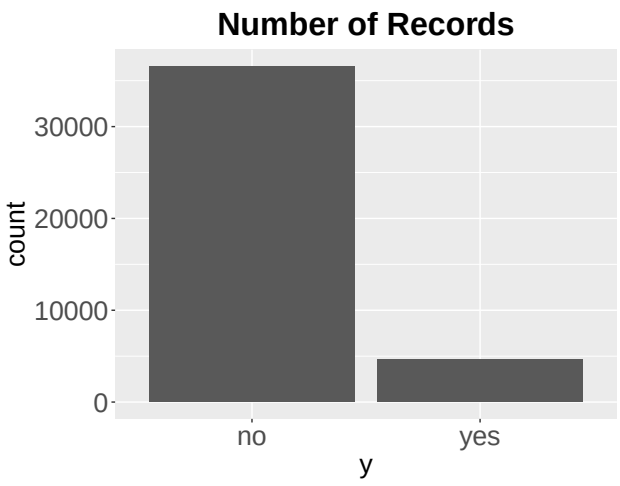
Performance of L2 Penalized Logistic Regression on Predicting the Success of Bank Telemarketing

1 Introduction

The project mainly focuses on a classification that helps the Portuguese retail bank to predict if a client would subscribe to a bank term deposit. We apply two different models; one is a logistic regression model, and another is a random forest tree model since we have a binary dependent variable. To better predict and classify the subscription condition, we are going to compare the performances of two different models and explain the differences in the performances. The data we use called bank-additional-full, which collected of 45211 client's information along with the subscription condition, ordered by date with 16 output retributes. The reduced dataset only contains 10% examples from the full data selected randomly. Thus, avoiding miss any vital information, we decided to use the full data. The data don't have any missing value.

2 Descriptive Analysis

- ***Unbalance Data*** Figure 1(a) shows that dataset "bank-additional-full" is unbalanced, as only 4640 records(11.26%) are related to 'yes', which means the client subscribed a term deposit. This unequal records for different classes need to be addressed since most machine learning classification algorithms are sensitive to unbalance in the predictor classes. An unbalanced dataset will bias the prediction model towards the more common class. Since we will use Random Forest Classification in the latter part, we'd like to deal with this imbalance from the beginning. The approach we adopt is under-sampling, where we randomly select a subset of samples from the class with more records('no') to match the number of records coming from less common class('yes'). We used the `downSample` function in the `caret` package, so that we have a new dataset of 9280 records in total, with the same number of records for both classes. We applied under-sampling out of two reasons: first, even after disregarding a large number of records, we still had plenty of data that gave sufficient information; second, by decreasing the number of records, the learning time was also decreased. One major disadvantage, which is obvious, is that undersampling discards are potentially useful data. After under-sampling, we check the levels of categorical variables, which indicates that all levels in the full dataset were included in the under-sampling dataset.



(a)

Variables	VIF
pdays	11.42
previous	5.22
poutcome.success	10.14
emp.var.rate	127.95
cons.price.idx	53.39
cons.conf.idx	5.24
euribor3m	140.39
nr.employed	144.04

(b)

Figure 1: (a)Number of Records in bank-additional-full (b)List of Variables with VIF>5

- ***Add Dummy Variables*** The dataset has 20 predictor variables, and 10 of them are categorical variables. For the convenience of L2 penalized logistic regression in the later section, we transformed the initial dataset by adding dummy variables. After transformation, we have 53 variables in total. A detailed explanation could be found in the Appendix.
- ***Split into Training/Testing Data*** Our full dataset had a total of 9280 records, and we adopt 80/20 rule, which gave us 7424 training records and 1856 testing records.
- ***Multicollinearity*** One assumption in logistic regression model is there should be no high multi-collinearity among the predictors. To quantify multicollinearity, we used `vif{car}` to calculated the variance inflation factor(VIF). One particular fact we noticed is that, two of the dummy variables, housing.unkown and loan.unkown, are linearly dependent: when housing.unkown is 'unknown', so is the loan.unkown. Thus to avoid aliased coefficients in the model, we dropped one column, in this case, loan.unknown. Figure 1(b) shows the list of variables whose VIF is larger than 5 when modeling with logistic regression. In practice, $max_k\ VIF_k > 10$ is often taken as an indication that multicollinearity is high. In this case, the maximum value is 144.04, which is much larger than 10. We will adjust for this high multicollinearity in the modeling part.

