Machine Learning approaches for selecting the best predictive model using the dataset for estimation of obesity levels based on eating habits and physical condition in individuals from Colombia, Peru, and Mexico

Pronob Barman

May 2021

1 Introduction

The data set used for this project is publicly available through the UCI Machine Learning Repository (https://archive.ics.uci.edu/ml/datasets/Estimation+of+obesity+levels+based+on+eating+habits+and+physical+condition+). This data set entitled "Estimation of obesity levels based on eating habits and physical condition" contains 17 variables and the values for these variables for 2,111 Mexican and South American participants. However, only 15 of the 17 variables are used in the current analysis, as the original output variable was excluded based on project guidelines. Moreover, a new output variable was computed as a function of two of the existing variables: height (in meters) and weight (in kilograms). This output variable is body mass index (BMI), and is computed as weight divided by height squared. Therefore, height and weight were left out of analyses as well. If these variables were left in analyses, the new output variable (BMI) would by mostly explained by these variables, not allowing for the entire variance in BMI to be explained by the other candidate inputs, and biasing all results.

Descriptions of the candidate inputs are as follows: sex; age; family history of an overwight BMI; FAVC - eating high caloric food frequently; FCVC - frequency of eating vegetables; NCP - amount of daily meals; CAEC - eating food between meals; SMOKE - whether or not a person smokes; CH2O - how much water ingested daily; SCC - calorie monitoring; FAF - physical activity; TUE - time spent on technological devices per day; CALC - frequency of alcohol consumption; and MTRANS - what form of transportation they most often use. The dichotomous inputs are: sex, family history of overweight, eating high caloric food, smoking, caloric monitoring, and mode of transportation. These variables are scored as '1' if 'yes' or 'male' or an active form of transportation

was indicated (e.g. walking or biking) and '0' if 'no' or 'female' or a passive form of transportation was indicated (e.g. personal vehicle or public transportation). Ordinal variables were treated as interval variables. For example, alcohol ingestion was changed from frequencies to numeric values indicating frequency (e.g. not drinking was coded to '0', always drinking was coded to '3'). These recoded ordinal variables included eating vegetables, meals per day, eating between meals, amount of daily water intake, level of physical activity, daily time spent on technological devices, and alcohol ingestion. The remaining variables (age and the output BMI variable) are true continuous variables.

1.1 Part A. Partitioning the dataset

At first, we randomly divided the whole dataset into training and validation portions with probabilities $\frac{2}{3}$ and $\frac{1}{3}$ respectively.

1.2 Part B.1 Fit a full linear regression model to the training data

Let us consider
Y =`BMI`
X_1 =`Gender`
X_2 =`Age`
X_3 =`Family history with overweight`
X_4 =`FAVC`
X_5 =`FCVC`
X_6 =`NCP`
X_7 =`CAEC`
X_8 =`SMOKE`
X_9 =`CH2O`
X_{10} =`SCC`
X_{11} =`FAF`
X_{12} =`TUE`
X_{13} =`CALC`
X_{14} =`MTRANS`

Our fitted full model using the training dataset is:
$\hat{Y} = -0.014 - 0.052X_1 + 0.097X_2 + 0.374X_3 + 0.100X_4 + 0.233X_5 + 0.044X_6 - 0.233X_7 - 0.011X_8 + 0.063X_9 - 0.067X_{10} - 0.133X_{11} - 0.038X_{12} + 0.113X_{13} + 0.023X_{14}$

1.3 Part B.2 Use the fitted model to predict outputs in the validation data

We have used R programming to predict the outputs in the validation data using the fitted model.

1.4 Part B.3 Report R^2 for the training data and for the validation data

R^2 for Training data	R^2 for Validation data
0.4422	0.4199

1.5 Part C.1 Use best subsets to obtain a sequence of competing linear regression models. For each competing model, report R^2 for the training data and for the validation data

The best subsets algorithm chose the order in which variables are to be added to the model as follows: $X_3, X_5, X_7, X_{11}, X_{13}, X_2, X_4, X_9, X_{10}, X_{14}, X_{12}, X_6, X_1$, and lastly X_8. For example, the first model, labelled M_1, is given by $Y = \beta_3 X_3 + \varepsilon$; the second model, M_2, is $Y = \beta_3 X_3 + \beta_5 X_5 + \varepsilon$, and so forth.

