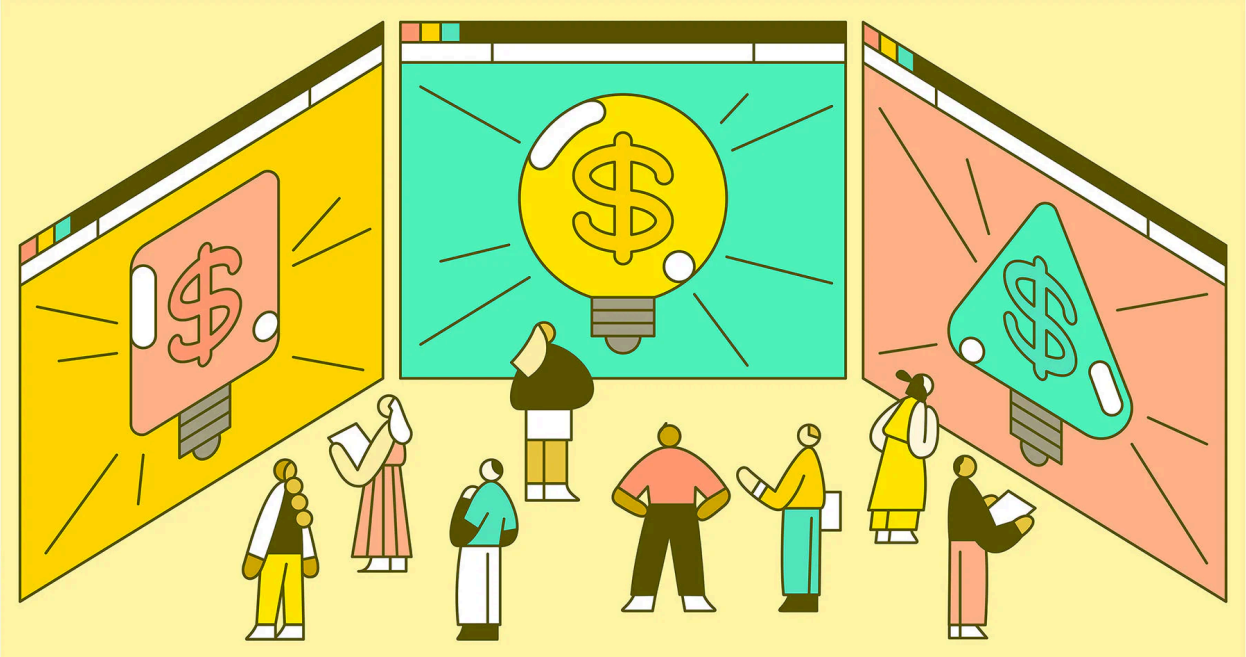
CROWDFUNDING SUCCESS ANALYSIS

The influence of the digital space on crowdfunding



Aleksandra Zografska

23.03.2025

Università degli Studi di Milano

INTRODUCTION

This paper takes a look at data from different crowdfunding platforms in Turkey. By analysing the different columns, I will try to make an inference on what influences success in crowdfunding projects. After variable and cluster analysis, I will build models that predict the success of a project, which is accomplished if the project met the funding target.

HYPOTHESIS

The main hypothesis is that the digital space: social media reach and websites, have a positive influence in a project's success.

FEATURE PREPROCESSING

The dataset consists of 1628 observations on 38 features. By doing a quick summary on

the dataset, I concluded that there are no missing values in the columns. Since the initial one is in Turkish, the first step is to clean it up by translating the features to English. The category variable in the dataset contained 16 different categories. In order to simplify, I reduced them to 4 more general categories: `culture_and_education, lifestyle, technology and other`. I made dummies for these categories to make it more readable in the model result output. All the other binary values were transformed into factors, with values 1 and 0. I engineered a dummy variable called `in_big_cities` by aggregating the big cities in Turkey in a list[2] and comparing the location variable in the dataset. In another dummy variable I extracted whether the owner is a female for value 1, and 0 for the owner being male or unknown. Taking a look at the `funding target` variable, I noticed 0 values. I removed them since they seem like a data entry error. For both numeric variables `num_social_followers` and `funding_target` I introduced variables of their log transformation. Since `num_social_followers` can be 0, I replaced that value with 1 to be able to compute the log transformation of this variable. To measure if the project even had social media followers I introduced a dummy called `have_followers`.

FEATURE EXPLORATION

The proportion of successful against failed projects is 376:1252. By plotting the numerical values, it's evident that there is no high correlation between them.

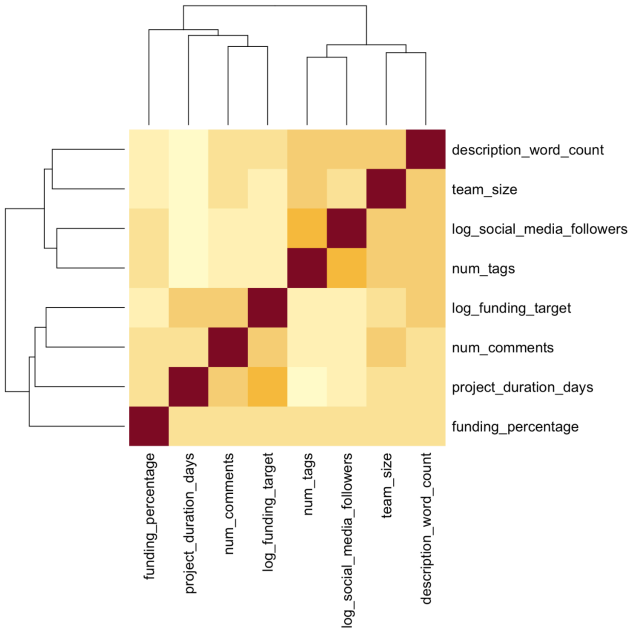


Figure 1. Correlation Heatmap

The next steps explore some properties of the numerical variables. The funding target doesn't look to have a normal distribution, according to the visual assessment of the normal QQ plot. In theory, normality would suggest a straight diagonal line. In this case it looks convex. The low p-value from the shapiro test on the `log_funding_target` further proves that we cannot assume normality for the variable. On the other hand, since we have a big amount of observations, we can rely on the CLT to assume normality.

