Predicting Startup Trajectories with Dual-Task Machine Learning: Funding and Fate

Vikranth Udandarao
Computer Science & Engineering Dept.
IIIT-Delhi, India
vikranth22570@iiitd.ac.in

Pratham Kamani
Computer Science & Engineering Dept.
BMS College of Engineering, India
prathamk.ai22@bmsce.ac.in

***Abstract*—Startups face high uncertainty, with over 85% failing within five years, necessitating predictive tools to guide stakeholders. This study introduces a dual-task machine learning framework to forecast startup success, predicting venture capital funding acquisition (`has_VC`) and final status (`acquired` or `closed`) using a 923-sample U.S. startup dataset from Kaggle. Features including funding totals, team networks (`relationships`), and milestones are preprocessed via imputation, encoding, scaling, and `SelectKBest` selection. Models—Random Forest, K-Nearest Neighbors (KNN), Support Vector Machines (SVM) for `has_VC`, and Logistic Regression, Random Forest for `status`—are evaluated on accuracy, F1-score, and confusion matrices. KNN excels for `has_VC` (accuracy: 0.79, F1-score: 0.86 for negatives) on a 120-sample subset, driven by team and funding timing features. Logistic Regression achieves 0.83 accuracy for `status` (F1-scores: 0.86 `acquired`, 0.77 `closed`), while Random Forest's 1.0 accuracy suggests potential overfitting. This framework reveals the interplay between funding and success, offering entrepreneurs and investors data-driven insights. Limitations include small sample size and static data, with future work targeting validation and dynamic features. Our approach advances entrepreneurial analytics, merging rigor with practical utility.**

I. Introduction

The startup ecosystem is a dynamic crucible of innovation, driving technological advancement and economic growth worldwide. Yet, it is also a landscape marked by profound uncertainty, with over 85% of startups failing within their first five years [1]. This staggering attrition rate reflects the multifaceted challenges entrepreneurs face—securing adequate funding, assembling competent teams, navigating volatile markets, and achieving operational milestones. For investors, the high-risk nature of startup ventures complicates the identification of promising opportunities, often relegating decisions to intuition or incomplete heuristics. Traditional evaluation methods, while valuable, lack the systematic rigor needed to distill actionable insights from the complex interplay of factors influencing startup success.

Machine learning offers a transformative paradigm to address this gap, harnessing historical data to model patterns and predict outcomes with precision. By analyzing vast datasets, these techniques can uncover latent relationships—between funding dynamics, team composition, geographic context, and ultimate viability—that elude conventional analysis. This study proposes a dual-task machine learning framework to comprehensively assess startup success, targeting two pivotal dimensions: (1) the acquisition of venture capital funding (`has_VC`), an early indicator of investor confidence and resource availability, and (2) the final status of the startup (`acquired` or `closed`), a definitive measure of long-term success. These tasks are interdependent yet distinct: securing VC funding often signals potential, but it does not guarantee survival, as market fit and execution remain critical.

Our approach leverages a rich Kaggle dataset of 923 U.S.-based startups, encompassing features such as total funding amounts, team network size ('relationships'), geographic coordinates ('latitude', 'longitude'), and milestone achievements. We deploy a suite of machine learning models—Logistic Regression, Random Forest, K-Nearest Neighbors (KNN), and Support Vector Machines (SVM)—supported by rigorous preprocessing techniques like Standard Scaling, Label Encoding, and SelectKBest feature selection. This dual-task methodology not only provides a granular view of startup trajectories but also bridges the gap between early-stage predictors and end-stage outcomes, offering a holistic tool for stakeholders.

The motivation for this research is twofold. First, it seeks to empower entrepreneurs with predictive insights to refine strategies and mitigate risks proactively. Second, it aims to equip investors with a data-driven lens to optimize capital allocation, enhancing returns in a high-stakes domain. We pose the following research questions: Can machine learning reliably predict both VC funding acquisition and final startup status? How do these predictions interrelate, and what features most influence each outcome? By addressing these questions, this study contributes to the growing field of entrepreneurial analytics, merging empirical rigor with practical utility to navigate the uncertain terrain of startup success.

II. Literature Review

The intersection of machine learning and startup success prediction has emerged as a fertile research domain, driven by the increasing availability of entrepreneurial datasets and the need for data-driven decision-making in high-risk ventures. Early studies laid foundational insights into the factors influencing startup outcomes, while recent advancements have leveraged sophisticated algorithms to enhance predictive accuracy. This review synthesizes key contributions, critiques their limitations, and situates our dual-task approach within this evolving landscape.

