PREDICTING WHETHER A STARTUP WILL ACHIEVE A LIQUIDITY EVENT WITHIN 5 YEARS

TEAM MEMBERS:

KRISH GUPTA
PARTH SANGHVI
ARSHVEER CHHABRA
PRATHAM RAJ BHARDWAJ

GITHUB LINK- https://github.com/Arshveer05/Startup_ipo_prediction


01
PROBLEM STATEMENT
02
LITERATURE SURVEY
03
DATASET AND FEATURES PREPROCESSING
04
ML METHODOLOGY

PROBLEM STATEMENT

Despite the critical role startups play in economic growth and job creation, the majority fail before reaching a liquidity event.

Traditional startup evaluation methods, which rely heavily on qualitative assessments and historical financial data, often fail to capture the complexity of the modern startup landscape.

The ML revolution presents an opportunity to develop predictive models providing likelihood of startup success by analysing structured and unstructured data.

This research aims to develop a multi-feature ML model that integrates diverse data sources to improve startup success prediction, aiding investors, entrepreneurs, and policymakers in more effective decision-making and capital allocation.

LITERATURE SURVEY

Ünal & Ceasu (2020): A Machine Learning Approach Towards Startup Success Prediction (https://econpapers.repec.org/paper/zbwirtgdp/2019022.htm)

- Compared 6 models for predicting startup outcomes
- XGBoost, Random Forest and Ensemble
- **XGBoost: 94.5%**
- RF: 94.1% accuracy
- Oversampling approach, ADASYN to handle imbalance
- **Crunchbase data**

Zhang et al. (2020): Financial Sentiment Analysis: Techniques and Applications (https://dl.acm.org/doi/10.1145/3649451)

- Social Media Sentiment in Finance
- **Added sentiment analysis to financial forecasting**
- NLP + ML models for sentiment-informed prediction
- 8–12% improvement over baseline models

Okeleke et al. (2024): Predictive analytics for market trends using AI: A study in consumer behaviour (https://www.researchgate.net/publication/383410055_Predictive_analytics_for_market_trends_using_AI_A_study_in_consumer_behavior)

- AI for Market Trend Prediction
- Predicted consumer behaviour & market trends using AI
- Deep Learning & traditional ML on consumer data
- ~90% accuracy in market trend forecasting

LITERATURE SURVEY

Current Landscape

Most existing literature focuses on structured data sourced from tools like Crunchbase and Pitchbook—covering funding history, revenue growth, team size, and investor profiles. These features offer strong predictive power for assessing startup success, but they primarily reflect historical and static company attributes.

“These structured metrics provide strong predictive power, but are limited to past performance and static attributes.” (Wang, 2019)

Core Limitation

While structured data is valuable, it lacks the ability to account for dynamic external factors like market sentiment, media trends, or consumer interest. For instance, two startups with identical funding profiles may have very different trajectories depending on public perception and real-time market shifts—factors not captured by traditional datasets.

The Gap

- Predictive models often miss out on real-time signals like Google Trends, social media, and online search activity. These unstructured data sources reflect public interest and sentiment shifts that structured data cannot capture.
- Since startups operate in fast-changing environments, relying only on historical data reduces prediction accuracy.
- Adding real-time market signals helps models detect early signs of success or failure more effectively.
- As Zaremba, Bąk, and Kowalski (2020) suggest, future models should combine structured metrics with unstructured data to improve prediction accuracy.

DATASET

Nature of the Data

- The Crunchbase 2013 dataset captures startup and venture capital funding information. It includes:
- Types of funding rounds (e.g., angel, venture, private equity).
- State location codes, information about investors
- A target column (for a binary classification task, such as success/failure of the startup).
- 1254 datapoints (rows).
- 25 features (columns),

Features

- The dataset includes eight binary indicators representing different types of funding rounds .
- It also captures the frequency of funding rounds through features labeled from funding_round_0 to funding_round_14.
- Additional numerical features include trend scores, state codes, and the number of investors.

How the dataset was collected?

Data was scraped or downloaded regarding:

- Companies
- Funding rounds
- Investors
- Acquisitions

The data was then structured into relational tables and merged into usable machine learning formats.