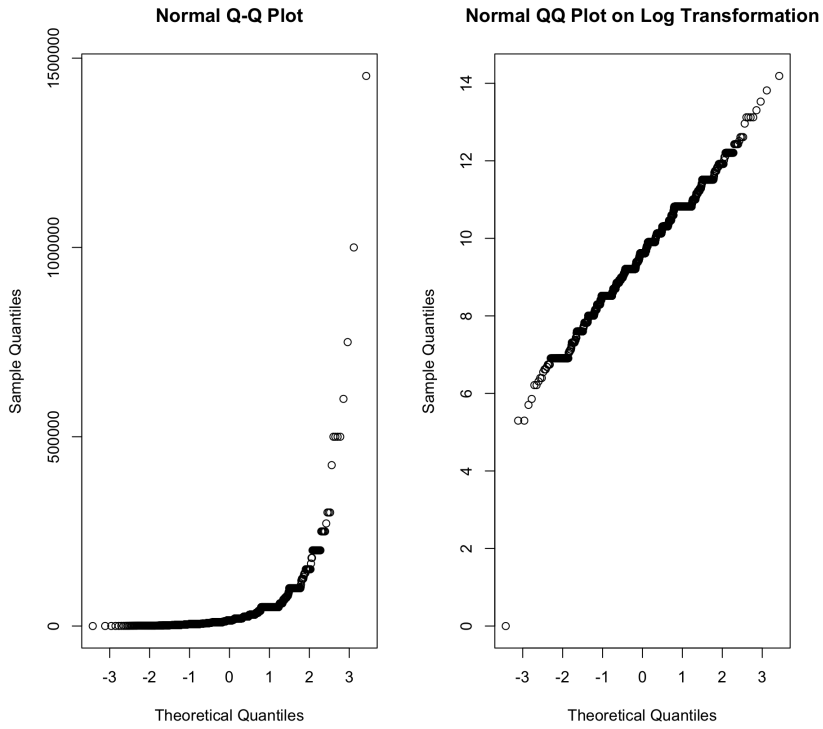


Figure 2. funding_target vs log_funding_target

In Figure3 the box plot gives insight on how the `log_funding_target` varies on `success_status`. To add on, the effect of cleaning up the data where the funding target is unrealistically 0, is really evident. Doing a test of the means of the log_funding_target when success_status differs, gives a `p-value = 0.05063`. We can say we reject the null hypothesis that the means are equal at the 5% significance level, furthermore this would suggest that the success status is associated with the funding target.

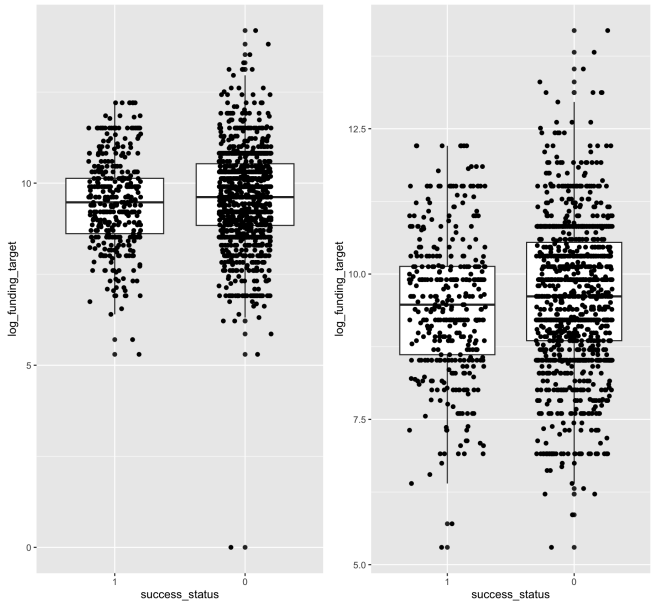


Figure 3. log_funding_target on success_status boxplots

Doing a similar analysis for the `log_social_media_followers`, we plot them in a histogram and notice that they are fairly normally distributed, when those with 0 followers are excluded. From the boxplot we can infer that there is high variation, and a noticeable difference in means. The same is proven by doing a t-test, in which we got the p-value of `2.2e-16`, hence rejecting the null hypothesis that the means of `log_social_media_followers` are the same by success_status.

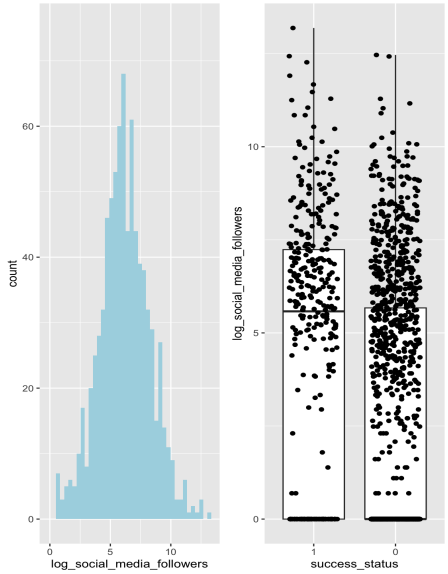


Figure 4. Log_social_media_followers histogram and boxplot

Another question I asked is if no followers variable can be explained by not having social media

```
> table(dataset$have_followers, dataset$has_social_media)

     1   0
1 825   0
0  10 792
```

Figure 5. Have_followers on has_social_media table

From the results in Figure 5, we can infer that that is true, since only 10 accounts that have social media don't have followers.

To assess the association of `success_status` and `has_social_media`, we need to look at a contingency table of the two variables, shown at Figure 5.

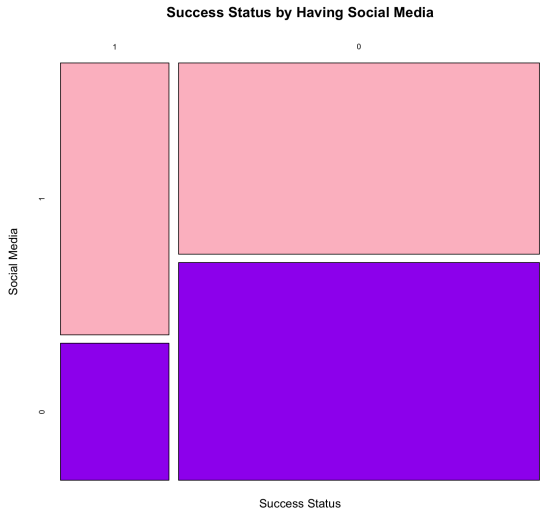


Figure 6. Mosaic plot: success_status and has_social_media

Doing a chi-square test gives `p-value = 2.894e-11` for which we can conclude significant association between the variables at the 5% significance level.