3 Analysis

3.1 Logistic Regression

This project is a case-control study, and we focus on a classification problem. Under this circumstance, logistic regression is a choice because the calculated sample odds ratio performs well as an estimator of the population odds ratio for moderate and large samples. Thus, We use logistic regression to model the conditional probability $Pr(Y = 1|\mathbf{X} = \mathbf{x}) = E[Y|\mathbf{X} = \mathbf{x}]$ as a function of the predictors and use maximum likelihood estimation to estimate unknown parameters in the function. In the data analysis section, we found some variables correlate with each other, so we try to build the L2 penalized logistic model for solving this problem. Then, we will do a sensitivity analysis and evaluate the performance of the model.

3.1.1 L2 Penalized Logistic Regression

$$logit(\pi_i) = \beta_0 + \beta_1 X_{1,i} + \beta_2 X_{2,i} + \cdots + \beta_{53} X_{53,i}$$

$\pi_i = Pr(Y = 1|\mathbf{X} = \mathbf{x_i}) = E[Y|\mathbf{X} = \mathbf{x_i}]$; logit(p) denotes logit function or the log-odds, which is defined as $\frac{p}{1-p}$.

$X_{k,i}$ denotes the value of the kth independent variable in the ith sample.$k = 1, 2, \cdots, 53; i = 1, 2, ..., 7424$.

The explanation of $X_k, k = 1, 2, \cdots, 53$ is shown in the Appendix 1.

Y_i denotes whether the client has subscribed a term deposit.

If the client has subscribed a term deposit, $Y_i = 1$. Otherwise, $Y_i = 0$.

β_k denotes the coefficient of the kth predictor.$k = 1, 2, \cdots, 53; \beta_0$ denotes intercept.

Estimate the coeficients by maximum likelihood estimation:

$$\hat{\boldsymbol{\beta}} = argmin_{\boldsymbol{\beta}} \left[\frac{1}{7424} \left\{ -Y_i(\beta_0 + \sum_{k=1}^{53} \beta_k X_{k,i}) + log(1 + exp(\beta_0 + \sum_{k=1}^{53} \beta_k X_{k,i})) \right\} + \lambda \sum_{k=1}^{53} \beta_k^2 \right]$$

3.1.2 Model Assumptions for Logistic Regression

- ***Independence*** Observations to be independent of each other. This dataset is a case-control study and we randomly select a subset of samples from the class with more records('no'). It is reasonable to believe that this assumption holds.
- ***Binary outcome*** The response variable should be binary data. This assumption satisfied since our response variable is if the client subscribed to a term deposit, which is a binary data that includes "yes" and "no" two outcomes. This assumption satisfied since our response variable is if the client subscribed to a term deposit, which is a binary data that includes "yes" and "no" two outcomes.

- ***No influential values*** There should be no influential values like extreme values or outliers in the predictors
- ***No multicollinearity*** There should be no high multicollinearity among the predictors.

We will test the influential values assumptions and multicollinearity assumptions in the Model Diagnostic section.

3.1.3 Model Fitting

In this project, we consider all independent variables include 20 predictor variables. Among them, 10 are categorical variables. After transforming all categorical variables by adding dummy variables, we included 53 variables in the project. We use 'cv.glmnet' to choose λ and estimate the coefficients. The estimated coefficients are listed in Appendix 1. The fitting results showed that the response variable was significantly affected by the variables of job, marital, education, housing loan, loan, contact communication type, duration, campaign, pdays, previous and outcome.

