Data Science with Kaggle´s Competition "Don´t Get kicked!"

Article · May 2014

CITATIONS
0

READS
5,066

1 author:

Oswaldo Figueroa Domejean
Universidad Católica Boliviana "San Pablo"

5 PUBLICATIONS **0** CITATIONS

Data Science with Kaggle´s Competition “Don´t Get kicked!”

Oswaldo F. Domejean
Universidad Católica Boliviana “San Pablo”
La Paz, Bolivia
ofigue@ucb.edu.bo

Abstract

In this paper an analysis of the application of Machine learning models has been done to a dataset related to vehicle purchase made at auctions, this dataset was obtained from Kaggle competitions entitled "Do not Get Kicked!". The purpose of the study is to create a model to predict the vehicle purchase be in the best condition possible, hence reduce the risk of buying a car in poor condition. We used several Machine Learning models for analysis but one in particular gave a big leap in the level of accuracy.

Keywords: Machine Learning, Data Mining, Imbalanced Data, Feature Selection, Neural Nets, Support Vector Machines, Random Forest, Correlation, Over and Under – Sampling.

1. Introduction

This paper describe the analysis of an dataset called "Do not Get Kicked!", a past competition from Carvana published at www.kaggle.com, which refers to the considerations a buyer has to make in order to buy a used car that is in the best possible conditions. The buyer is an enterprise that purchases cars from auctions, to sell them to the clients it has.

The risk that exists when a company buys used vehicles at auctions is high, when the purchased vehicle is not in the best conditions it is considered a “kick”, for example in the case that vehicle has mechanical, electrical or any other problems not notice at the time of purchasing them.

In order to reduce the risk of acquiring vehicles that were not in good condition, we have this dataset which has lost of data related to different elements that can influence the conditions of vehicles that were not in good shape. All this data help us to make more accurate prediction of the elements to consider when buying a vehicle from an auction.

It’s been used the software Rapidminer®, a software platform that provides an integrated environment for machine learning, data mining, text mining, predictive analytics and business analytics [1].

In this paper a standard process called CRISP-DM (Cross Industry Standard Process for Data Mining) is used for every step of the process, from exploration to the insight descriptions. It’ll be used five of the steps CRISP-DM has, Business Understanding, Data Understanding, Data Preparation, Modeling and Evaluation Standard [2].

2. Business understanding

The competition site is:

http://www.kaggle.com/c/DontGetKicked,

it has a very specific description in relation to the risks involved in both the acquisition of companies that sell these vehicles and selling them to customers. There may be cases of altered odometers, hardly perceptible mechanical or electrical problems, which difficult proper identification of the status of the vehicles. The challenge in this competition is that dealerships could have some mean of predicting whether vehicles purchased from an auction doesn’t come out to be Kicks.

3. Data understanding

The competition dataset has 72.983 instances described with 32 features, the feature *IsBadBuy* that indicates whether a vehicle has the condition of “kicked”or not, Kicked in this case has the label "1" otherwise "0". It also has an identifier called *RefID*, the features are categorical, numeric and date type. The description of each feature is found in a file named “Data Dictionary” that is in the competition website. Here it’s been described the features from an analytical perspective.

The information age of vehicles has *PurchDate*, *VehicleAge* and *VehYear* as shown by the names of features. There is a relationship between the age of the vehicle related to the year of it and the year of purchase.

The features *Make*, *Model*, *Trim* and *SubModel* represent the vehicle’s brand characteristics, in this case it can be seen the variety of values those features have. The feature *WheelTypeID* is the Id of *WheelType*, in this case only *WheelType* will be used. It’s been seen that the features Color, Transmission, VehOdo (i.e. vehicle’s odometer), Nationality, Size, TopThreeAmericanName (e.g. CHRYSLER, FORD, etc.) are very specific don’t need further explanation.

In the case of the following features:

MMRAcquisitionAuctionAveragePrice,
MMRAcquisitionAuctionCleanPrice,
MMRAcquisitionRetailAveragePrice,,
MMRAcquisitonRetailCleanPrice,
MMRCurrentAuctionAveragePrice,
MMRCurrentAuctionCleanPrice,
MMRCurrentRetailAveragePrice y
MMRCurrentRetailCleanPrice.