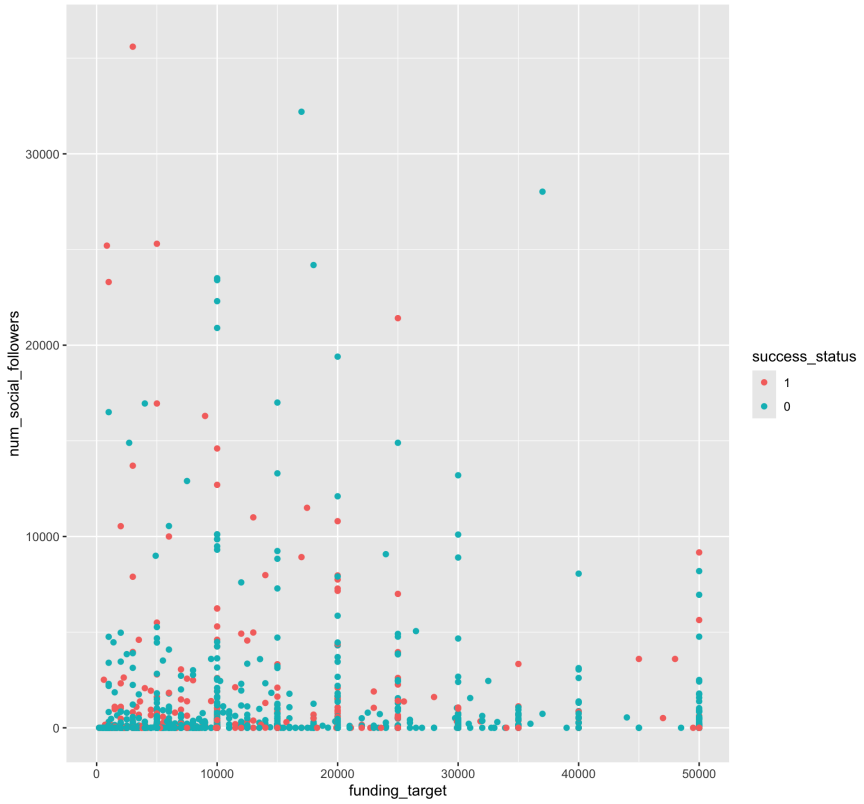


Figure 7. Funding_target against number of followers

The plot on Figure 7 suggests that the success varies between the combination of different social media followers and funding targets.

Doing a similar analysis for the `description_word_count`, we plot it on a histogram and notice that it looks like the distribution has a long tail on the right. From the boxplot we can infer that there is high variation, and a noticeable difference in means. The same is proven by doing a t-test, in which we got `p-value = 1.99e-09`, hence concluding that the mean of `description_word_count` differs for the two success values.

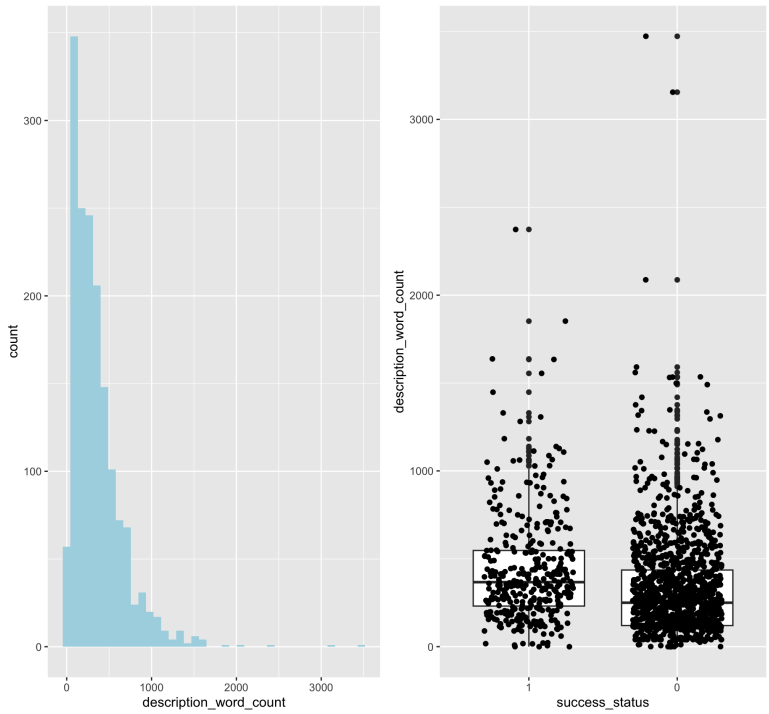


Figure 8. Description word count histogram and box plot on success

To measure the association between `in_big_city` and `success_status` we plot a mosaic plot and we do a chi-square test that proves association with `p-value=3.55e-06` hence significance of 0.05. From the plot, we conclude a positive relation between the variables, i.e. being in a big city improves the possibility of success.

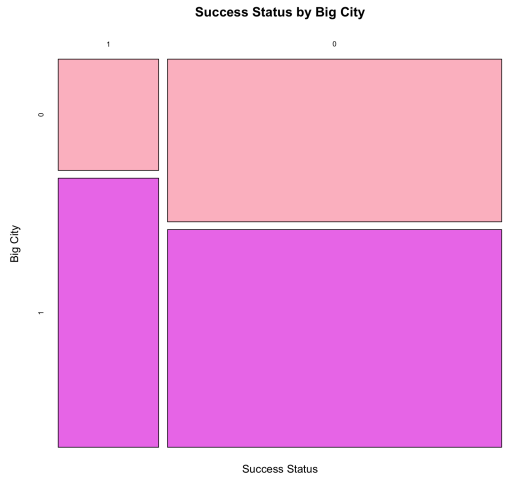


Figure 9. Mosaic plot: in_big_city and success_status

There is association between the success status and the generalized category seen on Figure 10, which is proven by the chi-square test, with `p-value = 3.831e-08`.