3.1.4 Model Diagnostics

- ***No influential values*** Since we employ L2 penalized logistics model to do classification, there is no common way to detect influential observations. Even though L2 penalized logistics model is not very robust to outliers, we can not find an efficient way to detect influential outliers and deal with them. Thus, in this project, we just accept that there is no influential value.

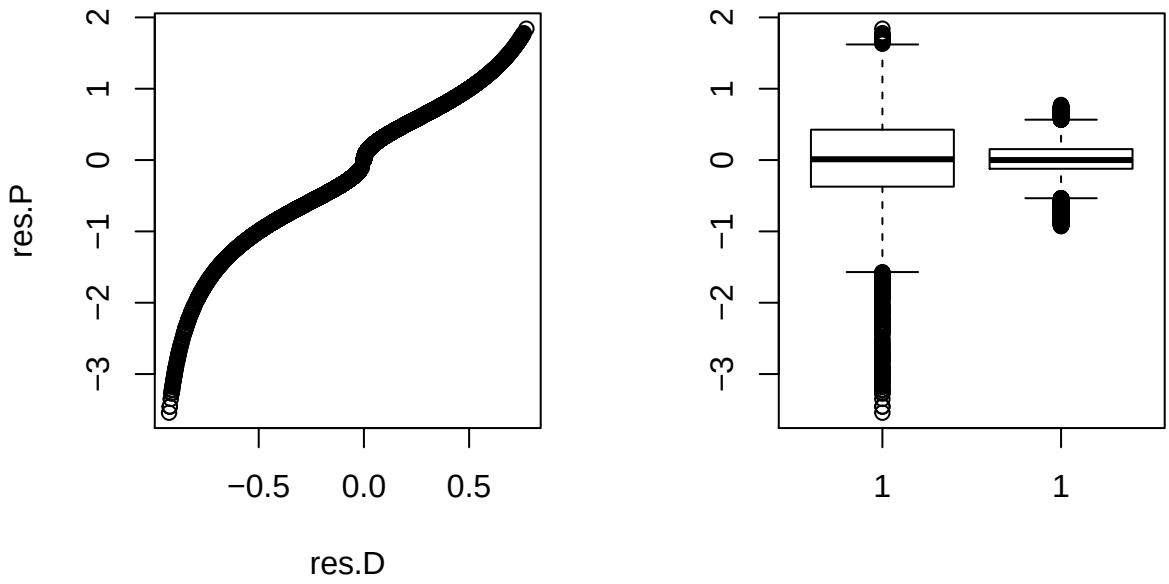
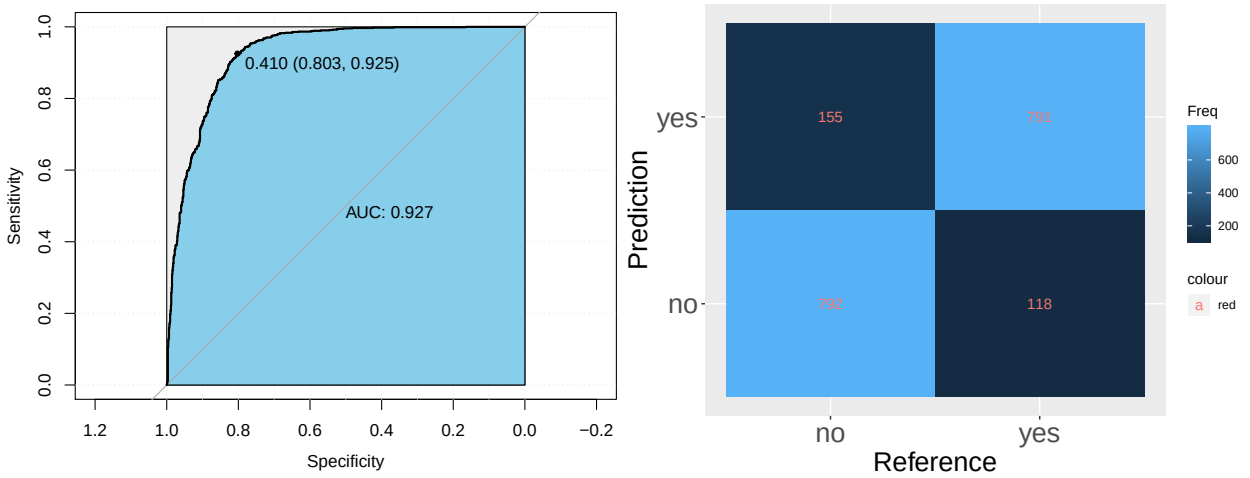


