A Comparison of Feature Selection Methods to Optimize Predictive Models Based on Decision Forest Algorithms for Academic Data Analysis

Antonio Jesús Fernández-García(✉), Luis Iribarne, Antonio Corral, and Javier Criado

Applied Computing Group, University of Almeria,
Ctra. Sacramento, s/n, 04120 La Cañada, Almería, Spain
ajfernandez@ual.es

Abstract. Nowadays, Feature Selection (FS) methods are essential (1) to create easy-to-explain predictive models in shorter periods of time, (2) to reduce overfitting and (3) avoid sparsity of data. The suitability of using these techniques is studied in this paper. Furthermore, a comparison of some widely extended techniques is performed to know which one is more appropriated to create predictive models using decision forest algorithms. For this comparison, experiments are conducted in which predictive models for each FS method are built to foresee if students will finish their degree after finishing their first year in college. A real dataset with students' data provided by the University of Almería is used to generate the predictive models. By comparing the accuracy of the built models, we can measure the effectiveness of each FS method, being the *Chi-Square statistic* the method that leads to better results in our experimental study.

Keywords: Feature selection · Machine learning · Decision forest

1 Introduction

The set of attributes of a dataset should be as significant as possible to describe the nature of the reality they represent. It may seem that a large number of features can better describe a problem and (with them) better predictive models can be built, but this is not entirely true. FS methods can help to reduce overfitting [11] and avoid high-dimensional spaces and the sparsity of data [6] that datasets with a large set of attributes may have. Also, a feature subset of a dataset implies shorter training times building the models by removing redundant or irrelevant features.

We are currently working on a project that consists of applying machine learning techniques to create a *decision model to predict if a student at the*

Á. Rocha et al. (Eds.): WorldCIST'18 2018, AISC 745, pp. 338–347, 2018.
https://doi.org/10.1007/978-3-319-77703-0_35

University of Almería (Spain) will graduate after finishing the first year of college studies. There are many reasons to know if a student is going to be graduated: (a) Plan next year enrollments; (b) Put the necessary means to improve the rate of graduates; (c) Identify and assist students who may require it; and (d) Make an economic incoming forecasting based on future enrollment fees.

To achieve that, we follow the same methodology that we proposed for developing adaptive evolutionary interfaces [3,5]. In that proposal, we carry out a feature engineering process [4], that transforms raw data to create features that have better representation. Therefore, the transformed dataset, allows the creation of better predictive models. In the case of the prediction of graduates, we propose to apply some feature selection methods in addition to applying the feature engineering techniques of the mentioned methodology.

In this work, we perform an experimental study with a comparative of the effectiveness of several Feature Selection (FS) methods available in the Microsoft Azure Machine Learning Studio, AzureML [9], over a real-world dataset provided by the University of Almería. The dataset includes data obtained across several years of 6867 students during their first year in college labeled as graduated or not. Feature engineering techniques have been previously applied to the dataset. Table 1 shows the features that make it up.

Table 1. Set of features of the "first year in college performance" dataset provided by the University of Almería (Spain)

Feature name	Description	Type
`Age`	Age	Numeric
`BirthProvince`	Place of Birth	String
`Nationality1`	First Nationality	String
`DoubleNationality`	Has the student more than a nationality?	Boolean
`Degree`	Degree Name	String
`DegreeField`	Degree Field	String
`Faculty`	Faculty Name	String
`UniversityAccessType`	How the student access to the degree	String
`CreditsEnrolled`	Number of Credits Enrolled	Numeric
`AverageScore`	Average Score in the Subjects Enrolled (0–10)	Numeric
`SuccessRate`	Percentage of Credits Pass	Numeric
`Graduated`	Has the student graduated? *(Label)*	Boolean

There are several surveys and experimental evaluations of feature selection methods in the bibliography. For instance, in Molina et al. [10] the authors work on a criteria that enables to adequately decide which algorithm to use in certain situations; in Chandrashekar et al. [2] the authors describes and apply some machine learning algorithms to standard datasets to analyze and compare feature selection

methods; and as example of work centered on a certain field, in Lazar et al. [7] the authors focus on filter feature selection methods for informative feature discovery in gene expression microarray (GEM) analysis.

