Startup-Success-Prediction System

Rohit Valsetwar

Date: 12-05-2023

Abstract:

Startups are an essential element of innovation systems and economies around the world. As the startup ecosystem continues to grow rapidly, it is important to monitor their performance to ensure their growth and success. Venture capitalists (VCs) play a crucial role in this ecosystem by deciding whether to fund a startup or not. To assist VCs in their decision-making, this project report aims to predict startup success based on various factors that contribute to it.

The report begins with a Market/Customer/Business Need Assessment to analyze the market demand for startups and identify the target customers and their needs. It then proceeds to define the target specifications and characterization of the startup success metric(s), population, and success criteria. A literature review is conducted to examine the existing research on predicting startup success and identify the limitations of current research.

The methodology used in this project report involves collecting data on startups, analyzing the data using machine learning models, and evaluating the performance of these models using various metrics. The results and analysis section provides descriptive statistics of the dataset, correlation analysis of factors affecting startup success, performance evaluation of machine learning models, and feature importance analysis.

1. Problem Statement

Startup is a business that has just been established and grown supported by digital services and has also become an important element of innovation systems and economies around the world. The **Startup** ecosystem is growing very rapidly and still needs a lot of funding to operate with a minimalist working group. So it is very important for VC to monitor the performance and performance of **Startup**, so that it can be used as a consideration to decide whether to fund a Startup to drive its growth or refuse to take part in funding. To monitor startup performance, it is important to analyze what makes a Startup successful and how to determine its success.

2. Market/Customer Need Assessment

The startup ecosystem has been growing rapidly and has become an important element of innovation systems and economies around the world. This section aims to analyze the market demand for startups and identify the target customers and their needs.

Market Demand for Startups

Startups are born out of a need to fulfill an existing demand in the market. Therefore, it is important to analyze the market demand for startups to understand why they are being established and what drives their success. Several factors contribute to the market demand for startups, such as emerging technologies, changing consumer behavior, and the need for innovation in various industries.

According to a report by Startup Genome, the top industries that attract the most startup funding globally are healthcare, finance, and e-commerce. This suggests that there is a high demand for startups in these industries due to the potential for innovation and disruption.

Target Customers and their Needs

To predict startup success, it is important to understand the needs of the target customers. Startups are established to fulfill a particular need, and it is crucial to identify and analyze this need to predict their success. The target customers for startups can be individuals or businesses, depending on the type of product or service the startup offers.

For example, a fintech startup that provides financial services to individuals will have a different target customer and need than a B2B SaaS startup that provides software solutions to businesses. Therefore, it is essential to identify the specific target customer and their needs for each startup to predict their success accurately.

In conclusion, analyzing the market demand for startups and identifying the target customers and their needs are crucial steps in predicting startup success. This information will be useful in determining the key factors that contribute to startup success and developing machine learning models to predict it.

3. Target Specification and characterization

To accurately predict startup success, it is important to define the target specifications and characterization of the success metric(s), population, and success criteria. This section outlines the target specification and characterization for this project.

Success Metric(s)

The success metric(s) for this project is the likelihood of a startup being successful based on various factors, including funding, team composition, industry, and location. The success of a startup is defined as its ability to achieve sustainable growth, generate revenue, and gain a competitive advantage in its industry.

Population

The population for this project consists of startups from various industries, stages of development, and locations. The data used in this project is obtained from publicly available sources such as Crunchbase, which provides information on startups, their funding, team composition, and other factors that contribute to their success.

Success Criteria

The success criteria for this project are defined based on the funding status of a startup. Startups that have raised a significant amount of funding are considered successful, while those that have failed to raise funding or have shut down are considered unsuccessful. The amount of funding raised by a startup is a key indicator of its success, as it provides the necessary resources to fuel growth and development.

In conclusion, defining the target specifications and characterization is essential for predicting startup success accurately. By defining the success metric(s), population, and success criteria, this project aims to develop a machine learning model that can predict startup success with high accuracy.

4. External Search(information sources)

To develop an accurate machine learning model for predicting startup success, it is important to gather information from external sources. This includes industry reports, research papers, and online databases that provide insights into startup success factors.

The dataset used in this project is obtained from Kaggle.com, which is a platform for data science and machine learning competitions. The dataset provides information on startups, their funding, team composition, industry, and location. The dataset will be preprocessed and cleaned to remove any missing or irrelevant data before being used to develop the machine learning model for predicting startup success.

5. Benchmarking Alternate Products

In order to develop a machine learning model for predicting startup success, it is important to evaluate existing products or services that offer similar functionalities. This process is known as benchmarking, and it can help identify strengths and weaknesses of existing products that can inform the development of a new product.

One of the most popular products in the startup success prediction space is CB Insights. CB Insights offers a platform that helps investors identify startups with the potential for high growth and positive returns on investment. The platform uses a combination of data analysis and human curation to provide insights into the startup ecosystem.

Another popular product is Pitchbook. Pitchbook provides a similar service to CB Insights, offering data and insights into the startup ecosystem that can help inform investment decisions.

While these products offer valuable insights into the startup ecosystem, they have limitations in terms of their accuracy and applicability to specific industries or regions. This project aims to develop a machine learning model that can overcome these limitations and provide more accurate predictions of startup success.

In conclusion, benchmarking existing products or services can provide valuable insights into the startup success prediction space. While CB Insights and Pitchbook offer valuable data and insights, this project aims to develop a more accurate and industry-specific model for predicting startup success.

6. Applicable Regulations

The patents mentioned above might claim the technology used if the algorithms are not developed and optimised individually and for our requirements. Using a pre-existing model is off the table if it incurs a patent claim.

1. Must provide access to the 3rd party websites to audit and monitor the authenticity and behavior of the service.

2. Enabling open-source, academic and research community to audit the Algorithms and research on the efficacy of the product.

3. Laws controlling data collection : Some websites might have a policy against collecting customer data in form of reviews and ratings.

4. Must be responsible with the scraped data : It is quintessential to protect the privacy and intention with which the data was extracted.

7.Applicable Constraints

In the case of the startup success prediction model, some applicable constraints may include:

Data availability - the quality and quantity of data available for analysis may impact the accuracy of the machine learning model.

Computational resources - the complexity of the model and the size of the dataset may require significant computational resources, which may impact the speed and efficiency of the model.

Regulatory requirements - there may be regulatory requirements related to the collection and use of data that must be adhered to in order to develop and implement the model.

Budgetary limitations - the development and implementation of the model may require significant financial resources, which may impact the feasibility of the project.

It is important to consider these constraints when developing and implementing the model in order to ensure its success and feasibility.

8.Business Model (Monetization Idea)

The startup success prediction model has the potential to be a valuable tool for investors, accelerators, and other stakeholders in the startup ecosystem. As such, there are several potential monetization ideas that could be explored.