Model	R^2 for Training data	R^2 for Validation data
M_1	0.2392	0.2228
M_2	0.2967	0.2871
M_3	0.3617	0.3452
M_4	0.3890	0.3593
M_5	0.4062	0.3877
M_6	0.4177	0.3970
M_7	0.4279	0.4108
M_8	0.4322	0.4151
M_9	0.4359	0.4155
M_{10}	0.4384	0.4163
M_{11}	0.4401	0.4164
M_{12}	0.4415	0.4180
M_{13}	0.4420	0.4202
M_{14}	0.4421	0.4199

Table 1: Model R^2 Values

1.6 Part C.2 Rankings of the competing models based on C_p from the training data

Model	C_p Values	Validation data R^2	Ranking based on C_p	Ranking based on R^2
M_1	498.814	0.2228	14	14
M_2	356.251	0.2871	13	13
M_3	195.102	0.3452	12	12
M_4	128.434	0.3593	11	11
M_5	87.232	0.3877	10	10
M_6	60.446	0.3970	9	9
M_7	36.666	0.4108	8	8
M_8	27.880	0.4151	7	7
M_9	20.614	0.4155	6	6
M_{10}	16.487	0.4163	5	5
M_{11}	14.044	0.4164	3	4
M_{12}	12.510	0.4180	1	3
M_{13}	13.294	0.4202	2	1
M_{14}	15.000	0.4199	4	2

Table 2: Model ranking based on C_p and R^2

1.7 Part D. Forward Selection and Backward Elimination

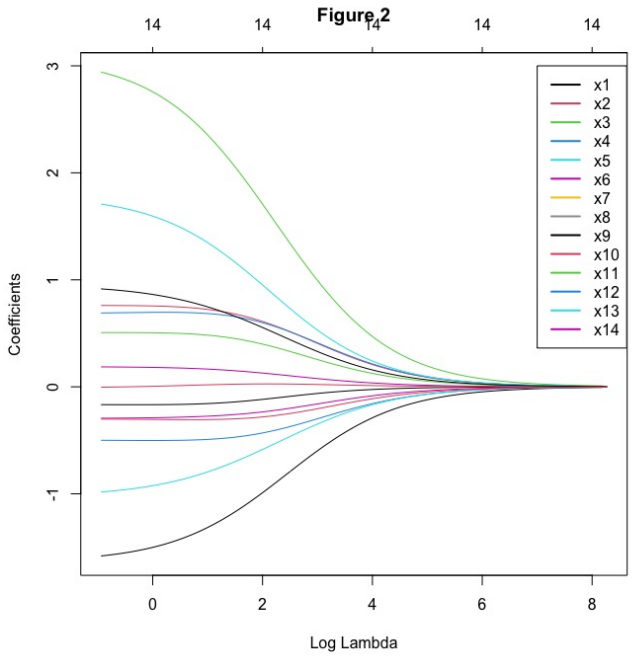
Part D asked to repeat the best subsets analysis using both the 'forward selection' and 'backward elimination' methods. These methods were computed similarly to the best subsets method, with all inputs standardized. Interestingly, both of these methods yielded the exact same models with the same variable entry as the best subsets method did. When this occurs, the results (e.g. R^2 for training and validation, and C_p for training) from this method are identical to the results from best subsets. Therefore, in order to save space and reduce redundancy, Table 2 may also be used to assess the results of obtaining a series of competing regression models for predicting BMI using the forward selection and backward elimination methods.

Using the rankings of C_p for the training data and R^2 for the validation data from Table 2, it can also be gleaned that, generally, validation R^2 increased and training C_p decreased as more variables were entered into the model. However, that trend stops towards the end of the table, where it can be seen that model 12 is ranked best for training C_p, and model 13 is ranked best for validation R^2. Therefore, these rankings of models are mostly in congruence, and the "best" model obtained by forward selection and backward elimination may be model 12 or model 13.

In summary, it seems that these three methods are similarly effective at predicting the validation data and produce the exact same models.

1.8 Part E. Ridge Regression

Following the backward elimination and forward selection procedures, ridge regression was also used. All inputs were standardized prior this method. Ridge regression is particularly useful in instances when there may be multicollinearity among variables, so first, it may be useful to assess the degree of multicollinearity present among inputs. Using the vif() function in R, within the package car(), the variance inflation factors among inputs were relatively low (all values were less than 1.3), indicating that multicollinearity may not be an issue within the inputs for the training data. Regardless, we continued with ridge regression, and the ridge trace is included below.