Ethical Concerns

Bias: Crunchbase data favoured startups from tech hubs or English-speaking regions.

funding_round_11	funding_round_14	trends	target	state_code	num_investors

funding_round_2	funding_round_3	funding_round_4	funding_round_5	funding_round_6	funding_round_7	funding_round_8	funding_round_9	funding_round_10

round_angel	round_other	round_private-equity	round_series-a	round_series-b	round_series-c+	round_unknown	round_venture	funding_round_0	funding_round_1

DATA PREPROCESSING

Self-Collected Data (Google Trends):

- We manually collected Google Trends data for each company in our dataset, covering the period from 2013–2018.
- This data was used to create a binary sentiment variable indicating if public interest crossed a threshold (1 or 0).
- Considerations included aligning search terms to company names and ensuring uniform time windows.
- Ethical Note: As Google Trends provides anonymised and aggregated data, there were no personal privacy concerns involved.

External Datasets:

- We used and merged 4 publicly available startup datasets (e.g., Crunchbase and similar sources) containing company metadata.
- These datasets included features like founding date, funding rounds, location, industry category, number of investors etc.
- After merging and filtering for common companies across sources, we obtained a unified dataset of ~1700 companies with 31 features.

DATA PREPROCESSING

Feature Selection

- Rather than engineering new features, we performed correlation matrix analysis to evaluate the relationship between each feature and the target variable (startup success).
- Features with low or no correlation were removed to streamline the model and reduce noise.
- Since the dataset was already structured and interpretable, dimensionality reduction techniques like PCA or LDA were not required.

Addressing Class Imbalance

- Initially, there was a strong imbalance toward unsuccessful startups. To mitigate this, we sourced additional data on successful startups.(label = 0).
- The improved ratio reduced the need for synthetic oversampling techniques like SMOTE.
- However, XGBoost's scale_pos_weight was still fine-tuned (via Optuna) to reflect the updated class distribution and improve sensitivity to the minority class.
- The recall for successful startups increased, and we observed a 37% improvement in F1-score for the positive class.

Data Cleaning

- Records with excessive missing or corrupt values were removed.
- Minor missing values were handled using statistical imputation (mean or median).

