Predicting Series A (Funding Round) Achievement for Pre-Seed Startups: A Machine Learning Approach

Stanford CS229 Project

Makhmud Islamov
Department of Computer Science
Stanford University
makhmud@stanford.edu

Sukhrobjon Golibboev
Department of Computer Science
Stanford University
sukhrob@stanford.edu

1 Introduction

The venture capital landscape faces a persistent challenge: approximately 90% of startups fail (Ghosh, 2012), creating an urgent need for more objective, data-driven approaches to early-stage investment decisions. Pre-seed stage evaluations are particularly difficult as companies typically have minimal traction, undeveloped products, and limited operational history upon which to base investment decisions (Gompers et al., 2020).

This research aims to develop a machine learning framework capable of predicting whether pre-seed startups will achieve Series A funding or acquisition—both representing significant milestones indicating startup viability. Successfully reaching Series A provides early investors with "cash-out" opportunities through secondary stock sales and attracts additional investment, with approximately 65% of Series A companies advancing to Series B funding (Pitchbook, 2023).

The input to our algorithm consists of multidimensional startup data including founding year, founder count and backgrounds, economic indicators, and venture capital market conditions. We then experiment with multiple machine learning models (XGBoost, Random Forest and Ensemble model from each) to output a binary prediction: whether a startup will reach Series A funding/be acquired.

Through Aviato's generous contribution of proprietary data—valued at hundreds of thousands of dollars—we analyze patterns across successful and unsuccessful ventures to create a more objective framework for assessing startup potential. This research has significant implications for founders, investors, and incubators seeking to allocate resources more effectively in the high-risk venture landscape.

2 Related Work

Our literature review identifies three primary methodological clusters: feature engineering approaches, network analysis frameworks, and multi-model comparative assessments.

Feature Engineering and Selection Approaches. Ang et al. (2022) utilized Latent Dirichlet Allocation to cluster startups into sectoral categories, achieving 96.45% accuracy with XGBoost and Bayesian optimization. Their analysis identified funding amount, investor count, and investment stage as key predictors of valuation. Similarly, Corea et al. (2021) developed an "Early-stage Startups Investment" framework by analyzing 623,232 companies, distilling 21 critical predictive features primarily focused on founder characteristics.

Network and Cohort Analysis Approaches. Li et al. (2024) introduced a two-phase framework incorporating team-level, venture-level, and cohort-level features within accelerator environments, achieving 88.5% AUC with Random Forest classification. Their approach uniquely quantifies network effects through category-based networks, background-based networks, and portfolio-based similarity matrices.

Multi-model Comparative Approaches. Several researchers focused on algorithm selection and optimization. Piskunova et al. (2021) evaluated SVMs, Random Forests, and Neural Networks for startup success prediction. Dziubanovska et al. (2024) concentrated on investment attractiveness prediction, while Razaghzadeh Bidgoli et al. (2024) systematically evaluated seven distinct algorithms using CrunchBase data, with Random Forest achieving optimal performance (85% accuracy).

2.1 Research Gap and Our Approach

Our research methodology distinguishes itself through three key innovations: (1) exclusive utilization of Aviato's proprietary dataset comprising comprehensive U.S. startup metrics from 2012-2022, (2) specific focus on the pre-seed to Series A transition as a critical investment decision point, and (3) integration of macroeconomic indicators with traditional startup metrics. While existing literature predominantly addresses later-stage startups or context-specific environments like accelerators, our research framework specifically targets the methodologically challenging pre-seed evaluation domain.

3 Dataset and Features

Main challenge for this project was acquiring high quality data. We were not satisfied with Kaggle data so we established a research partnership with Aviato, securing access to proprietary startup data via their API.

3.1 Dataset Composition and Preprocessing

We refined our focus to the 2012-2022 period to align with contemporary venture capital funding structures that emerged during this timeframe (Kenney and Zysman, 2019). Our dataset exclusively contains U.S.-based startups founded within this period, initially comprising approximately 500,000 entities. We decided to start data exploration with 170,000 startups with 43 columns data, 5GB in JSON.