I’s been found in the competition Forum post **“Request for more information on fields”** the description of them:

> *“The MMR price field are Manheim Market Report (MMR) prices for the specific vehicle. They are Manheim's best estimate of the market price for that specific vehicle. Keep in mind this is not the price that the buyer actually paid for the specific vehicle, just an index for where similar vehicles are usually priced.”*

The name of the features that includes "*Acquisition*" means the price of the vehicle MMR at which it was sold at auction. When the feature name includes "*Clean*" refers to the price of the vehicle in good condition therefore it should be higher than the MMR feature that includes the word “*Average*”.

When the feature name includes "*Auction*" refers to the expected price of the vehicle at the auction. When the feature name includes "*Retail*" refers to the expected price of the vehicle to which the customer is willing to pay at the dealership.

In the case of feature *AUCGUART, PRIMEUNIT* these are related because *PRIMEUNIT* identifies the level of demand with respect to a standard purchase and the feature *AUCGUART* identifies the risk that can be run with the vehicle, the later has three values Green, Yellow and Red, the value Green is less risky and Red the riskiest. These two features have around 98% as missing data, which limits their use.

In the case of the features *VNST* and *VNZIP1*, they have a very close relationship since the former is the Zip code and the later is the corresponding state in USA.

In the case of the feature *BYRNO*, it’s simply a code assigned to the purchaser of the vehicle. The feature *VehCost* is the price of the vehicle. The feature *IsOnlineSale* is described by itself. Finally *WarrantyCost* is the price of the warranty. The features listed in the data dictionary *AcquisitionType* and *KickDate* are not in the training dataset.

4. Data preparation

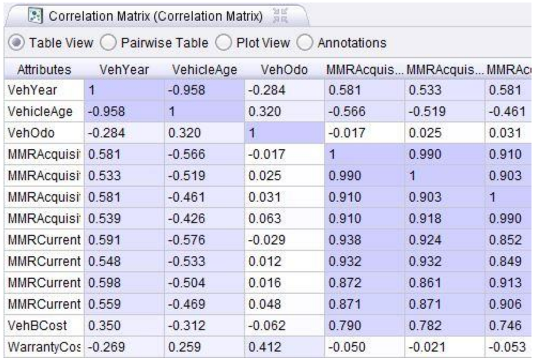
In this step the data was explored, analyzing the central tendency and dispersion of the features, the data has also been visualized to get some intuition about the characteristics of the features individually and between them. This dataset is imbalanced, in this case initially an Under-Sampling had been used, and then an Over-sampling technique was also used. It’s been made some processes related to feature selection in order to identify the most proper ones for this problem [3].

4.1 Data Exploration

The analysis had begun with the study of features individually, in which it was observed that the features as *VehYear*, *VehAge*, *VehOdo*, *IsOnlineSale* and *Nationality* and some others have a distribution mainly Gaussian in which no element of greater relevance was observed. In cases of features like *VehCost*, *WarrantyCost*, a right skewed distribution was observed, which indicates that these features usually have some outlier values. In the cases of the features *Auction*, *Color*, *size*, *Transmission* and *TopThreeAmericanName* nothing special is observed.

Among the features that have a very large range of values is *SubModel*, about 864 different values, in the case of *Model* has 1.063 values, *Trim* has 134 values.

It is observed that the *MMR ...* features have a right skewed distribution, which means the presence of outliers. It's very interesting to notice that there is a correlation between some numerical features, in the Figure 1 can it can be seen the high correlation between *MMR…* features and also with *VehAge* and *VehYear*, this characteristic is a justification for reducing use of the features is presented *MMR* ...