Even though there are many works in the bibliography related to feature selection techniques, as far as the authors are concerned, there are no contributions that compare the implementations deployed on AzureML. This platform is widely used today not only for prototyping and experimentation but to create predictive models that are deployed in the industry. For that reason, we have performed a small comparison of the AzureML feature selection methods.

In our experiments, a two-class decision forest classification algorithm is used to generate the predictive models. The resulting datasets, with a subset of features from the original dataset after applying feature selection methods, are the input to the algorithm that creates the predictive models. One predictive model is created as a result of applying each feature selection method. To measure the effectiveness of the FS methods, the *accuracy*, *precision* and *recall* of the predictive models built are compared. Also, the *accuracy*, *precision* and *recall* of the original dataset without applying any FS method is taking into account. As an experimental deduction of the methods applied, we conclude that applying FS methods before creating predictive models is beneficial. In our experiments, the *Chi-Squared statistic* method gets the best results.

The rest of the paper is organized as follows. Section 2 describes the FS methods applied in the experimental study. Section 3 analyzes the results of the experiments conducted after the creation of predictive models using a two-class decision forest classification algorithm and the FS methods described in Sect. 2. Finally, Sect. 4 concludes and provides future directions.

2 Feature Selection Methods

In this work, we differentiate selection methods that support only numeric features from those supporting all data types features. Table 2 shows the FS methods compared in this experimental study and their supported features. As commented above, we make use of the AzureML implementation of these algorithms.

Table 2. Feature selection methods available in AzureML

Feature selection method	Supported features
Pearson's correlation	Numeric
Kendall's correlation coefficient	Numeric
Spearman's correlation coefficient	Numeric
Mutual information score	All data types
Chi-squared statistic	All data types

The next subsections describe the FS methods according to the features they support: numeric or all data types.

2.1 Methods for Dataset with Numeric Features

These methods measure how well two (or more) numeric feature are related. Thus, the relation between the *Graduated* feature and each one of the others numeric features presented in the dataset (*Age*, *CreditsEnrolled*, *AverageScore*, *SuccessRate*) is measured.

Pearson's Correlation:

$$R(i) = \frac{(cov(x_i, Y))}{\sqrt{var(x_i)\ var(Y)}}; \tag{1}$$

where x_i is the i_{th} variable, Y is the output (class label), $cov()$ is the covariance and $var()$ is the variance [2].

Kendall's Correlation Coefficient:

$$K(X, Y) = \frac{(P - Q)}{\sqrt{((P + Q + X_0)\ (P + Q + Y_0))}}; \tag{2}$$

where P is the number of concordant pairs, Q is the number of discordant pairs, X_0 is the number of pairs tied only on the X variable, Y_0 is the number of pairs tied only on the Y variable. Two observations $(X_i Y_i)$ and $(X_j Y_j)$ are concordant if they are in the same order with respect to each variable, that is, if $X_i < X_j$ and $Y_i < Y_j$, or if $X_i > X_j$ and $Y_i > Y_j$. Two observations $(X_i Y_i)$ and $(X_j Y_j)$ are discordant if they are in reverse ordering, that is, if $X_i < X_j$ and $Y_j > Y_j$, or if $X_i > X_j$ and $Y_j < Y_j$ [12].

Spearman's Correlation Coefficient:

$$rs = 1 - \frac{6\ \sum_{i=1}^{n} d_i^2}{n\ (n^2 - 1)}; \tag{3}$$

where n is the number of samples and d_i is the difference between two ranks of each instance, defined as follow:

$$d_i = rg(X_i) - rg(Y_i); \tag{4}$$

being X and Y two attributes of a dataset [1].