Figure 2: Diagnostic plots. Left panel: Pearson residual vs Deviance residual. Right panel: Boxplot of Pearson residual and Devianc residual

- ***No multicollinearity*** The variance inflation factor (VIF) is an indicator estimates how the variance of an estimated regression coefficient increase due to the collinearity. If the VIF value is larger than 10, it indicates a problematic amount of collinearity. In our analysis, We originally had six variable have VIF values greater than 10, but we used L2 penalized logistic regression which prevents problems arising due to collinearity.
- ***Goodnedd-of-fit*** The Deviance Residuals and Pearson Residuals plots suggest that if the two kinds of residuals are not entirely similar to each other, the model may suffer from potential lack-of-fit. The two kinds of residuals are quite similar to each other wich menas the model does not suffer from potential lack-of-fit.

3.1.5 L2 Penalized Logistic Regression performance evaluation



(a) AUC plot

(b) Confusion Matrix

Figure 3: L2 Penalized Logistic Regression performance evaluation

To evaluate the performance of our L2 penalized logistic regression model, we use the Receiver Operating Characteristics (ROC) Area Under the Curve(AUC) curve as our visual indicator. ROC is a probability curve that bases on the ratio of the true-positive rate against the false-positive rate, and AUC represents the degree of separability under different threshold settings. It tells how confident the model could do the classification. The AUC value ranges from 0 to 1. Higher the AUC value, the better the model is at predicting. In the AUC plot, the AUC value is 0.927, which is close to 1. It means we have a 92.7% chance that the model will classify a client who subscribed to a term deposit as 'yes' and a client who didn't subscribe to a term deposit as 'no.'

3.2 Random Forests

3.2.1 Random Forests Model Built

Random Forests is an ensemble learning algorithm. It was designed based on the decision trees, and it combines the predications of several base decision trees. The base decision trees are built independently; then, their predictions are averaged as the final prediction. The base decision trees are created as a diverse set of classifiers so that randomness is introduced during this construction.

In the random forest tree algorithm, we used two ways to add randomness. One is from the samples used for each tree; we drew data for each base tree with replacement from the input data. The other one is the features used to splitting each node; we used a random subset of features for splitting the node of each tree.

We used the grid-search, ten-fold cross-validation method to tune the two parameters, one is the number of trees, and the other is the number of features randomly sampled as a candidate to split the node for each tree. The Fig 4 shows the accuracy for each parameter; from it, we can see, when the number of trees is 25, and the number of features is around 16, the best accuracy can be obtained as 0.890.

The cross-validation accuracy is calculated upon the validation part, using the model trained with the training part. So it can reflect the algorithm's testing performance. We found 25 trees are the best, and we won't go beyond 30 trees since more trees mean more complex of the algorithm. The Random Forest does not increase generalization error when more trees are added to the model. But the model with full trees likely has lower train error but higher test error than the model with pruned trees.

While there are other hyperparameters that can affect the model, due to limited ability, we won't tune them. However, we do some reference search to report their relationship with the model; interested readers are welcome to test by themselves.

