UNIVERSITÄT
BIELEFELD

Report third project

Yasmin Bouhlada, Annalisa Dettori, Andrea Schinoppi

January 2024

2 Introduction

3 Data Set

3.1 First data set

The first data set (named *real world data*) was a real data set regarding the prices of some houses. It was made of ten columns $X_1, \ \ldots, X_{10}$ for the independent variables and one column for the dependent variable y. The data set had 1095 numeric samples and it had no NaNs.

We could resume the main information of the variables with this table:

	`LotArea`	`TotalBsmtSF`	`1stFlrSF`	`2ndFlrSF`	`GrLivArea`	`WoodDeckSF`
mean	10722.41	1159.84	0	338.71	1505.13	$91,06$
std dev	11054.40	376.46	34900	432.04	514.24	120.64
min	1300	0	343	0	334	0
max	215245	3206	3228	1872	4676	670

	`OpenPorchSF`	`3SsnPoarch`	`ScreenPorch`	`PoolArea`	y
mean	47.26	2.78	15.09	2.14	179984.82
std dev	66.79	25.18	56.55	35.79	77610.06
min	0	0	0	0	34900
max	547	407	480	738	755000

3.2 Second data set

The first data set was made of thirteen columns X for the independent variable and one column for the dependent variable y. The data set had 110 numeric samples and it had no NaNs. We didn't know what the variables represented, but we could resume the main information of the variables with this table:

	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
mean	-0.28	-0.33	-0.25	0.02	-0.03	-0.31	-0.48	0.08	-0.19	-0.31
std dev	0.92	0.79	0.99	1.06	1.26	1.13	1.00	1.13	1.15	0.8
min	-1.49	-1.23	-3.31	-2.42	-1.95	-2.62	-2.61	-1.84	-2.5	-1.51
max	2.13	2.17	1.79	3.31	4.05	2.44	1.71	2.76	2.84	2.23

	X_{11}	X_{12}	X_{13}	y
mean	0.36	-0.39	-0.34	1.82
std dev	0.90	1.16	0.85	0.38
min	-1.6	-2.88	-1.46	1
max	2.33	2.04	1.96	2

3.3 Third data set

The third data set was made of ten columns X for the independent variable and one column for the dependent variable y. The data set had 1000 numeric samples, it had no NaNs and in the dependend variable y we had boolean values $\{0, 1\}$ referring to two classes. We didn't know what the variables represented, but we could resume the main information of the variables with this table:

	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
mean	0.66	-0.06	0.02	-0.2	0.04	-0.1	0.12	0.06	0.08	0.06
std dev	2.60	5.80	3.50	5.76	2.69	5.75	5.78	5.90	3.31	5.79
min	-7.32	-9.98	-8.44	-9.95	-9.13	-9.98	-9.98	-9.99	-7.91	-9.98
max	7.4	9.97	7.43	9.97	8.92	9.98	9.98	9.97	8.26	9.97

4 Foundations

4.1 Methods

4.1.1 Filter methods, wrapper methods and embedded methods

Given the regression (or classification) tasks $(\vec{x}_i, y_i)_{i=1}^n$, we decided to apply three methods to do feature selection : filter methods, wrapper methods and embedded methods.

- A filter method consists in assigning a relevance value $r(I)$ for every feature I based on the data only and then take the most relevant features only. In this case we decided to use F measure.

  Given a training set (X, Y), the F measure computes the Pearson correlation of the inputs to the output for each feature.

- A wrapper methods selects features iteratively based on their impact on the chosen classifier/regressor. We decided to be more complete to use both forward (adding a new variable starting from a model with zero variables) selection and backward (starting with the full model and always removing the worst value) selection.
- A embedded method integrates and adapts feature relevance terms within the model itself. We used Lasso that uses the formula $\min_{\vec{w}} NLL + \lambda||\vec{w}||_1$, where NLL refers to the negative data log likelihood.

4.1.2 Random forest classifier

Random forest classifier implements a bagging of decision trees with randomization of the input variable selection.

4.1.3 Naive Bayes classifier

The naive Bayes classifier estimates probabilities of the optimum Bayes classifier based on data and relying on the assumption of conditional independence of the observed data coefficients.