2.2 Methods for Dataset with All Data Types Features

These methods measure how well two (or more) feature are related, regardless the feature datatype. Thus, the relation between the **Graduated** feature and each one of the others features presented in the dataset (*Age*, *BirthProvince*, *Nationality1*, *DoubleNationality*, *Degree*, *DegreeField*, *Faculty*, *UniversityAccessType*, *CreditsEnrolled*, *AverageScore*, *SuccessRate*) is measured.

Mutual Information Score:

$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y)\ log\left(\frac{p(x,y)}{p(x)\ p(y)}\right); \tag{5}$$

where $p(x,y)$ is the joint probability function of X and Y, and $p(x)$ and $p(y)$ are the marginal probability distribution functions of X and Y respectively [2].

Chi-Squared Statistic:

$$X^2 = \sum_{i=1}^{n} \sum_{j=1}^{m} \frac{\left(O_{(i,j)} - E_{(i,j)}\right)^2}{E_{(i,j)}}; \tag{6}$$

where $O_{(i,j)}$ is the observed value of two nominal variables and $E_{(i,j)}$ is the expected value of two nominal values. The expected value can be calculated with the next formula:

$$E_{(i,j)} = \frac{\sum_{i=1}^{c} O_{(i,j)}\ \sum_{k=1}^{c} O_{(k,j)}}{N}; \tag{7}$$

where $\sum_{i=1}^{c} O_{(i,j)}$ is the sum of the i_{th} column and $\sum_{k=1}^{c} O_{(i,j)}$ is the sum of the k_{th} column [8].

3 Experimentation

This section presents the experiments conducted over the dataset after applying the FS methods commented above. Once the FS methods are applied, a predictive model is created for each method. We have used a decision forest algorithm to create these predictive models.

In order to test the utility of the FS methods, a comparison of the obtained results of the predictive models is performed. The next subsections describe the results of the FS methods and the *accuracy*, *precision* and *recall* of the predictive models created with them.

3.1 Feature Selection Results

In this subsection, the results of the experiments are shown to analyze the most relevant FS methods. Table 3 shows the numeric results of the methods and Fig. 1 graphically illustrates these results.

As a consequence of the experiments, methods for numeric attributes agree on the relevance of the features. The three methods (*Pearson's correlation*, *Kendall's correlation coefficient* and *Spearman's correlation coefficient*), show the same order of relevance of the features, which are: (1) *SuccessRate*, (2) *AverageScore*, (3) *CreditsEnrolled* and (4) *Age*. These methods even coincide in their relevance, as we can observe in Fig. 1. A subset of each method with the three most relevant

Table 3. Feature selection methods results

Feature/Method	Spearman correlation	Kendall correlation	Pearson correlation	Chi squared	Mutual information
SuccessRate	0.689132	0.579855	0.690387	2695.3404	0.290957
AverageScore	0.651426	0.539055	0.632768	2299.6431	0.257800
CreditsEnrolled	0.610896	0.521004	0.593288	2255.6441	0.240281
Age	0.128957	0.113682	0.104909	99.6425	0.009077
BirthProvince	-	-	-	55.4150	0.004235
Nationality1	-	-	-	52.7556	0.003316
UniversityAccessType	-	-	-	47.8643	0.004271
Degree	-	-	-	993.0825	0.095093
DegreeField	-	-	-	570.0683	0.055430
Faculty	-	-	-	897.7714	0.086082
DoubleNationality	-	-	-	0.2340	0.000022

features is selected to build a predictive model. These features are: *SuccessRate*, *AverageScore*, *CreditsEnrolled*.