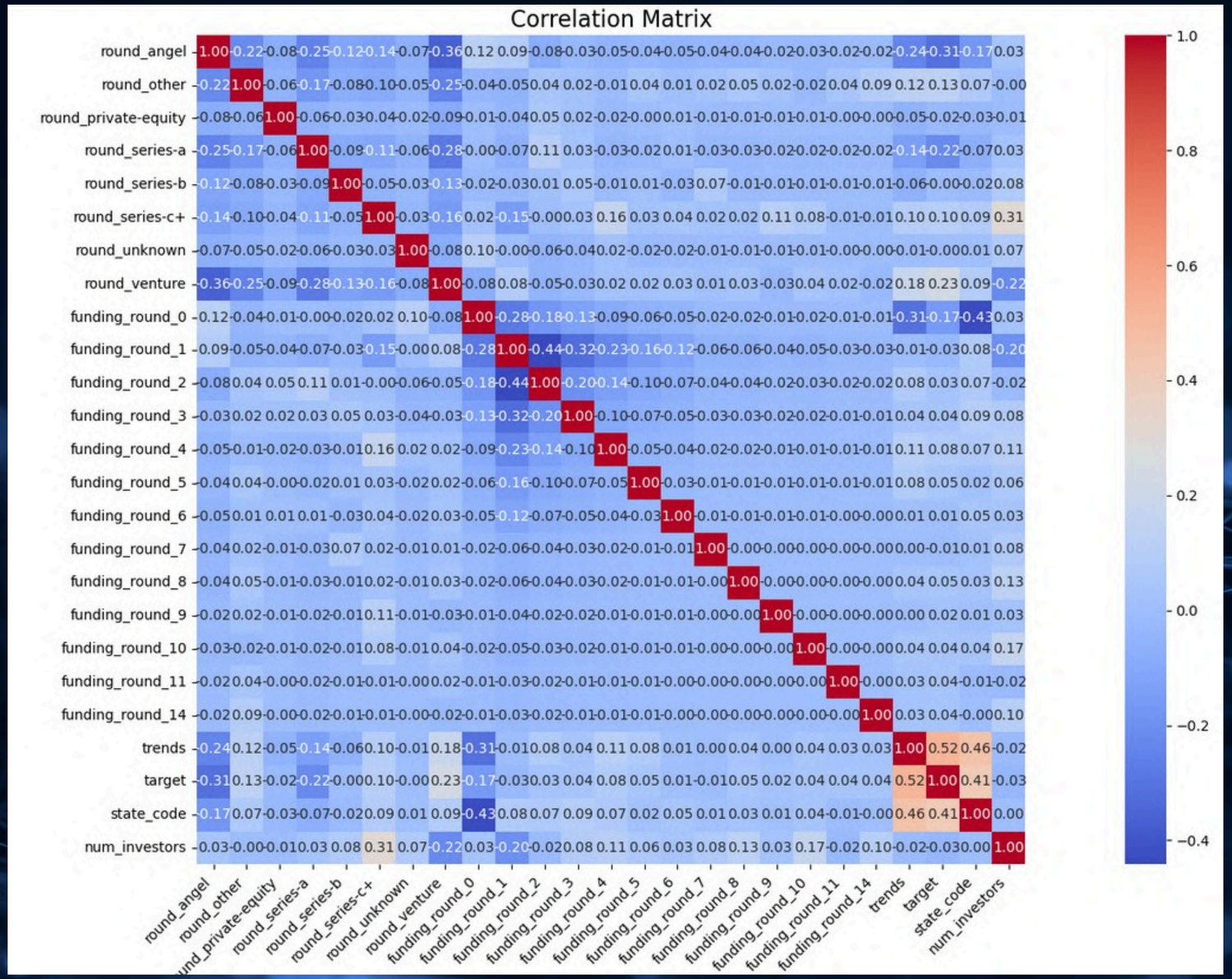
Outlier Handling

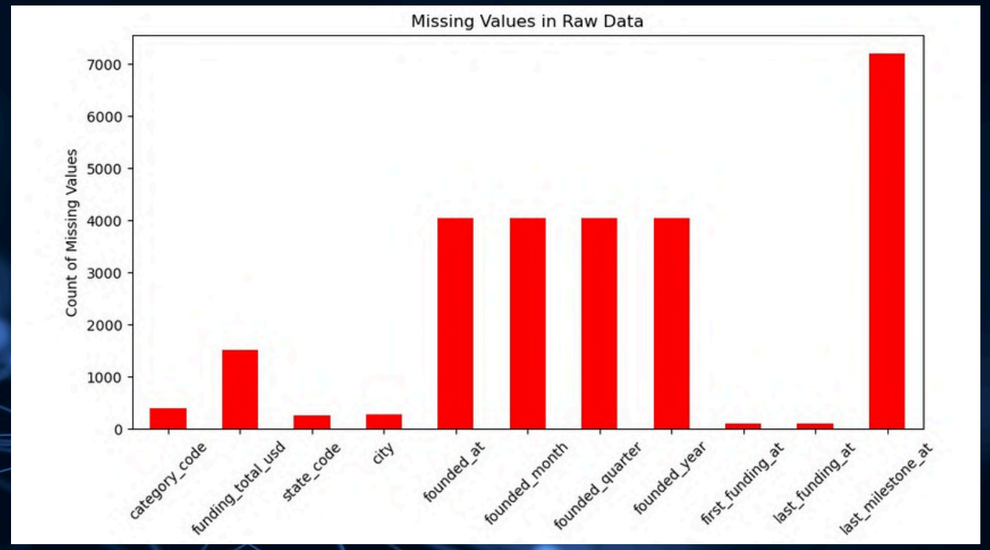
- Removed outlier records for features like team size and funding rounds

Encoding Categorical Features

- Target Encoding: Applied to high-cardinality features (state_code).
- One-Hot Encoding: Used for lower-cardinality features (e.g., funding_round).