- ***Pre-pruning Threshold*** This measures the threshold that a node will be split if this split induces a decrease of the impurity greater than or equal to this threshold value. This threshold is between (0.0-1.0); the bigger the value, the more aggressive pruning. Pruning in decision tree-based classification methods is very important. The goal is to iteratively split to minimize the "impurity" of the partitioned dataset, meaning a leaf node contains samples that all belong to one class, it is "pure" and thus has an impurity of 0. The ultimate goal of decision tree-based models is to split the tree such that each leaf node corresponds to the prediction of a single class, even if there is only one sample in that class. However, this can lead to the tree radically overfitting the data; it will grow in a manner such that it will create a leaf node for every sample if necessary. So the vital thing is about tree pruning. There are different pruning methods. In general, the training error immediately increased after pruning significantly from thousands of trees to less than 100, on the contrary, the test/cv error decreased, both of them keep falling until some nodes, then starts increased if pruning too much and leads to underfitting.
- ***Max depth*** Another feature to limit tree growth in some way is the max depth. The tree grown with a small max depth has a relatively high error rate since a tiny tree could be underfitting. Thus, it's essential to maintain a balance in tree size. However, trees with too many nodes overfit easily.

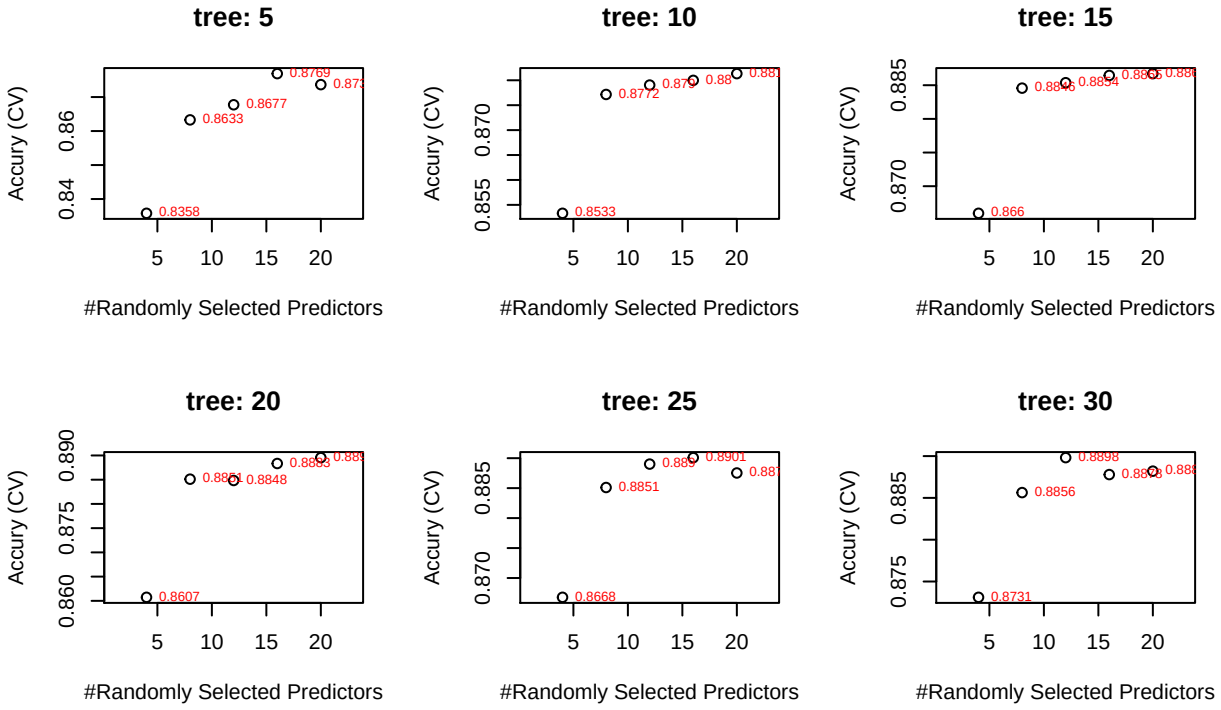


Figure 4: grid search results

3.2.2 The learning curve, ROC, and confusion matrix

Learning curves were plotted after setting a reasonable number of trees of 25, and the number of features sampled to split the node is 12 from the previous analysis. The train, test errors vs. various training examples are reflected in Fig **??