Figure 26: Coefficient of the ridge regression

```
> # c) Ridge
> cv.ridge <- cv.glmnet(X1[,c(1,4,6,7,8,12,13)], Y , alpha=0)
> cv.ridge$lambda.min
[1] 0.08684103
> ridge.pred <- predict(cv.ridge, s = "lambda.min", newx = X2[,c(1,4,6,7,8,12,13)])
> mse.cv.ridge <- mean( (ridge.pred - Ytest) ^ 2)
> mse.cv.ridge
[1] 0.2693751
> coef(cv.ridge)
8 x 1 sparse Matrix of class "dgCMatrix"
                             1
(Intercept) -1.324340e-15
age         -2.176885e-01
neck         1.117481e-01
abdomen     -6.788204e-01
hip          5.648165e-03
thigh       -1.837222e-01
forearm     -1.013551e-01
wrist        1.851080e-01
```

Figure 27: Ridge regression on the reduced model

itself and how to get the best model out of our data. Clearly multicolinearity is an issue which such a dataset, so we diagnosed the sources of such problem and introduced some solutions with data selection and data reduction with standard methods. We finally got

the best model with ridge regression applied to our reduced dataset with MSE_{red}=0.2693.

5 Conclusion

In this project, we applied different methods for fitting a linear model to our data. Several standard methods have been introduced and a thorough analysis of our dataset has been proposed.
An idea to improve the performance of our model and mitigate the effect of multicolinearity would be to look for additional data.