In the same way, FS methods for all data types features (*Mutual Information*, *Chi Squares*) show similar results. In this case, given the existence of eleven features, we consider that a subset of the eight most relevant features is enough to test if the selected features are suitable to create an accurate predictive model. As it can be observed in Table 3, the eight most relevant features of the Mutual Information method are SuccessRate, *AverageScore*, *CreditsEnrolled*, *Degree*, *Faculty*, *DegreeField*, *Age* and *BirthProvince*; and the eight most relevant

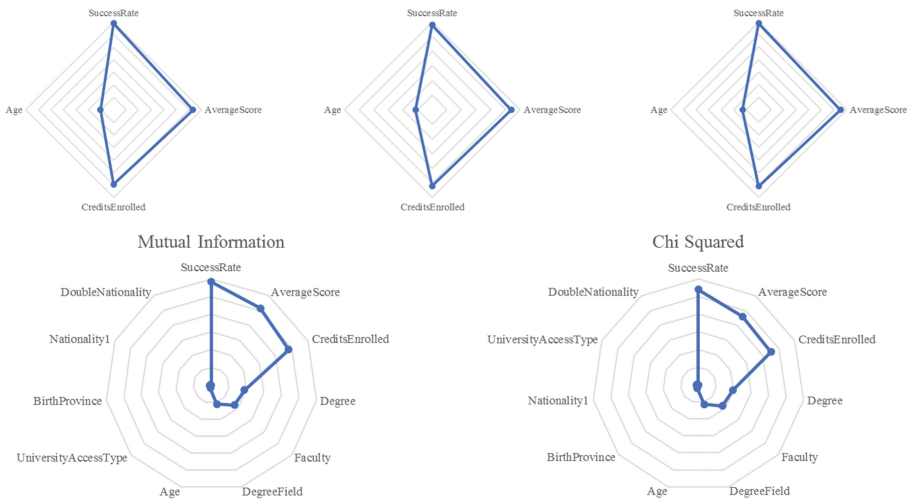

Fig. 1. Graphic illustration of the feature selection methods results.

features of the *Chi Squared* method are *SuccessRate*, *AverageScore*, *CreditsEnrolled*, *Degree*, *Faculty*, *DegreeField*, *Age* and *UniversityAccessType*. The only difference is the eighth most relevant feature that in the case of the *Mutual Information* method is *BirthProvince* and in the case of the*Chi Squared* method is *UniversityAccessType*.

3.2 Predictive Model Results

We use the *Decision Forest* algorithm to build the predictive models used to test the effectiveness of the FS methods commented above.

The *decision forest* algorithm creates several decision trees and votes the most popular output of them. The implementation used in this paper does not directly count the output of them, but it takes the sum of the normalized frequency of each output in each tree to get the label with more "probability" as it can be seen in the next formula:

$$f = \frac{1}{T} \sum_{t=1}^{T} f_t(x); \tag{8}$$

where T is the total number of trees and $f_t(x)$ is the probability of class x in the t tree.

The way in which the algorithms can be parameterized may have a great influence on their behavior. For that reason, all the experiment has been conducted with the same configuration of the *Decision Forest* algorithm. The setup of the algorithm is shown in Table 4.

Table 4. Decision forest algorithm configuration

Parameter	Value
Number of decision trees	8
Maximum depth of the decision trees	32
Number of random splits per node	128
Minimum number of samples per leaf node	1
Resampling method	Bagging

The results of the predictive models created with the algorithm for each FS method are shown in Table 5 and graphically illustrated in Fig. 2. The metrics used are *Accuracy*: ratio of correctly predicted observations; *Precision*: ratio of correct positive observations [True Positives/(True Positives + False Positives)]; and *Recall*: ratio of correctly predicted positive events [True Positives/(True Positives + False Negatives].

