WHITE PAPER

ROLE OF MACHINE LEARNING IN FINANCIAL SERVICES ORGANIZATIONS

Abstract

The Financial Services industry is probably the most important economic sector as it provides flow of capital and its performance is generally an index of a nation's prosperity. FSI is also quite likely the most data-intensive sector in the global economy. These vast volumes of data (in YTD April 2021, SWIFT recorded an average of 42.4 million FIN messages per day) makes it the perfect ground for applying machine learning techniques to harness value.

This whitepaper introduces key machine learning (ML) tasks, then elaborates how each of them are being applied by financial services organizations in the spheres of managing customers, trades, portfolios, compliance, and resilience. Business relevance of ML applications is provided for each sphere, followed by a comparison of ML to traditional approaches, a set of key aspects to be kept under consideration, and the path forward.

Infosys®
Navigate your next

Introduction

Machine learning is the domain of applying statistical algorithms to learn approximation models from a training dataset of past observations and apply the models to make predictions on new observations. Machine learning techniques are most relevant where rules are not immediately evident and vast amounts of multimodal data must be analyzed to discover them. Going by this guideline, almost every decision-led business process in financial institutions can benefit from machine learning. The challenges of high compute requirements, performance-explainability tradeoff, and bias are being surmounted through active research and through increasing adoption of such approaches in information technology.

Family	Description	Key Algorithms	Key considerations	Hyperparameters	Regularization
Similarity based	Labeling of new instances based on votes from most proximal points in latent feature space	Supervised: K-Nearest Neighbors Unsupervised: K-Means	Curse of dimensionality, features need scaling, prone to overfitting, delayed processing (lazy learners)	Number of neighbors or clusters, distance metric (Euclidean, Manhattan), centroids (unsupervised)	
Error based	Broad family that attempt to discover an optimal-fit function on prior data	Linear Regression, Logistic Regression, Support Vector Machines	Affected by multicollinearity, prone to overfitting (polynomial regression) and outliers	Regularization strength (C), alpha (lasso), kernel (linear, rbf), penalty parameter for SVM	Ridge (L2), Lasso (L1), ElasticNet
Information based	White-box supervised approach of learning decision rules from prior data	Decision Trees	Prone to overfitting, can handle imbalanced datasets, handles categoricals, less susceptible to noise	Disorder measure function (e.g. entropy), tree depth, number of trees (ensamble)	Pruning, bagging (Random Forests), boosting (CatBoost / XGBoost)
Probability based	Classifiers based on applying Bayes theorem assuming features are independent	Naïve Bayes, Bayesian Networks	Affected by multicollinearity, cannot handle unobserved scenarios, need to discretize continuous variables	None	Ridge (L2), Lasso (L1), ElasticNet
Connections based	Interconnected groups of nodes in layers that can simulate any discriminant or predictive function	Supervised: ANN, CNN, RNN-LSTM Unsupervised: VAE, GAN	Prone to overfitting, need large datasets, resource intensive training, can approximate any function, lack of transparency	Number of hidden layers, units per layer, iterations, learning rate, momentum, mini-batch size	Early stopping, dropouts, batch normalization, network pruning
Density based	Identifying clusters based on density of instance points	DBSCAN, hierarchical agglomerative clustering, Gaussian Mixture Models	Continuous features, curse of dimensionality, features need scaling,	Distance threshold, minimum number of neighbors, distance metric	
Fitness based	Searching for optimal solutions based on a fitness function	Adaptive genetic algorithms	Better than random local search for optimization problems, computationally intensive	Population size, number of generations, fitness function	
Rewards based	Learning actions that maximize cumulative rewards	Markov decision process, Q Learning, Deep Q Network	Sequential decision making, based on agent's interactions with environment	Actions, states, rewards, policy, value	

Table 1: ML algorithm families

Broadly speaking, ML models discover decision boundaries in the latent feature space by drawing correlations between predictor and predicted variables. Data scientists have worked out a wide range of approaches to solve such problems. Table 1 lists the regularly encountered approaches, with sets of related techniques bundled together into algorithm families. The No Free Lunch theorem states that no single approach will work best for all scenarios, hence multiple approaches must be explored for finding the most optimal solution to a given use case. That being the case, there are benefits of each approach, and some of these are mentioned under key considerations column. Hyperparameters are parameters that cannot be learnt from the training dataset, and hence much be explicitly set. Hyperparameters increase the size of the search space but help by providing fine-grained control over the performance of a model. Overfitting is the primary reason for failure of a model to generalize and hence regularization mechanisms are listed that can prevent overfitting.

Table 2.A provides a summary of ML tasks listed in the order of relevance to financial services organizations. A set of sub-tasks are listed in Table 2.B. These sub-tasks are frequency applied behind the scenes within the context of the overall tasks and can go a long way in improving model performance.