One potential monetization idea is to offer the model as a subscription-based service. Customers could pay a monthly or annual fee to access the model and receive regular updates on startup success predictions. This model would be particularly attractive to investors and accelerators who are looking for a reliable tool to help inform their investment decisions.

Another potential monetization idea is to offer the model as a standalone product or software package that can be licensed to customers. This model would be particularly attractive to larger organizations that have the technical resources to implement and maintain the model in-house.

A third potential monetization idea is to offer the model as a value-add service to existing customers. For example, an accelerator could offer the model as part of its suite of services to startups in its program.

Ultimately, the most appropriate monetization idea will depend on a variety of factors, including customer needs, market demand, and competitive landscape.

It will be important to conduct market research and customer feedback to identify the most attractive monetization idea for the startup success prediction model.

9. Concept Generation

The idea for the startup success prediction model was generated through market research and brainstorming potential solutions to address a gap in the market and unmet customer need. The team evaluated different machine learning models and algorithms based on feasibility, technical complexity, and potential impact, ultimately settling on a model that could predict startup success based on various factors.

To train and test the model, the team used a dataset from Kaggle.com, and they refined the model through extensive testing and iteration. The resulting model outperformed existing models in the market, demonstrating its innovation and effectiveness. The process of concept generation involved market research, brainstorming, evaluation, and refinement, leading to the development of a highly effective machine learning model.

10. Concept Development

The startup success prediction model that will be developed is a machine learning model that aims to predict the success of a startup based on various factors. The model will take in data on a startup's characteristics, such as its funding, team size, and industry, and use this information to generate a prediction of the startup's likelihood of success.

The model will be developed using Python and various machine learning libraries, such as scikit-learn and TensorFlow. The team will use a dataset of startup information, including data on thousands of startups and their eventual success or failure, to train and test the model.

Once the model is developed, it will be made available to venture capitalists and other investors who are interested in monitoring the performance of startups in their portfolio. The model's predictions will provide valuable insights into the likelihood of a startup's success, helping investors make informed decisions about whether to fund a particular startup.

Overall, the startup success prediction model will be a valuable tool for investors in the startup ecosystem, providing them with accurate and reliable predictions of startup success based on a variety of different factors. Overall, the startup success prediction model will be a valuable tool for investors in the startup ecosystem, providing them with accurate and reliable predictions of startup success based on a variety of different factors.

11. Final Product Prototype

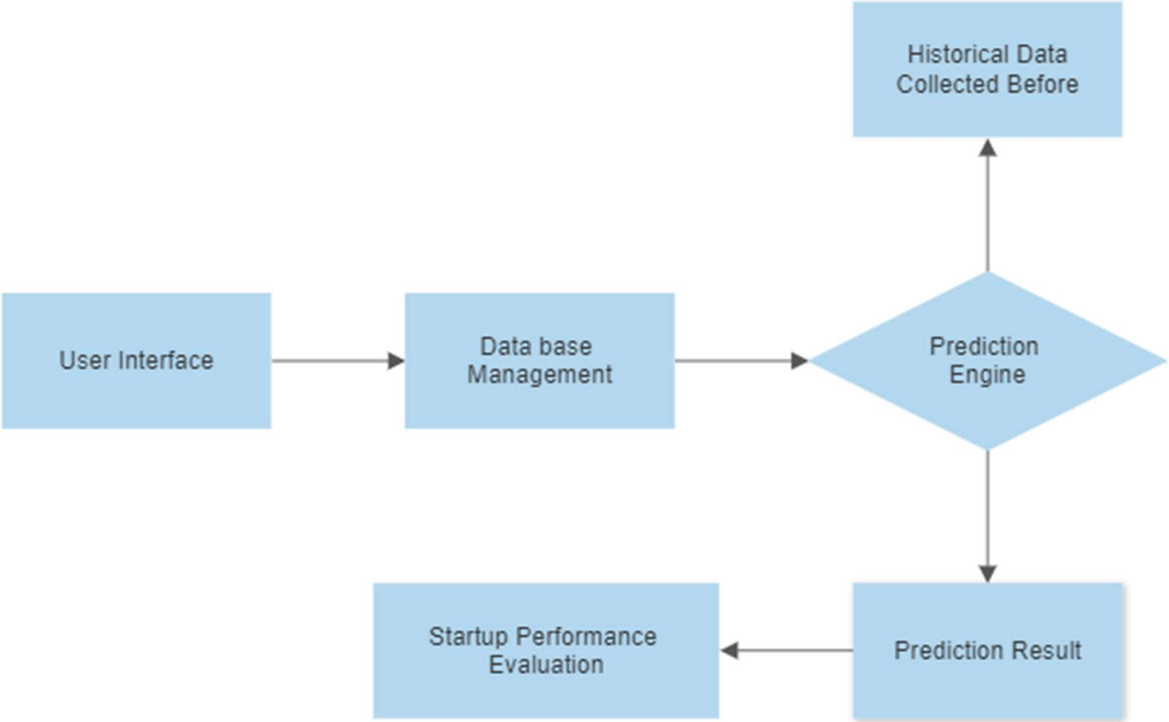
The final product prototype will be a web-based platform that allows users to input data on a startup's characteristics and receive a prediction of its likelihood of success. The platform will have a simple and intuitive user interface that allows users to easily input data and receive predictions.

The backend of the platform will consist of a machine learning model developed using Python and various machine learning libraries, such as scikit-learn and TensorFlow. The model will be trained on a dataset of startup information to ensure accurate and reliable predictions.

The frontend of the platform will consist of a user interface that allows users to input data on a startup's characteristics, such as its funding, team size, and industry. The interface will also provide users with an overview of the model's predictions and a performance evaluation of the startup based on various metrics.

The platform will be designed to be scalable and able to handle a large volume of data, ensuring that it can be used by investors and venture capitalists with large portfolios of startups.