● Table View ○ Pairwise Table ○ Plot View ○ Annotations

Attributes	VehYear	VehicleAge	VehOdo	MMRAcquis...	MMRAcquis...	MMRAc
VehYear	1	-0.958	-0.284	0.581	0.533	0.581
VehicleAge	-0.958	1	0.320	-0.566	-0.519	-0.461
VehOdo	-0.284	0.320	1	-0.017	0.025	0.031
MMRAcquisi	0.581	-0.566	-0.017	1	0.990	0.910
MMRAcquisi	0.533	-0.519	0.025	0.990	1	0.903
MMRAcquisi	0.581	-0.461	0.031	0.910	0.903	1
MMRAcquisi	0.539	-0.426	0.063	0.910	0.918	0.990
MMRCurrent	0.591	-0.576	-0.029	0.938	0.924	0.852
MMRCurrent	0.548	-0.533	0.012	0.932	0.932	0.849
MMRCurrent	0.598	-0.504	0.016	0.872	0.861	0.913
MMRCurrent	0.559	-0.469	0.048	0.871	0.871	0.906
VehBCost	0.350	-0.312	-0.062	0.790	0.782	0.746
WarrantyCos	-0.269	0.259	0.412	-0.050	-0.021	-0.053

Figure 1: Correlation between numerical features.

4.2 Data transformation

With the results of the previous exploration and depending of the semantics of features some transformation had been done to get the most suitable ones for the analysis and prediction.

The feature *PurchDate* is of type date, initially it had been processed in order o have just year and month. It has been changed to categorical the features *IsBadBuy*, *VehYear* and *BYRNO*. The feature *WheelTypeID* has been removed because *WheelType* is its identification. It's been deleted the attributes PRIMEUNIT and AUCGUARD because only approximately 2% has data, the rest is missing.

The feature *VNZIP1* has been removed which is the ZIP code, and *VNST* has been kept because is more general feature that has the US state. The features that have a very large range of values *SubModel*, *Model* and *Trim* has been removed. Only the feature *Make* was kept

In the case of features that have missing values, *TRIM* had been replaced by the most common value that is "Bas". In the case of the missing feature values *TypeWheel* has been replaced by another most common that is "Alloy". The same in the case of *SubModel* it's been replaced by "4D SEDAN".

The feature *Color* has been replaced with the most common value "SILVER", also *Transmission* with value "AUTO", *Nationality* with value "AMERICAN". *Size* with "MEDIUM", *TopThreeAmericanName* with "GM".

4.3 Imbalanced data

The total number of instances in this dataset is 72.983, of which only the 8.976 have the label "1" corresponding to the vehicle that was kicked, which means bad purchase. This feature indicates that the dataset is unbalanced the most instances have labeled "0".

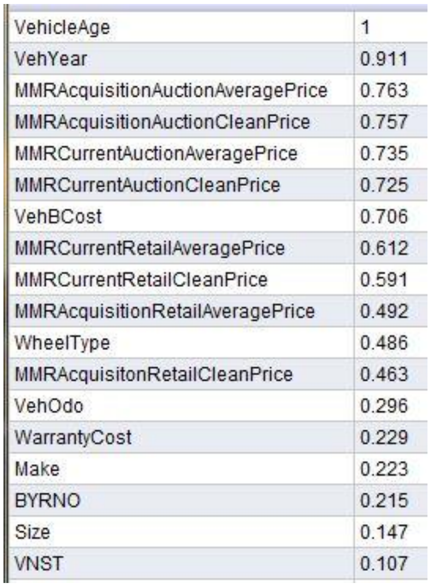
It has been used initially Under-sampling [4] as a technique to balance the dataset of positive and negative instances, it's been taken the dataset with majority label "0" and reduced the dataset in order to match with the minority label "1". This reduction in the number of instances reduced a considerable amount of information that is relevant to improve the fit of the prediction.

Por tal motivo se optó por la alternativa del Over-Sampling [4] como una via adecuada. Se ha creado un dataset que cuente con un número de ejemplos con la etiqueta "0" que sea mayor al dataset que cuenta con la etiqueta "1", para este último se utilizó el operador de Bootstrap para generar una muestra que cuente con el mismo número de ejemplos de la etiqueta "0". De esta forma se crearon diversos datasets balanceados que contenían el mismo número de ejemplos de ambas etiquetas. La alternativa del over-sampling brindó mejores resultados a nivel de ajuste en las predicciones.

