Dataset description

Dataset describing 472 startups and their founders as well as information on the company status described a success or a failure. *The objective of the analysis is to identify and explain driving factors behind startup **success** or **failure**.*

Feature type	Description and count
Categorical features (object)	*Ordinal, Nominal features.* *Count = 74. Information about startups, such as companies name, description of company profile, industry of company and etc.*
Numerical features (float, int)	*Numerical information about startups.* *Count = 42*
Dependent class	*Success or failure.* *Count = 472*
N = 472 observation, Features = 116	

Determination of problem

Feature Selection is one of the core concepts in machine learning which hugely impacts the performance of model. The data features that uses in train machine learning models have a huge influence on the achieved performance. Irrelevant or partially relevant features can negatively impact on model performance. Feature selection and Data cleaning should be the first and most important step of model designing. There are many ways to select a function. The most common way is conducting hypothesis testing on feature significance or choosing features based on low p-value and VIF (Variance inflation factor) (This methods work well for specific algorithms such as linear regression). However, that methods can be time costly in case of large data and many features. In my analysis I will try to find important features through machine learning techniques.

Data preprocessing

Machine learning models can find patterns on numerical datas. For that reason dataset needs some preparation. Initially, a large number of categorical values were presented in the data. The biggest part of ordinal / nominal features had different number of categories. In this case it is not efficient using build in encoders since model can learn incorrect patterns (For example, in case with ordinal encoder). I separated nominal features with binary (yes/no) values and the rest (with more values) for conducting one hot encoding. To learn the right order in ordinal features I encode them by hand. Also, initially in dataset there were many NA's and unknown values. Since filling that values with mean or median of each feature were able to bring high bias in information, I decided to fill unknown values with frequentist datapoints of each feature (mode). Features that had many unique values had removed from dataset because conducting one hot encoding or dummy variables on that features can be costly from modelling hand point. Other features were converted to the desired format.

Methods

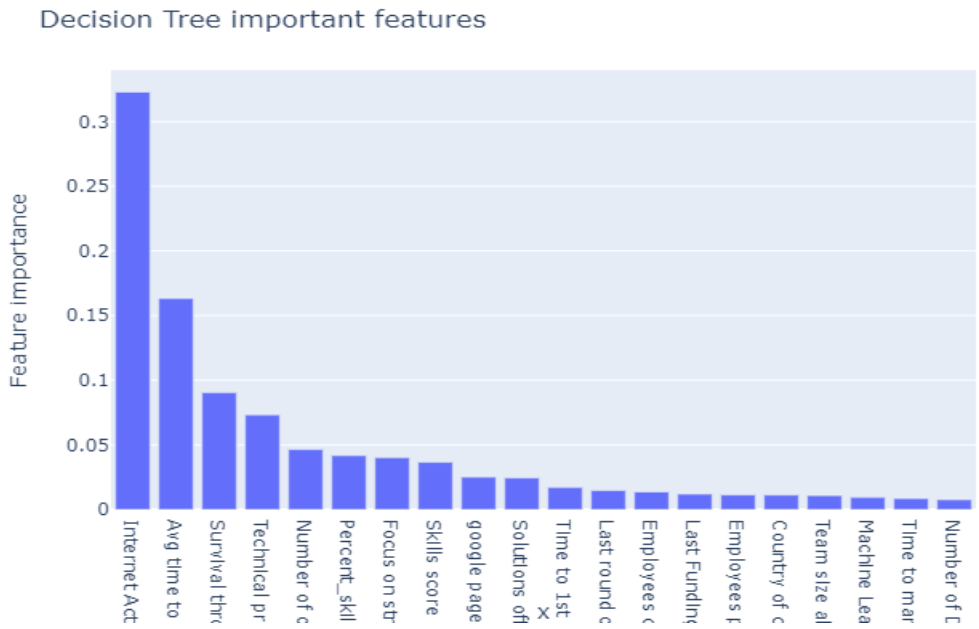
Decision Trees (DTs) is a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features (SciKit Learn explanation). The idea of tree is the next: Algorithm finds features on which it can split on nodes and get minimal impurity/entropy on that nodes. The most important feature maximally can explain output of target. Decision trees feature importance method orders features by importance. I divided dataset on train and test sets with 0.8/0.2 and train decision tree classifier with max_depth = 10 ((The maximum depth of the tree. If None, then nodes are expanded until all leaves are pure or until all leaves contain less than min_samples_split samples.) SciKit Learn explanation) and rest with default hyperparameters. Goal is to predict feature that maximally explain Success of startup.