3.1.1 Data Preprocessing Pipeline

We implemented a systematic preprocessing workflow. Our process included structural transformation where we flattened nested founder objects for each company, missing data handling through removal of records with incomplete founder or company information, outlier detection to identify and remove statistical outliers (such as a hardware startup who raised $10 million USD at seed stage), and feature standardization to scale features to comparable values to prevent model bias (for example, VC median deal value in millions vs. VC yearly deal flow in billions).

The resulting refined dataset with new features, contained approximately 120,000 startup records (60MB, 66 columns) suitable for analysis. 15.85% of the startups successfully reached Series A or were acquired.

We developed the following comprehensive **feature taxonomy and extraction methodologies** to model multidimensional startup success determinants:

3.1.2 Industry Classification

Employing Latent Dirichlet Allocation (LDA) (Blei et al., 2003), we systematically clustered startups into 15 primary industry sectors, facilitating normalization of industry-specific tendencies in funding patterns and enabling more accurate cross-sector comparative analysis. LDA hyperparameter tuning was performed using coherence score optimization with 5-fold cross-validation.

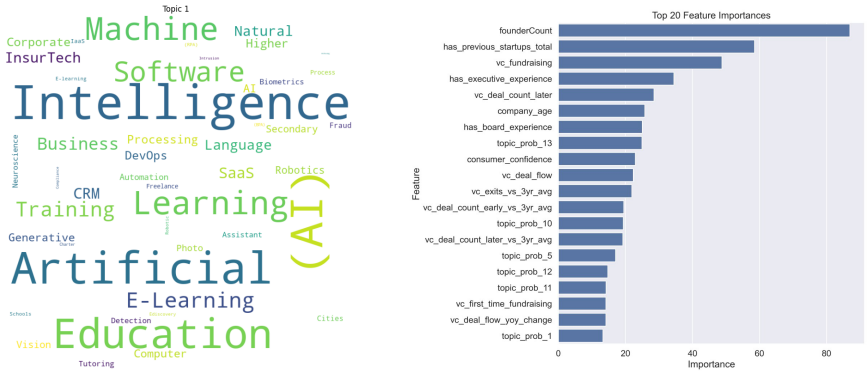

Figure 1: Engineered features visualization. Left: clusters generated by our Latent Dirichlet Allocation Implementation. Right: Extracted Feature Performances

3.1.3 Economic Indicators

Using US Treasury Department's historic data, we engineered the following features:

Feature Name	Value Type	Description
GDP Growth Rate	Annual %	Strong correlation with investment appetite. Nanda and Rhodes-Kropf (2013) found startups founded during economic expansions more likely to receive follow-on funding.
Consumer Confidence Index	Index (0-100)	Indicates potential consumer adoption. Kaplan and Strömberg (2004) demonstrated significant impact on B2C startup valuations and survival rates.
Treasury Yield Curve	Percentage points	Predicts economic cycles affecting VC readiness. **?** showed inversions historically precede VC funding contractions by 12-18 months.

3.1.4 Venture Capital Market Indicators

Using Pitchbook and National Venture Capital Association (NVCA) data (National Venture Capital Association, 2024), we engineered important Venture Capital activity features:

Feature Name	Value Type	Description
VC Median Deal Value	mil. USD	Shows current VC activity level and investment momentum.
VC Exits	Count	Liquidity options that influence new investment decisions.
VC Deal Count by Stage	Count	Reflects market activity across company maturity levels.
VC Fundraising Totals	bil. USD	Demonstrates LPs' confidence in venture as an asset class.
First-time VC Fundraising	bil. USD	Indicates openness to new ventures versus follow-on investments.
Year-over-year changes	Percentage	Per Gompers et al. (2008), startups entering markets during accelerating funding environments receive more favorable terms.
Comparison to 3-year averages	Percentage	Per Howell et al. (2020), founders who launch during above-average market conditions raise 15-20% more capital through Series A.