DATA PREPROCESSING





MACHINE LEARNING METHODS EXPLORED

BINARY CLASSIFICATION TASK

RANDOM FOREST

- Builds multiple decision trees on random subsets
- Reduces overfitting
- Handles noise well
- Handles mixed data types well

XGBOOST

- Boosting model that corrects previous errors
- Regularization prevents overfitting
- Handles noise and small datasets
- Supports class imbalance via scale_pos_weight

CATBOOST

- It’s designed for categorical features and noisy datasets
- Reduces Overfitting
- Handles missing values

MODEL VALIDATION & PERFORMANCE SUMMARY

Model Validation Process

- Used 5-fold cross-validation
- Trained on 4 subsets, validated on 1
- Repeated 5 times → robust & unbiased metrics
- Test set 20%

Performance Summary

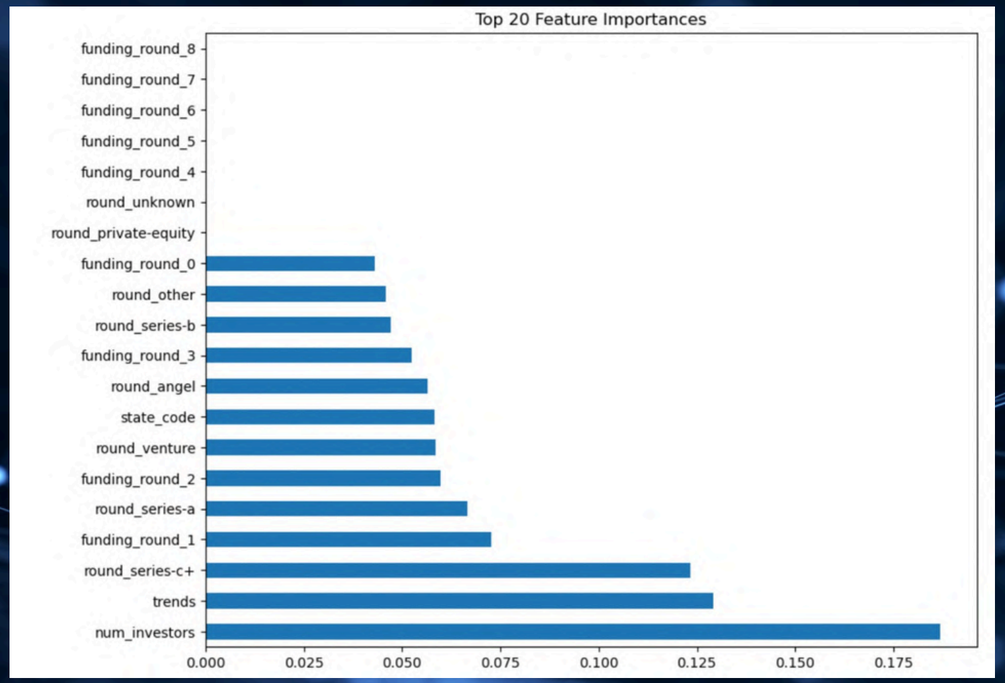
- Optuna-tuned model:
- Tuned parameters: learning rate, tree depth, subsampling ratio, class weights, scale_pos_weight
- Random forest was not able to predict any successes
- Catboost performance was very low (48% accuracy)

