We can do it!

Determinants of women participation to the American labor force

Pietro Carlotti, Laura Pittalis, Filippo Salmaso

June 12, 2022

Abstract

This work drives out an inquiry concerning the determinants of women participation to the American labor force in the mid-70s, whose data are collected into the Mroz dataset. Along with a first assessment of the main sources of variability through the implementation of unsupervised techniques (PCA, clustering), the supervised analysis provides a detailed test for the best-accuracy model (both classification and cross validation) and an evaluation via Ridge/Lasso for catching crucial variables. Further research paths for implementing the analysis are eventually underlined.

Contents

Introduction: the Dataset, the Aim, the Techniques

Women's labour market dynamics has always been object of great debate from both a sociological and economic point of view. Particularly, assessing the main factors that determine the position of a woman in terms of wealth and accessibility to working careers can assume quite of a faceted analysis. In this work, we employed a dataset concerning female labour dynamics, supported by several socioeconomic variables of interest. Particularly, we made use of the Mroz dataset (Mroz 1987) which covers the U.S. labour market for the year 1975. It shows a cross-sectional analysis of 753 women in terms of participation to the national labour force (represented by our dependent variable, *lfp*; it is our object of interest in the supervised analysis); the other variables are reported downwards:

Table 1: Mroz Variables

Variable's id	Description
taxableinc	Taxable income for household
federaltax	Federal income taxes
hsiblings	Husband's number of siblings
hfathereduc	Husband's father's education level
hmothereduc	Husband's mothers's education level
siblings	Wife's number of siblings
lfp	Dummy variable = 1 if woman worked in 1975, else 0
kidsl6	Number of children less than 6 years old in household
kids618	Number of children between ages 6 and 18 in household
age	Wife's age
educ	Wife's educational attainment, in years
wage76	Wife's wage reported at 1976 interview, for 1976
hhours	Husband's hours worked in 1975
hage	Husband's age
heduc	Husband's educational attainment, in years
hwage	Husband's wage, in 1975 dollars
faminc	Family income, in 1975 dollars
mtr	Marginal tax rate facing the wife, includes Soc Sec taxes
mothereduc	Wife's mother's education level
fathereduc	Wife's father's education level
unemployment	Unemployment rate in county of residence
largecity	Dummy variable = 1 if live in large city (SMSA), else 0
exper	Actual years of wife's previous labor market experience

Therefore, we are dealing with 22 variables (excluding *lfp*), each one characterized by its own variability and impact on the dependent variable. The work is then structured as follows: in Section 2 we started the preprocessing part of data analysis, in order to have data suitable for the statistical methods we make use of; in Section 3 we employed PCA to assess the main sources of variability in the data; a first glance on potential grouping (and on the determinants of such) can be found in Section 4, centered to clustering; in Section 5 the reader can find several classification techniques applied to predict our main variable, *lfp*. In the Appendix, extra techniques and analyses are left to the reader intended to go deeper into the matter. Ulterior tools (i.e., smoothing and bootstrap) can be found there too.

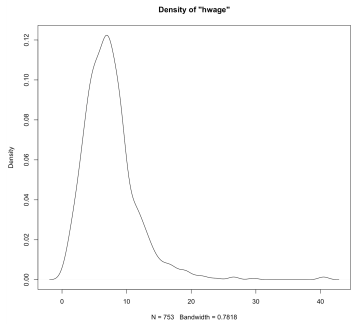

Figure 1: Example of a Skewed Variable (hwage)

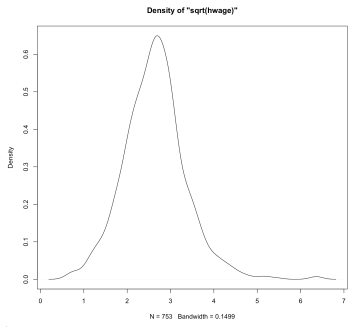

Figure 2: hwage Corrected through Square Root

1 Data Preprocessing

For what concerns the data-cleaning part, this dataset appears to be quite manageable due to the complete absence of missing values among all the variables. The main intervention here concerned the variables' normalization, in order to correct the skewness affecting most of them. See Figure (29) in the Appendix to have an idea of the initial densities. We furthermore scaled data in the 0-1 interval for clustering, and normalized in a more classical way, through mean and standard deviation, for other techniques. The creation of a range 0-1 ensures the fact that distances between centroids in the various linkage methods of clustering are not affected by the different variables' magnitude orders. The mean-standard deviation scaling, on the other side, stresses the accent more on the gaussianity of the densities, in order to grant the robustness of the results. As a last step for these preliminary considerations, we created a correlation plot to understand potential underlying associations between the variables. As the reader can observe from Figure (3), most variables don't display any relevant correlation; actually, *educ*, *mtr*, *faminc*, *hwage* and *heduc* show the greatest correlation.
The next Sections are aimed at visualizing data from different perspectives: we try to capture the main sources of structure and dynamics and we try to group observation to assess the main determinants of variability among these groups. These unsupervised techniques should provide a much greater understanding of the dataset and the relations among its variables, and they can be thought as a first approach to the data before employing supervised techniques too.

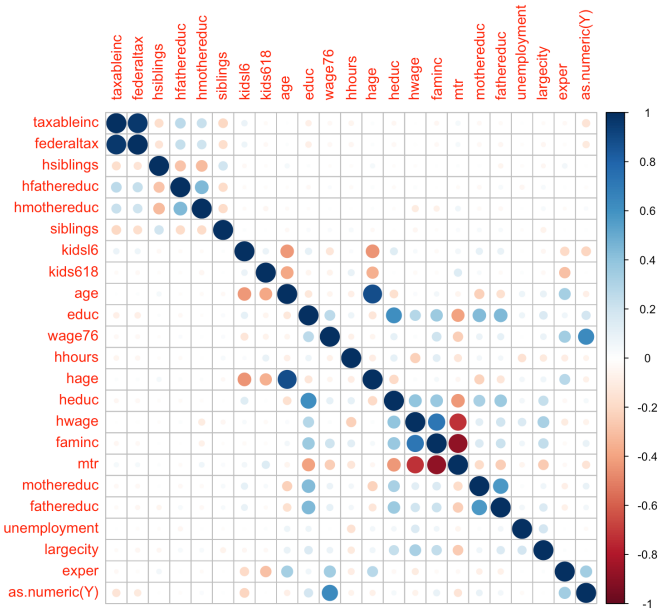

Figure 3: Correlation Plot of the Variables

2 Unsupervised techniques

2.1 Principal component analysis

Proceeding to the exploratory data analysis, PCA provides a useful dimension reduction tool to verify the main sources of variability among observations. This technique, in fact, consists of projecting each data point onto only the first few principal components (i.e., the directions with the greatest variance of the projected data) to obtain lower-dimensional data while preserving as much of the data's variation as possible. Principal components are, mathematically, eigenvectors of the data's covariance matrix. Thus, we computed by eigen decomposition of the data covariance matrix (see Table 4). We can immediately see that the first two components (which show the highest eigenvalues) explain less than 30% of the variance in the data. Figure (4) plots the cumulative variance explained when including more and more components: it seems that the first 11 dimensions (out of 22) are required to reach 80% of PVE, hence dimensions 1 and 2 are not much effective in capturing a good level of variability in the data. In the Appendix the reader can also find in Figure (30) a screeplot with a good representation of the PVE. If we focus our analysis on the first two components, a biplot comes as a useful tool to represent with efficacy the role of the variables in terms of contribution to the distances between the observations in our dataset. As it results from Figure (5), the marginal tax rate, along with the family income and the husband's education and wage, seems to contribute most to the variance explained by the first dimension; on the other side, the second one sees both woman's and husband's age and the number of kids under the age of 6 as the most important contributors. Further visualizations of the correlations between the variables and the principal components are displayed in the Appendix, Figure (31).

We also ran a robust-to-outlier PCA to correct the potential measurement error of the loadings due to outliers. We set the trimming proportion $\alpha = \frac{h}{n}$ (the fraction of points with the highest distance that do not contribute to the objective function) at 0.75. The eigen decomposition and the following screeplot in Figure (6 confirm the reduction to 10 dimensions, with the first two components grabbing around 40% of the PVE.