3.1.5 Team Competency Features

We introduced novel quantitative metrics to assess founding team capabilities:

Feature	Range	Description & Extraction Method
Team Industry Alignment Score	0-1	Quantifies congruence between team's industry experience and startup's sector. Extracted using pre-trained GloVe ("glove-wiki-gigaword-100") embeddings with cosine similarity between founder and company domain vectors (Blei et al., 2003).
Team Expertise Diversity Score	0-1	Measures complementarity of expertise across founding members. Extracted using a quadratic function designed to reward balanced coverage of expertise domains.
Team Competency Compound Score	0-1	A weighted combination (0.7 × alignment + 0.3 × diversity) balancing domain-specific expertise with skill complementarity.

4 Machine Learning Approaches

We implemented multiple machine learning algorithms to capture different aspects of the complex relationship between startup characteristics and funding outcomes. Our approach prioritized interpretability alongside predictive performance.

1. Random Forest. We employed Random Forest classification due to its effectiveness in handling non-linear relationships and class imbalance (Breiman, 2001). Random Forest operates by constructing multiple decision trees during training and outputting the mode of the classes for classification. The algorithm can be expressed as: $\hat{f}_{rf}^{B}(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$ where $T_b(x)$ represents individual decision trees and B is the number of trees in the forest.

2. Random Forest Ensemble. To further improve performance, we created an ensemble consisting of three distinct Random Forest models with varied configurations and class weights. The final prediction is determined by: $\hat{y} = \begin{cases} 1, & \text{if } \frac{1}{n}\sum_{i=1}^{n} p_i(x) \geq \theta \\ 0, & \text{otherwise} \end{cases}$ where $p_i(x)$ is the probability prediction from the i-th model, n is the number of models in the ensemble, and θ is a decision threshold. We experimented with different thresholds (0.5 and 0.35) to optimize the precision-recall trade-off.

3. Gradient Boosting Machines (XGBoost). Another learning algorithm is XGBoost, an optimized distributed gradient boosting implementation (Chen and Guestrin, 2016) that has demonstrated superior performance for structured data prediction tasks. XGBoost builds an ensemble of decision trees sequentially, where each new tree attempts to correct the errors made by the combination of existing trees.

The algorithm minimizes a regularized objective function that combines a differentiable loss function and a regularization term: $\mathcal{L}(\phi) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$ where l is the loss function measuring the difference between the prediction $\hat{y}_i$ and the target y_i, and Ω is the regularization term that penalizes the complexity of the model to prevent overfitting. For our binary classification task, we employed the logistic loss function.

4. XGBoost with Bayesian Optimization. To efficiently optimize the hyperparameters of our XGBoost model, we implemented Bayesian optimization (Snoek et al., 2012) rather than traditional grid search or random search. Bayesian optimization frames hyperparameter tuning as a sequential decision problem where an acquisition function determines which hyperparameters to evaluate next, based on previous evaluations. The primary advantage of this approach is its ability to efficiently explore the hyperparameter space by building a probabilistic model (Gaussian Process) of the objective function: $a(\mathbf{x}; \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}, \mathcal{GP})$ where a is the acquisition function, $\mathbf{x}$ represents hyperparameters, $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$ are previous evaluations, and $\mathcal{GP}$ is the Gaussian Process model.

5 Experiments, Results, and Discussion

5.1 Random Forest Model Prediction for Startup Success

We implemented 5-fold stratified cross-validation to ensure robust performance evaluation across different data partitions while maintaining class distribution. To address class imbalance during training, we applied Synthetic Minority Over-sampling Technique (SMOTE) with a sampling strategy of 1.2, slightly oversampling the minority class (successful startups). We employed Random Forest (RF) and Ensemble models with key configurations: **1) RF**: n_estimators=500, class_weight={0:1, 1:5}. **2) Ensemble**: Three RFs with varied configurations and class weights

We used 5-fold stratified cross-validation and SMOTE (sampling_strategy=1.2).