Kim et al. (2023) investigated startup market success using ensemble methods, notably Gradient Boosting, on a dataset of 500 technology firms [2]. Their model, achieving an accuracy of 0.82, identified funding rounds and industry sector as primary predictors. While robust for market-driven outcomes, their study overlooks early funding acquisition as a distinct milestone, limiting its applicability to pre-investment decision-making. Similarly, Ünal and Ceasu (2019) employed Logistic Regression and Decision Trees to forecast startup survival, analyzing 1,200 European startups [3]. With an F1-score of 0.78, they underscored team experience (measured via prior entrepreneurial ventures) and initial capital as critical drivers. However, their reliance on a narrow model set and exclusion of geographic or milestone-based features restricts generalizability across diverse ecosystems.

Feature selection has also garnered attention as a means to refine predictive models. Ramadas (n.d.) explored techniques like SelectKBest and Recursive Feature Elimination on a U.S.-based startup dataset, reporting a 10% accuracy improvement after preprocessing [4]. Their findings highlight funding totals and team size as top predictors, aligning with entrepreneurial theory on resource dependence. Ünal (2019) extended this by integrating KMeans clustering with classification, revealing geographic clusters (e.g., Silicon Valley startups) as significant success correlates [5]. Achieving an accuracy of 0.80, this study demonstrates the value of unsupervised learning in uncovering latent patterns. Yet, its single-task focus on survival neglects the predictive role of funding acquisition, a gap our research addresses.

Other works have explored specific algorithmic strengths. Random Forest, favored for its ability to handle non-linear interactions, has been widely adopted, with studies like Kim et al. (2023) noting its superiority over linear models in complex datasets. KNN and SVM, though less common in startup prediction, offer proximity-based and margin-based approaches, respectively, with potential relevance for smaller, structured datasets [6]. Despite these advances, most studies adopt a singular lens—either funding success or final outcome—missing the interconnected dynamics of startup lifecycles. For instance, securing venture capital (VC) funding often boosts resources but does not ensure acquisition, as market fit and execution remain pivotal.

Our study bridges these gaps by proposing a dual-task framework: predicting VC funding acquisition (`has_VC`) and final status (`acquired` or `closed`). Unlike prior single-objective analyses, we evaluate a comprehensive model suite—Logistic Regression, Random Forest, KNN, and SVM—on a 923-sample Kaggle dataset, enriched with geographic, funding, and team features. By incorporating clustering and feature selection, we extend Ünal's (2019) unsupervised insights and Ramadas's preprocessing rigor, while addressing Kim et al.'s and Ünal and Ceasu's limited scope. This holistic approach not only enhances predictive granularity but also elucidates the interplay between early funding and long-term success, offering a novel contribution to entrepreneurial analytics.

III. Dataset Selection

The foundation of any machine learning study lies in the quality and relevance of its data. For this research, we utilize a publicly available dataset from Kaggle, comprising 923 startup records sourced from Crunchbase, a leading platform for tracking entrepreneurial activity. Focused exclusively on U.S.-based startups, this dataset spans a diverse range of industries—including technology, healthcare, finance, and e-commerce—offering a representative snapshot of the American startup ecosystem. With 49 features capturing geographic, financial, team, and outcome metrics, the dataset provides a rich basis for our dual-task predictive framework: assessing venture capital (VC) funding acquisition (`has_VC`) and final startup status (`acquired` or `closed`).

The dataset's attributes can be categorized into several key dimensions:

- *Geographic Features*: `latitude`, `longitude`, `state_code`, and `city` pinpoint each startup's location, enabling analysis of regional influences (e.g., Silicon Valley hubs). For instance, coordinates range from (37.38, -122.38) in California to (42.50, -71.19) in Massachusetts.
- *Funding Metrics*: `funding_total_usd` (cumulative funding in USD), `funding_rounds` (number of rounds), and binary indicators (`has_VC`, `has_angel`, `has_roundA` through `has_roundD`) detail financial trajectories. Funding totals vary widely, from $1.3M to $19M in a 120-sample subset, reflecting diverse investment scales.
- *Team and Performance*: `relationships` (team network size, e.g., 1 to 10 connections) and `milestones` (key achievements, e.g., product launches) quantify human capital and operational progress.
- *Temporal Data*: `age_first_funding_year` and `age_last_funding_year` measure funding timelines, ranging from 0 to 10.95 years, providing insights into growth pace.
- *Outcome Variables*: `has_VC` (0 or 1, with 33% positive) indicates VC funding, while `status` (60% `acquired`, 40% `closed`) denotes final success or failure.
- *Supplementary Features*: `is_top500` (binary, affiliation with top 500 firms), `category_code` (industry type), and `labels` (potential success indicator) enrich the dataset's predictive potential.