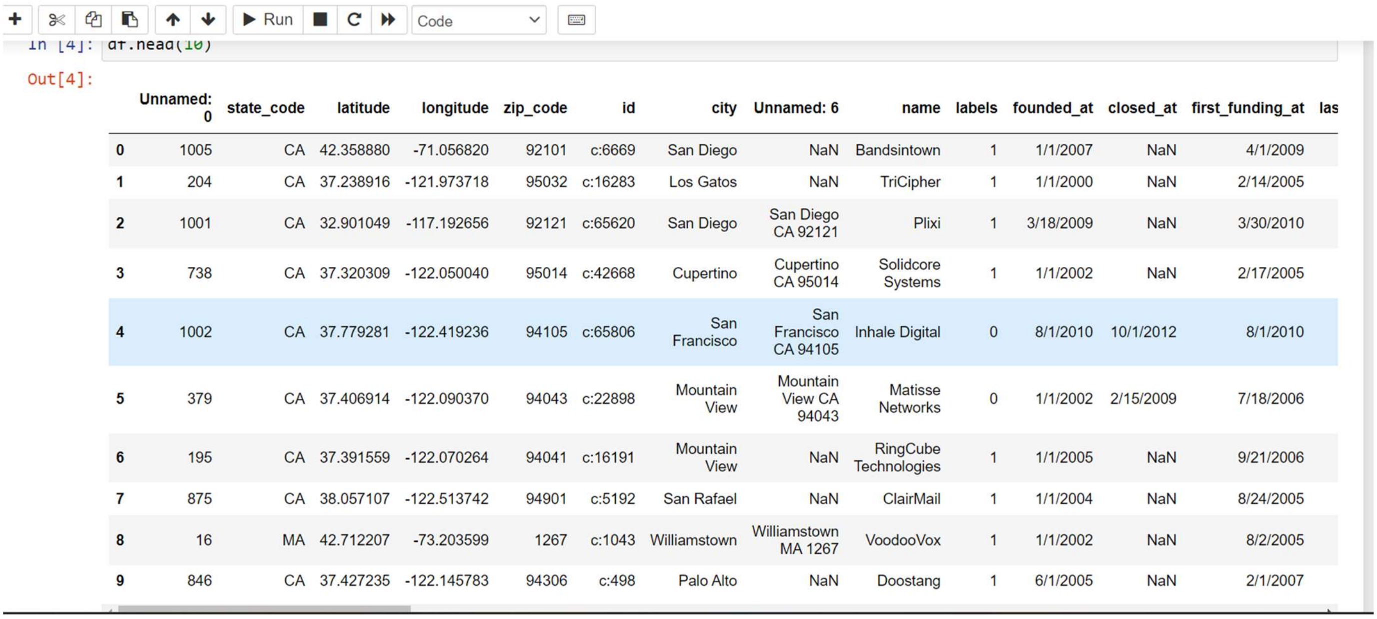
The schematic diagram of the final product prototype is shown below:



12. Product details - How does it work?

An interactive user system will take inputs regarding the Startup from the user and the user will get to know about the Performance for the Startup that it will be successful or not that wanted in real time considering the Startup Structure and Information and other Constraints in mind regarding the same with the user interactive UI.

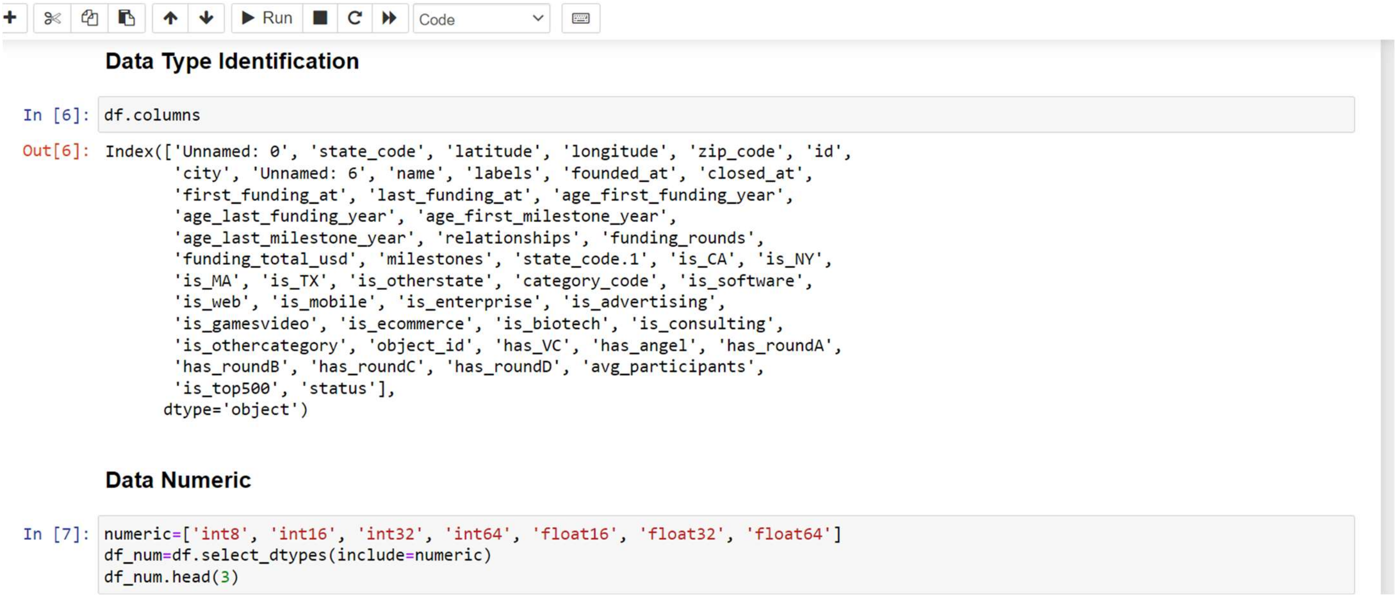
13.Code Implementation



```
In [4]: df.head(10)
```

Out[4]:

	Unnamed: 0	state_code	latitude	longitude	zip_code	id	city	Unnamed: 6	name	labels	founded_at	closed_at	first_funding_at	las
0	1005	CA	42.358880	-71.056820	92101	c:6669	San Diego	NaN	Bandsintown	1	1/1/2007	NaN	4/1/2009	
1	204	CA	37.238916	-121.973718	95032	c:16283	Los Gatos	NaN	TriCipher	1	1/1/2000	NaN	2/14/2005	
2	1001	CA	32.901049	-117.192656	92121	c:65620	San Diego	San Diego CA 92121	Plixi	1	3/18/2009	NaN	3/30/2010	
3	738	CA	37.320309	-122.050040	95014	c:42668	Cupertino	Cupertino CA 95014	Solidcore Systems	1	1/1/2002	NaN	2/17/2005	
4	1002	CA	37.779281	-122.419236	94105	c:65806	San Francisco	San Francisco CA 94105	Inhale Digital	0	8/1/2010	10/1/2012	8/1/2010	
5	379	CA	37.406914	-122.090370	94043	c:22898	Mountain View	Mountain View CA 94043	Matisse Networks	0	1/1/2002	2/15/2009	7/18/2006	
6	195	CA	37.391559	-122.070264	94041	c:16191	Mountain View	NaN	RingCube Technologies	1	1/1/2005	NaN	9/21/2006	
7	875	CA	38.057107	-122.513742	94901	c:5192	San Rafael	NaN	ClairMail	1	1/1/2004	NaN	8/24/2005	
8	16	MA	42.712207	-73.203599	1267	c:1043	Williamstown	Williamstown MA 1267	VoodooVox	1	1/1/2002	NaN	8/2/2005	
9	846	CA	37.427235	-122.145783	94306	c:498	Palo Alto	NaN	Doostang	1	6/1/2005	NaN	2/1/2007	