Five values of λ were used to generate a series of competing linear regression models: 5.281158 (the 72nd value tried), 3.640098 (the 76th value), 2.508978 (the 80th value), 1.308183 (the 87th value), and 0.3903157 (the 100th value). With the exception of the last value, the 100th value tried, the other values were selected based upon the ridge trace. We attempted to space out values across the X-axis of the ridge trace and use values that would generate different estimates. The last value listed (0.3903157) was identified as being the "optimal" lambda value by performing a k-fold cross-validation, and is the value that minimizes the mean squared error. The following table includes the results from these five models with these λ values. The R^2 for the validation data was obtained by predicting the output values in the validation data using the coefficients for each training model based upon the λ value.

Table 3 - Results from Ridge Regression			
λ Value	R^2 Training	C_p Training	R^2 Validation
72nd	0.3790	128.34	0.3969
76th	0.3966	86.19	0.4175
80th	0.4078	59.20	0.4314
87th	0.4176	35.67	0.4445
100th	0.4221	24.99	0.4519

Given these results, the rankings of the five competing models using ridge regression, based on the C_p from the training data, would show a trend that the model improves as the λ value gets smaller. Similarly, ranking the models with the R^2 from the validation data reveals an identical pattern. The R^2 gets larger with the smaller λ values. Comparing these two, it appears that the model with the smallest λ value is the best fit. Interestingly, the R^2 for the validation data at each value of λ is slightly higher than the corresponding R^2 value for the training data. However, if we consider the largest R^2 and the smallest C_p for ridge regression against the results of previous methods, we see a slightly divergent pattern. The lowest C_p for ridge regression is over 1.5x that of the smallest C_p for best subsets. On the other hand, the largest R^2 for the validation data from ridge regression is nearly three percent greater than the largest R^2 for the validation data from best subsets. Therefore, we can highlight that ridge regression might be slightly better than best subsets for predicting the validation data, although it is definitely worse when fit to the training data.

1.9 Part F. Lasso Regression

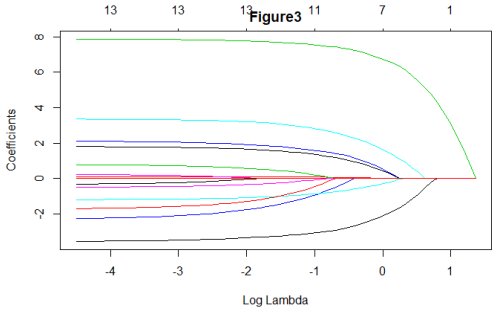


Table 4 - Results from Lasso Regression			
λ Value	R^2 Training	C_p Training	R^2 Validation
10th	0.2255	24.91	0.2117
20th	0.3766	24.92	0.3832
30th	0.4138	24.93	0.4319
40th	0.4210	24.93	0.4454
50th	0.4224	24.94	0.4504

As the λ decreases, C_p of train data increases as the R^2 of validation data

increased. However, C_p is very close to each other for the 5 λ we choose.

1.10 Part G. PCA

$PC1 : \mathbb{E}[Y|R_1 = r_1] = \beta_0 + \beta_1 r_1$
$PC2 : \mathbb{E}[Y|R_1 = r_1, R_2 = r_2] = \beta_0 + \beta_1 r_1 + \beta_2 r_2$
$\vdots$
$PC14 : \mathbb{E}[Y|R_1 = r_1, ..., R_{14} = r_{14}] = \beta_0 + \beta_1 r_1 + ... + \beta_{14} r_{14}$

Model	C_p Values	Training data R^2	Validation data R^2	Rankings based on training data C_p	Rankings based on validation data R^2
$PC1$	1	0.2523	0.2587	1	14
$PC2$	2	0.2533	0.2676	2	13
$PC3$	3	0.3144	0.3390	3	12
$PC4$	4	0.3598	0.4113	4	11
$PC5$	5	0.3603	0.4123	5	10
$PC6$	6	0.3637	0.4200	6	9
$PC7$	7	0.3927	0.4458	7	8
$PC8$	8	0.3931	0.4470	8	7
$PC9$	9	0.4015	0.4491	9	6
$PC10$	10	0.4093	0.4631	10	5
$PC11$	11	0.4097	0.4633	11	4
$PC12$	12	0.4099	0.4651	12	3
$PC13$	13	0.4224	0.4685	13	2
$PC14$	14	0.4226	0.4687	14	1

According to the footnote 72 of lecture notes chapter 3, we use the number of parameters as approximate C_p value. The rank of C_p value of train data is opposite to the rank of R^2 of validation data.