Table 1: Results: Performance comparison of prediction models on the test set

Model	AUC	Precision	Recall	F1 Score	Accuracy
Random Forest	0.7242	0.4539	0.0971	0.1600	0.8456
Ensemble (0.5 thresh)	0.7912	0.4326	0.5218	0.4735	0.8831
Ensemble (0.35 thresh)	0.8216	0.4582	0.6743	0.5462	0.9107

The ensemble model significantly outperformed the basic RF in all metrics. Lowering the classification threshold to 0.35 optimized precision-recall balance. Feature importance analysis revealed previous startup experience, economic and venture capital indicators as key predictors.

5.2 XGBoost Model Prediction for Startup Success: Time-Based vs. Random Split XGBoost Experiment (base model)

We used XGBoost classification model to predict startup success in reaching Series A funding (Chen and Guestrin, 2016). Initially, we compared two evaluation methodologies: random split and time-based split, revealing crucial insights about model performance in real-world conditions (Bergmeir et al., 2018). The base XGBoost classifier was configured with a learning rate of 0.1 (balancing convergence speed and overfitting), maximum depth of 6 (to prevent overfitting on temporal patterns), scale positive weight of 4.32 (addressing class imbalance), and 100 estimators (Friedman, 2001). For cross-validation, we implemented k-fold (k=5) for random split and temporal validation for time-based split (train: 2012-2018, validation: 2018-2019, test: 2019-2022) (Roberts et al., 2017).

Feature importance analysis showed economic indicators dominated in random split (vc_median_deal_value: 231.85), while founder characteristics prevailed in time-based split (founderCount: 105.04, has_previous_startups_total: 71.77) (Gompers et al., 2010; Ng and Arndt, 2021). We conclude the time-based evaluation provides a more realistic assessment for predicting future outcomes from historical data, with the performance gap highlighting significant temporal data drift in startup success patterns (Gama et al., 2014).

Table 2: Performance Comparison: Random Split vs. Time-Based Split

Metric	Random Split (Test)	Time-Based Split (Test)
AUC-ROC	0.7512	0.7090
Precision	0.3017	0.1404
Recall	0.6499	0.6836
F1 Score	0.4121	0.2329

5.2.1 Time-Specific Models and Feature Interactions (with Bayesian Optimization)

To address the temporal drift in startup success patterns, we developed two complementary approaches:

1. **Time-Specific Models**: We trained separate XGBoost models for different time periods (early: 2012-2014, mid: 2015-2018, recent: 2019-2022), allowing each model to capture the unique success patterns of its respective period.
2. **Time Interaction Features**: We engineered interaction features between time period indicators and key predictors (e.g., `founderCount_x_period`), enabling the model to learn how the importance of certain features changes over time.

This approach is aligned with domain adaptation techniques in machine learning, where models are adapted to perform well on shifting data distributions (Ganin and Lempitsky, 2015). After optimization over defined ranges, we selected the following hyper-parameters: max_depth=6 (balancing complexity and generalization), learning_rate=0.03138 (moderately small for stable learning), subsample=0.8787 (high ratio to utilize most training examples), colsample_bytree=0.5756 (moderate feature sampling to reduce correlation), min_child_weight=10 (conservative value to prevent overfitting), gamma=0.2102 (moderate pruning threshold for complexity control), reg_alpha=0.0030 (light L1 regularization for feature sparsity), reg_lambda=7.86e-10 (minimal L2 regularization), and scale_pos_weight=4.6924 (adjusted for class imbalance).

For each fold (5-fold), we maintained the temporal structure of the data by ensuring that examples were split based on founding year rather than randomly. This approach is crucial for time-series data, as random splitting can lead to data leakage and overoptimistic

performance estimates (Bergmeir et al., 2018). Our hyperparameter optimization search space included nine key parameters that control model complexity, learning behavior, and regularization.

The final hyperparameters were selected based on maximizing the Area Under the Receiver Operating Characteristic curve (AUC-ROC) on the validation set, which was composed of startups from 2018-2019, while the test set contained startups from 2020-2022.