As a general analysis, it seems that what initially emerged through the correlation plot (Figure 3) is confirmed from a different point of view: the correlation among the variables is quite scarce as much as it is the variables' contribution to the structure of the data. In both cases, we count only a few

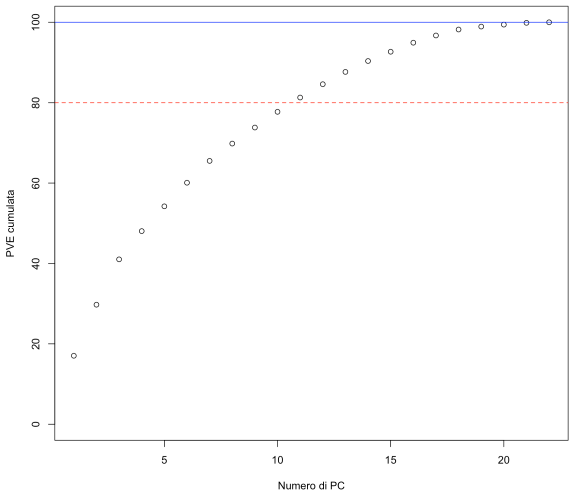

Figure 4: Percentage of Variance Explained

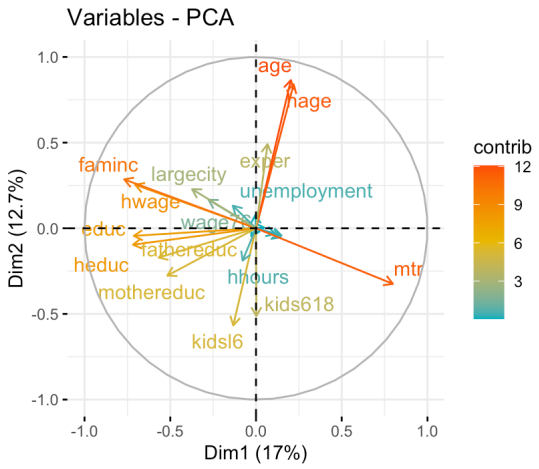

Figure 5: Biplot of the First Two Components

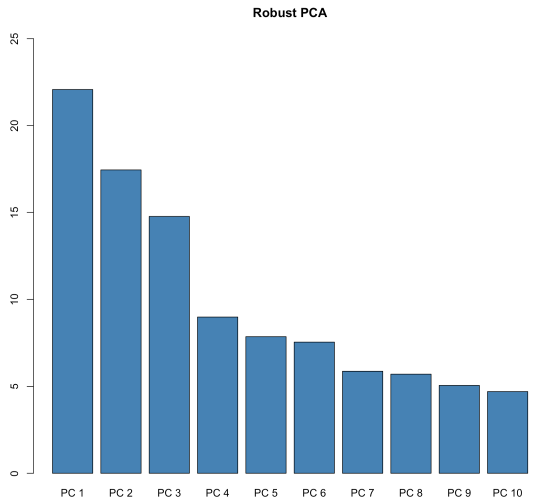

Figure 6: Screeplot of Robust PCA

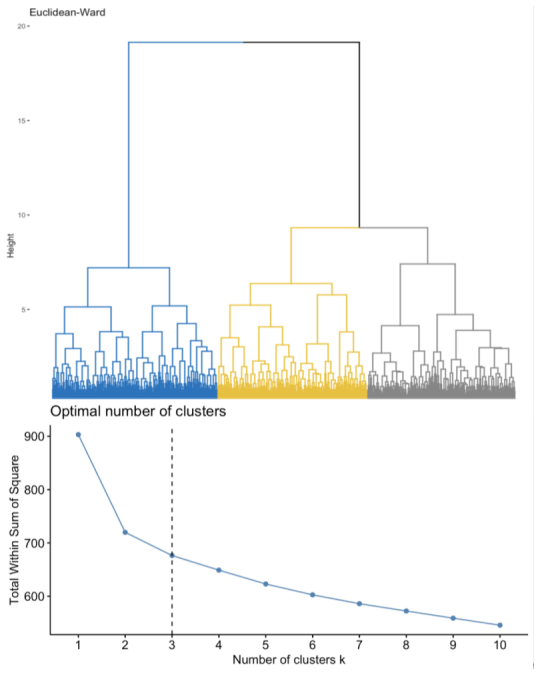

Figure 7: Top: Hierarchical Clustering Dendrogram; Bottom: Optimal Clustering Solution (Elbow Method)

exceptions, which shows both moderate-to-high correlation between each other and which constitute the main contributors to the first two components in the PCA.

2.2 Clustering

It would not be surprising, at this point, to find out that the same variables depicted in the last paragraph are the leading sources of grouping in clustering analysis. This technique allows to form groups of observations that are similar within groups and distinct across groups. Clearly, the dissimilarity has to be related to key features whose variability grants the grouping process. In this setting, we employed the euclidean distance to measure the distance between the observations in the multidimensional (p=22) space. Then we proceeded with a agglomerative hierarchical clustering solution, which forms a nested partition of the data by merging consecutively larger clusters according to a specific linkage function verifying the criterion of minimum distance (i.e., maximum similarity). We employed as a linkage function the ward distance, whose criterion for choosing the pair of clusters to merge at each step is based on the minimum error sum of squares. The dendrogram in Figure (7) depicts the results, which stand in favor of a three-cluster partition (also a two-cluster solution seem to work well). What matters, as we evidenced above, is trying to derive the sources of grouping of the data, i.e., the variables with the greatest variance among the three clusters. See the Appendix, Figure (32), for the complete cluster means results. As expected, *hage, age, kidsl6* are the most varying features (note that are all very relevant for PC2); the fact that *largecity* may also be a great determinant of grouping in this analysis can be criticized by the fact that a dummy variable 0-1 displays very large distances, compared to all the other variables (scaled and ranged to 0-1, but also assuming intermediate values). However, following the results emerged, we may say that the three clusters define women mostly accordingly to the their age, their husbands' age, the number of kids under the age of 6 and the place where they live (large or small city).
We also performed a k-means clustering: the algorithm reconstruct clusters starting from an assigned number of centroids (in this case, 3) and group data accordingly to the closest centroid available (the procedure comes at multiple steps, each time updating the centroids as new points are added). The results are shown in Figure (8), where the three clusters are plotted in the first two principal components.

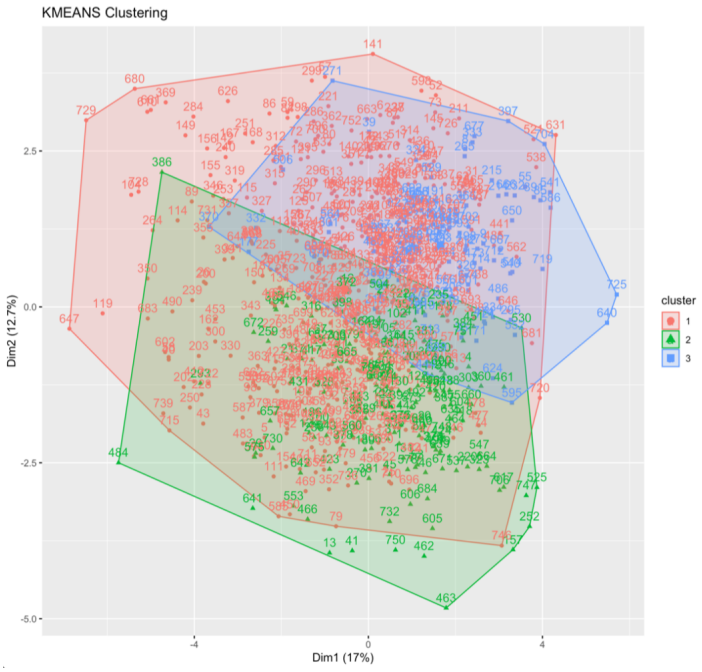

Figure 8: K-means Clustering Plotted in the First Two Components

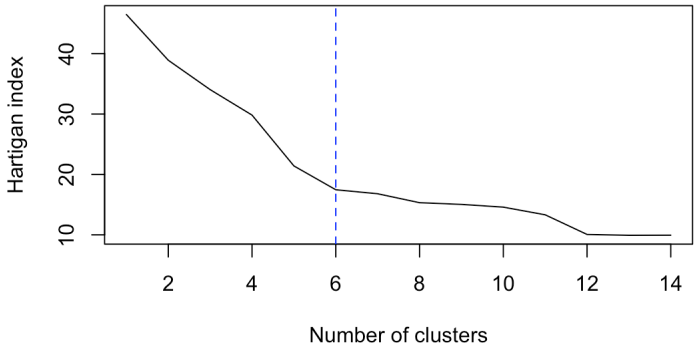

Figure 9: Hartigan Index for k-means Clustering

Generally, k-means' clusters shows a considerable degree of overlapping, which is confirmed by an average silhouette width of 0.16 (see Figure 33 and 34 in the Appendix for a more complete visualization). As a final test for k-means, you can find the Hartigan Index (Figure 9), which measures the relative change of fitness as the number of clusters changes. Results report a preference for a six-cluster solution. However, considering a bunch of 24 indexes from which we can derive the optimal number of clusters, 8 give preference for two clusters, 8 for three clusters, and then the other indexes give quite diversified solutions. As a conclusion, we can assert with a certain degree of confidence that partitioning the data into three or two clusters does its job properly.