4.1.4 Logistic regression

Given the two class classification problem we had with classes 0, 1, logistic regression realizes the model

$$\mathbb{P}(y = 1|\vec{x}, \vec{w}) \sim Ber(y = 1|sgd(\vec{w}^T\vec{x}))$$

where $\vec{w}$ is the model parameter, $sgd(t) = (1 + exp(-t))^{-1}$ is the sigmoidal function, and Ber is the Bernoulli distribution.

4.1.5 k nearest neighbours

For fixed $k > 0$, the k-nearest neighbor classifier provides the output

$$\mathbb{P}(y = c|\vec{x}, \mathcal{D}, k) = \frac{1}{k}|\{\vec{x}_i|(\vec{x}_i, y_i) \in \mathcal{D}, \vec{x}_i \text{ within } k \text{ closest points of } \vec{x} \text{ and } y_i = c\}|$$

Then the class is determined for $f_{kNN}(\vec{x}) = argmax_c \mathbb{P}(y = c|\vec{x}, \mathcal{D}, k)$.

4.2 Evaluation

4.2.1 Cross validation

For our methods we always used the cross validation for which we splitted the data set into 5 folds. To do it we trained the model on four folds and we tested on the fifth one to compute the our accuracy measures. After doing it for all folds, we took the mean of the five accuarcy measure obtained.

4.2.2 Accuracy measures

As accuracy measures for random forest classifier, naive Bayes classifier, logistic regression and k-nearest neighbors we used the accuracy score which is the sum of the true positive and true negative over all cases.

5 Experiments

5.1 Set-Up

Our experiments were carried on a jupyter notebook that we converted in a python file. The data sets' extension is `.npz`.

5.2 Analysis and Results

5.2.1 First data set

On the first dataset we did did a variable selection with the previous explained methods and using the random forest regressor the *F measure* we selected the following features

- Two features: `['TotalBsmtSF' 'GrLivArea']`
- Six features: `['2ndFlrSF' 'OpenPorchSF' 'WoodDeckSF' '1stFlrSF' 'TotalBsmtSF' 'GrLivArea']`

With the *forward feature selection* we selected the following features

- Two features: `['TotalBsmtSF' 'GrLivArea']`
- Six features: `['LotArea' 'TotalBsmtSF' '1stFlrSF' '2ndFlrSF' 'GrLivArea' 'OpenPorchSF']`

With the *backward feature selection* we selected the following features

- Two features: `['TotalBsmtSF' '2ndFlrSF']`
- Six features: `['LotArea' 'TotalBsmtSF' '1stFlrSF' '2ndFlrSF' 'OpenPorchSF' '3SsnPorch']`

With *Lasso* we selected the following features

- Two features: `['2ndFlrSF' '1stFlrSF']`
- Six features: `['2ndFlrSF' '1stFlrSF' 'TotalBsmtSF' 'WoodDeckSF' 'OpenPorchSF' 'ScreenPorch']`

METTERE UN COMMENTO SU COME SONO CAMBIATE LE FEATURES

With the selection of two variables we obtained the following results:

method	R^2
F measure	0.22
Forward selection	0.22
Backward selection	0.20
Lasso	0.18

COMMENTA CHE SONO BASSI

With the selection of six variables we obtained the following results:

method	R^2
F measure	0.47
Forward selection	0.46
Backward selection	0.44
Lasso	0.44

COMMENTA CHE SONO BASSI

5.2.2 Second data set

We worked on the first dataset which was a classification task with the following methods:

- Random forest classifier
- Naive Bayes
- Logistic regression
- kNN

and with all of them we used cross validation with five folds. The results we obtained for the accuracy are the following:

method	accuracy
Random Forest Classifier	0.91
Naive Bayes Classifier	0.9
Logistic regression Classifier	0.54
k-nearest neighbors Classifier	0.65

Then we decided to use forward selection. We selected the model with the highest value for the R^2 on the test set which was the model complete of the second, third, sixth and nineth variable to obtain the following results:

method	accuracy
Random Forest Classifier	0.91
Naive Bayes Classifier	0.9
Logistic regression Classifier	0.51
k-nearest neighbors Classifier	0.86

5.2.3 Third data set

6 Bonus task

eheh

A Appendix

banana