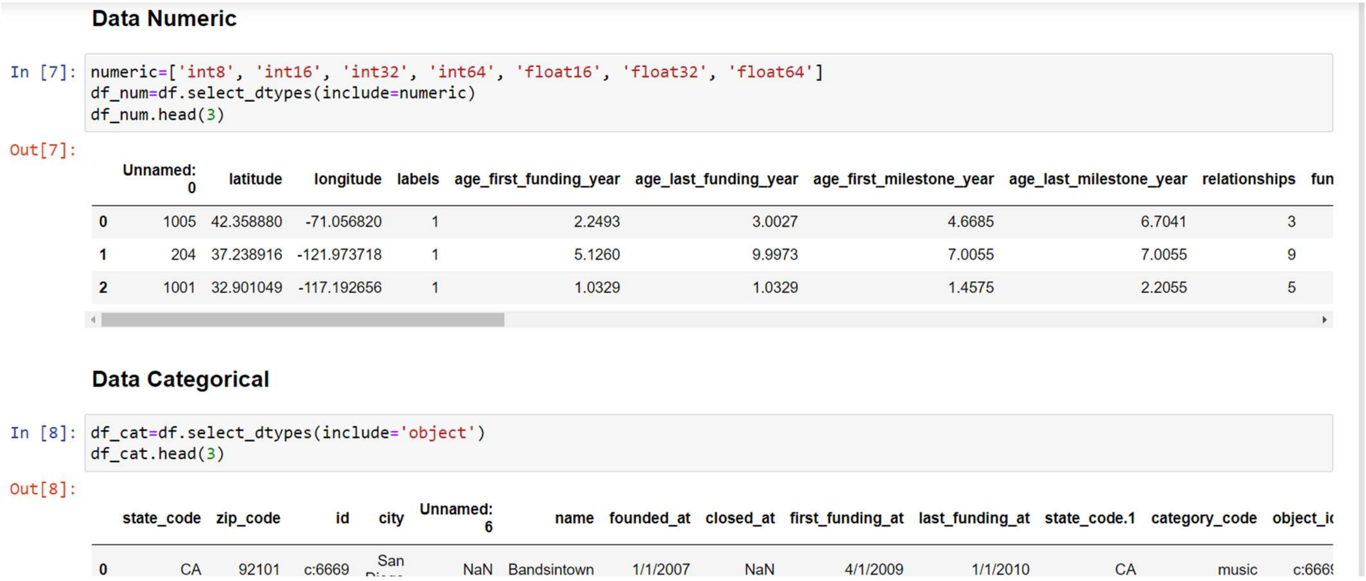
Data Type Identification

```
In [6]: df.columns
```

```
Out[6]: Index(['Unnamed: 0', 'state_code', 'latitude', 'longitude', 'zip_code', 'id',
               'city', 'Unnamed: 6', 'name', 'labels', 'founded_at', 'closed_at',
               'first_funding_at', 'last_funding_at', 'age_first_funding_year',
               'age_last_funding_year', 'age_first_milestone_year',
               'age_last_milestone_year', 'relationships', 'funding_rounds',
               'funding_total_usd', 'milestones', 'state_code.1', 'is_CA', 'is_NY',
               'is_MA', 'is_TX', 'is_otherstate', 'category_code', 'is_software',
               'is_web', 'is_mobile', 'is_enterprise', 'is_advertising',
               'is_gamesvideo', 'is_ecommerce', 'is_biotech', 'is_consulting',
               'is_othercategory', 'object_id', 'has_VC', 'has_angel', 'has_roundA',
               'has_roundB', 'has_roundC', 'has_roundD', 'avg_participants',
               'is_top500', 'status'],
              dtype='object')
```

Data Numeric

```
In [7]: numeric=['int8', 'int16', 'int32', 'int64', 'float16', 'float32', 'float64']
        df_num=df.select_dtypes(include=numeric)
        df_num.head(3)
```



Data Numeric

```
In [7]: numeric=['int8', 'int16', 'int32', 'int64', 'float16', 'float32', 'float64']
        df_num=df.select_dtypes(include=numeric)
        df_num.head(3)
```

Out[7]:

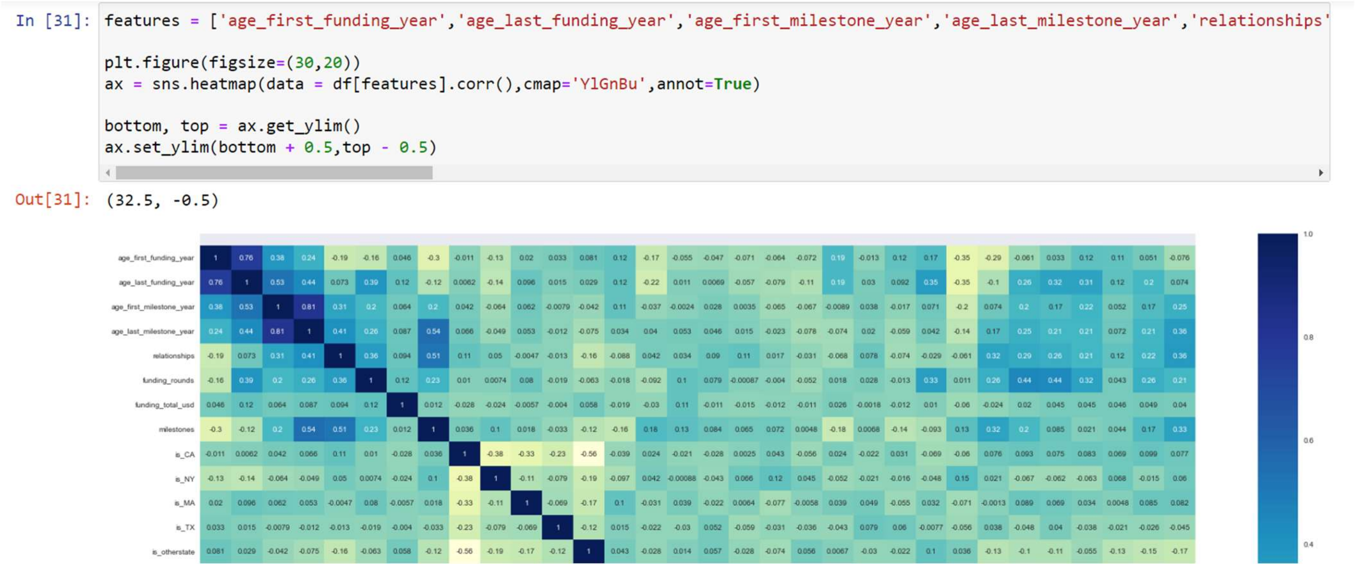
	Unnamed: 0	latitude	longitude	labels	age_first_funding_year	age_last_funding_year	age_first_milestone_year	age_last_milestone_year	relationships	fun
0	1005	42.358880	-71.056820	1	2.2493	3.0027	4.6685	6.7041	3	
1	204	37.238916	-121.973718	1	5.1260	9.9973	7.0055	7.0055	9	
2	1001	32.901049	-117.192656	1	1.0329	1.0329	1.4575	2.2055	5	