3 Supervised techniques

3.1 Classification

For what we have seen in the previous Sections, the unsupervised analyses stressed the role of certain features in contributing to the structure of the data. We now present as a first supervised approach several classification techniques applied to predict our categorical variable *lfp*. First of all, we estimated a logistic regression in order to find a benchmark for all the other classifiers. The difference between this regression with a stepwise logistic one is not terrific, however we still can see an improvement in terms of ability to fit the data (the reader can find the estimates of both models in the Appendix, Figure 35). Such stepwise logit algorithm is useful to select the most effective and relevant features when predicting the response. The aim of this process is to find the most efficient model with the best value for a specific information criterion. In fact, the AIC scored 372.32 for the complete logit, with respect to a slightly better 357.79 performed by the stepwise logit. Also the accuracy (the ratio of correctly classified data compared to the total observations) did not improve considerably with the stepwise model, according to the data reported in Table (2).

Model	AIC	Accuracy
Logic	372.32	0.5212766
Stepwise Logit	357.79	0.5265957

Table 2: AIC and Accuracy results for Logit and Stepwise Logit

We also performed LDA analysis, whose algorithm relies on the Bayes classifier rule, which works greatly with normalized data. Figure (10) shows a quite good classification of the response variable according to the LDA algorithm. However, note that some variables in this context may appear as "easy" predictor, as long as they are related to obvious fiscal dynamics. The reader may find as an object of interest the fact that *taxableinc* is very good at classifying *lfp*, because if the latter has 0 as a value, the taxable income is more probably to be closer to zero (it is actually zero if the husband doesn't work either); see Figure (36). We propose a series of confusion matrices aimed at classifying the response variable employing several techniques. As the reader can see from Figure (11), we can say that the levels of accuracy are reasonable for every technique employed.

3.2 Cross-validation

The results presented in the previous subsection were the outcome of a first attempt at crafting an algorithm to predict our variable of interest, which is *lfp*, through a suitable exploitation of all the others features included in the dataset. The confusion matrices for each classifier where computed following a standard Validation Set Approach. The data was split in two subsets: a training set and a test set. The first one was used to appropriately tune the parameters incorporated in the classifiers as to maximize in-sample fitting of the data, whereas the second subset was employed to assess the out-of-sample performance of each classifier. To make sure that our classification were effective, we took two significant precautions. First of all, it was necessary to ensure that the distribution of the data in the two subsets mimicked the original distribution in the whole dataset, at least for the dependent variable. In order to do so, we performed a stratified sampling to so that lfp was equally distributed in the training set and the test set. Furthermore, since estimating the parameters that make up a

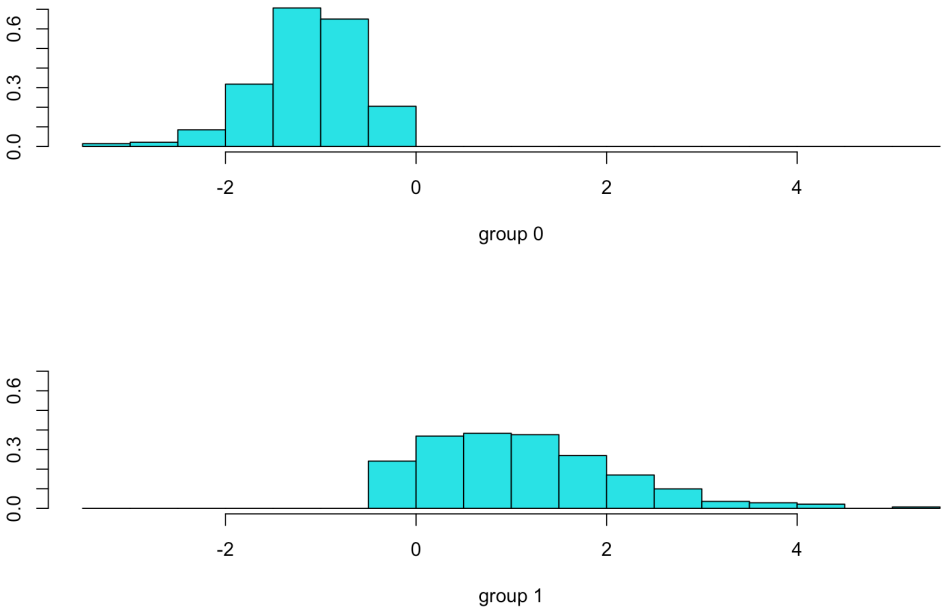

Figure 10: Classification of the Response Variable by the Linear Discriminant

```
> Logit_confusion_matrix$table
          Reference
Prediction  0  1
         0 77 13
         1  4 94
> Logit_stepwise_confusion_matrix$table
          Reference
Prediction  0  1
         0 70 19
         1 11 88
> LDA_confusion_matrix$table
          Reference
Prediction  0  1
         0 78 17
         1  3 90
> QDA_confusion_matrix$table
          Reference
Prediction  0  1
         0 69 14
         1 12 93
```

```
> KNN_confusion_matrix$table
          Reference
Prediction  0  1
         0 69 22
         1 12 85
> PC_logit_confusion_matrix
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0  76   5
         1   5 102
```

```
> SVM_confusion_matrix$table
          Reference
Prediction  0  1
         0 71 21
         1 10 86
> Naive_Bayes_confusion_matrix$table
          Reference
Prediction  0  1
         0 72 21
         1  9 86
```

Figure 11: Confusion Matrices - Different Methods Applied

classifier often requires a large sample size, we fixed the dimension of the training set to be larger than that of the test set. To be more precise, the training set exploited 75% of the original sample size, whereas the test set only used 25%.

This procedure, albeit effective, was still quite coarse. We reckoned that the best way to improve the performance of our classification was to employ a cross-validation. The underlying philosophy of this technique is similar to that of the Validation Set Approach: we split the data in two parts, one to estimate the model and one to test its out-of-sample performance. The difference is that in order to cross-validate a classification model the training set need to be broken down further. First of all, we chose to leave out only 50 observations (approximately 7% of the original sample size) to make up the test set. To evaluate whether a classifier has a good performance on new observations that were not used to estimate its parameters one does not need that many observations. The training set, which consisted of the remaining 703 observations, was split into 10 more subsets, which are usually known in jargon as "folds". The classifier is then estimated only on a subset of such folds through an iterative procedure: for each step of the cross-validation one fold is taken out of the training set and the classifier is estimated on the remaining 9 folds; the resulting model is then tested on the left out fold to evaluate its classification performance.

There are many different evaluation metrics that one could use to assess the performance of a classifier. Among the many available options we decided to employ the accuracy, which is defined according to the following formula:

$$ACCURACY = \frac{TP+TN}{TP+TN+FP+FN}$$

where TP and TN stand for True Positives and True Negatives, while FP and FN indicate False Positives and False Negatives. This metric can be roughly interpreted as the mean prediction error, since it is computed as the ratio of correct predictions over the total number of predictions. There are other evaluation metrics, such as Precision and Recall, which assign a higher penalization depending on whether the prediction error is a FP or a FN, which is sometimes a desirable property. However, in our case there is no theoretical reason which indicates that these two cases should be treated differently, therefore for our purposes it is reasonable to use Accuracy.

One of the greatest advantages of performing a cross-validation of a classification algorithm is that it does not generate just one set of parameters, as in the Validation Set Approach, but a whole family of models with their evaluation metrics. The larger is the number of folds, the larger is the number of estimated models. The best model that comes out of the cross-validation procedure (that is the one that best fits the left-out fold) is then selected as the output model and it is evaluated on the test set to asses the out-of-sample performance of the classifier. Therefore, the final output of a cross-validation on one classifier is the the value of the evaluation metric computed on the test set. One could simply assess the performance of different classifiers by comparing the individual value obtained by cross-validating each one of them; however, in order to have a deeper understanding of how a classifier performs on new observations it would be useful to have a whole distribution of out-of-sample evaluation metrics. For this reason, we decided to repeat the whole cross-validation procedure 20 times. In principle, the higher is the number of iterations of the cross-validation, the better is our understanding of the out-of-sample performance of the classifier, but we needed to find a trade-off with computation time.