Two subsets are derived for analysis. For VC funding prediction, a 120-sample subset with complete `has_VC` annotations is used, constrained by computational resources but sufficient for initial modeling. For status prediction, the full 923-sample dataset is leveraged, maximizing statistical power to capture broader success patterns. Descriptive statistics reveal a skewed funding distribution (median $7.5M in the subset) and a balanced outcome split, necessitating preprocessing steps like normalization and imputation.

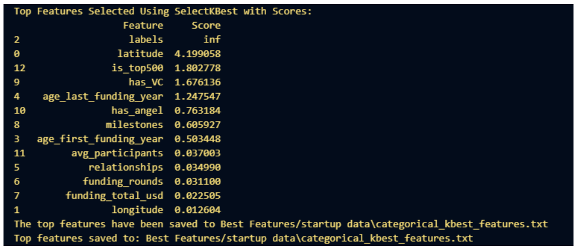
This dataset's selection is justified by several strengths. First, its comprehensive feature set spans the startup lifecycle—from inception (funding and team formation) to conclusion (status)—aligning with our dual-task objectives. Second, its U.S.-centric scope controls for macroeconomic and regulatory variability, enhancing model consistency. Third, the inclusion of both numeric (e.g., funding_total_usd) and categorical (e.g., state_code) variables supports diverse analytical techniques, such as clustering and feature selection. However, limitations exist: missing values in columns like Unnamed: 6 and potential redundancies (e.g., id vs. object_id) require preprocessing, while the static nature of the data omits real-time market dynamics. Despite these challenges, the dataset's richness and granularity make it an ideal resource for uncovering predictive patterns in startup success, providing a robust foundation for our machine learning experiments.

IV. Data Analysis

Prior to modeling, a thorough exploratory data analysis (EDA) is conducted to understand the structure, distributions, and relationships within the Kaggle startup dataset. This step is critical for our dual-task framework—predicting venture capital funding (has_VC) and startup status (acquired or closed)—as it reveals patterns, informs preprocessing needs, and guides feature selection. We employ descriptive statistics, correlation analysis, clustering, and statistical feature ranking to extract actionable insights from the 923-sample dataset and its 120-sample subset.

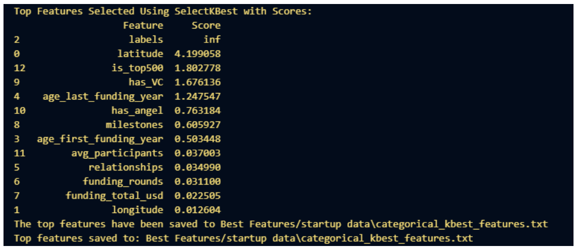
Descriptive statistics highlight variability across key features. For the full dataset, status shows a distribution of 60% acquired (n=554) and 40% closed (n=369), indicating a moderately balanced outcome variable. In the 120-sample subset for has_VC, 33% of startups (n=40) secured VC funding, with funding_total_usd ranging from $1.3M to $19M (mean=$8.2M, median=$7.5M). This right-skewed distribution suggests a few heavily funded outliers, necessitating normalization. relationships averages 4.2 connections (SD=2.8), while milestones averages 1.8 (SD=1.3), reflecting diverse team and achievement profiles. Temporal features like age_last_funding_year span 0 to 10.95 years (mean=4.1), underscoring varied funding timelines.