Task	Description	Algorithms	Evaluation Methods
Regression	Supervised approach for predicting value of a continuous variable	Linear Regression, decision trees, artificial neural networks	Mean square error, RMSE, mean absolute error
Classification	Supervised approach for labeling new instances into one of more discrete classes	Logistic Regression, KNN, SVM, Naïve Bayes, Decision Trees, ensemble models, artificial neural networks	Confusion matrix, F1 score, area under ROC curve, PR curve, log-loss
Clustering	Unsupervised approach to segregate instances in a dataset into distinct groups called clusters	K-Means, DBSCAN, hierarchical clustering, gaussian mixture models	Silhouette coefficient, Dunn's index, elbow method
Anomaly Detection	Detecting unusual or novel (called novelty detection) values in one or more features of new instances	Isolation Forests, DBSCAN, One-class SVM, Local Outlier Factor, autoencoders, hidden Markov models	AUC (supervised)
Time Series Forecasting	Forecasting value of a continuous variable based on time series pattern detection	KNN, RBF, vector autoregression, autoregressive neural networks	RMSE, mean absolute percentage error (MAPE)
Recommendation	Recommending optimal action from a predetermined set of options based on similarity metrics	KNN, Matrix factorization (collaborative filtering), association rule mining, autoencoders, Q learning	Mean average precision at k (number of items), mean average recall, coverage, personalization, intra-list similarity

Table 2.A: ML tasks

Task	Description
Dimensionality Reduction	Principal component analysis, linear discriminant analysis
Representation Learning	Variational autoencoders, generative adversarial networks
Feature Selection	Lasso, random forests, XGBoost
Model Selection	Genetic algorithms
Outlier Detection	Isolation Forests

Table 2.B: ML sub-tasks

In this whitepaper we study key use cases of ML in financial services organizations. However, the opportunities are endless as almost every financial services business process encounters vast amounts of data. The insights from this paper can be applied to related ML use cases.

Managing Customers

The cost of financial intermediation has hovered around 2% for over a century now. This relatively high cost is a prime reason for the trend towards disintermediation. To counter this move, banks are deploying robo advisors (which are algorithms that optimize a customer's investment portfolio in accordance to their risk profiles, as we will see in section Managing Portfolios below) to lower costs and improve the quality of financial advice.

Functional Task	Description	Applicable ML Tasks	Key Algorithms	Data Sources
Robo advisors	Recommending most relevant products or next-best-action based on assessment of customer interactions and prior decisions	Recommendation	KNN, collaborative filtering	Customer demographics, actions and preferences
Credit scoring	Analyzing credit worthiness of a client	Regression, classification	Neural networks, deep belief networks	Customer demographics, transaction history
KYC based recommendation	Devising an investment strategy based on customer's know-your-client parameters	Clustering	K-means clustering	Customer KYC records and transactions
Counterparty default	Predicting likelihood of counterparty default in consumer or corporate banking	Classification	ANN + NLP, CNN + feature selection, LSTM + Event2Vec	Defaultable securities

Table 3: ML applications for managing customers

Financial institutions traditionally relied on a single credit score to assess credit worthiness, typically applied during loan issuance. Now credit scoring is conducted across multiple dimensions and applied to all aspects of financial services. A radical transformation is the use of non-traditional data sources (such as socio-economic, occupational, personal data of customers) to assess credit worthiness for low-or-no credit history applicants (called thin files), increasing inclusivity and allowing banks to reach out to unbanked populations. While these models, particularly based on neural networks, are capable of achieving good accuracies in prediction, they are contingent on banks having access to these alternate datasets. Thus, banks should first embark on a data modernization strategy before exploring these opportunities. Model explainability is another concern hence models that provide feature relevance are preferred.

Advisors are required to provide financial recommendations that are suitable to customers. Suitability is achieved by understanding the customer's needs and objectives through a process called know-your-client (KYC). KYC data, that consists of demographics and financial situation, is used to generate a customer's risk aptitude level. Cluster analysis of financial transactions suggests that groups of customers with similar KYC attributes tend to follow similar transaction behaviors. Such analysis can be used to recommend suitable products to customers. Recommendations can be driven by customer information (called content-based filtering) or by the customer transaction behaviors (called collaborative filtering). Techniques such as matrix factorization and cosine similarity are applied to identify suitable recommendations.

Counterparty default can lead to credit risk. Banks need to keep aside capital to provision for unexpected losses. Approaches to calculating capital requirements are covered in Managing Compliance section below. Loan default prediction is critical for banks, both at loan origination and during servicing.

