Contents

1. Introduction

“You got yourself a deal!!!”

If this sounds familiar to you, you must have heard about “Shark Tank”. “Shark Tank” is an American business television series in which entrepreneurs pitch their companies, ideas, or products to investors. Originating in Japan, the show has become such a major television success that several international versions have been broadcast around the world. With thousands of applications each year, the show invokes an interesting question:

“Would my deal be successful on Shark Tank?” – A contestant might ask

The “Shark Tank Pitches” dataset might be able to provide some answers to this question. Collected from Shark Analytics, the dataset accumulates data from 6 seasons of Shark Tank, which consists of 122 episodes and 495 companies. This project will aim at developing a predictive model to estimate the probability of success of a pitch, as well as a model that allows producers to understand the casting region knowing the criteria they would like to look at.

2. Data description

The dataset contains 495 observations spanning 6 seasons of Shark Tank. The variables are described in the following parts.

2.1. Quantitative variables

The dataset contains 5 quantitative variables. For this project, the quantitative variables of episode and season will not be included in the model. The reasoning behind this is that this information would only be decided after the deal decision is made, which means that they bear no effect on the outcome the deal.

Each variable is individually explored to understand its distribution. The findings are summarized in the following table:

Variable	Minimum	Median	Mean	Maximum	Skewness
askedFor	10000	150000	258491.00	5e+06	Left
exchangeForStake	3	15	17.54	1e+02	Left
valuation	40000	100000	2165615.00	3e+07	Left

***Table 1**: Descriptive analytics of quantitative variables*

An analysis of the 3 quantitative variables reveals that 463 out of 495 companies asked for less than 500,000 USD from the investors and only 15 asked for more than 1,000,000 USD. In terms of the percentage of stake, 380 offered 20% or less of their company, and 105 offered between 20% and 40% of their companies. An outlier company, "My Cold Snap", whose product is a beverage holder that keeps drinks cold, asked for 50,000 USD in exchange for 100% of the company ownership. Since valuation is a function of askedFor and exchangeForStake, a similar trend applies to valuation. 448 out of 495 companies are valued at 5,000,000 or less. The company with the highest valuation at 3,000,000 USD is "SynDaver Labs", which makes synthetic body parts used in medical education.

I also ran a collinearity test for the quantitative variables and the results are displayed in the following matrix.

Correlation matrix

	askedFor	exchangeForStake	valuation
askedFor	1	-0.009	0.761
exchangeForStake	-0.009	1	-0.321
valuation	0.761	-0.321	1

***Table 2**: Correlation matrix of quantitative variables*

As evident from the matrix, valuation and askedFor are highly correlated. As a result, the valuation variable will be excluded from the model, as valuation is a function between askedFor and exchangeForStake.

2.2. Categorical variables

There are 19 categorical variables in the dataset. For the purpose of this research, the following descriptive variables will not be included: description, entrepreneurs, website, title, episode.season.

For deal and Multiple.Entreprenuers, they are dummified.

For the location variable, the state of each location is extracted. Subsequently, a new categorical variable called "region" is created. The region variable represents the 4 statistical regions decided by the United States Census Bureau: Northeast, Midwest, South, and West. The value of the region variable of each observation depends on the state of that observation. For instance, the state of California (CA) would belong to the West region.

For the category variable, there are 54 unique values. In order to reduce the number of unique values, I have filtered out those that have more than 24 observations. Those with fewer than 24 observations in the dataset were put into the general category Other. As a result, the predictor category now has 4 selections: Specialty Food, Novelties, Baby and Child Care, and Other.

For the 5 variables that indicate the name of the investors present in the episode – shark1, shark2, shark3, shark4, shark5, there are 10 unique sharks that have appeared on the show during the first 6 seasons. Each of the sharks would be assigned a binary variable with their name, with 1 indicating present in the episode and 0 indicating absent. As a result, 10 extra binary variables are created with the shark's name: barbara, lori, robert, kevin_o, steve, daymond, jeff, mark, kevin_h, john and nick.