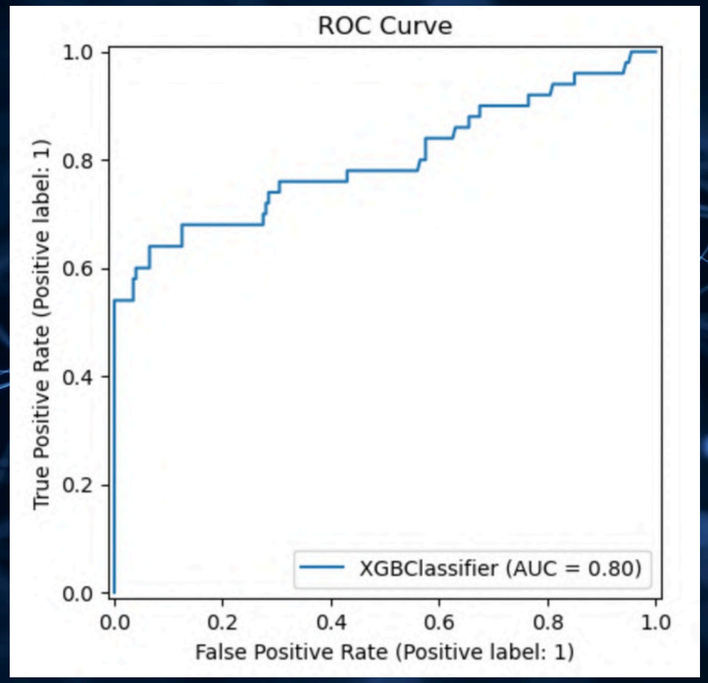
OUR FINAL MODEL

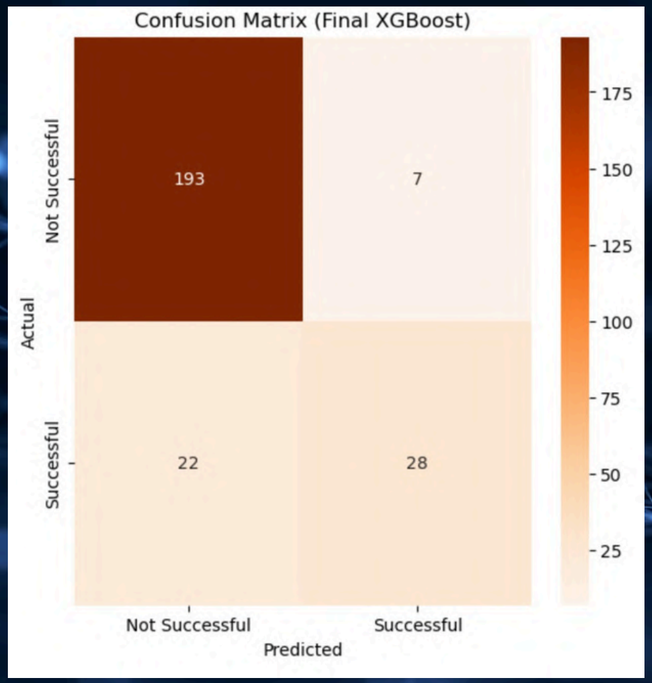
XGBoost

- XGBOOST Performed the best out of all models
- Accuracy: 88 percent

- Precision for 0: 0.9
- Precision for 1 : 0.8
- Weighted average Precision :0.88







CHALLENGES & WORK-AROUNDS

LOW ACCURACY

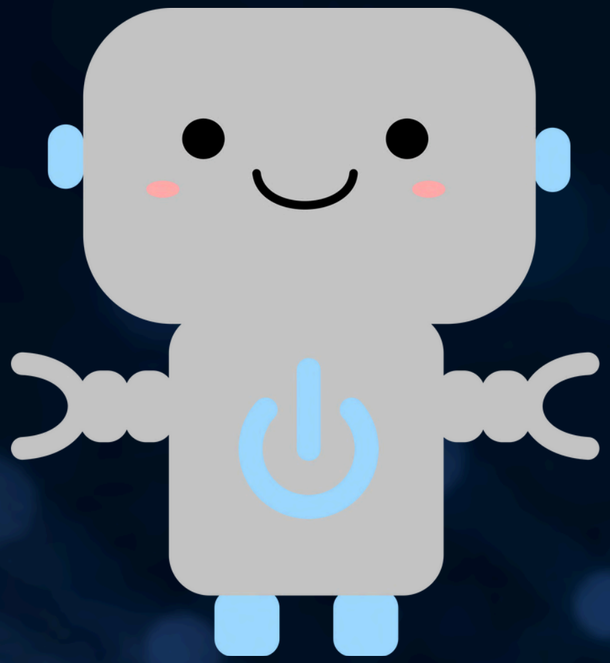
- Hyperparameter Optimization using Optuna:
- Automated search for best hyperparameters
- Tuned: learning_rate, max_depth, subsample, scale_pos_weight

ADDRESSING CLASS IMBALANCE:

- Dataset had few successful startups (originlly l0 percent)
- We added more data for successful startups
- increased successful startups by l5%
- Used XGBoost's scale_pos_weight
- Used oversampling and undersampling techniques.

DIFFERENT TIME PERIODS

- Since people use google more often the relative scale of google searches increases with time that must be taken into account while training model
- Funding amount is also relative to its time period due to inflation and other such factors.



THANK YOU!