The purpose of our supervised analysis was to find the classifier that best predicts whether a woman is part of the labor force depending on all the features included in the dataset. In order to do so, we experimented numerous classifiers. This is a complete list of the techniques we employed:

- Logit regression
- Logit regression with a stepwise feature selection using the AIC
- Linear Discriminant Analysis
- Quadratic Discriminant Analysis
- K Nearest Neighbors classifier
- Principal component logit

- Support Vector Machine classifier
- Naive Bayes classifier
- Decision tree
- Random forest

Some of the classifiers we used were not covered during the course and therefore we invite the reader to consult the bibliography for further information on how these classification algorithms work. In particular, we recommend reading Durgesh and Lekha (2010) to understand the functioning of SVM classifiers; Swain and Hauska (1997) and Liaw and Wiener (2002) could be helpful for understanding how Decision trees and the Random forest work.

The other advantage of performing a cross-validation is that it allows to tune some of the parameters that need to plugged into the classifier. Some algorithms require the researcher to set some parameters at will. How should the researcher make their choice? One simple way to de-randomize this decision is to cross-validate the tuning parameter. In addition to the usual k-fold cross-validation, the researcher performs the full cross-validation procedure for different values of the tuning parameter and then selects the value with the best out-of-sample performance. This methodology can become even more complex when there are multiple parameters that need to be cross-validated. For our purposes, we decided to tune only one parameter per classifier (obviously only for those classifier that include tuning parameters). To be more specific, this is a list of the parameters we tuned for each classifier:

- KNN: number of K nearest neighbors
- PC logit: number of principal components
- SVM: kernel function applied to the data
- Decision tree: maximal depth of the decision tree
- Random forest: number of variables to choose from at each node of the decision tree

The final output produced by the cross-validation procedure can be summed up in the two following two figures. Figure 12 shows for each classifier the distribution of all accuracies computed during the cross-validation procedure. It is quite evident that all classifiers share a similar performance in terms of training-set accuracies. Their median accuracies are almost aligned at a very high level (above 80%); their maximal accuracy can go well over 90%; their distributions are not that dispersed. The only visible exception to this last assertion is the PC logit, which has a very long left tail; however, this circumstance is very easy to grasp once we remember that the cross-validation on such classifiers also tunes the number of principal components to include in the regression equation, starting from just 1 component. It is very likely that as the number principal components increases, at least to some point, the accuracy rises from a mediocre 50% up to 80%/90%: hence the long left tail.

Although these classifiers all display very good accuracies on the training set, we know that what ultimately matters is their performance on the test set. Figure 13 shows the distribution of all 20 test-set accuracies for each classifier. Here the picture is more diversified. There are some salient aspects that emerge from this plot:

- All classifiers still have very high accuracies on the test set (they range approximately from 85% to 95%); however, some appear to perform better than others.
- The classifier with the highest accuracy is the KNN classifier, however, its distribution displays a high degree of variability.
- The classifier with the lowest accuracy is the QDA, which probably overfits the data, since its test-set accuracy are considerably lower than the LDA.
- A stepwise feature selection on the Logit classifier using the Aikaike criterion looks like a good solution to improve the test-set accuracy of the Logit. As a matter of fact, the such classifier is the one with the highest median accuracy.
- Similarly, the Random forest seems to improve to improve the Decision tree's performance, since its distribution is slightly shifted to the right.

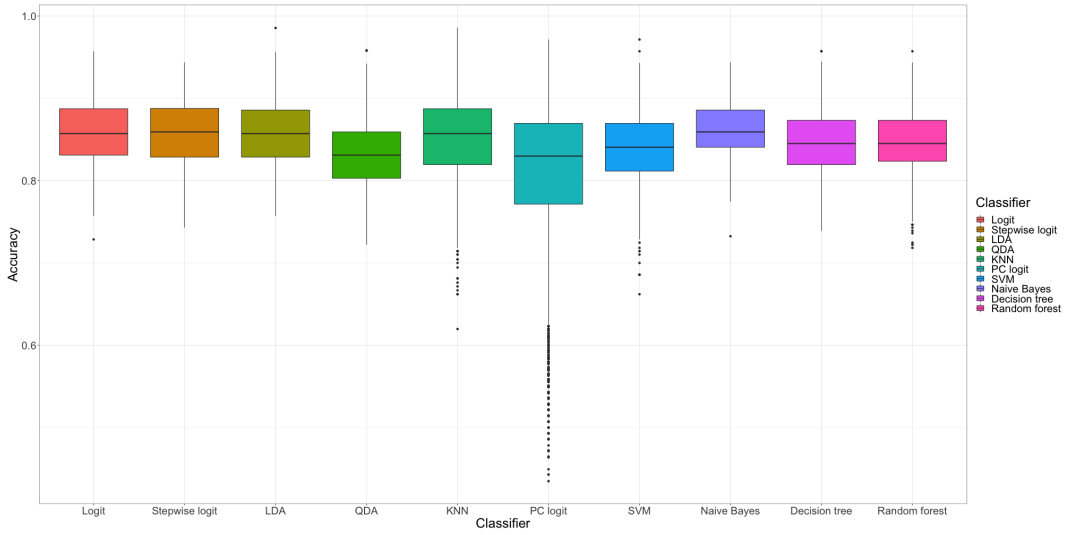

Figure 12: Cross-validation accuracies on the training set.

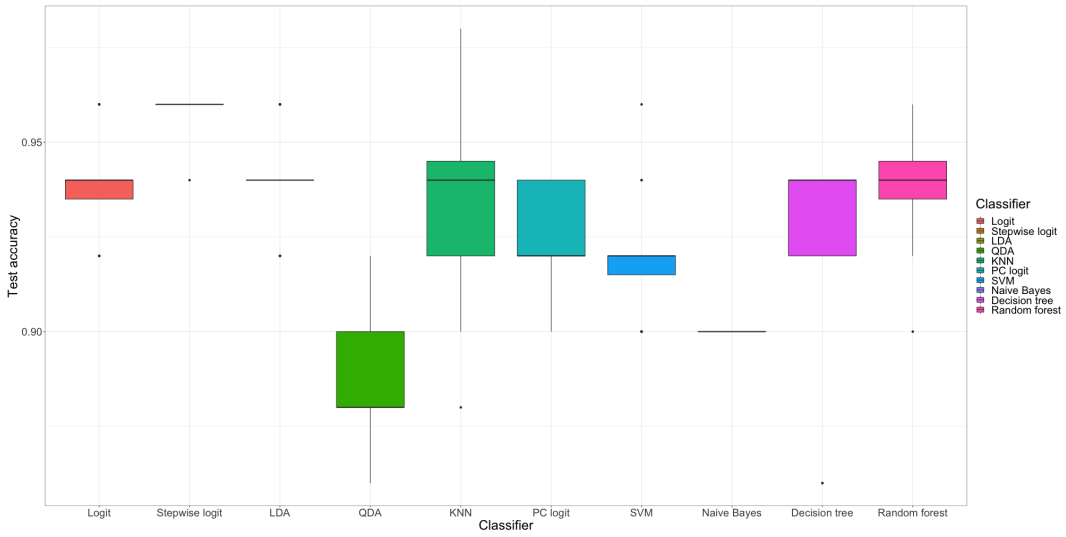

Figure 13: Cross-validation accuracies on the test set.

4 Ridge and Lasso

The classification problem we focus on aims to predict whether women do find jobs based on a set of 22 regressors. Given the large set of predictors, it is important to assess whether there is any issue of multicollinearity among them. We refer to as multicollinearity the presence of linear dependencies among predictors. The higher the number of predictors, the higher the probability that some of them contain the same information and thus show linear dependencies and correlation. The fact that in a linear model some predictors shows linear dependencies may be particularly problematic since it introduces variance in the effect estimates, resulting in predicted values being far away from the actual values. Some of the most common ways to diagnose multicollinearity is to look at the correlation matrix or to plot pairwise scatter plots that help to visualize how variables are distributed. As we have already mentioned, the correlation matrix (figure 3) does not show strong and frequent correlations among our predictors. However, in order to get a better understanding of the mutual dependencies between them we compute the Variance Inflation Factor. This is defined as follows:

$$VIF_j = \frac{1}{1 - R^2_{X_j|otherXs}} \tag{1}$$

We compute 22 different regressions by running each time one of the predictors on all the others. In this way, we are able to compute a measure of the variance of each regressor explained by the others. In table 3 you can find the VIF values we obtained. As it clearly emerges, three are the variables that show an high VIF: the taxable income, the federal tax and the marginal tax rate.