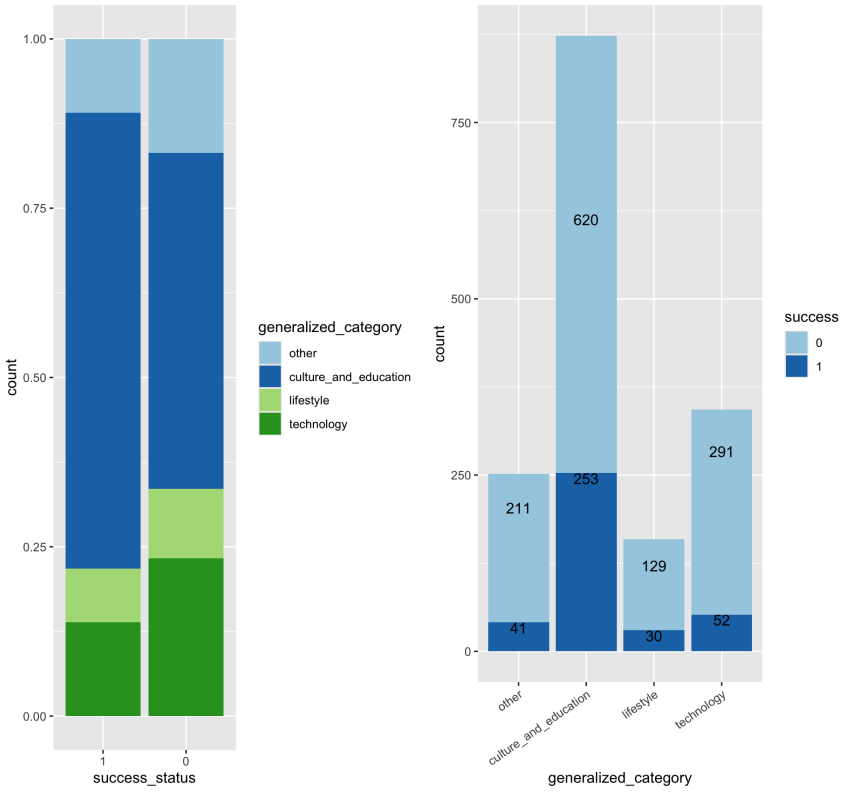


Figure 10. Boxplots for generalized category and success status

UNSUPERVISED LEARNING

To learn more about the data, before continuing on to the supervised learning, I experiment with the unsupervised approach.

FAMD

Factor Analysis for Mixed Data (FAMD) is an extension of the principal component analysis that works with data that is numerical and categorical, in comparison to PCA which only works with numerical. I applied the factor analysis on the whole dataset, and 19 variables. The initial results (Figure 11A) show pretty disappointing results, since not a lot of the variance is explained.

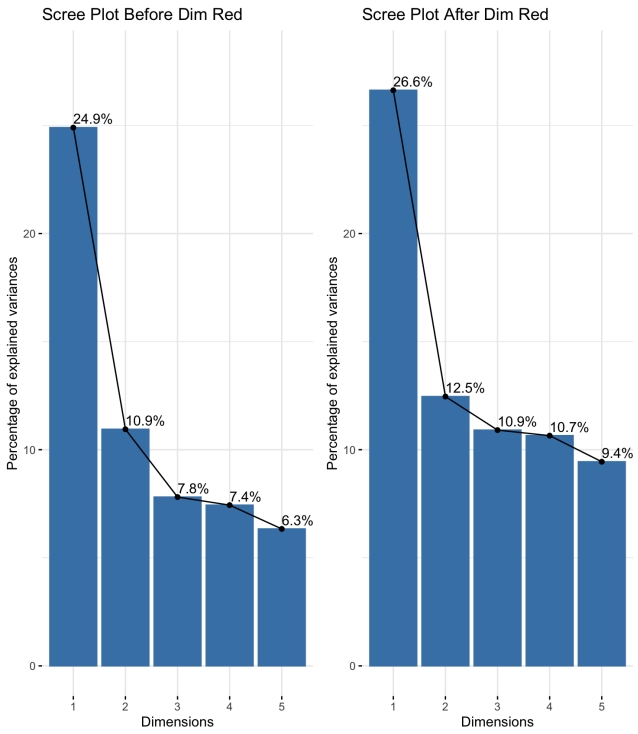


Figure 11A - left, 11B-right

This could be a side effect of the curse of dimensionality. In order to get better results, I removed the least contributing variables (Figure 12) that included: `description_word_count, promo_video_length, is_all_or_nothing, has_website, has_social_media, have_followers.`