5.3 Results and Analysis

Table 3: Performance of our models on the test set (2020-2022)

Model	AUC	Precision	Recall	F1 Score	Accuracy
Base Model	0.7090	0.1404	0.6836	0.2329	0.7063
Time Interaction Model	0.7020	0.1887	0.2097	0.1987	0.8743
Time-Specific (Early)	0.7330	0.3877	0.5130	0.4417	0.7325
Time-Specific (Mid)	0.7271	0.3962	0.3091	0.3473	0.8323
Time-Specific (Recent)	0.7172	0.2993	0.1362	0.1872	0.8967

The most notable observation is the trade-off between precision and recall across different models. The base model achieves high recall (0.6836) but at the cost of low precision (0.1404), whereas the time-specific models generally improve precision at the expense of recall. This trade-off is particularly important in the context of startup investment, where false positives (incorrectly predicting success) can be more costly than false negatives.

5.3.1 Confusion Matrix Analysis

Table 4: Confusion matrix for the base model on the test set

	Predicted Negative	Predicted Positive
Actual Negative	14,619	8,918
Actual Positive	674	1,456

The confusion matrix reveals a significant number of false positives (8,918), which is expected given the model's high recall but low precision. This suggests that while the model is effective at identifying most successful startups (high recall), it also incorrectly flags many unsuccessful startups as successful (low precision).

5.3.2 Feature Importance Analysis

The most important features were founder-related attributes (`founderCount`, `has_previous_startups_total`, `has_executive_experience`), followed by VC ecosystem indicators (`vc_median_deal_value`, `vc_exits_vs_3yr_avg`) and industry sectors (various `topic_prob` features). This aligns with prior research by Gompers et al. (2010), who found that founder characteristics are among the strongest predictors of startup success.

5.4 Discussion and Implications

The temporal drift in startup success patterns presents a fundamental challenge for predictive modeling in this domain. Our time-specific models and time interaction features demonstrate promising approaches to address this challenge, though performance on recent startups remains lower than for earlier cohorts.

5.5 Comparison: RF Ensemble vs. XGBoost

Our comparative analysis revealed that ensemble methods with optimized thresholds outperformed individual models across the key metrics. While the time-specific XGBoost models showed promising results for earlier periods (Early period F1: 0.4417), they couldn't match the balanced precision-recall trade-off achieved by our ensemble approach. The implementation of 5-fold stratified cross-validation, SMOTE sampling, and class weighting proved effective in addressing the inherent class imbalance in startup success prediction.

6 Conclusion and Future Work

Future research will refine our success definition with valuation metrics and time-bound objectives (e.g., Series A within four years, $8M+ funding) while expanding our dataset temporally (1999-2022) and geographically (e.g., ecosystems in India and the Middle East). Drawing on Schmidt et al.'s Schmidt et al. (2018) findings that robust generalization requires significantly more training data, we will enhance model robustness and interpretability through increased sample size and ensemble methodologies across diverse entrepreneurial contexts.

7 Contributions

Aviato (https://www.aviato.co/): Provided proprietary startup data via their API, including comprehensive information on US-based startups founded between 2012-2022. This data contribution, valued at hundreds of thousands of dollars, enabled the development of our predictive models. Aviato's dataset provided founding team information, startup characteristics, and funding outcomes that served as the foundation for this research.

Makhmud Islamov: Designed and engineered the economic indicator features and venture capital market indicators from multiple data sources including US Treasury data, Pitchbook, and NVCA reports. Implemented the Random Forest models, including development of the ensemble approach with varying configurations and decision thresholds. Conducted model training, validation, and performance analysis for Random Forest and ensemble models.

Sukhrobjon Golibboev: Led the development of the data preprocessing pipeline, including structural transformation of nested JSON data, handling of missing values, and outlier detection. Implemented and optimized the XGBoost models, including configuration of base parameters and implementation of Bayesian optimization for hyperparameter tuning. Conducted model training, validation, and performance analysis for the XGBoost models.

8 Acknowledgement

The students would like to thank Aviato (`https://www.aviato.co/`) for giving academic access to proprietary data for this project. The students confirm that there is no conflict of interest.

9 Data Availability

The students do not have permission to share the data due to NDA with Aviato.

References

Yoke Qi Ang, Agnes Chia, and Soroush Saghafian. 2022. Using machine learning to demystify startups' funding, post-money valuation, and success. pages 271–296. Springer International Publishing.