Table 3: VIF

taxableinc	18.484
federaltax	17.806
hsiblings	1.278
hfathereduc	1.266
hmothereduc	1.341
siblings	1.132
kidsl6	1.471
kids618	1.522
age	5.486
educ	2.025
wage76	1.125
hhours	1.743
hage	5.082
heduc	1.995
hwage	3.875
faminc	5.507
mtr	6.687
mothereduc	1.664
fathereduc	1.585
unemployment	1.107
largecity	1.198
exper	1.300

Since these are all predictors related to tax measures, they are likely to bring in the same information to our classification problem and to be linear interdependent. Thus, we firstly try to assess this problem by dropping out two of those three variables from the model. The coefficients estimates are reported in the appendix. Figure 14 shows the boxplot for the accuracies computed by crossvalidating the training set in 10 folds and in Figure 15 is reported the boxplot for the accuracies obtained in the test set by reiterating the crossvalidation 20 times. By comparing these boxplots with the ones reported above, we are able to conclude that the new model specification succeed in improving the Logit performance. Thus, removing the variables that show linear dependencies represents a first sound solution for the multicollinearity problem in our model and help us improving the performance of our benchmark classifier.

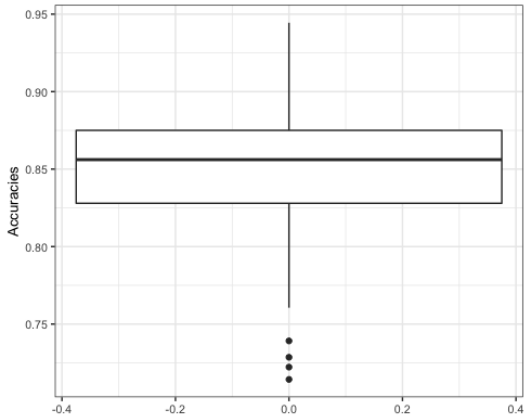

Figure 14: CV accuracies

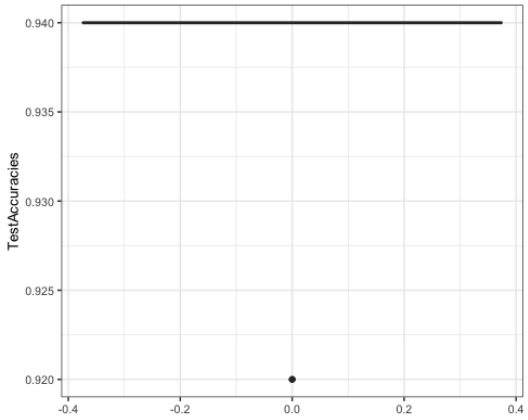

Figure 15: Test Accuracies

Secondly, we have implemented a Ridge model in order to stabilize our fit. Ridge regression is a model tuning method that is used to analyse data that suffers from multicollinearity. The first step is to choose the best lambda: our penalization metric. The higher the values of lambda, the bigger is the penalty and therefore the magnitude of coefficients is reduced. The penalization path plotted in figure 16 shows how the coefficients of the model change when the log-lambda increases and so the penalty. In order to find the best lambda, we have used a cross validation and we have tried to fit our Ridge model for 100 different values of lambda. We compute two different goodness-of-fit metrics - the binomial deviance and the misclassification error - in order to be able to compare the results. In figure 17, you can find the plots for these metrics when the log-lambda changes.

We highlight two values of lambda, the minimum lambda (the lambda that minimizes the metrics) and the maximum lambda within one standard error from the minimum value. This helps us understanding how much our fit is influenced by the value of the lambda. On one hand, by using the binomial deviance we are push to think that the lambda is highly influential since for an increase in 1 se we obtain a very small change in the lambda value. On the other hand, the misclassification error shows that an increase in 1 se leads to a large change in the value of lambda and thus our estimates seem to be less influenced by the value of the penalization metric. This different conclusion points out that depending on which metrics you decide to use, the choice of the right lambda may seem more or less influential. However, the value of the minimum lambda is very similar between the two metrics (around 0.031) and we use it to compute our Ridge model. In the appendix, the new coefficients are reported. They are all close to 0 apart from the value of wage76, the variable of the wage of the preceding year.

We have run a cross-validation to compute the accuracies of the model and we obtain the two boxplots reported in figure 20. The first boxplot shows the accuracy computed in the 10 folds of the original cross validation on the training set. The second boxplot shows the value of the accuracy on the test set for a cross validation reiterated 20 times. The distribution of the accuracies do not show any real improvement with respect to the other classifiers used above.

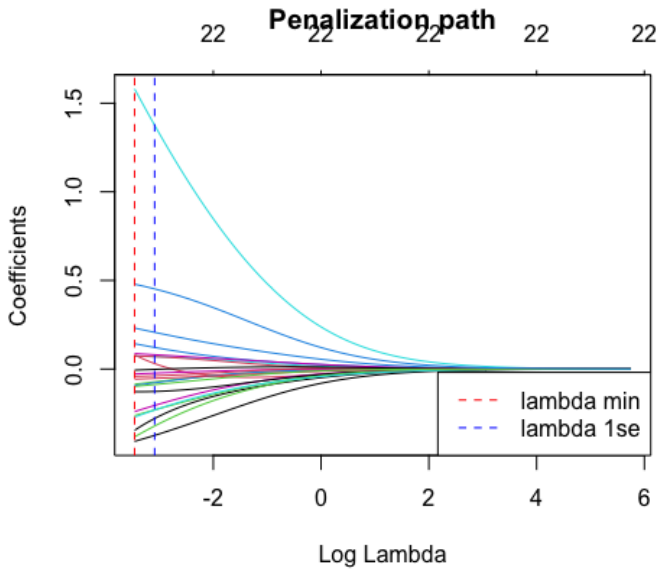

Figure 16: Penalization path Ridge

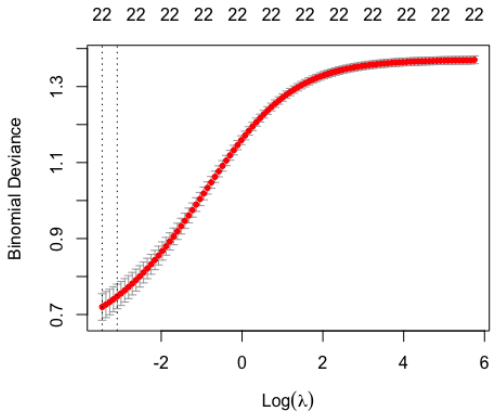

Figure 17: Binomial Deviance path

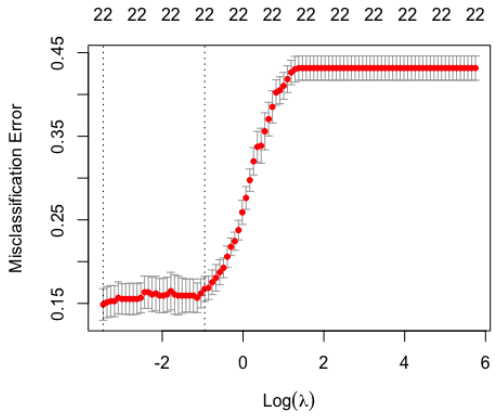

Figure 18: Misclassification Error path

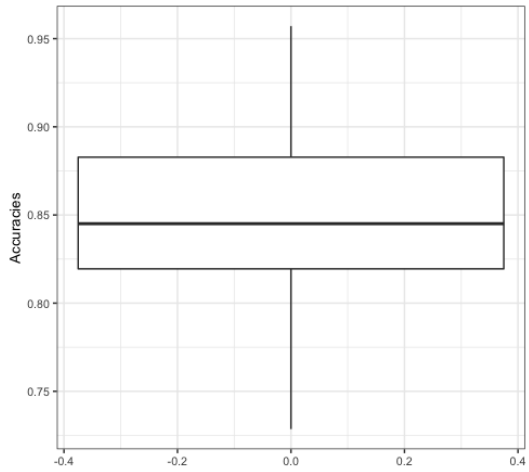

Figure 19: CV accuracies

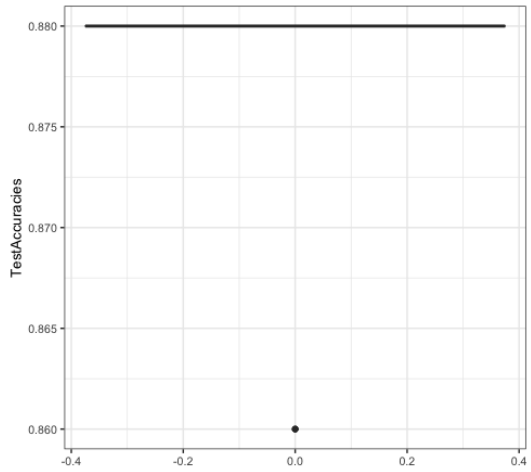

Figure 20: Test Accuracies

As a third solution, we have implemented a Lasso model in order to eliminate some predictors and make our model more parsimonious. Lasso is both a regularization method and a model selection technique and it can help us tuning our classification problem. Here too, we have cross-validated in order to obtain the best lambda. In figure 21, you can find the penalization path. In this case, as log-lambda increases the coefficients are shrinked to 0. As you can see in figure 23, we obtain similar results for the two goodness-of-fit metrics (around 0.0018 for the binomial deviance and 0.0027 for the missclassification error).