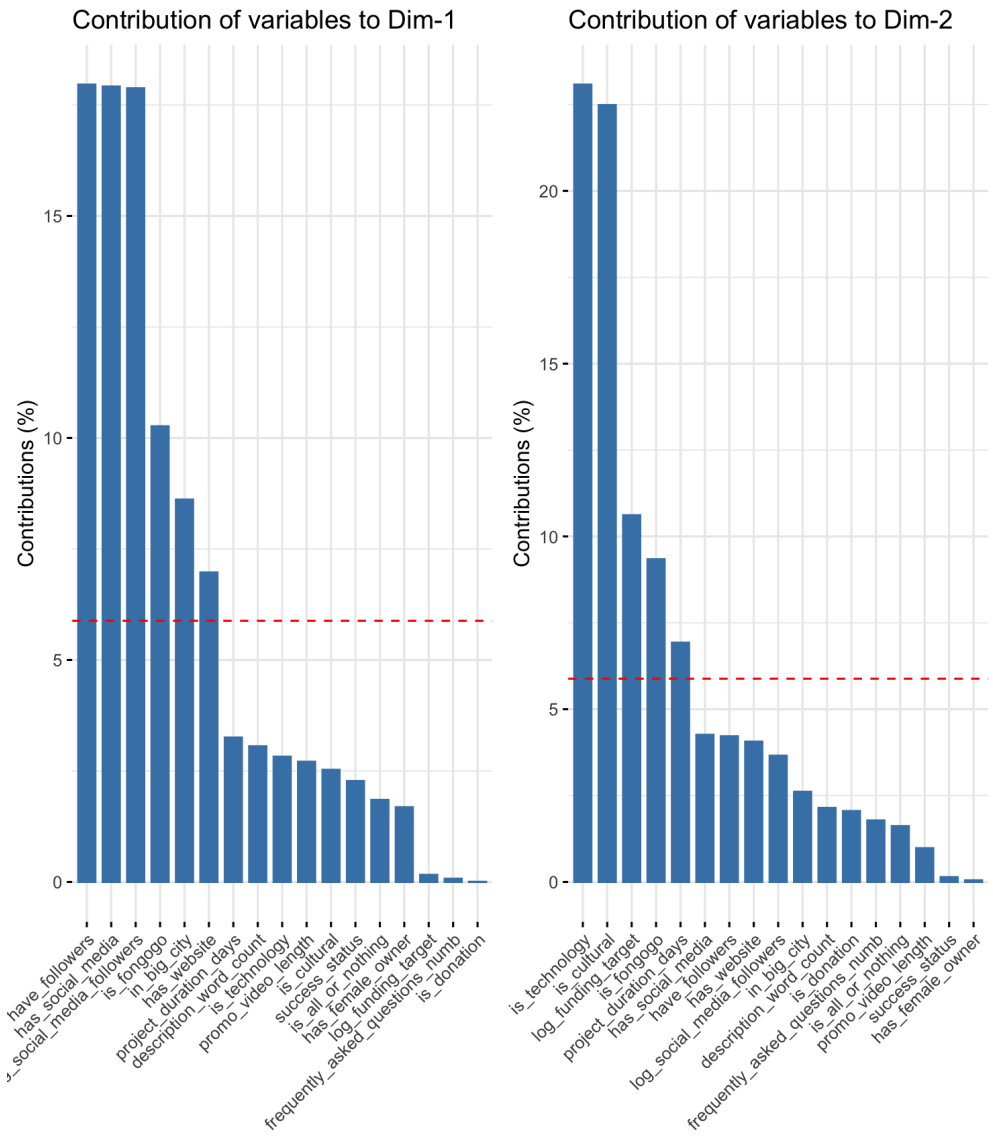


Figure 12. Plotted measures of variable contributions for Dim1 and Dim2

Even though there was a slight improvement, the model doesn't really explain the majority of the variance since the first two dimensions only explain `39,09%` of it (Figure 11B).

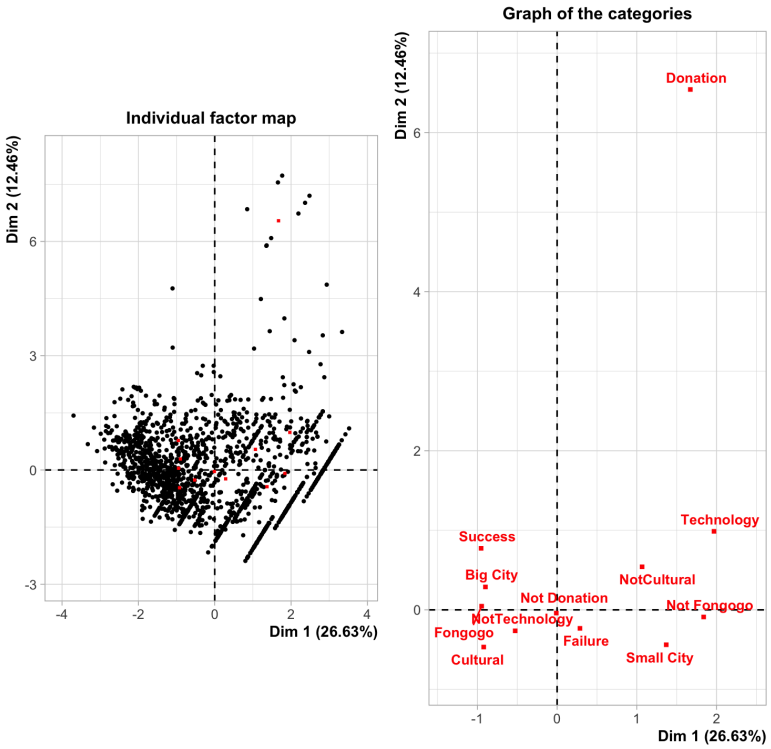


Figure 13. A: plotted observations in 2 dimensions. B: plotted qualitative variables

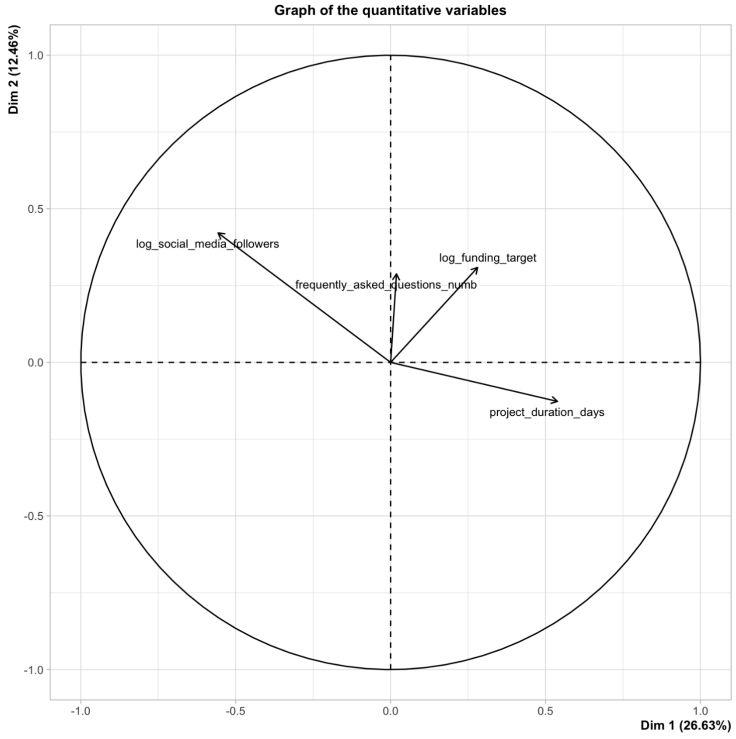


Figure 14. Correlation circle for quantitative variables

From the graph of the categories (Figure 13B), we can see that on the left side `Success, BigCity, Fongogo,Cultural and NotTechnology` are pretty close together which implies some sort of relationship between these variables, which also correspond with the increase in `social media followers` from the quantitative graph. Still, these conclusions are pretty vague, on account of the low variance explanation.