After fitting I get the next result:

Accuracy and performance metrics	**Value**
Accuracy on cross validation with 3 fold	0.817058201058201
Precision_score	0.8464730290456431
Recall_score	0.864406779661017
Accuracy on test set	0.8947368421052632

Result is almost 81% on validation sets and 89% on test set. Test size is very small so in case of larger test set its accuracy can drop a bit. Algorithms top 10 (high quality graphs in code)

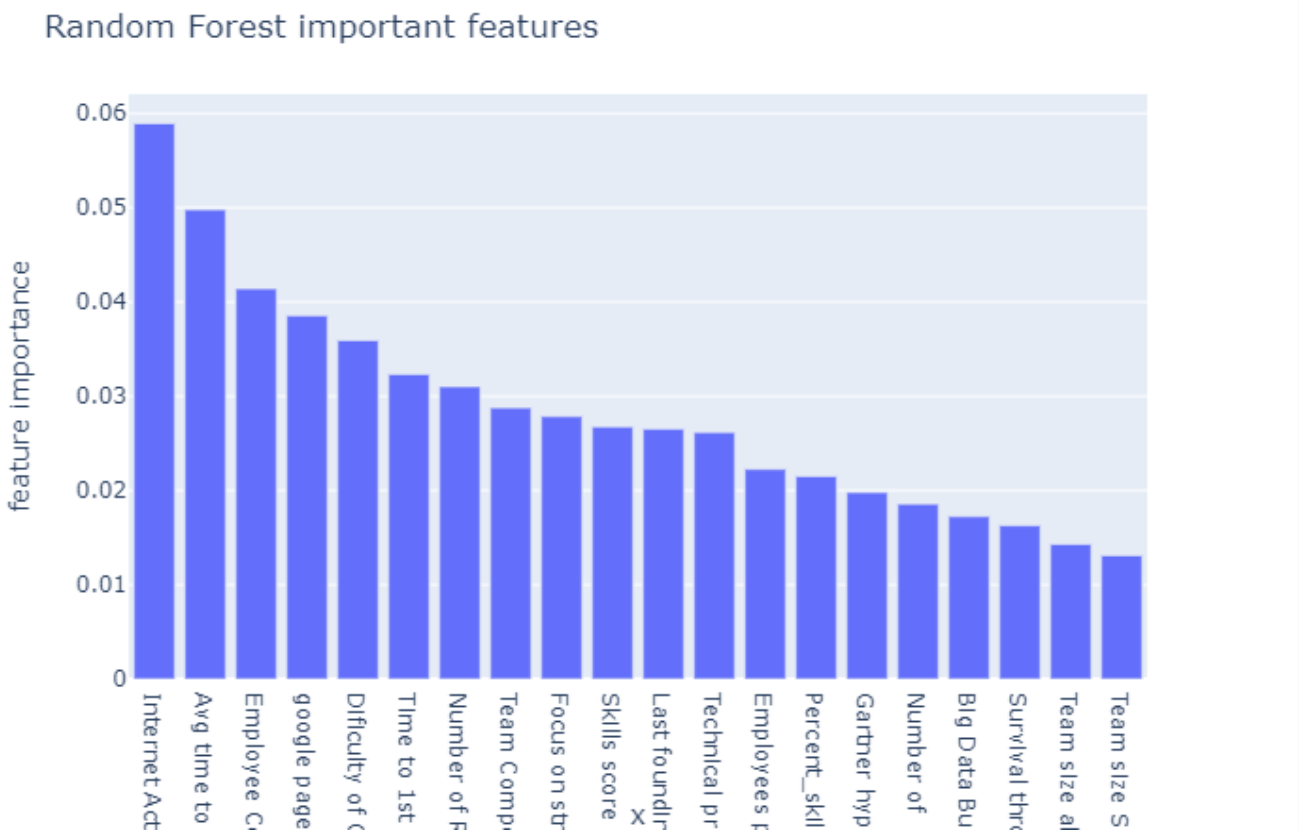
Important features are the next:



After, I tried to train random forest with different hyperparameters to see which features it will split. The result shows that during changing hyperparameters the main feature of decision tree doesn't change but are being changed only features with almost the same percentage share. For example random forest classifier with 500 trees and rest by default hyperparameters show the next results:

Accuracy and performance metrics	**Value**
Accuracy on cross validation with 3 fold	0.9071957671957671
Precision_score	0.8939393939393939
Recall_score	0.9711934156378601
Accuracy on test set	0.968421052631579

We see performance metrics and accuracy essentially improved. But the main goal of train Random Forest is to see whether after changing hyperparameters will be changed as well the features by which it splits trees. Top 20 Important features of Random Forest with default hyperparameters but 500 trees are the next (because of scale graph is low quality, you can find high quality graph in code)



We see that the main features did not change. Changed only features with low percentage share by decision tree importance.

To show effect of decision trees important features I trained logistic regression. To speed up training I bring dataset to the same scale (Normal(0, 1) distribution) with Standard Scaler and trained logistic regression classifier on all dataset.

After fitting I got the next results:

Accuracy and performance metrics	**Value**
Accuracy on cross validation with 3 fold	0.8753015873015872
Precision_score	0.8888888888888888
Recall_score	0.9218106995884774
Accuracy on test set	0.8526315789473684

After I tried to train logistic regression classifier only with top 10 important features by decision tree to understand, whether they can give better accuracy than training on all dataset.

Top 10 important features

Order	Feature	%
N1	**Internet Activity Score**	32
N2	**Avg time to investment - average across all rounds, measured from previous investment**	16
N3	**Survival through recession, based on existence of the company through recession times_0**	9
N4	**Technical proficiencies to analyse and interpret unstructured data**	7
N5	**Number of of repeat investors**	4.6
N6	**Percent_skill_Sales**	4.1
N7	**Focus on structured or unstructured data_no**	4
N8	**Skills score**	3
N9	**google page rank of company website**	2.5
N10	**Solutions offered**	2.4

So, after fitting model accuracy and performance metrics significantly improved.

Accuracy and performance metrics	Value
Accuracy on cross validation with 3 fold	0.8859259259259259
Precision_score	0.921875
Recall_score	0.9076923076923077
Accuracy on test set	0.8842105263157894

Because of small number of observations and minority class 'Success', precision and recall show close values. In case of large number of observations this difference will increase.

In summary performance metrics and all dataset and only top 10 important features are the next:

Accuracy and performance metrics	All dataset	TOP 10 feature
Accuracy on cross validation with 3 fold	0.8753015873015872	0.8859259259259259
Precision_score	0.8888888888888888	0.921875
Recall_score	0.9218106995884774	0.9076923076923077
Accuracy on test set	0.8526315789473684	0.8842105263157894

Conclusion

So, in this case the main factors influenced on startups success is internet activity and average time between investments. An active online campaign raises trust among potential investors and consumers. Companies trying to get attention through active Internet activities are generally becoming more successful. Another attractive factor for investors is the difference in time between investments. From the point of view of investors, if the difference is large, then the start-up is not attractive enough and success is unlikely, and vice versa, a small difference indicates interest from other investors and the potential success of the company. Also, financial data and the company's experience during recession raises confidence among potential investors and consumers of the product. This indicates quality management and demand product. Judging by the data, startups are in the field of data science and the company's skills in processing unstructured data increase the likelihood of a company's success. An important role is played by the company's sales skill. Qualified sales agents provide growth in sales and profits of the company. The main factors influencing the success of a startup are the earlier.