3. Model selection and Methodology

3.1. Feature selection

In order to select which variables to include in my model, I used **Random Forest** in order to understand the importance of each variable. The model will run 500 trees on 500 bootstrapped models and the result is as follows:

Variable importance

	FALSE	TRUE	MeanDecreaseAccuracy	MeanDecreaseGini
category	6.888	4.813	8.185	14.022
askedFor	4.252	0.549	3.533	38.297
exchangeForStake	-2.666	6.242	2.633	25.358
Multiple.Entreprenuers	1.343	5.008	4.631	7.054
region	-1.360	1.962	0.460	16.901
barbara	-2.305	-1.662	-3.457	3.240
lori	-2.020	1.513	-0.380	3.734
robert	-0.340	2.465	1.645	1.776
kevin_o	-2.783	-1.369	-3.042	2.276
steve	-0.082	-4.452	-3.157	0.618
daymond	-1.276	-0.407	-1.177	2.083
jeff	-2.927	0.655	-1.432	1.056
mark	-0.885	3.205	1.865	2.737
kevin_h	0.293	2.385	2.202	2.636
john	-3.185	-0.088	-2.388	0.598
nick	1.458	4.764	4.713	1.289

***Table 3**: Variable importance for all predictors*

Evident from the table, there are several variables with a negative Mean Decrease Accuracy. Hence, I decided to exclude these variables from the predictive model. As a result, the final model will include

the following variables: *category, askedFor, exchangeForStake, Multiple.Entreprenuers, region, lori, robert, mark, kevin_h, nick*

3.2. Model selection

Since the purpose of this model is to allow entrepreneurs to predict whether their project would succeed or fail, I decided to choose a classification model that gives them the result when they input the parameters. As a result, the random forest model was utilized. The reason behind the selection of this model is that with limited time and resources to predict whether their project will succeed or fail, the random forest model provides a "True" or "False" response instead of a probability, allowing the entrepreneur to make crucial decisions on their businesses.

The shark_tank dataset is divided into the training and test set, with a ratio of 70% training set and 30% test set. The subsequent models will be trained using the training set and tested using data in the test set.

In order to optimize the accuracy score of the model, I ran several loops to determine the optimal mtry, ntree and maxnodes. In the end, the best random forest model is tat with mtry=1, ntree = 1000 and maxnodes = 20.

Aiming to increase the accuracy score of the random forest model, I ran the boosted model and using parameter tuning to identify the n.trees and interaction.depth with the highest accuracy score. The loop shows that the best accuracy score happens when n.trees = 2500 and interaction.depth = 2.

Running boosted model with the optimal hyperparameters, it was identified that 2 variables Robert and nick have extremely close to 0 importance. As a result, these 2 variables are eliminated from the model.

4. Results

4.1. Final Model

The final prediction model is:

$$\boldsymbol{gbm}(\boldsymbol{formula} = \boldsymbol{deal}\sim\boldsymbol{category} + \boldsymbol{askedFor} + \boldsymbol{exchangeForStake} + \boldsymbol{Multiple.Entreprenuers} + \boldsymbol{region} + \boldsymbol{mark} + \boldsymbol{kevin_h}, \boldsymbol{distribution} = \text{bernoulli}, \boldsymbol{n.trees} = \mathbf{2500}, \boldsymbol{interaction.depth} = \mathbf{2}, \boldsymbol{cv} = \mathbf{10}, \boldsymbol{n.cores} = \boldsymbol{NULL}, \boldsymbol{verbose} = \boldsymbol{FALSE})$$

The confusion matrix when the model is applied to the test dataset is as follows:

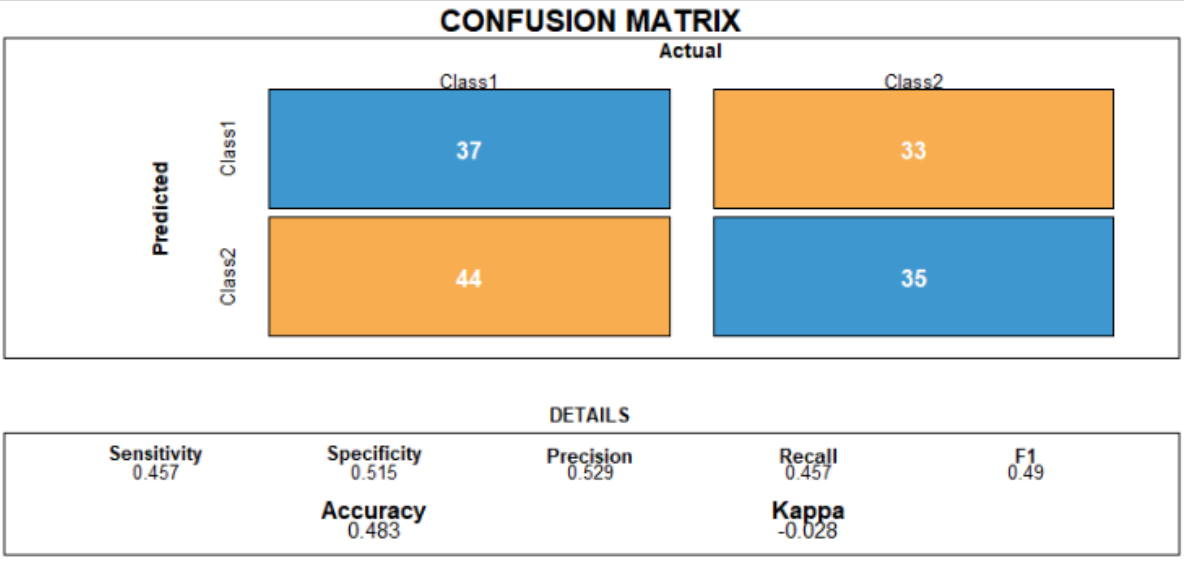

Figure 1: *Confusion Matrix of the final model*

4.2. Prediction

Let's make some prediction:

Imagine you are a sushi lover. You have lived in Japan for 5 years and fallen in love with the conveyor belt there. You wanted to establish a conveyor belt sushi chain in California and is asking for 100,000 USD in exchange for 10% of the company. A producer told you that Mark Cuban and not Kevin Harrington will be appearing on your show. Sadly, you must do this all alone because you do not have any supporting friends.

Another prediction involves Juan, who invented the phone jail that allows users to refrain from using their phone in an indefinite amount of time. Juan is trying to ask for 200,000 USD in exchange for 20% of the company. Juan is putting his company in Chicago and is looking forward to seeing Kevin Harrington. Similarly, Juan does not have any partners supporting him in this journey.

The prediction is revealed as follows:

Category	AskedFor	ExchangeForStake	Multiple	Region	Mark	Kevin_h	Prediction
Specialty Food	100000	10	FALSE	West	1	0	TRUE
Novelties	200000	15	FALSE	Midwest	0	1	TRUE

Table 4: Prediction result

Congratulations! Both of your projects will be likely to score a deal on the show.

5. Managerial conclusions and recommendations

5.1. Managerial conclusions

5.1.1. Potential benefits of the model

The model would be a useful tool that allows contestants to predict the success likelihood of their projects on the show. The producer could create an app that enables entrepreneurs to make predictions and give them a chance to adjust different parameters (this assumes that the details about the sharks are released in advance). The app could be monetized to generate more revenue for the show. In addition, by allowing the contestant to predict the outcome of their pitches, the show will undoubtedly attract more unique and innovative products that would undoubtedly help with general ratings and reviews. In addition, the show could also create a play-along app using the model to give the user a simulated result after inputting the parameters for their products. The app would be an interactive for Shark Tank, attracting users that could possibly become viewers and contestants on the show.

5.1.2. Limitations

5.1.2.1. Data

The dataset comprises of data on Shark Tank's first 6 seasons, with 495 observations. However, the show has been running for 14 seasons and has featured more than 1187 products. This number is nearly 3 times as much as the given number of observations. As a result, the accuracy could be tremendously improved given the additional observations.

Another important factor to take into consideration is the new addition of the "sharks" on the show. Throughout 14 seasons, the show has had 6 main investors, with guest investors appearing on different seasons. The abundant number of judges inevitably render the modeling challenging as it generates a large number of categorical variables with large variance. As a result, it is possible to only include the main judges in the model and not others in the model to increase the predictive power.

Finally, Shark Tank is considered to be a reality show, and a major part of reality shows is to show the authentic side of the personalities involved. As a result, the outcome of the deal could be manipulated by producers, the investors and even the contestants themselves (some companies only come on the show for publicity). Consequently, the human aspect of the data could hugely alter the outcome. Hence, the result of the prediction model could not be as accurate as expected.

5.1.2.2. Model

Boosted forest models are highly effective and can often outperform other machine learning algorithms on a variety of tasks. However, like all machine learning models, boosted forest models have their limitations.

One of the main limitations of boosted forest models is that they can be sensitive to noise in the data. Because they rely on multiple decision trees, any noise in the training data can potentially lead to overfitting, which can negatively impact the model's ability to make accurate predictions on new data. This can be particularly problematic if the training data is not carefully curated and cleaned before being used to train the model.

In addition, boosted forest models can also be difficult to interpret, especially when compared to simpler models like linear regression. Because they rely on complex combinations of decision trees, it can be challenging to understand how the model is making its predictions, which can make it difficult to diagnose and correct any issues that may arise.