PAM with GOWER DISTANCE

Partitioning Around Medoids (PAM) is a clustering algorithm that works well with mixed data since it works based on pairwise similarities. In this case, I chose the gower distance metric to calculate them. I obtained a silhouette score, by running the PAM algorithm for different values of k in the range of 2 to 10. From the results (Figure 15) we can see that the best score is for 2 clusters, following 3 and after that there is a big jump to 4 and five.

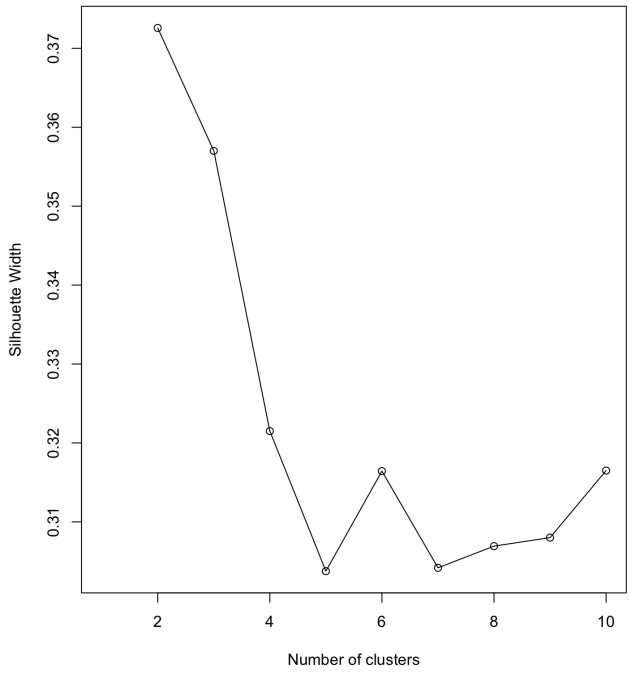


Figure 15. Silhouette plot

We notice that cluster 2 is most optimal since it has a higher silhouette score, which is a measure of how close the points are in the cluster. A quick summary of the clusters leads us to the analysis on Figure 16, which was generated, by first grouping the observations into the clusters, and applying the `summary()` function.

```
# first cluster: most of them were donations, most of them have social media, followers,
# and mostly in big cities, majority have a female owner
# project_duration_days mean: 48.11
# faq 0.2236
# log_social_media_followers 4.994
# log_funding_target: 9.476
# description_word_count: 416.5, high max 3473.0
# promo video length: 86.9
# 0.28% inner cluster success rate
# 0.73% of the successful observations are in this cluster

# second cluster: mostly not donations, no social media, low followers,
# mostly no website, mostly in small cities, by female owners
# project_duration_days mean: 57.55
# faq 0.26
# log_social_media_followers: 0.4761 - low social media folowers
# log_funding_target: 9.745
# description_word_count: 265.4
# promo video length: 40.21
# 0.14% success rate
# 0.27% of the successful observations are in this cluster
```

Figure 16. Summaries of cluster 1 and 2

From this summary we can conclude that the first cluster groups the majority of the successful observations, which also have a higher social media presence, lengthier promotional videos and descriptions as well as smaller project duration days than the second cluster. What is interesting is that there is not a big difference in the mean in the funding targets. Using the TSNE method a plot for the cluster is obtained, showing promising between-cluster separation, except for a small area in the center, where there is a small cluster circle of cluster 2 in the space of cluster 1. Also it's evident that still, the inner cluster variability is pretty high, since we have smaller cluster chunks fairly separated between each other. This might indicate that a bigger cluster number might be more appropriate.

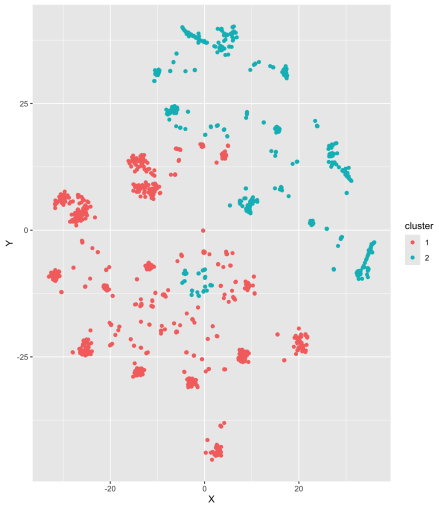


Figure 17. TSNE plot

CMDS

Classical Multidimensional Scaling (CMDS) aims at finding low-dimensional structure by preserving pairwise distances of data, in this case: the gower distance. We can use this method to see how a certain variable divides the observations based on binary data. Worth noting is that `has_social_media` and the two dummies from the category variable separate the data in the reduced dimension plot almost perfectly.