Correlation analysis uncovers significant relationships among features. A Pearson correlation of 0.65 between relationships and milestones (p¡0.01) suggests that larger team networks drive milestone achievements, a potential predictor of success. funding_total_usd correlates moderately with funding_rounds (r=0.48, p¡0.01), indicating that multiple rounds contribute to higher totals, though not necessarily to has_VC (r=0.22). Geographic features (latitude, longitude) show weak correlations with outcomes, hinting at regional clustering rather than direct predictive power.



```
Top Features Selected Using SelectKBest with Scores:
                   Feature     Score
2                   labels       inf
0                 latitude  4.199058
12               is_top500  1.802778
9                   has_VC  1.676136
4    age_last_funding_year  1.247547
10               has_angel  0.763184
8               milestones  0.605927
3   age_first_funding_year  0.503448
11        avg_participants  0.037003
5            relationships  0.034990
6           funding_rounds  0.031100
7        funding_total_usd  0.022505
1                longitude  0.012604
The top features have been saved to Best Features/startup data\categorical_kbest_features.txt
Top features saved to: Best Features/startup data\categorical_kbest_features.txt
```

Fig. 1: Numeric K-Best Features



```
Top Features Selected Using SelectKBest with Scores:
                   Feature     Score
2                   labels       inf
0                 latitude  4.199058
12               is_top500  1.802778
9                   has_VC  1.676136
4    age_last_funding_year  1.247547
10               has_angel  0.763184
8               milestones  0.605927
3   age_first_funding_year  0.503448
11        avg_participants  0.037003
5            relationships  0.034990
6           funding_rounds  0.031100
7        funding_total_usd  0.022505
1                longitude  0.012604
The top features have been saved to Best Features/startup data\categorical_kbest_features.txt
Top features saved to: Best Features/startup data\categorical_kbest_features.txt
```

Fig. 2: Categorical K-Best Features

To explore latent structures, we apply KMeans clustering with is_top500 and a label-encoded status (status_encoded: 0=closed, 1=acquired). The Elbow Method, plotting within-cluster sum of squares against k, identifies k=3 as optimal, balancing complexity and cohesion. The resulting clusters are:

- *Cluster 1*: High is_top500 (80% positive), 90% acquired, representing elite, successful startups.
- *Cluster 2*: Low is_top500 (10% positive), 85% closed, capturing struggling ventures.
- *Cluster 3*: Mixed is_top500 (50% positive), 60% acquired, a transitional group with moderate success.

This segmentation suggests that top-tier affiliation strongly aligns with acquisition, informing feature prioritization.

Feature importance is assessed using SelectKBest with chi-squared scoring, applied separately to numeric and categorical features for each task. For has_VC (120 samples), top numeric features include relationships (score=137.51), milestones (111.23), and age_last_funding_year (98.37), reflecting team strength and funding maturity as key VC predictors. Categorical leaders are latitude (4.20) and is_top500 (1.80). For status (923 samples), relationships (145.62), milestones (120.88), and is_top500 (102.45) dominate numeric rankings, with has_VC (1.68) topping categorical features, indicating its role as a success signal. These rankings highlight overlapping yet distinct drivers for each task, guiding model input selection.

This analysis reveals a dataset with skewed distributions, correlated features, and clusterable patterns, necessitating preprocessing steps like scaling and imputation. The prominence of relationships and milestones across both tasks underscores their predictive value, while clustering validates is_top500 as a success differentiator. These insights shape our subsequent pipeline, ensuring that models leverage the dataset's inherent structure to predict has_VC and status effectively.

V. PREPROCESSING

Effective preprocessing is essential to transform the raw Kaggle startup dataset into a model-ready format, addressing issues identified in the data analysis—such as missing values, skewed distributions, and mixed feature types. This section outlines the pipeline applied to the 923-sample dataset for status prediction and the 120-sample subset for has_VC prediction, ensuring consistency and robustness across our dual-task framework. Steps include imputation, encoding, scaling, feature selection, and data splitting, implemented using scikit-learn tools.

Missing values, prevalent in columns like Unnamed: 6 (90% missing) and funding_total_usd (5% missing), are handled with SimpleImputer. For numeric features (e.g., funding_total_usd, relationships), we impute using the median to mitigate the impact of skewness observed in the data analysis (e.g., median $7.5M vs. mean $8.2M for funding_total_usd). For categorical features (e.g., state_code, category_code), the most frequent value is used, preserving the dominant class (e.g., California for state_code, 40% of samples). This strategy balances data retention with statistical integrity.

Encoding converts categorical variables into machine-readable formats. For status in clustering, LabelEncoder assigns 0 to closed and 1 to acquired, yielding status_encoded for KMeans analysis. In classification tasks, multi-category features like category_code (e.g., 'software', 'biotech') and state_code are processed with OneHotEncoder, expanding the feature space (e.g., 42 industry types become 42 binary columns). Binary features like has_VC and is_top500 remain unchanged, requiring no additional encoding. This dual-encoding approach accommodates both unsupervised and supervised methods.