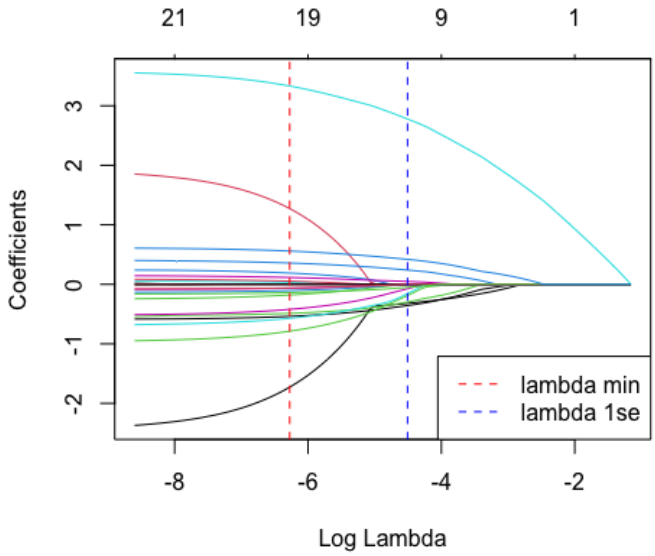

Figure 21: Penalization path Lasso

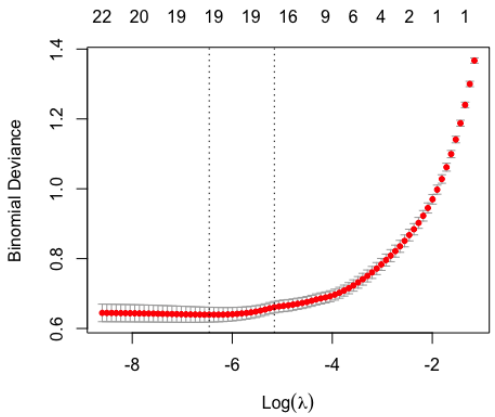

Figure 22: Binomial Deviance path

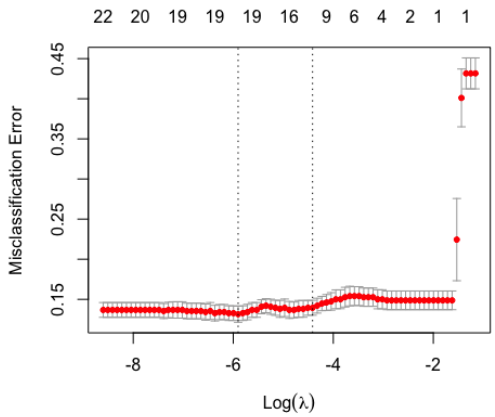

Figure 23: Misclassification Error path

The coefficients of our Lasso model are reported in the appendix. The model goes from 22 regressors to 20: the predictors of the husband age and of the father education are shrinked to 0. We run a performance evaluation of this model by cross validating and computing the accuracies. In figure 25, you can find the boxplot for the accuracies computed on the 10 folds of the training set and the boxplot of the metrics obtained on the test set by reiterating the cross validation 20 times. In this case, the performance results are comparable with the boxplots of the best classifiers shown above.

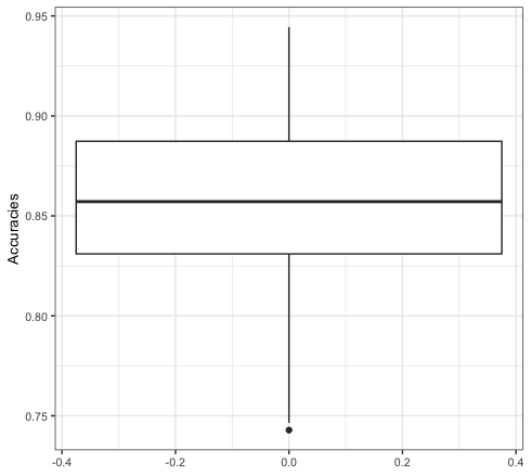

Figure 24: CV accuracies

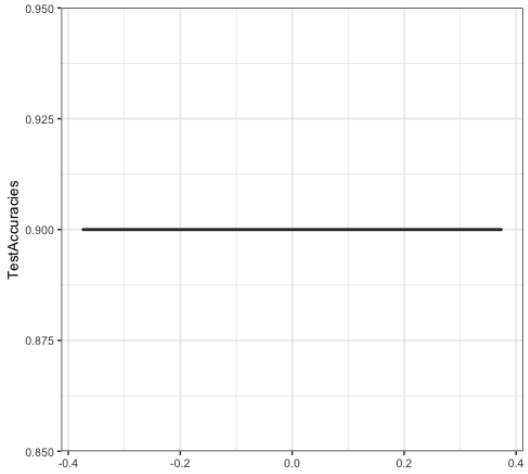

Figure 25: Test Accuracies

Finally, in order to further explore the best model specification for our classification problem and compare it with the previous results, we have implemented a Best Subset selection with the Logit model. The figures 26 27 28 show the difference results we obtained implementing 3 measures of model performance: binomial deviance, AIC and BIC. The binomial deviance suggests that the preferable model is the one with all the predictors. However, this might be due to the specific construction of this metric which tends to prefer complete model. The other metrics perform better with models with half of the original predictors (around 10). This is consistent with the plot of the Lasso shown in figure 23: both binomial deviance and the missclassification error show low values around 10 regressors.

As further steps, it might be useful to test a model with just half of the predictors as suggested by the Best Subset Selection. In parallel, it may be interesting to add interations between variables and squared values to the original model and try to perform again a Lasso model. Another suggestion may be to test new classifiers or to perform a supervised dimension reduction.