The model created with all the features get the highest accuracy as well as precision and recall, but the difference with the subset of features obtained

Table 5. Decision forest algorithm configuration results

Method	Accuracy	Precision	Recall
All features	0.821	0.815406977	0.826277372
Spearman correlation features subset	0.792	0.786647315	0.798245614
Kendall correlation features subset	0.792	0.786647315	0.798245614
Pearson correlation features subset	0.792	0.786647315	0.798245614
Mutual Information features subset	0.814	0.810218978	0.818313953
Chi Squared features subset	0.816	0.810144928	0.822840410

applying FS techniques is small. *Mutual Information* and *Chi Squared* methods, that have selected subsets with eight features (seven of them coincide), have close results between them (no far from the original dataset), being the *Chi Squared* method a bit better than the*Mutual Information* method. It is interesting to observe that the *Pearson's correlation*, *Kendall's correlation coefficient* and *Spearman's correlation coefficient* methods, which work with numerical features, selecting subsets of just three features (*SuccessRate*, *AverageScore*, *CreditsEnrolled*) have good results considering the reduced number of features used.

Figure 2 graphically shows the same results. The y-axis displays the percentage of *accuracy*, *precision* and *recall*, and it can be observed the small difference between the different methods and how close they are to get the results of creating a model using *all the features*. It highlights the reason why it is important to

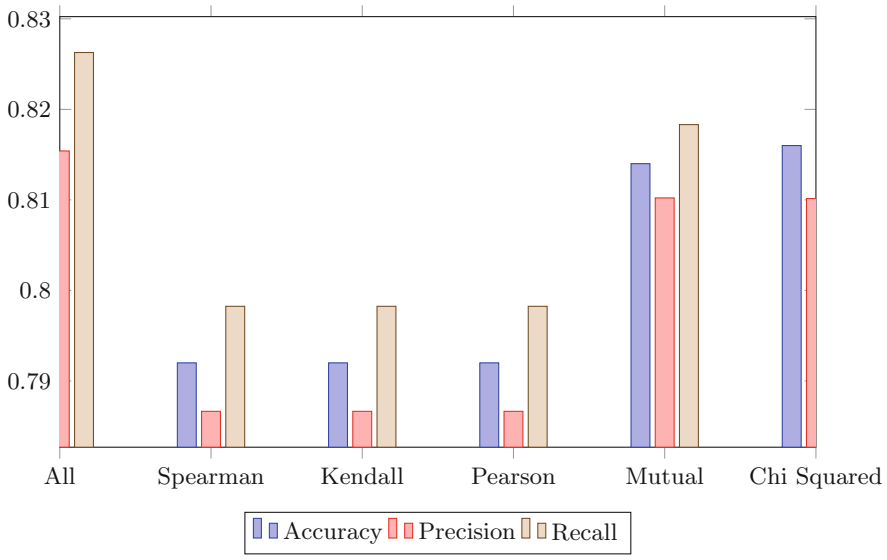

Fig. 2. Decision forest algorithm configuration results

use feature selection methods: they are capable of discovering the features with more relevance, extract them and create predictive models easy-to-explain to experts in short training times. Figure 2 shows in the same graphic the three effectiveness measure we use to compare (*accuracy*, *precision* and *recall*), they are not related to each other, but seeing these results together help to understand how FS methods proportionately affect all of them.

4 Conclusions and Future Works

In this paper, we consider the problem of analyzing FS methods. We aim to discover the best method to build simpler models discarding irrelevant or redundant features, reducing overfitting and with shorter training times. A model based on *Decision Forest* algorithms has been created for each FS method applied to a real-world dataset with academic data of the first year in college of students of the University of Almeria, Spain. The aim of the model is to predict if a student will graduate after finishing the first year of college studies.

The results obtained from the conducted experiments conclude that it is very important to apply FS methods because very good results can be obtained with a subset of features and simpler models easy to explain to field experts can be built. However, the importance of which FS method to use is relative since they all thrown similar results.