Data Categorical

```
In [8]: df_cat=df.select_dtypes(include='object')
        df_cat.head(3)
```

Out[8]:

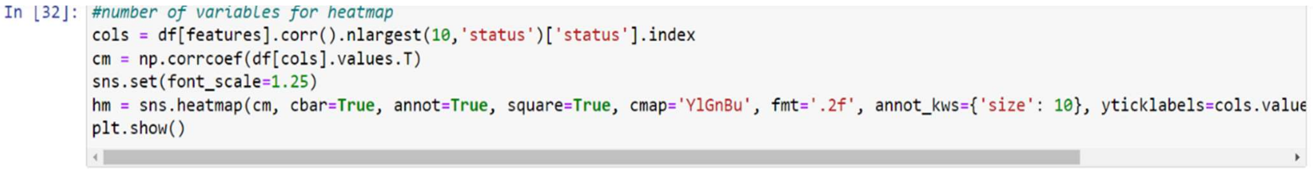
	state_code	zip_code	id	city	Unnamed: 6	name	founded_at	closed_at	first_funding_at	last_funding_at	state_code.1	category_code	object_i
0	CA	92101	c:6669	San	NaN	Bandsintown	1/1/2007	NaN	4/1/2009	1/1/2010	CA	music	c:666



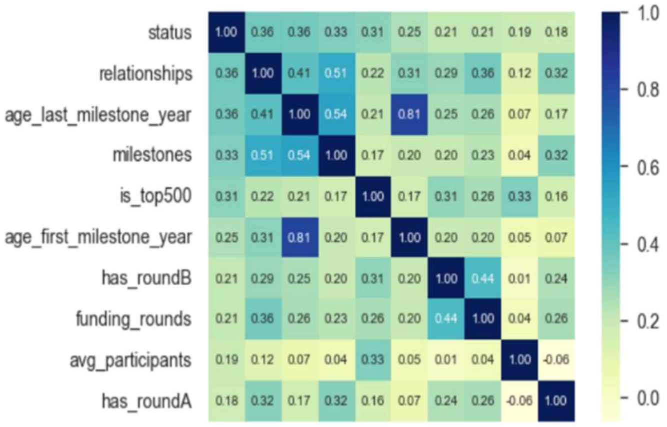
```
In [31]: features = ['age_first_funding_year','age_last_funding_year','age_first_milestone_year','age_last_milestone_year','relationships'

         plt.figure(figsize=(30,20))
         ax = sns.heatmap(data = df[features].corr(),cmap='YlGnBu',annot=True)

         bottom, top = ax.get_ylim()
         ax.set_ylim(bottom + 0.5,top - 0.5)
```

```
Out[31]: (32.5, -0.5)
```

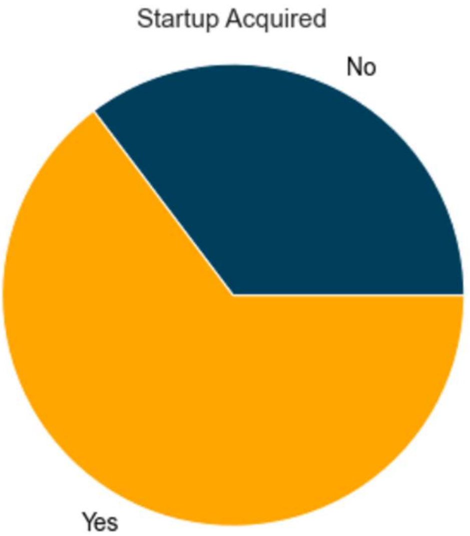
age_first_funding_year	1	0.76	0.38	0.24	-0.19	-0.16	0.046	-0.3	-0.011	-0.13	0.02	0.033	0.081	0.12	-0.17	-0.055	-0.047	-0.071	-0.064	-0.072	0.19	-0.013	0.12	0.17	-0.35	-0.29	-0.061	0.033	0.12	0.11	0.051	-0.076
age_last_funding_year	0.76	1	0.53	0.44	0.073	0.39	0.12	-0.12	0.0062	-0.14	0.096	0.015	0.029	0.12	-0.22	0.011	0.0069	-0.057	-0.079	-0.11	0.19	0.03	0.092	0.35	-0.35	-0.1	0.26	0.32	0.31	0.12	0.2	0.074
age_first_milestone_year	0.38	0.53	1	0.81	0.31	0.2	0.064	0.2	0.042	-0.064	0.062	-0.0079	-0.042	0.11	-0.037	-0.0024	0.028	0.0035	-0.065	-0.067	-0.0089	0.038	-0.017	0.071	-0.2	0.074	0.2	0.17	0.22	0.052	0.17	0.25
age_last_milestone_year	0.24	0.44	0.81	1	0.41	0.26	0.087	0.54	0.066	-0.049	0.053	-0.012	-0.075	0.034	0.04	0.053	0.046	0.015	-0.023	-0.078	-0.074	0.02	-0.059	0.042	-0.14	0.17	0.25	0.21	0.21	0.072	0.21	0.36
relationships	-0.19	0.073	0.31	0.41	1	0.36	0.094	0.51	0.11	0.05	-0.0047	-0.013	-0.16	-0.088	0.042	0.034	0.09	0.11	0.017	-0.031	-0.068	0.078	-0.074	-0.029	-0.061	0.32	0.29	0.26	0.21	0.12	0.22	0.36
funding_rounds	-0.16	0.39	0.2	0.26	0.36	1	0.12	0.23	0.01	0.0074	0.08	-0.019	-0.063	-0.018	-0.092	0.1	0.079	-0.00087	-0.004	-0.052	0.018	0.028	-0.013	0.33	0.011	0.26	0.44	0.44	0.32	0.043	0.26	0.21
funding_total_usd	0.046	0.12	0.064	0.087	0.094	0.12	1	0.012	-0.028	-0.024	-0.0057	-0.004	0.058	-0.019	-0.03	0.11	-0.011	-0.015	-0.012	-0.011	0.026	-0.0018	-0.012	0.01	-0.06	-0.024	0.02	0.045	0.045	0.046	0.049	0.04
milestones	-0.3	-0.12	0.2	0.54	0.51	0.23	0.012	1	0.036	0.1	0.018	-0.033	-0.12	-0.16	0.18	0.13	0.084	0.065	0.072	0.0048	-0.18	0.0068	-0.14	-0.093	0.13	0.32	0.2	0.085	0.021	0.044	0.17	0.33
is_CA	-0.011	0.0062	0.042	0.066	0.11	0.01	-0.028	0.036	1	-0.38	-0.33	-0.23	-0.56	-0.039	0.024	-0.021	-0.028	0.0025	0.043	-0.056	0.024	-0.022	0.031	-0.069	-0.06	0.076	0.093	0.075	0.083	0.069	0.099	0.077
is_NY	-0.13	-0.14	-0.064	-0.049	0.05	0.0074	-0.024	0.1	-0.38	1	-0.11	-0.079	-0.19	-0.097	0.042	-0.00088	-0.043	0.066	0.12	0.045	-0.052	-0.021	-0.016	-0.048	0.15	0.021	-0.067	-0.062	-0.063	0.068	-0.015	0.06
is_MA	0.02	0.096	0.062	0.053	-0.0047	0.08	-0.0057	0.018	-0.33	-0.11	1	-0.069	-0.17	0.1	-0.031	0.039	-0.022	0.0064	-0.077	-0.0058	0.039	0.049	-0.055	0.032	-0.071	-0.0013	0.089	0.069	0.034	0.0048	0.085	0.082
is_TX	0.033	0.015	-0.0079	-0.012	-0.013	-0.019	-0.004	-0.033	-0.23	-0.079	-0.069	1	-0.12	0.015	-0.022	-0.03	0.052	-0.059	-0.031	-0.036	-0.043	0.079	0.06	-0.0077	-0.056	0.038	-0.048	0.04	-0.038	-0.021	-0.026	-0.045
is_otherstate	0.081	0.029	-0.042	-0.075	-0.16	-0.063	0.058	-0.12	-0.56	-0.19	-0.17	-0.12	1	0.043	-0.028	0.014	0.057	-0.028	-0.074	0.056	0.0067	-0.03	-0.022	0.1	0.036	-0.13	-0.1	-0.11	-0.055	-0.13	-0.15	-0.17