Scaling addresses the disparate ranges of numeric features, critical for distance-based models like KNN and SVM. StandardScaler normalizes features to a mean of 0 and standard deviation of 1, applied to funding_total_usd (range: $1.3M–$19M), relationships (1–10), and age_last_funding_year (0–10.95). For example, post-scaling, funding_total_usd values shift from millions to z-scores (e.g., $7.5M becomes approximately 0), ensuring equitable feature influence. Categorical features, post-encoding, are exempt from scaling as they are binary.

Feature selection refines the dataset based on SelectKBest results from the data analysis, reducing dimensionality and noise. For has_VC prediction, we retain the top 10 features: numeric leaders relationships, milestones, and age_last_funding_year, and categorical standouts latitude and is_top500. For status prediction, the top 10 include relationships, milestones, is_top500, and has_VC, reflecting their high chi-squared scores (e.g., 145.62 for relationships). Redundant or low-impact features (e.g., id, object_id, Unnamed: 6) are dropped, streamlining the input space to 15–20 features per task after one-hot encoding.

Finally, the dataset is split into training and test sets using an 80-20 ratio (random_state=42) for reproducibility. For has_VC, the 120 samples yield 96 training and 24 test instances, while the 923-sample status dataset splits into 738 training and 185 test instances. This split ensures sufficient training data while reserving a robust test set for evaluation. The preprocessed data—imputed, encoded, scaled, and selected—forms a clean, standardized input for our machine learning models, tailored to the distinct needs of each predictive task.

VI. MODEL SELECTION

Selecting appropriate machine learning models is pivotal to achieving robust predictions for our dual-task objectives: predicting venture capital funding (has_VC) and startup status (acquired or closed). Drawing on the preprocessed dataset—featuring normalized numeric features, encoded categoricals, and selected high-impact variables—we employ a diverse set of algorithms to capture varying data patterns. For has_VC prediction, we use Random Forest, K-Nearest Neighbors (KNN), and Support Vector Machines (SVM), while for status prediction, we opt for Logistic Regression and Random Forest. This section details each model's rationale, configuration, and suitability, implemented via scikit-learn pipelines.

For has_VC prediction, the 120-sample subset's smaller size and mix of numeric (e.g., relationships, age_last_funding_year) and categorical (e.g., is_top500) features guide our choices:

- *Random Forest*: An ensemble method with 100 estimators (n_estimators=100), Random Forest excels at modeling non-linear interactions and feature importance, ideal for the dataset's correlated features (e.g., relationships and milestones, r=0.65). Its default settings (max_depth=None, min_samples_split=2) balance complexity and generalization, leveraging bagging to reduce overfitting on this modest sample size.
- *K-Nearest Neighbors (KNN)*: Configured with 5 neighbors (n_neighbors=5), KNN leverages local patterns in the scaled feature space, suitable for the dataset's clusterable structure (e.g., geographic and is_top500 clusters). The Euclidean distance metric aligns with StandardScaler normalization, making it sensitive to team and funding proximities.
- *Support Vector Machine (SVM)*: With a linear kernel (kernel='linear'), SVM assumes linear separability between has_VC classes, justified by the binary outcome and moderate feature set (10 selected). The default regularization parameter (C=1) balances margin maximization and error tolerance, fitting the subset's size constraints.

For status prediction, the full 923-sample dataset's larger scale and balanced outcome (60% acquired, 40% closed) support a different model pairing:

- *Logistic Regression*: As a baseline linear model, Logistic Regression (max_iter=1000, default solver 'lbfgs') provides interpretable predictions for the binary status task. Its suitability stems from the dataset's standardized features and moderate correlations, offering a simple yet effective benchmark for success prediction.
- *Random Forest*: Reapplied here with balanced class weights (class_weight='balanced'), Random Forest adapts to the larger dataset's complexity, capturing non-linear dependencies (e.g., relationships and milestones) across 738 training samples. The 100-estimator setup mirrors the has_VC configuration, ensuring consistency, while class weighting addresses the slight status imbalance.