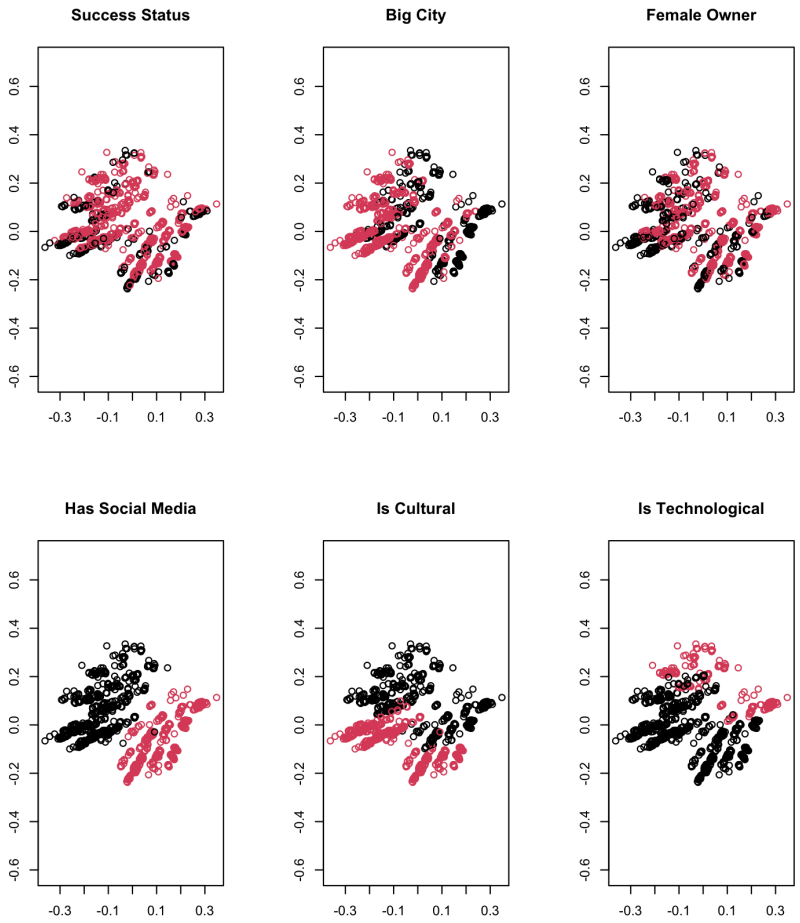


Figure 18. CMDS plots where data points are colored by corresponding dummy variable

SUPERVISED LEARNING

LOGISTIC REGRESSION

Since we're trying to predict a binary variable called `success_status` where 1 indicates success and 0 indicates failure for the regression, we need to do a logistic one. By running the `glm` function for the first time we can spot the relevant variables marked with the significance codes listed at the bottom.

```
glm(formula = success_status ~ ., family = binomial(link = "logit"),
    data = regression_variables)

Coefficients:
                                 Estimate Std. Error z value Pr(>|z|)
(Intercept)                    -2.3047041  0.8007524  -2.878 0.004000 **
is_donation1                    5.3397608  1.1360337   4.700 2.60e-06 ***
is_all_or_nothing1              1.6999260  0.3370930   5.043 4.59e-07 ***
project_duration_days          -0.0221228  0.0045433  -4.869 1.12e-06 ***
promo_video_length              0.0021051  0.0005659   3.720 0.000199 ***
frequently_asked_questions_numb 0.0063381  0.0430920   0.147 0.883066
has_website0                   -0.3427017  0.1600757  -2.141 0.032284 *
has_social_media0               0.7048638  1.0864598   0.649 0.516486
have_followers0                 0.5285058  1.1225351   0.471 0.637773
log_social_media_followers      0.2768320  0.0467790   5.918 3.26e-09 ***
description_word_count          0.0006173  0.0002147   2.876 0.004033 **
is_fongogo1                    -0.3810101  0.2064650  -1.845 0.064980 .
in_big_city1                    0.0007655  0.1708231   0.004 0.996425
has_female_owner1              -0.5850022  0.1371434  -4.266 1.99e-05 ***
log_funding_target             -0.0921504  0.0574078  -1.605 0.108453
is_cultural1                    0.7162207  0.2054718   3.486 0.000491 ***
is_technology1                  0.0700020  0.2541368   0.275 0.782971
is_lifestyle1                   0.0628206  0.2955866   0.213 0.831695
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 19. Glm regression on all variables

From here we can conclude there are 11 relevant variables for the model. This is still a big number, and determining the best ones for the prediction could be made easier with stepwise selection. To start off, I first divided the data into training (80% of the dataset) and test data. On the training data I used a method from the `MASS` package called `stepAIC` that performs the selection based on the best Akaike Information Criterion (AIC), which in short is a measure for a model's complexity and goodness of fit. Since I want to try out different combinations of 'forward', 'backward' and 'both' direction of the algorithm, and use different mechanisms to determine the cut-off value, I abstracted the fitting, prediction and confusion matrix inspection into separate methods. After the model fitting, I made predictions on the test data and analysed the confusion matrix. Comparing a 'both direction' feature selection on different cut-off techniques we get a

better result for sensitivity with the Euclidean index.

```
           Sensitivity Specificity Percision
Youden       0.6787149   0.8311688 0.9285714
Euclidian    0.7068273   0.7922078 0.9166667
```

Figure 20. Performance table on glms with different cutoff values

In the variance inflation (vif) analysis, there were two variables `has_social_media` and `have_followers` that had a high value > 4 and as a result I removed them from the model. The high vif could be a sign of multicollinearity between the variables, and removing them should improve the model. In this case it improved the specificity and precision as well as the area under the ROC curve

```
                 Sensitivity Specificity Percision
Youden             0.6787149   0.8311688 0.9285714
Cleaned Youden     0.7831325   0.7662338 0.9154930
Cleaned Euclidian  0.7831325   0.7662338 0.9154930
> # the sensitivity and precision improved for the cleaned model,
> # but not the specificity
>
> # compare auc
> auc(youden_predictions$roc_curve)
Area under the curve: 0.7996
> auc(cleaned_youden_predictions$roc_curve)
Area under the curve: 0.8119
```

Figure 21. Output of comparison of initial and cleaned models

Changing the step direction also changes the performance of the models (Figure 20), with backward selection giving the best results.

```
          Sensitivity Specificity Percision
Backward    0.6787149   0.8311688 0.9285714
Forward     0.7389558   0.7792208 0.9154229
Both        0.7831325   0.7662338 0.9154930
> # the both gives better results
>
> auc(cleaned_euclidean_predictions$roc_curve)
Area under the curve: 0.8119
> auc(forward_predictions$roc_curve)
Area under the curve: 0.8085
> auc(backward_predictions$roc_curve)
Area under the curve: 0.8119
```

Figure 22. Output of comparison of different stepwise selections

TREES

Another supervised learning method I tried was the decision trees. Running a classification tree on all the variables yielded a big tree with 58 inner nodes and 59 terminal nodes. To make it less complex, I decided to analyze the splits of the tree nodes based on the complexity parameter (cp).