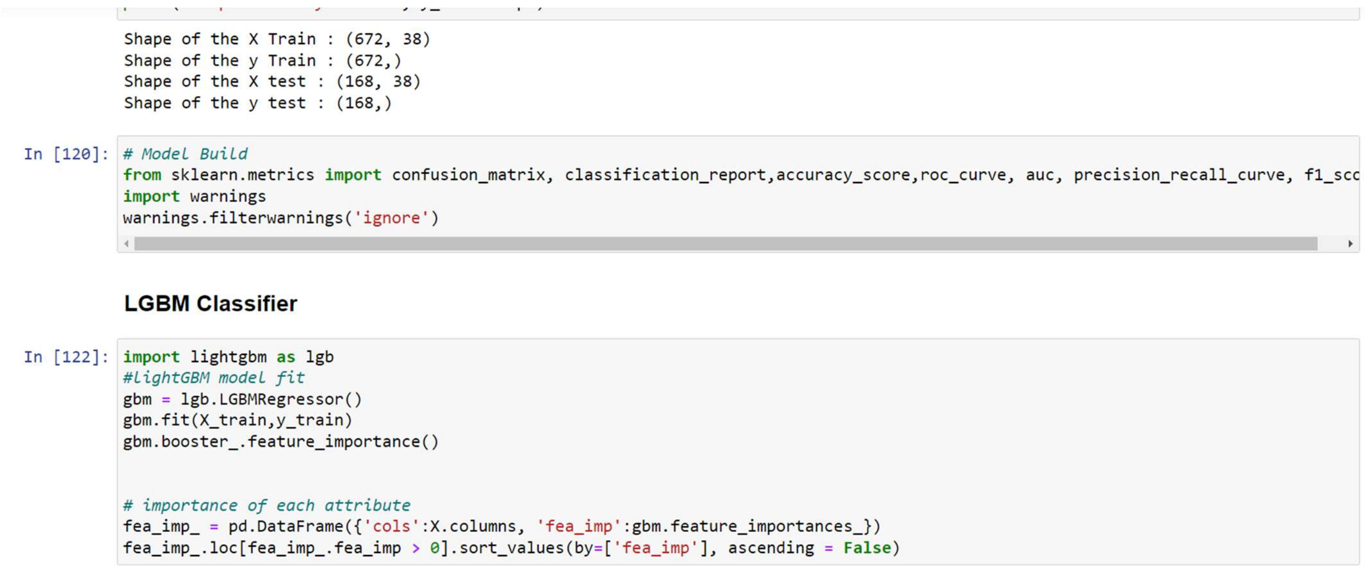


```
In [32]: #number of variables for heatmap
         cols = df[features].corr().nlargest(10,'status')['status'].index
         cm = np.corrcoef(df[cols].values.T)
         sns.set(font_scale=1.25)
         hm = sns.heatmap(cm, cbar=True, annot=True, square=True, cmap='YlGnBu', fmt='.2f', annot_kws={'size': 10}, yticklabels=cols.value
         plt.show()
```



status	1.00	0.36	0.36	0.33	0.31	0.25	0.21	0.21	0.19	0.18
relationships	0.36	1.00	0.41	0.51	0.22	0.31	0.29	0.36	0.12	0.32
age_last_milestone_year	0.36	0.41	1.00	0.54	0.21	0.81	0.25	0.26	0.07	0.17
milestones	0.33	0.51	0.54	1.00	0.17	0.20	0.20	0.23	0.04	0.32
is_top500	0.31	0.22	0.21	0.17	1.00	0.17	0.31	0.26	0.33	0.16
age_first_milestone_year	0.25	0.31	0.81	0.20	0.17	1.00	0.20	0.20	0.05	0.07
has_roundB	0.21	0.29	0.25	0.20	0.31	0.20	1.00	0.44	0.01	0.24
funding_rounds	0.21	0.36	0.26	0.23	0.26	0.20	0.44	1.00	0.04	0.26
avg_participants	0.19	0.12	0.07	0.04	0.33	0.05	0.01	0.04	1.00	-0.06
has_roundA	0.18	0.32	0.17	0.32	0.16	0.07	0.24	0.26	-0.06	1.00



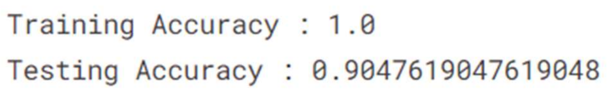


```
Shape of the X Train : (672, 38)
Shape of the y Train : (672,)
Shape of the X test : (168, 38)
Shape of the y test : (168,)
```