Christoph Bergmeir, Rob J Hyndman, and Bonsoo Koo. 2018. A note on the validity of cross-validation for evaluating autoregressive time series prediction. *Computational Statistics & Data Analysis*, 120:70–83.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Leo Breiman. 2001. Random forests. *Machine Learning*, 45(1):5–32.

Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794.

Francesco Corea, Giorgio Bertinetti, and Enrico Maria Cervellati. 2021. Hacking the venture industry: An Early-stage Startups Investment framework for data-driven investors. *Machine Learning with Applications*, 5:100062.

Nataliia Dziubanovska, Vladyslav Maslii, Oleksandr Shekhanin, and Sofiia Protsyk. 2024. Machine learning for assessing StartUp investment attractiveness.

Jerome H Friedman. 2001. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5):1189–1232.

João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. 2014. A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4):1–37.

Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*, pages 1180–1189. PMLR.

S. Ghosh. 2012. Why companies fail? And how to prevent it. *Wall Street Journal*.

Paul Gompers, Anna Kovner, Josh Lerner, and David Scharfstein. 2008. Venture capital investment cycles: The impact of public markets. *Journal of Financial Economics*, 87(1):1–23.

Paul Gompers, Anna Kovner, Josh Lerner, and David Scharfstein. 2010. Performance persistence in entrepreneurship. *Journal of Financial Economics*, 96(1):18–32.

Paul A Gompers, Will Gornall, Steven N Kaplan, and Ilya A Strebulaev. 2020. How do venture capitalists make decisions? *Journal of Financial Economics*, 135(1):169–190.

Sabrina Howell, Josh Lerner, Ramana Nanda, and Richard Townsend. 2020. Financial distancing: How venture capital follows the economy down and curtails innovation. Working Paper 27150, National Bureau of Economic Research.

Steven N Kaplan and Per E Strömberg. 2004. Characteristics, contracts, and actions: Evidence from venture capitalist analyses. *The Journal of Finance*, 59(5):2177–2210.

Martin Kenney and John Zysman. 2019. Unicorns, Cheshire cats, and the new dilemmas of entrepreneurial finance. *Venture Capital*, 21(1):35–50.

Yunan Li, Iman Zadehnoori, Asif Jowhar, Sean Wise, Andre Laplume, and Mojgan Zihayat. 2024. Learning from yesterday: Predicting early-stage startup success for accelerators through content and cohort dynamics. *Journal of Business Venturing Insights*, 22:e00490.

Ramana Nanda and Matthew Rhodes-Kropf. 2013. Investment cycles and startup innovation. *Journal of Financial Economics*, 110(2):403–418.

National Venture Capital Association. 2024. NVCA 2024 yearbook. Technical report. Venture Capital Indicators.

Walter Ng and Felix Arndt. 2021. Entrepreneurial traits and firm innovation: A machine learning approach. *Entrepreneurship Theory and Practice*, 45(5):1073–1103.

Olena Piskunova, Larysa Ligonenko, Roman Klochko, Tetiana Frolova, and Tetiana Bilyk. 2021. Applying machine learning approach to start-up success prediction. *Scientific Horizons*, 11(24):72–84.

Pitchbook. 2023. Venture capital funnel shows odds of raising capital. Analyst note, PitchBook.

Matin Razaghzadeh Bidgoli, Iman Raeesi Vanani, and Mohammadreza Goodarzi. 2024. Predicting the success of startups using a machine learning approach. *Journal of Innovation and Entrepreneurship*, 13(1):80.

David R Roberts, Volker Bahn, Simone Ciuti, Mark S Boyce, Jane Elith, Gurutzeta Guillera-Arroita, Severin Hauenstein, José J Lahoz-Monfort, Boris Schröder, Wilfried Thuiller, et al. 2017. Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. *Ecography*, 40(8):913–929.

Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Mądry. 2018. Adversarially robust generalization requires more data. In *Advances in Neural Information Processing Systems*, pages 5014–5026. NeurIPS.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. 2012. Practical bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, 25:2951–2959.