1.11 Part H. Partial Least Squares

The competing models using partial least squares are as follows:
$PLS1 : \mathbb{E}[Y|R_1 = r_1] = \beta_0 + \beta_1 r_1$
$PLS2 : \mathbb{E}[Y|R_1 = r_1, R_2 = r_2] = \beta_0 + \beta_1 r_1 + \beta_2 r_2$
$\vdots$
$PLS14 : \mathbb{E}[Y|R_1 = r_1, ..., R_{14} = r_{14}] = \beta_0 + \beta_1 r_1 + ... + \beta_{14} r_{14}$

The model with just R_1 performs poorly when evaluated with either C_p or R_2 of the validation data. The models with 4 to 14 components are increasingly overfit when considering the C_p statistics and relative lack of increase in R^2. Therefore, the models with 2 or 3 components appear to be the two best, as evident by the rankings in the final two columns above. Partial least

Model	C_p Values	Training data R^2	Validation data R^2	Rankings based on training data C_p	Rankings based on validation data R^2
PLS_1	99.52	0.4005	0.4337	14	14
PLS_2	53.17	0.4208	0.4582	3	1
PLS_3	51.07	0.4225	0.4542	1	2
PLS_4	52.82	0.4226	0.4539	2	3*
PLS_5	54.81	0.4226	0.4539	4	3*
PLS_6	56.81	0.4226	0.4539	5	3*
PLS_7	58.81	0.4226	0.4539	6	3*
PLS_8	60.81	0.4226	0.4539	7	3*
PLS_9	62.81	0.4226	0.4539	8	3*
PLS_{10}	64.81	0.4226	0.4539	9	3*
PLS_{11}	66.81	0.4226	0.4539	10	3*
PLS_{12}	68.81	0.4226	0.4539	11	3*
PLS_{13}	70.81	0.4226	0.4539	12	3*
PLS_{14}	72.81	0.4226	0.4539	13	3*

*11-way tie

squares has chosen components in a way that we are comfortably able to discard $R_4, R_5, ..., R_{14}$.

As a side note, the validation R^2 is slightly larger than the training R^2, and not strictly increasing with the addition of more model terms. This is likely due to having a particularly noisy training set.

1.12 Part I. Other methods and discussion.

A k-nearest neighbor regression was run using the knnregTrain function found in the caret package in R. All 14 variables were standardized and included in the algorithm. Using the same training and validation data as in prior parts, the R^2 for the predicted validation output corresponding to a few selected k-values are below.

Consulting the four nearest neighbors for each observation appears to be optimal, with about 81% of the variation in the predicted BMI explained by these four neighbors. This is quite good. The inclusion of fewer input values was explored, but all yielded lower R^2 values, so we left all explanatory variables in the model.

The validation R^2 for this non-parametric regression approach can be compared to the previously explored parametric models as long as the input variables are the same – that is, the models utilizing the standardized inputs. The k-nearest neighbor algorithm performed much better across the board when compared to the ridge regression, principle component regression, and partial least squares methods.

k	Validation R^2
1	0.7211
2	0.7952
3	0.8028
4	0.8112
5	0.8102
8	0.8027
12	0.7912
25	0.7538
50	0.6641

1.13 Part J. Summary of Methods

In conclusion, the methods used within this project (excluding nearest neighbors) were reasonably similar in predicting BMI for the validation data. These methods reveal two patterns: 1) models were poor when parameters were poorly specified (e.g. using large λ values in ridge or lasso regression) or when only a few variables were entered into the model (e.g. the first two or three models for best subsets, forward selection, and backward elimination), and 2) the validation R^2 tended to yield smaller and smaller increases when combating the pattern for 1 (e.g. validation R^2 only increased slightly when adding the 13th and 14th variables into the model or when specifying two similarly small values of λ). In contrast, the exploratory method used in Part I (nearest neighbors) performed very well regardless of the number of neighbors consulted, and had a much larger validation R^2. In this way, nearest neighbors may be the best method in comparison to the ones specified for use in this project.