Overall, while boosted forest models are powerful tools for making predictions, they do have their limitations. Careful data curation and a good understanding of the strengths and limitations of these models is essential for achieving the best results.

5.2. Recommendation

Another use of the data is to predict the region that the contestants might come from, based on certain criteria. This would allow producers and casting directors to target specifically companies in that region to increase rating. This allows the producers to manipulate the outcome of the deal, segment the show into different episodes and create show climax, attracting more viewers. As Shark Tank is a reality TV show and their main purpose is to attract more viewers, casting plays an important part in this process. A quick overview of the classification model using Linear Discriminant Analysis is included in the appendix.

Additionally, the dataset could be enhanced by including more information such as the shark that makes the deals, the gender of the investors, gender of the sharks accepting the deal, the duration of the contestant’s pitch. This information would enhance the accuracy of the prediction, as it allows the model to take into consideration the preference of the sharks and the performance of the contestants.

Appendix

Variable importance

	Midwest	Northeast	South	West	MeanDecreaseAccuracy	MeanDecreaseGini
deal	2.600	-0.395	2.687	3.856	4.403	10.244
category	4.348	4.538	5.853	6.826	10.600	17.558
askedFor	4.919	-3.355	0.663	6.655	5.273	43.976
exchangeForStake	-2.961	-2.871	3.062	9.957	7.860	34.414
Multiple.Entreprenuers	-2.600	1.156	2.241	-2.790	-0.998	8.960
barbara	-0.918	1.646	0.819	-4.004	-2.479	4.813
lori	0.179	0.369	2.039	-1.461	0.016	4.487
robert	-0.681	-0.807	-4.438	-0.889	-3.081	1.742
kevin_o	0.437	-0.374	-2.066	0.843	-0.324	2.962
steve	-1.728	1.001	-0.744	-3.249	-3.221	1.148
daymond	1.351	2.703	-1.025	-2.116	-1.382	2.873
jeff	1.731	-0.985	8.989	7.755	10.140	2.384
mark	7.470	1.369	0.212	5.112	6.757	4.144
kevin_h	6.896	-1.433	-4.953	4.157	1.370	3.463
john	-1.328	-3.169	-3.943	-3.642	-5.932	1.048
nick	-2.981	-1.095	-4.085	5.005	2.268	1.464

Table 1: *Variable importance for all predictors for the recommended lda model*

```
Call:
lda(region ~ deal + category + askedFor + exchangeForStake +
    Multiple.Entreprenuers + jeff + mark + nick, data = shark_tank1_train)

Prior probabilities of groups:
  Midwest Northeast     South      West
0.1271676 0.1647399 0.2861272 0.4219653

Group means:
          dealTRUE categoryNovelties categoryother categorySpecialty Food askedFor exchangeForStake Multiple.EntreprenuersTRUE       jeff
Midwest   0.5000000        0.06818182     0.6818182              0.2045455 180772.7         20.25000                  0.2954545 0.00000000
Northeast 0.4385965        0.05263158     0.7894737              0.1228070 207140.4         17.77193                  0.3333333 0.01754386
South     0.5656566        0.05050505     0.7272727              0.1111111 222626.3         18.80808                  0.2828283 0.06060606
West      0.5068493        0.04109589     0.7808219              0.1438356 217541.1         15.89041                  0.3082192 0.00000000
              mark       nick
Midwest   0.7045455 0.02272727
Northeast 0.8596491 0.01754386
South     0.7474747 0.01010101
West      0.8698630 0.01369863

Coefficients of linear discriminants:
                              LD1           LD2           LD3
dealTRUE                -4.370421e-01  1.701675e-01  1.205316e+00
categoryNovelties        1.907973e+00 -1.361882e+00 -1.445041e+00
categoryother            2.095001e+00 -4.716940e-01 -7.840738e-01
categorySpecialty Food   2.184511e+00 -1.767308e+00 -1.385244e-02
askedFor                -6.910187e-07  7.126299e-07  3.525702e-07
exchangeForStake        -4.824228e-02 -4.197957e-02 -5.609759e-02
Multiple.EntreprenuersTRUE  1.157641e-01 -1.037895e-01 -7.011318e-01
jeff                    -3.790778e+00  4.147671e+00 -2.314796e+00
mark                     7.098942e-01  1.480134e+00 -1.129795e+00
nick                     1.306149e-01 -1.474021e+00 -1.240801e+00

Proportion of trace:
   LD1    LD2    LD3
0.6272 0.3019 0.0708
```

Figure 1: Result of the lda model