In the case of the experimental study we have conducted over the student dataset, the Chi-Squared method gets the better results *(Accuracy = 0.816*, *Precision = 0.810144928*, *Recall = 0.822840410*) with a small significant difference that the model created using all the features (*Accuracy = 0.821*, *Precision = 0.815406977*, *Recall = 0.826277372)*. The feature selection methods that accept only numeric features have the same results because they select the same features. It is patent that feature selection methods are important and it can be seen that the difference between creating a model with all the features and creating a model with the worst methods applied in this experiment (*Spearman*, *Kendall* and *Pearson*) is very small (*Accuracy = 0.029*, *Precision = 0.028759662*, *Recall = 0.028031758*). The reason why this occurs is because in the dataset (and this is frequent), only a small feature subset concentrates the significance to construct the predictive model.

As future work, it would be interesting to compare the obtained results of predictive models created with decision forest algorithm to other widely extended algorithms such as *SVM*, *Neural Networks* or other *Classification Trees*. We intend to use all these algorithms in our current work that consists in create an accurate decision model to predict if a student at the University of Almería (Spain) will graduate after finishing the first year of college. In the same way, it would also be interesting to develop a tool to determine which subset is the best to build the predictive model by comparing all FS techniques and configurations transparently. Thus, given a dataset, it would make it easier to find its optimal features subset.

Acknowledgement. This work has been funded by the EU ERDF and the Spanish Ministry of Economy and Competitiveness (MINECO) under Projects TIN2013-41576-R and TIN2017-83964-R. A.J. Fernández-García has been funded by a FPI Grant BES-2014-067974.

References

1. Campbell, M. (eds.): Statistics at square one, 9th edn. University of Southampton, Copyright BMJ Publishing Group (1997)
2. Chandrashekar, G., Sahin, F.: A survey on feature selection methods. Comput. Electrical Eng. **40**(1), 16–28 (2014). ISSN:0045-7906
3. Criado, J., Rodriguez-Gracia, D., Iribarne, L., Padilla, N.: Toward the adaptation of component-based architectures by model transformation: behind smart user interfaces. Softw. Pract. Experience **45**(12), 1677–1718 (2015). ISSN:0038-0644
4. Fernández-García, A.J., Iribarne, L., Corral, A., Wang, J.Z.: A Microservice-based Architecture for Enhancing the User Experience in Cross-device Distributed Mashup UIs with Multiple Forms of Interaction, Universal Access in the Information Society (2017). Special Issue on Distributed UIs: Distributing Interactions
5. Fernández-García, A.J., Iribarne, L., Corral, A., Wang, J.Z.: Evolving mashup interfaces using a distributed machine learning and model transformation methodology. In: Proceedings of On the Move to Meaningful Internet Systems: OTM 2015. International Workshop on Information Systems in Distributed Systems (ISDE). LNCS, vol. 9416, Rhodes, Greece, 26-30 October, pp 401–410. Springer, Cham (2015)
6. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd edn. Springer, New York (2009)
7. Lazar, C., et al.: A survey on filter techniques for feature selection in gene expression microarray analysis. IEEE/ACM Trans. Comput. Biol. Bioinf. **9**(4), 1106–1119 (2012)
8. Mantel, N.: Chi-square tests with one degree of freedom; extensions of the Mantel-Haenszel procedure. J. Am. Stat. Assoc. **58**(303), 690–700 (1963)
9. Microsoft Corporation: Microsoft Azure Machine Learning Studio. https://studio.azureml.net. Accessed 8 Aug 2017
10. Molina, L.C., Belanche. L., Nebot, A.: Feature selection algorithms: a survey and experimental evaluation. In: IEEE International Conference on Data Mining, 2002, Proceedings, pp. 306–313 (2002)
11. Salem, A., Jiliang, T., Huan, L.: Feature Selection for Clustering: A Review. In: Data Clustering: Algorithms and Applications. Chapman & Hall/CRC (2013). ISBN: 978-1-4665-8674-1. eBook, ISBN: 978-1-4665-8675-8
12. Sharp, T., Lengerich, R., Bai, S.: Online. STAT 509. Eberly College of Science. Penn State. https://onlinecourses.science.psu.edu/stat509/node/158. Accessed 8 Aug 2017