**. Here learning curves are used to evaluate the underfitting or overfitting of the overall algorithm. The training error starts from zero because a function can always be found that touches those number of points precisely. The training error starts increased as the training set gets larger, and the error value will plateau out after a particular training set size. On the contrary, the test errors start from high, which is because of the weak classifier trained from a small portion of instances, then slowly decreased. It showed that once the training size reached to a level, the error $error_{train}(\theta)$ and error $error_{test}(\theta)$ is close, this could indicate a bias of the algorithm. However, it may not be a high bias case since the $error_{test}(\theta)$ is still decreasing very slowly.

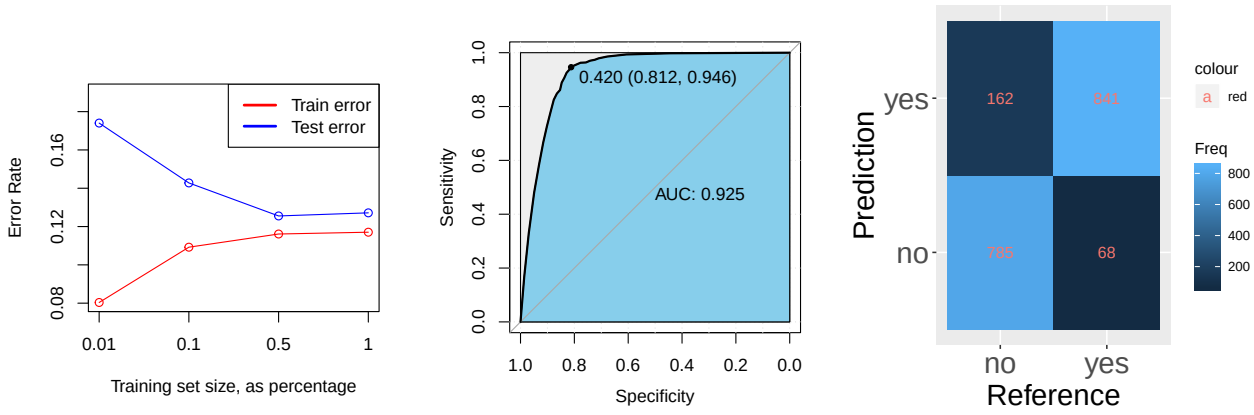


Figure 5: Performance Plot. Left panel: Random Forests Learning Curve. Middle panel:ROC, AUC.Right panel:

We used our final model to predict the test data set. The ROC, AUC, and confusion matrix are shown in the below Fig 5. We can see the AUC is close to the AUC of the L2 penalized logistic model.

4 Discussion

4.1 Under-sampling and Over-sampling

Both under-sampling and over-sampling could deal with class imbalance. Apart from the reasons for choosing under-sampling given in the data processing section, we also consider the disadvantages and feasibility of using over-sampling. When over-sampling, we randomly duplicate samples from the class with fewer records. Even though this process avoids losing information, one major disadvantage is that we might overfit the model and overestimate the performance since we are more likely to have the same samples in training and testing data. Besides, over-sampling increases the size of the training dataset greatly and makes it more time consuming to implement learning algorithms. Compared with these disadvantages, losing partial information from under-sampling is more acceptable, thus lead to the decision of using under-sampling.