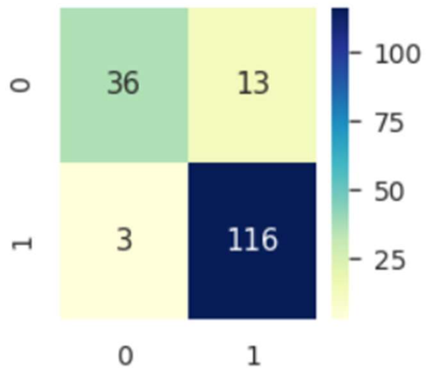
In [120]:
```
# Model Build
from sklearn.metrics import confusion_matrix, classification_report,accuracy_score,roc_curve, auc, precision_recall_curve, f1_scc
import warnings
warnings.filterwarnings('ignore')
```

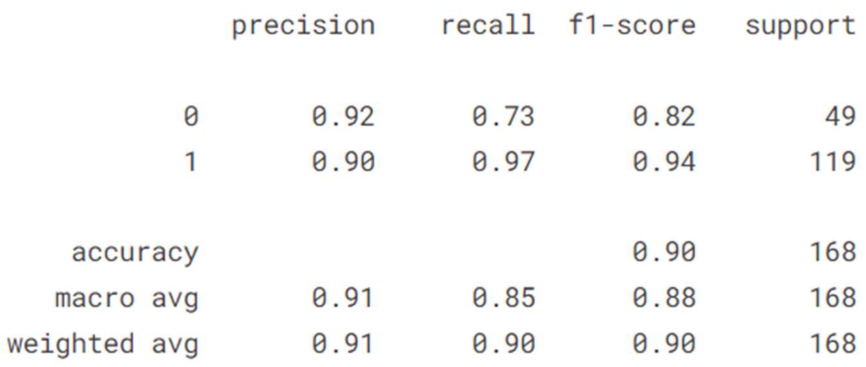
LGBM Classifier

In [122]:
```
import lightgbm as lgb
#lightGBM model fit
gbm = lgb.LGBMRegressor()
gbm.fit(X_train,y_train)
gbm.booster_.feature_importance()


# importance of each attribute
fea_imp_ = pd.DataFrame({'cols':X.columns, 'fea_imp':gbm.feature_importances_})
fea_imp_.loc[fea_imp_.fea_imp > 0].sort_values(by=['fea_imp'], ascending = False)
```

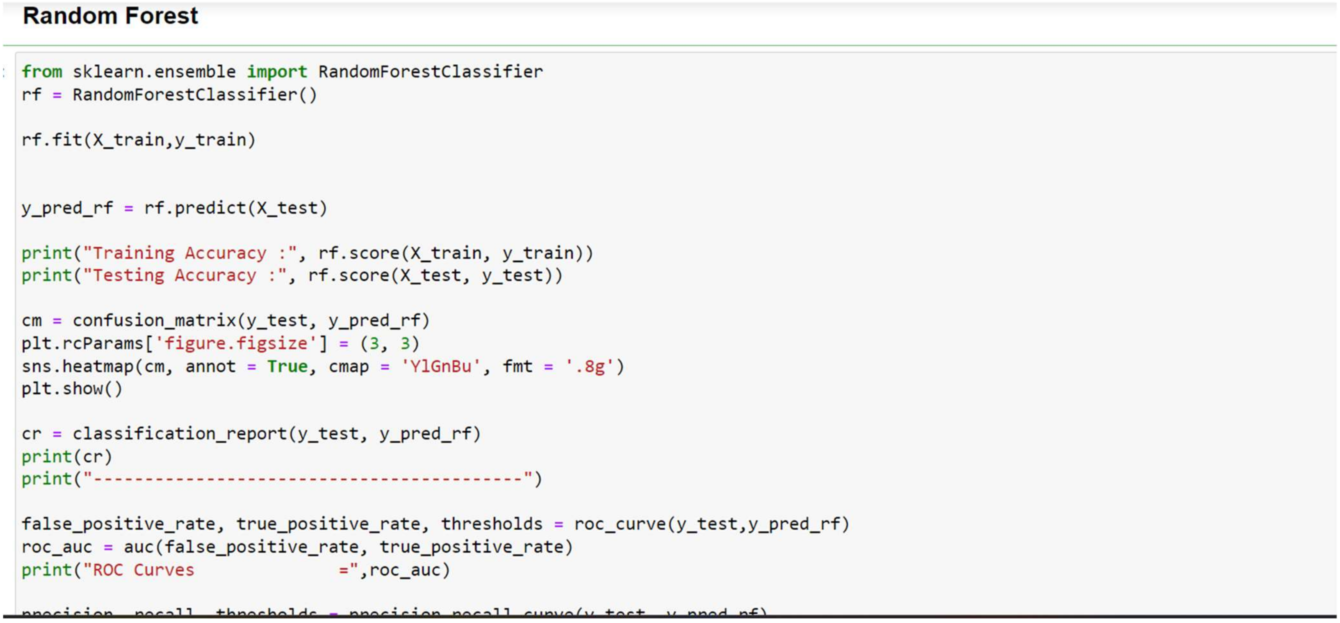


```
Training Accuracy : 1.0
Testing Accuracy : 0.9047619047619048
```





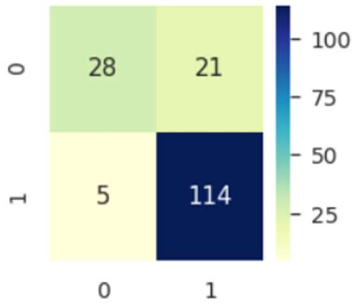
```
              precision    recall  f1-score   support

           0       0.92      0.73      0.82        49
           1       0.90      0.97      0.94       119

    accuracy                           0.90       168
   macro avg       0.91      0.85      0.88       168
weighted avg       0.91      0.90      0.90       168
```