```
Classification tree:
rpart(formula = success_status ~ ., data = data.train, cp = 0.005)

Variables actually used in tree construction:
 [1] description_word_count
 [2] frequently_asked_questions_numb
 [3] has_female_owner
 [4] is_all_or_nothing
 [5] is_cultural
 [6] is_other
 [7] is_technology
 [8] log_funding_target
 [9] log_social_media_followers
[10] project_duration_days
[11] promo_video_length

Root node error: 299/1301 = 0.22982

n= 1301

         CP nsplit rel error  xerror     xstd
1 0.0326087      0   1.00000 1.00000 0.050753
2 0.0234114      4   0.86957 0.98662 0.050513
3 0.0180602      5   0.84615 1.01003 0.050930
4 0.0167224     10   0.75585 0.96321 0.050083
5 0.0100334     15   0.66890 0.93980 0.049641
6 0.0083612     16   0.65886 0.90635 0.048988
7 0.0066890     20   0.62542 0.89632 0.048787
8 0.0050167     24   0.59532 0.89298 0.048720
```

Figure 23. Classification tree summary and cp table

From the results we conclude that a `cp=0.0109649` is a good value to choose for pruning the tree, i.e at 11 splits. Here the relative error stabilizes as well as the xerror which measures the cross-validation error that tells us how good the tree generalizes. The result from the prune is still a big tree, but it gives a more intuitive overview on how different variables influence the success outcome. It's valuable to note that the number of social media followers plays a role in the tree construction. The second node, when splitting `log_social_media_followers >= 4.8` the left child node has a bigger proportion of success variables than the right hand side. This is concluded even though both child nodes have 0 as a predicted class, the probability of 0 on the left side is higher than the other. Outputting the contribution score variables, it's evident that this variable has a high contribution score of `75.796623`.

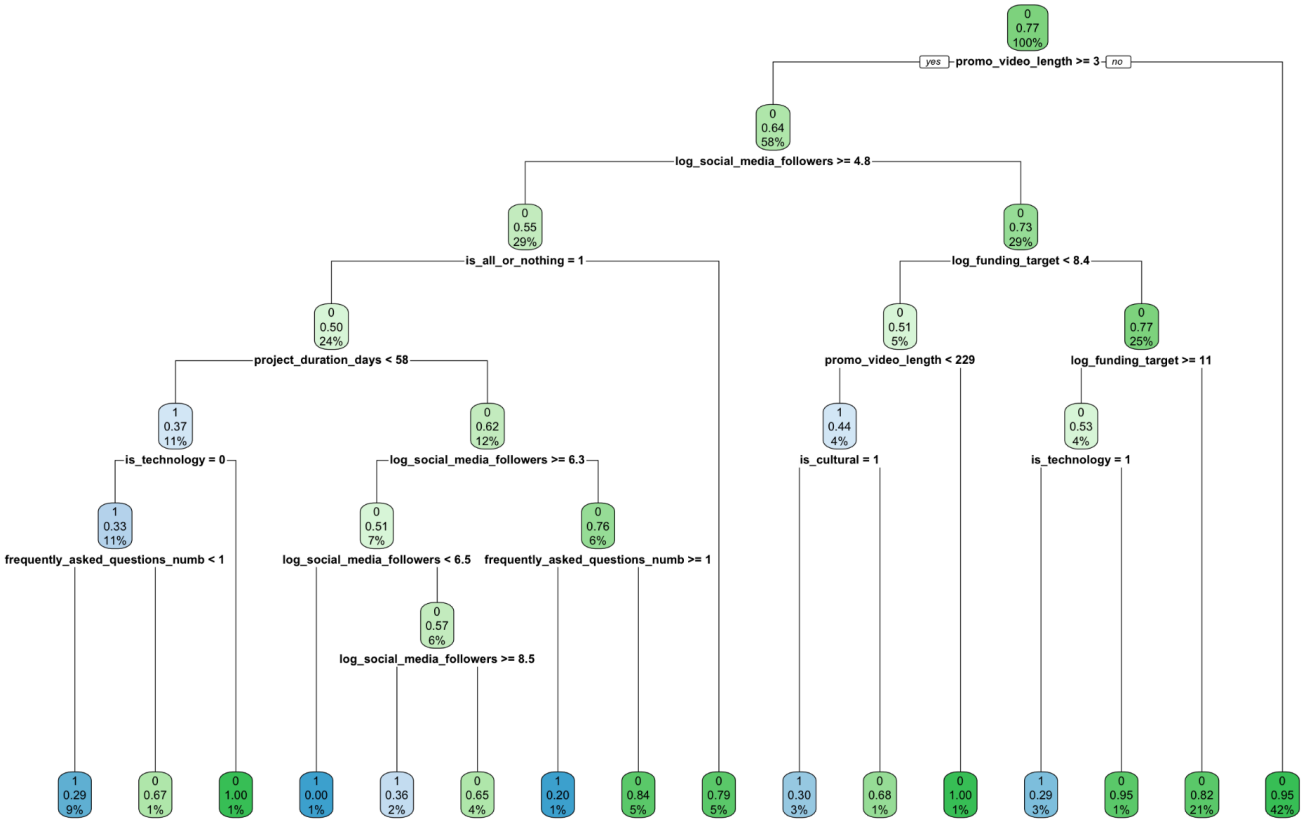


Figure 24. Final pruned classification tree

	Overall
description_word_count	51.838417
frequently_asked_questions_numb	20.882727
has_female_owner	7.033316
has_social_media	10.374774
has_website	6.703781
have_followers	17.854787
in_big_city	6.176999
is_all_or_nothing	9.213337
is_cultural	30.138381
is_fongogo	12.491394
is_other	4.496339
is_technology	24.367866
log_funding_target	44.289860
log_social_media_followers	64.353249
project_duration_days	42.903052
promo_video_length	81.334788
is_donation	0.000000
is_lifestyle	0.000000

Figure 25. Variable importance output

GLM VS TREE

Training and testing the models, using the same train and test data, resulted in GLM scoring better in sensitivity, while tree in specificity and precision.

	Sensitivity	Specificity	Percision
Tree	0.5324675	0.9036145	0.6307692
GLM	0.7662338	0.7831325	0.5221239

Figure 26. Tree VS GLM scores

RESULTS & CONCLUSION

By combining the results from the supervised and unsupervised methods we can conclude that social media, from the number of followers, as well as thorough description and promo videos, positively influence the success of a crowdfunding project. The argument is supported through the presence of those variables as relevant parts of the trained models.

REFERENCES & SOURCES

1. Dataset: https://archive.ics.uci.edu/dataset/1025/turkish+crowdfunding+startups
2. List of largest cities in Turkey
   https://en.wikipedia.org/wiki/List_of_largest_cities_and_towns_in_Turkey