For the mentioned reason we chose the alternative of Over-Sampling as a suitable route. It’s been created a sample that has a number of examples with the "0" label greater than the instances that has the label "1", for the latter the operator Bootstrap was used to generate a sample that has the same number of samples labeled "0" chosen before. Hence, diverse balanced datasets containing the same number of examples of both labels were created. The alternative of over-sampling provided better results in order to raise the accuracy of predictions.

4.4 Feature selection

In order to select the most relevant features for the prediction, “Chi-square” and “Information Grain” has been used to identify the relevance of the dataset features. In Figure 2 the result of Chi-square statistic is shown in order of relevance, the results were very similar in the case of “Information Grain”. It’s important to notice in Figure 2 that the features *MMR…* have predictive power but they are closely correlated, this situation have to be taken into account at the time of choosing the final set of features to run models.



VehicleAge	1
VehYear	0.911
MMRAcquisitionAuctionAveragePrice	0.763
MMRAcquisitionAuctionCleanPrice	0.757
MMRCurrentAuctionAveragePrice	0.735
MMRCurrentAuctionCleanPrice	0.725
VehBCost	0.706
MMRCurrentRetailAveragePrice	0.612
MMRCurrentRetailCleanPrice	0.591
MMRAcquisitionRetailAveragePrice	0.492
WheelType	0.486
MMRAcquisitonRetailCleanPrice	0.463
VehOdo	0.296
WarrantyCost	0.229
Make	0.223
BYRNO	0.215
Size	0.147
VNST	0.107

Figure 2: Feature selection using the operator Weight by Chi Squared Statistic

5. Modeling and evaluation [6]

The models that have been chosen to analyze the dataset are RandomForest, Neutral Nets and SVM with different training sets that vary in number of features and number of instances. The original dataset contains 72,983 training instances with 32 features.

In the case of RandomForest some parameters were changed in order to generate small trees, otherwise the trees were very large and difficult to interpret and in some cases also to fit in main memory. The dataset has a label proportion of positive to negative class around 1 to 7. RandomForest has been used with Under-Sampling that had a proportion of the positive to negative class around 1 to 1.5 generated a non-significant result since the model tends to classify the dominant label in this case "0" (vehicle not kicked).

There had been run various tries with different proportions going down gradually to 1 to 1.4, 1 to 1.3 and so on until the sample dataset was completely balanced at a ratio of 1 to 1, which generated a lower result in prediction accuracy but higher in predictive power for the minority class (vehicle kicked), however the results was as high as 58% accuracy which is not a great one, see Figure 3.

Then, it’s been tried to use Over-Sampling in which case a sample from the majority class was taken that had more instances than the minority class, and in order to balance the training dataset the minority class was increased using Bootstrap in order to get the same number of instances of the majority class.

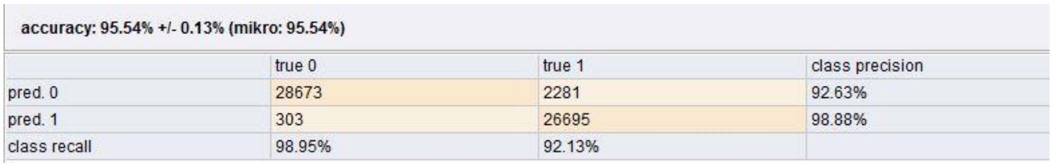
The next model used was Neural Nets, for which the Rapidminer® operator “Nominal to Numerical” was used in order to transform categorical features values in features that have the value "1" or "0" depending on the presence or not of the value, this process generated 137 features in the Dataset .

This process began with a dataset of around 25.000 instances which generated 65% of accuracy using Neural Nets, in this case with just one hidden layer in the model. The balanced dataset with Over-Sampling was increased up to 57,952 instances and it also has been increased the number of hidden layer until the model was run with three hidden layers which it worked out to 70% of accuracy, see Figuere 3.