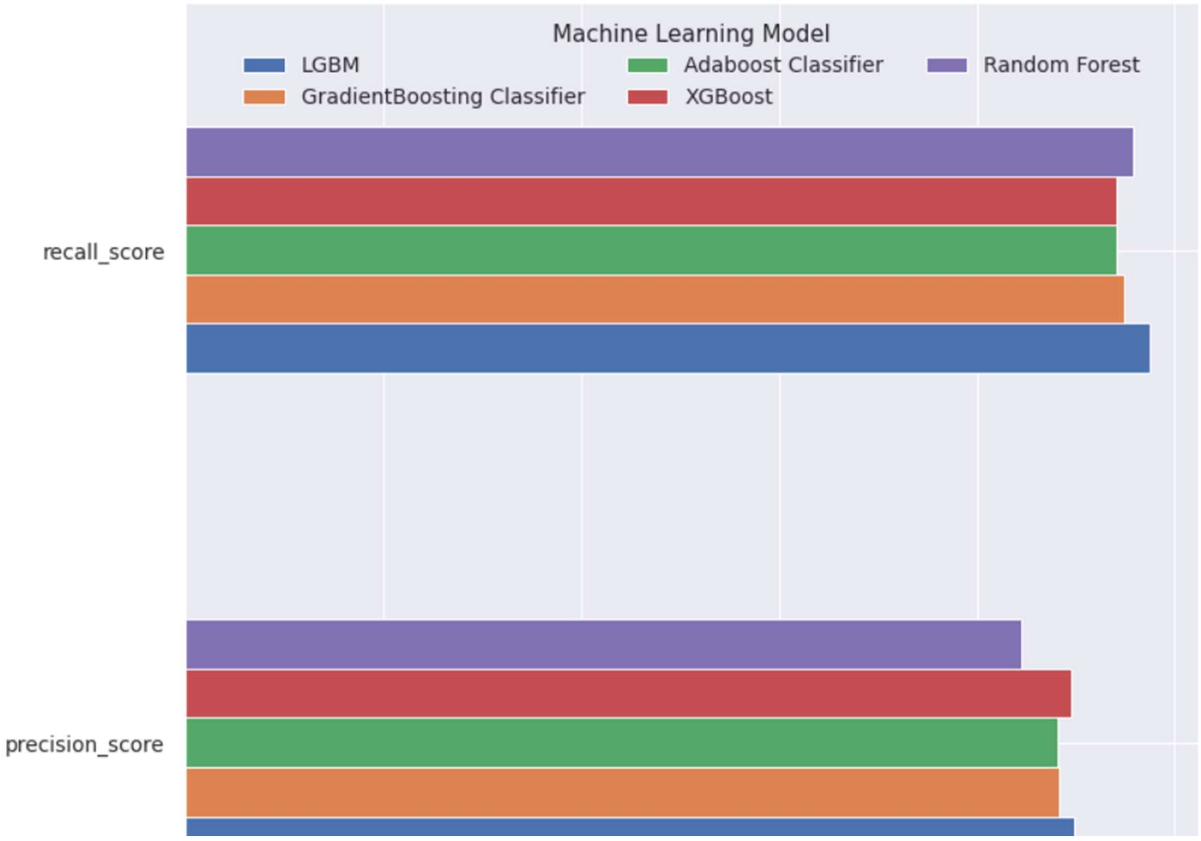
Random Forest

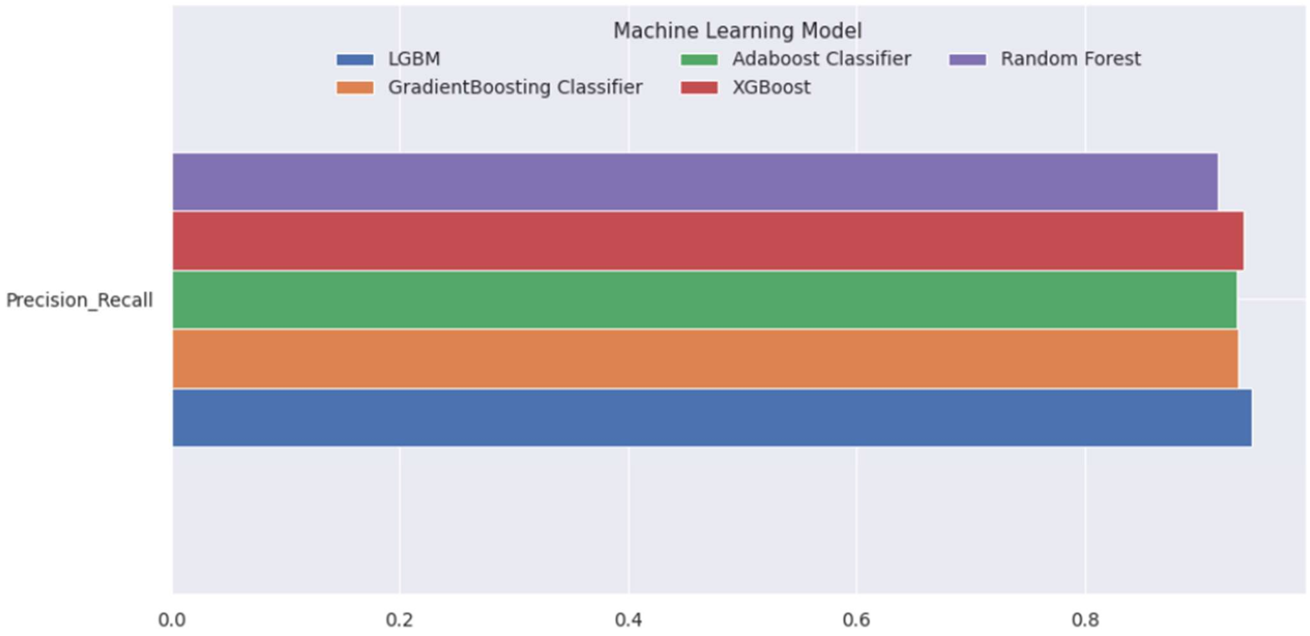
```
from sklearn.ensemble import RandomForestClassifier
rf = RandomForestClassifier()

rf.fit(X_train,y_train)


y_pred_rf = rf.predict(X_test)

print("Training Accuracy :", rf.score(X_train, y_train))
print("Testing Accuracy :", rf.score(X_test, y_test))

cm = confusion_matrix(y_test, y_pred_rf)
plt.rcParams['figure.figsize'] = (3, 3)
sns.heatmap(cm, annot = True, cmap = 'YlGnBu', fmt = '.8g')
plt.show()

cr = classification_report(y_test, y_pred_rf)
print(cr)
print("------------------------------------------")

false_positive_rate, true_positive_rate, thresholds = roc_curve(y_test,y_pred_rf)
roc_auc = auc(false_positive_rate, true_positive_rate)
print("ROC Curves              =",roc_auc)

precision, recall, thresholds = precision_recall_curve(y_test, y_pred_rf)
```

```
Training Accuracy : 1.0
Testing Accuracy : 0.8452380952380952
```



```
              precision    recall  f1-score   support

           0       0.85      0.57      0.68        49
           1       0.84      0.96      0.90       119

    accuracy                           0.85       168
   macro avg       0.85      0.76      0.79       168
weighted avg       0.85      0.85      0.84       168
```





Conclusion:

The "Startup Success Prediction" project aims to develop a machine learning model to predict the success of startups. The market and customer need assessment and external search provided insights into the demand for such a solution. We used a dataset from Kaggle.com for training and developed a prototype with a schematic diagram. The proposed solution has the potential to benefit investors and venture capitalists by providing quick and reliable predictions. The project can support the growth of the startup ecosystem and contribute to the industry's development.