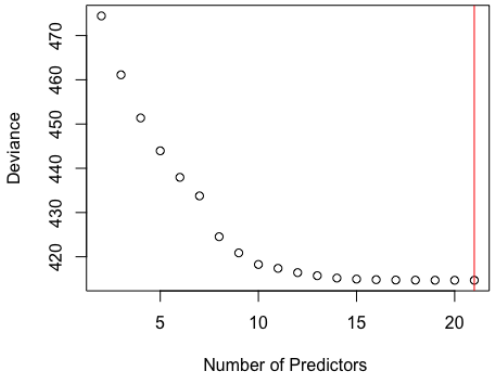

Figure 26: Binomial deviance Best Subset selection

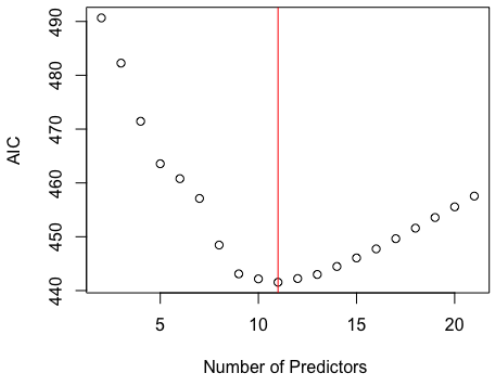

Figure 27: AIC Best Subset selection

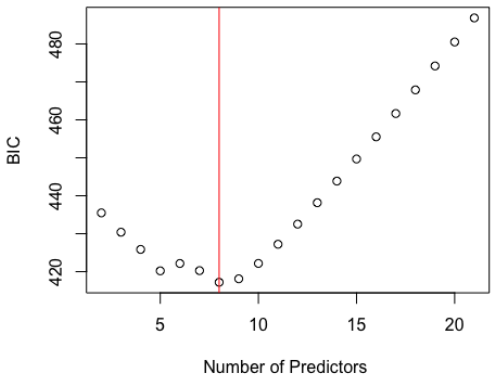

Figure 28: BIC Best Subset selection

5 Appendix

Table 4: Eigen Decomposition and PVE

dimension	eigenvalue	variance.percent	cumulative.variance.percent
Dim.1	3.74804735	17.0365789	17.03658
Dim.2	2.79423939	12.7010881	29.73767
Dim.3	2.48197112	11.2816869	41.01935
Dim.4	1.54603477	7.0274308	48.04678
Dim.5	1.35651205	6.1659639	54.21275
Dim.6	1.29165945	5.8711793	60.08393
Dim.7	1.19497374	5.4316988	65.51563
Dim.8	0.94821722	4.3100783	69.82570
Dim.9	0.87775412	3.9897915	73.81550
Dim.10	0.86056709	3.9116686	77.72716
Dim.11	0.78155209	3.5525095	81.27967
Dim.12	0.72890453	3.3132024	84.59288
Dim.13	0.67333647	3.0606203	87.65350
Dim.14	0.59616976	2.7098625	90.36336
Dim.15	0.50837165	2.3107802	92.67414
Dim.16	0.49398866	2.2454030	94.91954
Dim.17	0.39524152	1.7965524	96.71610
Dim.18	0.32647181	1.4839628	98.20006
Dim.19	0.16100986	0.7318630	98.93192
Dim.20	0.10618837	0.4826744	99.41460
Dim.21	0.09805915	0.4457234	99.86032
Dim.22	0.03072985	0.1396811	100.00000

```
   taxableinc      federaltax        hsiblings       hfathereduc      hmothereduc        siblings          kids16          kids618           age       
 Min.   : 1500   Min.   :    0   Min.   : 0.00   Min.   : 0.000   Min.   : 0.000   Min.   : 0.000   Min.   :0.0000   Min.   : 0.000   Min.   :30.00  
 1st Qu.:13900   1st Qu.: 1428   1st Qu.: 2.00   1st Qu.: 7.000   1st Qu.: 7.000   1st Qu.: 1.000   1st Qu.:0.0000   1st Qu.: 0.000   1st Qu.:36.00  
 Median :18822   Median : 2426   Median : 3.00   Median : 7.000   Median : 9.000   Median : 3.000   Median :0.0000   Median : 1.000   Median :43.00  
 Mean   :21152   Mean   : 3276   Mean   : 3.45   Mean   : 8.525   Mean   : 9.242   Mean   : 3.254   Mean   :0.2377   Mean   : 1.353   Mean   :42.54  
 3rd Qu.:25780   3rd Qu.: 3947   3rd Qu.: 5.00   3rd Qu.:10.000   3rd Qu.:12.000   3rd Qu.: 5.000   3rd Qu.:0.0000   3rd Qu.: 2.000   3rd Qu.:49.00  
 Max.   :96000   Max.   :31386   Max.   :8.00   Max.   :17.000   Max.   :17.000   Max.   :8.000   Max.   :3.0000   Max.   :8.000   Max.   :60.00  
      educ           wage76          hhours           hage            heduc            hwage            faminc            mtr          mothereduc    
 Min.   : 5.00   Min.   :0.00   Min.   : 175   Min.   :30.00   Min.   : 3.00   Min.   : 0.4121   Min.   :  1500   Min.   :0.4415   Min.   : 0.000  
 1st Qu.:12.00   1st Qu.:0.00   1st Qu.:1928   1st Qu.:38.00   1st Qu.:11.00   1st Qu.: 4.7883   1st Qu.:15428   1st Qu.:0.6215   1st Qu.: 7.000  
 Median :12.00   Median :0.00   Median :2164   Median :46.00   Median :12.00   Median : 6.9758   Median :20880   Median :0.6915   Median :10.000  
 Mean   :12.29   Mean   :1.85   Mean   :2267   Mean   :45.12   Mean   :12.49   Mean   : 7.4822   Mean   :23081   Mean   :0.6789   Mean   : 9.251  
 3rd Qu.:13.00   3rd Qu.:3.58   3rd Qu.:2553   3rd Qu.:52.00   3rd Qu.:15.00   3rd Qu.: 9.1667   3rd Qu.:28200   3rd Qu.:0.7215   3rd Qu.:12.000  
 Max.   :17.00   Max.   :9.98   Max.   :5010   Max.   :60.00   Max.   :17.00   Max.   :40.5090   Max.   :96000   Max.   :0.9415   Max.   :17.000  
   fathereduc      unemployment      largecity          exper       
 Min.   : 0.000   Min.   : 3.000   Min.   :0.00000   Min.   : 0.00  
 1st Qu.: 7.000   1st Qu.: 7.500   1st Qu.:0.00000   1st Qu.: 4.00  
 Median : 7.000   Median : 7.500   Median :1.00000   Median : 9.00  
 Mean   : 8.809   Mean   : 8.624   Mean   :0.64280   Mean   :10.63  
 3rd Qu.:12.000   3rd Qu.:11.000   3rd Qu.:1.00000   3rd Qu.:15.00  
 Max.   :17.000   Max.   :14.000   Max.   :1.00000   Max.   :45.00  
```

Figure 29: Descriptive Statistics of the Variables

Figure 30: Screeplot of the PC

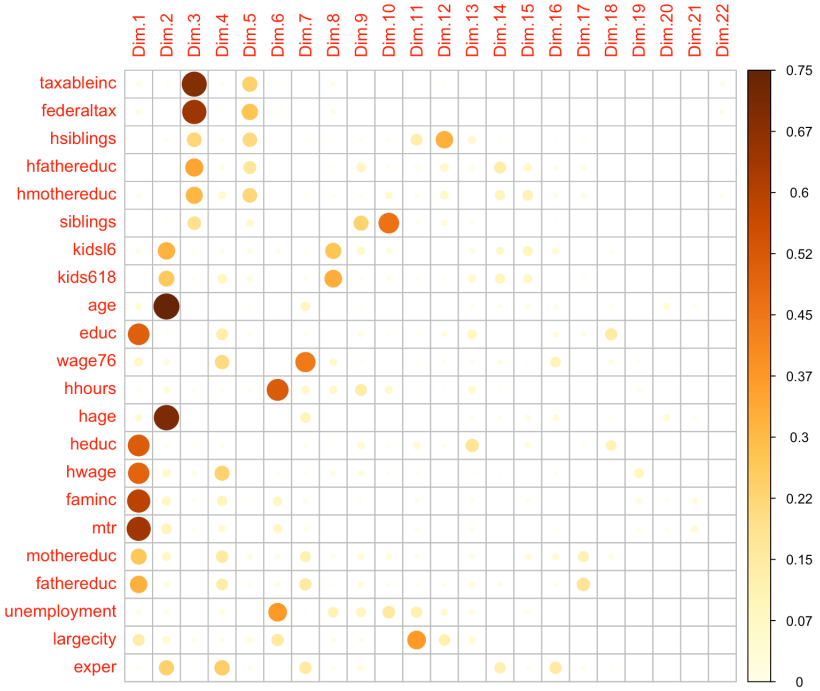

Figure 31: Correlation Plot between the Variables and the Principal Components

```
Cluster means:
  taxableinc federaltax hsiblings hfathereduc hmothereduc  siblings      kidsl6   kids618       age      educ
1  0.2021151 0.09796666 0.4511494   0.4969574   0.5418075 0.3668582 0.006385696 0.1048851 0.6333333 0.5973819
2  0.2114937 0.11109303 0.4484201   0.4915810   0.5342226 0.4326208 0.089219331 0.1765799 0.3831475 0.5666047
3  0.2105266 0.10376481 0.3873318   0.5185967   0.5571089 0.4220852 0.152466368 0.2354260 0.2077728 0.6677877
     wage76    hhours      hage     heduc     hwage    faminc       mtr mothereduc fathereduc unemployment
1 0.1870484 0.4086399 0.7120051 0.6647510 0.2040847 0.2588481 0.4374444  0.4938021  0.4825332    0.5628701
2 0.1569384 0.4513257 0.4796778 0.6104620 0.1305800 0.1859812 0.5318216  0.5261316  0.4753991    0.4413653
3 0.2176146 0.4385084 0.2899851 0.7748238 0.1990155 0.2438178 0.4494888  0.6249011  0.6114482    0.5350591
      exper  largecity
1 0.2911877  0.7450148
2 0.2335399 -1.3404727
3 0.1751868  0.7450148
```

Figure 32: Cluster Means for Variables

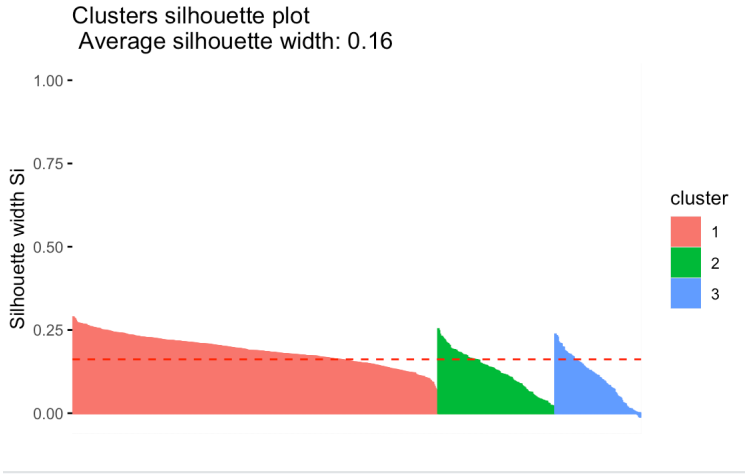

Figure 33: Silhouette Plot for k-means Clustering (negative silhouette: wrong cluster; close-to-zero silhouette: correct cluster but overlap with another cluster; close-to-one silhouette: correct cluster)