Technique	Parameters	Run time	Accuracy	Balanced dataset
RandomForest	50 trees	3 hours 20 minutes	54.32	Under-samplig
RandomForest	35 trees	1 hours 40 minutes	58.45	Over-Sampling
Neural Net	One hidden layer	6 hours 30 minutes	65.3	Over-Sampling
Neural Net	Two hidden layer	11 hours 20 minutes	68.65	Over-Sampling
Neural Net	Three hidden layers	15 hours 10 minutes	70.62	Over-Sampling
SVM	Kernel Type: dot	5 hours 30 minutes	63.71	Over-Sampling
SVM	Kernel Type: Radial	4 hours 9 minutes	**95.54**	Over-Sampling

Figure 3: Various model runs with some additional information

Then, SVM was used with a kernel type "dot" in which the accuracy was reduced to 63% comparing to Neural Nets. But when SVM was used with kernel type "Radial" the increase in accuracy was huge because it rose up to 95.54%, see Figure 4.



accuracy: 95.54% +/- 0.13% (mikro: 95.54%)

	true 0	true 1	class precision
pred. 0	28673	2281	92.63%
pred. 1	303	26695	98.88%
class recall	98.95%	92.13%	

Figure 4: The confusion matrix of SVM with kernel "Radial".

The difference in accuracy shows a very suitable predictive model is the radial type of the SVM. That fact is confirmed with the AUC curve that reached 96.5%, see Figure 5, This Outcome tells us a lot about the distribution of the dataset and the difference with the other models used.

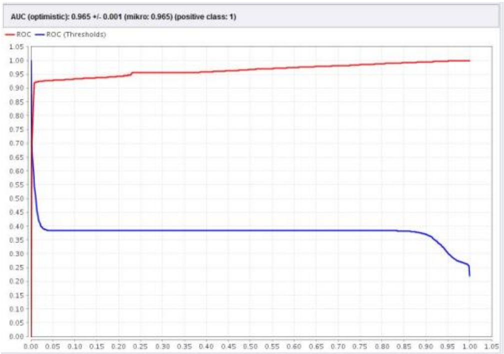


Figure 5: The AUC of SVM with kernel "Radial".

6. Conclusions

In this study we have used several models from the field of Machine Learning, with remarkably different results that were getting better gradually until very high levels of accuracy had been reached. At the beginning of the model use some samples were generated using the Under-Sampling method, which did not reached very encouraging results in terms of accuracy; the alternative was the use of Over-Sampling method resulting in better performance in terms of models accuracy.

Initially the model RandomForest was used with default parameters, which generated very large trees and then the parameters were changed to get a reduction in the size of the trees, which allowed increasing the number of trees. With these settings the result of accuracy improved but still remained low. Then the application of Neural Nets resulted in a substantial increase in accuracy when we raise the number of hidden layers from just one gradually to three, for which about 70% of accuracy was reached.

The SVM model did not perform well when "dot" was used as the type of kernel, which resulted in a lower accuracy compared to Neural Nets. But when the kernel was changed to "Radial", the leap was huge reaching a level of accuracy as far as 95%, the difference was too high which suggested that the resulting model was overfitted. One element of relevance was the number of hours it took the execution of the models, from about an hour to fifteen hours, even using parallelism in RapidMiner ®, with an i5 processor with 8GB of RAM.

The variability in the results obtained suggests that further study will most likely be required in terms of selection of features in order to identify the most suitable ones according to a deeper understanding of the distributions of the features.

7. Acknowledgements

I want to thank Carvana and Kaggle for providing the dataset.; and also Rapidminer for this amazing software.

References

[1] Rapidminer®: Statistical Analysis, Data Mining and Predictive Analytics. Website:http://rapidminer.com/

[2] Shearer C., *The CRISP-DM model: the new blueprint for data mining*, J Data Warehousing (2000); 5:13—22

[3] Chisholm, A.: "Exploring Data with Rapidminer". Ed. Packt Publishing Ltd., 2013.

[4] Chawla, Nitesh V.: "Data Mining for Imbalanced Datasets: An Overview". University of Notre Dame. Indiana – USA.

[5] Guyon, I., Elisseeff, A.: "An Introduction to Variable and Feature Selection". Journal of Machine Learning Research". 2013.

[6] Jane, G., Witten, D., Hastie, T., Tibshirani, R.: "An Introduction to Statistical Learning". Ed. Springer. 2013.