4.2 Comparison Between L2 Penalized Logistic Regression and Random Forests

In this project, we used two classification models, L2 Penalized Logistic Regression and Random Forests. Comparing the two models, L2 Penalized Logistic Regression has the advantage of fitting a model that tends to be easily understood by humans and less time consuming. When compared with logistic regression, Random Forests is more flexible as we have no probabilistic model, but just binary split. We might not need to make any assumption except sampling is representative and independent. However, Random Forests is hard to be interpreted and time consuming in learning.
Using AUC to evaluate the performance of classification algorithms, we compare the performance of two models. As shown in Figure 3 and Figure 5, AUC of L2 Penalized Logistic Regression is 92.7%, which outperformed Random Forests by 0.2%. However, we only run the whole analysis process once, this result might not be stable. If time allowed, we would apply resampling approach, such as cross validation, to generate a more stable result.

5 Reference

1. https://www.r-bloggers.com/dealing-with-unbalanced-data-in-machine-learning/
2. https://rpubs.com/shienlong/wqd7004_RRookie
3. http://www.sthda.com/english/articles/36-classification-methods-essentials/148-logistic-regression-assumptions-and-diag
#logistic-regression-assumptions

4. https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5
5. https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf
6. https://www.alexejgossmann.com/auc/
7. https://towardsdatascience.com/all-the-annoying-assumptions-31b55df246c3
8. A data-driven approach to predict the success of bank telemarketing, Sergio Moro, et al.(2014)

6 Appendix 1: List of Predictors and Estimated Coefficient

The table below listed the predictors used in L2 Penalized Logistic Regression, and their estimated coefficient.

X_k.i	Denote	Beta_k
	(Intercept)	20.80
X_1,i	age of record i	0.00
X_2,i	job.blue-collar of record i	-0.19
X_3,i	job.entrepreneur of record i	0.03
X_4,i	job.housemaid of record i	-0.12
X_5,i	job.management of record i	-0.14
X_6,i	job.retired of record i	0.24
X_7,i	job.self-employed of record i	-0.10
X_8,i	job.services of record i	-0.11
X_9,i	job.student of record i	0.25
X_10,i	job.technician of record i	0.00
X_11,i	job.unemployed of record i	0.17
X_12,i	job.unknown of record i	-0.12
X_13,i	marital.married of record i	-0.01
X_14,i	marital.single of record i	0.06
X_15,i	marital.unknown of record i	0.33
X_16,i	education.basic.6y of record i	-0.06
X_17,i	education.basic.9y of record i	-0.13
X_18,i	education.high.school of record i	-0.06
X_19,i	education.illiterate of record i	0.47
X_20,i	education.professional.course of record i	0.05
X_21,i	education.university.degree of record i	0.13
X_22,i	education.unknown of record i	0.17
X_23,i	default.unknown of record i	-0.26
X_24,i	default.yes of record i	-1.37
X_25,i	housing.unknown of record i	-0.01
X_26,i	housing.yes of record i	-0.02
X_27,i	loan.unknown of record i	-0.01
X_28,i	loan.yes of record i	-0.05
X_29,i	contact.telephone of record i	-0.10
X_30,i	month.aug of record i	0.13
X_31,i	month.dec of record i	0.05
X_32,i	month.jul of record i	0.14
X_33,i	month.jun of record i	0.08
X_34,i	month.mar of record i	1.20
X_35,i	month.may of record i	-0.75
X_36,i	month.nov of record i	-0.36
X_37,i	month.oct of record i	0.77
X_38,i	month.sep of record i	-0.01
X_39,i	day_of_week.mon of record i	-0.04
X_40,i	day_of_week.thu of record i	-0.02

X_41,i	day_of_week.tue of record i	0.04
X_42,i	day_of_week.wed of record i	0.15
X_43,i	duration of record i	0.00
X_44,i	campaign of record i	-0.03
X_45,i	pdays of record i	0.00
X_46,i	previous of record i	0.01
X_47,i	poutcome.nonexistent of record i	0.23
X_48,i	poutcome.success of record i	0.72
X_49,i	emp.var.rate of record i	-0.22
X_50,i	cons.price.idx of record i	0.04
X_51,i	cons.conf.idx of record i	0.02
X_52,i	euribor3m of record i	-0.17
X_53,i	nr.employed of record i	0.00