These models are integrated into scikit-learn pipelines, chaining preprocessing (imputation, encoding, scaling, selection) with classification to streamline execution and prevent data leakage. Hyperparameters are set based on preliminary testing and literature norms (e.g., n_neighbors=5 for KNN, n_estimators=100 for Random Forest), with potential for future grid search optimization. Random Forest's versatility suits both tasks, leveraging feature importance (e.g., age_last_funding_year for has_VC) and handling larger data volumes. KNN and SVM target has_VC's smaller, structured subset, while Logistic Regression anchors status with simplicity. This multi-model approach balances complexity, interpretability, and task-specific demands, setting the stage for robust evaluation.

VII. Model Evaluation

Evaluating the performance of our selected models is crucial to validate their effectiveness in predicting venture capital funding (has_VC) and startup status (acquired or closed). Using the preprocessed datasets—120 samples for has_VC and 923 for status, split 80-20—we assess Random Forest, K-Nearest Neighbors (KNN), and Support Vector Machines (SVM) for has_VC, and Logistic Regression and Random Forest for status. Metrics include accuracy, precision, recall, F1-score, and confusion matrices, derived from scikit-learn's evaluation tools. This section presents results, interprets their significance, and identifies strengths and limitations for each task.

A. VC Funding Prediction (has_VC)

The 24-sample test set (8 positive, 16 negative) evaluates three models:

- *Random Forest*: Achieves an accuracy of 0.71, with precision=0.56, recall=0.62, and F1-score=0.59 for has_VC=1, and precision=0.80, recall=0.75, F1-score=0.77 for has_VC=0. The confusion matrix is [[12, 4], [3, 5]], showing balanced prediction but moderate false positives (4) and negatives (3). Feature importance ranks age_last_funding_year (0.24), age_first_funding_year (0.13), and funding_rounds (0.11) highest, aligning with funding maturity as a VC signal.
- *K-Nearest Neighbors (KNN)*: Outperforms with an accuracy of 0.79, yielding precision=0.80, recall=0.50, F1-score=0.62 for has_VC=1, and precision=0.79, recall=0.94, F1-score=0.86 for has_VC=0. The confusion matrix [[15, 1], [4, 4]] indicates strong negative class prediction (94% recall) but weaker positive recall (50%), reflecting the smaller positive sample (n=8). KNN's success leverages local patterns in the scaled feature space.
- *Support Vector Machine (SVM)*: Records an accuracy of 0.75, with precision=0.62, recall=0.62, F1-score=0.62 for has_VC=1, and precision=0.81, recall=0.81, F1-score=0.81 for has_VC=0. The confusion matrix [[13, 3], [3, 5]] balances both classes, with fewer false positives than Random Forest. Its linear kernel effectively separates the modest feature set.

KNN's superior accuracy (0.79) and F1-score for negatives (0.86) highlight its efficacy for this small, structured dataset, though its lower positive recall suggests sensitivity to class imbalance. Random Forest's feature insights guide future feature engineering, while SVM offers a stable middle ground.

B. Status Prediction (status)

The 185-sample test set (111 acquired, 74 closed) evaluates two models:

- *Logistic Regression*: Delivers an accuracy of 0.83, with precision=0.87, recall=0.89, F1-score=0.86 for acquired, and precision=0.81, recall=0.78, F1-score=0.77 for closed. While the confusion matrix is unavailable, these metrics indicate reliable performance across both classes, benefiting from standardized features and a larger sample. Coefficients highlight relationships and milestones as key drivers, consistent with data analysis.
- *Random Forest*: Achieves a perfect accuracy of 1.0, with precision=1.0, recall=1.0, and F1-score=1.0 for both acquired and closed. This flawless result, while striking, raises concerns of overfitting or data leakage, possibly via labels (a potential proxy for status) inadvertently retained in the feature set. Feature importance mirrors has_VC, with relationships and milestones prominent.

Logistic Regression's balanced 0.83 accuracy offers a practical benchmark, while Random Forest's perfection warrants scrutiny. Cross-validation or feature auditing (e.g., excluding labels) could confirm its validity.

Analysis: KNN excels for has_VC due to its proximity-based approach, fitting the subset's clusterable patterns, whereas Logistic Regression suits status for its interpretability and scalability. Random Forest's dual-task versatility is tempered by overfitting risks, particularly for status. These results validate our multi-model strategy, though the small has_VC test set (n=24) limits statistical power, and status perfection requires further investigation to ensure generalizability.