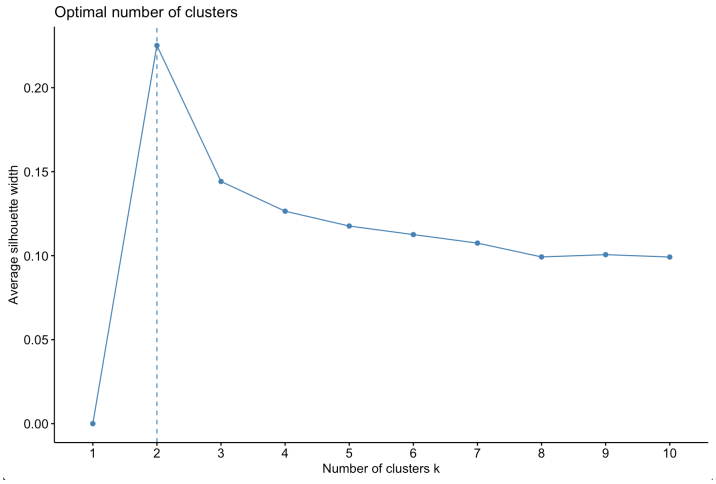

Figure 34: Optimal Number of Clusters for Maximizing the Silhouette Width

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.81000    0.29145   6.210 5.29e-10 ***
taxableinc   -2.15855    0.56733  -3.805 0.000142 ***
federaltax    1.65927    0.52774   3.144 0.001666 **
hsiblings    -0.36551    0.16999  -2.150 0.031536 *
hfathereduc  -0.10066    0.16354  -0.615 0.538231
hmothereduc  -0.04079    0.16363  -0.249 0.803143
siblings      0.06182    0.14459   0.428 0.668979
kidsl6       -0.65957    0.18595  -3.547 0.000390 ***
kids618       0.03865    0.16402   0.236 0.813717
age          -0.38574    0.32071  -1.203 0.229068
educ          0.23578    0.20618   1.144 0.252803
wage76        3.56348    0.40910   8.711  < 2e-16 ***
hhours       -0.42959    0.17741  -2.421 0.015459 *
hage         -0.03643    0.31424  -0.116 0.907719
heduc        -0.02423    0.19835  -0.122 0.902787
hwage        -0.90450    0.30217  -2.993 0.002759 **
faminc        0.06612    0.31674   0.209 0.834652
mtr          -0.80165    0.36787  -2.179 0.029320 *
mothereduc    0.33434    0.18820   1.777 0.075638 .
fathereduc   -0.09599    0.19012  -0.505 0.613623
unemployment -0.04578    0.14243  -0.321 0.747914
largecity    -0.08312    0.15015  -0.554 0.579890
exper         0.50650    0.15596   3.248 0.001164 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

(a) *Logit Estimates*

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.7226     0.2676   6.436 1.22e-10 ***
taxableinc   -2.0497     0.5635  -3.637 0.000276 ***
federaltax    1.5171     0.5289   2.868 0.004125 **
kidsl6       -0.5994     0.1602  -3.742 0.000183 ***
age          -0.6348     0.1697  -3.741 0.000183 ***
educ          0.3047     0.1806   1.688 0.091454 .
wage76        3.3999     0.3813   8.917  < 2e-16 ***
hhours       -0.4702     0.1777  -2.646 0.008139 **
hwage        -1.0466     0.2838  -3.688 0.000226 ***
mtr          -0.9651     0.2734  -3.530 0.000416 ***
mothereduc    0.2639     0.1625   1.624 0.104405
exper         0.6764     0.1676   4.037 5.42e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

(b) *Stepwise Logit Estimates*

Figure 35: Stepwise/Vanilla Logit - Estimates (note the consistent reduction of the number of features predicting the response)

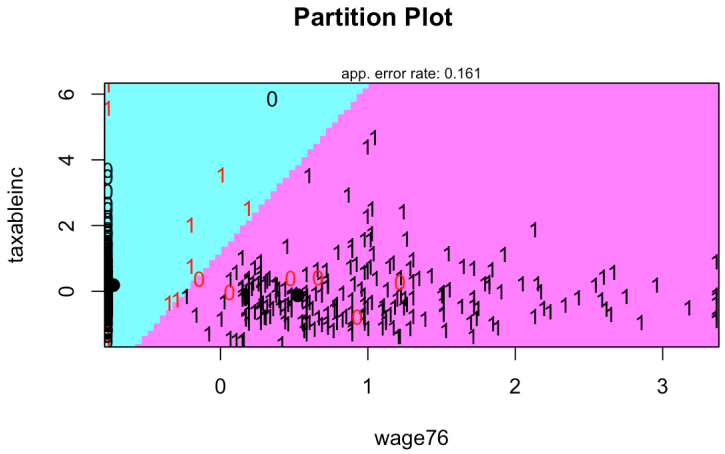

Figure 36: Partition Plot of the Response Variable, Classified accordingly to *taxableinc* and *wage76*

Table 5: Logit with only one tax variable

	Dependent variable:
	lfp
hsiblings	−0.189
	(0.158)
hfathereduc	−0.025
	(0.164)
hmothereduc	0.010
	(0.163)
siblings	0.028
	(0.152)
kidsl6	−0.733***
	(0.176)
kids618	0.074
	(0.159)
age	−0.752**
	(0.310)
educ	0.587***
	(0.208)
wage76	3.378***
	(0.408)
hhours	−0.620***
	(0.197)
hage	−0.026
	(0.307)
heduc	−0.102
	(0.199)
hwage	−0.957***
	(0.310)
faminc	0.337
	(0.330)
mtr	−0.684*
	(0.358)
mothereduc	0.167
	(0.180)
fathereduc	−0.034
	(0.189)
unemployment	−0.090
	(0.145)
largecity	−0.212
	(0.152)
exper	0.815***
	(0.173)
Constant	1.854***
	(0.295)
Observations	565
Log Likelihood	−165.463
Akaike Inf. Crit.	372.925
Note:	*p<0.1; **p<0.05; ***p<0.01

```
(Intercept)   0.72862180
taxableinc   -0.32890420
federaltax    0.09439418
hsiblings    -0.03954700
hfathereduc  -0.11409241
hmothereduc   0.10674335
siblings     -0.02000602
kidsl6       -0.41434821
kids618       0.10376023
age          -0.30504744
educ          0.19011423
wage76        1.57958476
hhours       -0.30812579
hage         -0.11067974
heduc         0.00207064
hwage        -0.37027908
faminc        0.09517143
mtr          -0.12652498
mothereduc    0.10630340
fathereduc    0.01351770
unemployment -0.06521322
largecity    -0.08110368
exper         0.45722393
```

Figure 37: Ridge coefficients

```
(Intercept)   1.79713152
taxableinc   -1.41707851
federaltax    1.00243934
hsiblings    -0.12781629
hfathereduc  -0.19922906
hmothereduc   0.10118867
siblings     -0.04872464
kidsl6       -0.57419284
kids618       0.04908124
age          -0.55986586
educ          0.32703043
wage76        3.47930598
hhours       -0.49125279
hage          .
heduc         0.05868827
hwage        -0.69387248
faminc        0.22298429
mtr          -0.21052734
mothereduc    0.18764200
fathereduc    .
unemployment -0.08452289
largecity    -0.15189722
exper         0.52782743
```

Figure 38: Lasso coefficients

References

K Srivastava Durgesh and B Lekha. Data classification using support vector machine. *Journal of theoretical and applied information technology*, 12(1):1–7, 2010.

George D. Greenwade. The Comprehensive Tex Archive Network (CTAN). *TUGBoat*, 14(3):342–351, 1993.

Andy Liaw, Matthew Wiener, et al. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.

Thomas A. Mroz. The sensitivity of an empirical model of married women's hours of work to economic and statistical assumptions. *Econometrica*, 55(4):765–799, 1987. ISSN 00129682, 14680262. URL `http://www.jstor.org/stable/1911029`.

Philip H Swain and Hans Hauska. The decision tree classifier: Design and potential. *IEEE Transactions on Geoscience Electronics*, 15(3):142–147, 1977.