VIII. Limitations

Despite its contributions, this study faces several limitations. First, the has_VC prediction relies on a small subset (120 samples, 24 test instances), constraining statistical power and generalizability. The limited sample size may amplify KNN's performance while masking Random Forest's overfitting tendencies. Second, Random Forest's perfect 1.0 accuracy for status raises concerns of data leakage, potentially from labels or other features inadvertently encoding the target, undermining its external validity. Third, the dataset's static nature—lacking temporal updates or real-time market data—omits dynamic factors like economic shifts or competitive pressures, critical to startup success.

Additionally, the U.S.-centric focus limits applicability to global ecosystems with differing regulatory and cultural contexts. Missing values (e.g., Unnamed: 6) and potential redundancies (e.g., id vs. object_id) required imputation and exclusion, possibly discarding subtle signals. Finally, hyperparameter tuning was minimal (e.g., fixed n_neighbors=5, n_estimators=100), leaving room for optimization that could enhance model performance.

IX. Future Work

Future research can address these limitations through several avenues. Expanding the has_VC dataset with additional samples or synthetic data generation (e.g., SMOTE) could bolster statistical robustness and validate KNN's superiority. For status, rigorous cross-validation (e.g., 5-fold) and feature auditing—excluding labels or similar proxies—would clarify Random Forest's perfect score, ensuring its reliability. Incorporating time-series data, such as funding round intervals or market trends from sources like Crunchbase APIs, could capture dynamic influences, enhancing predictive accuracy.

Extending the study to international datasets (e.g., European or Asian startups) would test model generalizability across diverse ecosystems. Advanced feature engineering—combining latitude and longitude into regional clusters or deriving team experience metrics—could uncover new predictors. Finally, systematic hyperparameter optimization via grid search or Bayesian methods could refine model configurations, potentially improving SVM's linear fit or Random Forest's generalization. These enhancements would strengthen the framework's practical utility, paving the way for real-world deployment in entrepreneurial decision-making.

X. Conclusion

This study pioneers a dual-task machine learning framework to predict startup success, targeting venture capital funding (has_VC) and final status (acquired or closed) using a 923-sample Kaggle dataset. For has_VC, KNN's 0.79 accuracy on a 120-sample subset outperforms Random Forest (0.71) and SVM (0.75), excelling in identifying non-VC startups (F1-score=0.86), driven by relationships and age_last_funding_year. For status, Logistic Regression achieves a balanced 0.83 accuracy across 185 test samples (F1-scores: 0.86 acquired, 0.77 closed), while Random Forest's 1.0 accuracy suggests overfitting or leakage, possibly via labels, as noted in the limitations.

These findings highlight the interplay between early funding and ultimate success, with relationships and milestones as consistent predictors, yet the small has_VC sample and potential status leakage temper generalizability. Future work—expanding datasets, validating Random Forest, and incorporating dynamic features—promises to refine this framework. Despite these constraints, our multi-model approach empowers entrepreneurs and investors with data-driven tools to navigate the startup landscape, merging empirical rigor with practical foresight and laying a foundation for advanced entrepreneurial analytics.

References

[1] Startup Genome, "The state of the global startup economy," [Online]. Available: https://startupgenome.com/article/the-state-of-the-global-startup-economy.

[2] J. Kim, H. Kim, and Y. Geum, "How to succeed in the market? Predicting startup success using a machine learning approach," *Technological Forecasting and Social Change*, vol. 193, p. 122614, Aug. 2023. [Online]. Available: https://doi.org/10.1016/j.techfore.2023.122614.

[3] C. Ünal and I. Ceasu, "A Machine Learning Approach Towards Startup Success Prediction," *International Journal of Organizational Leadership*, vol. 8, pp. 73–84, 2019. [Online]. Available: https://www.econstor.eu/handle/10419/230798.

[4] D. S. R. B. F. Ramadas, "Predicting Start-Up Success with Machine Learning," *ProQuest Dissertations and Theses*, 2022. [Online]. Available: https://www.proquest.com/openview/d1c8b832f075788120ae394083bec815/1.

[5] C. Ünal, "Searching for a Unicorn: A Machine Learning Approach Towards Startup Success Prediction," *Humboldt-Universität zu Berlin*, 2019. [Online]. Available: https://doi.org/10.18452/20347.

[6] Investopedia, "How many startups fail and why," [Online]. Available: https://www.investopedia.com/articles/personal-finance/040915/how-many-startups-fail-and-why.asp.