LightGCN is a model based on graph convolutional networks that has been successfully applied to collaborative filtering for improving recommendations.

Managing Trades

World's stock markets encompass enormous wealth. Trading involves making investment decisions that maximize returns for the investors while keeping the costs to a minimum. Traditionally this decision-making was largely driven by developing market insights through personal experience. However, with the explosion in amount of financial data available as well as due to the extreme rapid pace of trade executions, it is no longer possible to rely on individual judgement. Traders have turned to machine learning in almost all functions of managing trades.

Functional Task	Description	Applicable ML Tasks	Key Algorithms	Data Sources
Stock price movements	Predicting future price of an asset (stock, forex, commodity) based on sentiments expressed by people or in media coverage	Time series forecasting	RNN + stock2vec, CNN, ANN + autoencoder	Price, volume, technical indicators, transaction data
Algorithmic trading	Pre-trade analysis and trade execution for high-frequency trading systems	Classification	ANN, Q learning, GA + RNN	Daily close/high price, volumes, technical index
Trade execution	Determining optimal trading strategies that minimize costs while executing within a specified time period	Classification, regression	Bayesian networks, performance weighted random forests	Market data
Market volatility	Predicting volatility, which is the standard deviation of unexpected outcomes	Time series forecasting	GARCH	

Table 4: ML applications for managing trades

Stock price fluctuations are variously attributed to macroeconomic factors, consumer confidence, market expectations, corporate actions, geopolitical situation in the country, or sometimes to an unforeseen black swan event such as bursting of a bubble. With the advent of advanced data processing technologies, each of these factors is now getting tracked regularly, with the result that massive amounts data for financial and economic metrics is available to machine learning algorithms for detecting patterns underlying stock market trends. Deep learning-based time-series forecasting models are regularly being applied for this purpose. Forecasting or forex rates or other macroeconomic metrics can be forecasted using similar time-series approaches.

Time series forecasting is challenging in that the prediction depends entirely on endogenous (as opposed to exogenous) features as an observation at time t would depend upon similar observations in the past. It is challenging as there can only be a single sample for each combination (one observation for each stock at any point in time), unlike for other machine learning tasks. Also, time series data is non-stationary (stationarity is when mean and variance stay the same over complete set of observations). Hence, time series models perform better when frequently trained (tending towards online learning). Creating holdout validation datasets and stratification should be avoided as the chronological relationship between observations has to be preserved. Instead, probabilistic statistical measures such as Bayesian Information Criteria (BIC) should be applied on the training dataset itself to quantify model performance.

Algorithmic trading, or automated trading systems, applies machine learning algorithms to make extremely fast trading decisions, often executing millions of transactions in a day in a sub-domain called high-frequency trading (HFT). HFT is cited to improve market liquidity and reduce bid-ask spreads.

Flash crashes have been attributed to algorithms based on volatility quantitative strategies. Research indicates that humans lose confidence in algorithms faster than they do in other humans (called algorithm aversion). It is important to design customizable algorithms for traders to regain some amount of control.

FinRL is a deep reinforcement learning library for automated stock trading in quantitative finance.

Managing Portfolios

Portfolio managers are always looking to optimize asset allocation to achieve a defined financial objective, such as maximizing expected returns, minimizing risk exposure, or maximizing Sharpe ratio. There are two challenges with the traditional Markowitz framework of modeling portfolios. First, optimal asset weights are highly sensitive to estimates of future expected returns, which are often uncertain and can yield unstable weights. Second, estimating covariance matrix, which is central to this approach, requires large time-series data and susceptible to asset correlations. AI has been shown to produce better estimates of expected returns and generate more accurate weights that yield performing portfolios.

Functional Task	Description	Applicable ML Tasks	Key Algorithms	Data Sources
Expected returns	Predicting alpha and sigma and covariances for fundamental analysis of assets	Time series forecasting, clustering	Lasso, neural networks, hierarchical clustering	Financial data of assets
Portfolio optimization	Generating portfolio weights for producing optimized portfolios with better out-of-sample performance	Classification	ListNet, RNN + Q learning, genetic algorithms, neural networks	Financial data of assets
Option pricing	Forecasting option prices based on non-parametric models	Time series forecasting	Neural networks	

Table 5: ML applications for managing portfolios

AI approaches for optimization of portfolios are applied at two levels – once while allocating weights to distribution across assets classes (such as equity vs. bonds) and then while allocating weights to assets within same class. For portfolio selection, neural network-based approaches can estimate expected returns of investment in a portfolio. A tree-based representation through hierarchical clustering requires fewer estimates than traditional covariance matrix, and hence can yield better prediction of variances and covariances. Furthermore, genetic algorithms coupled with deep neural networks can solve optimization problems, including multi-objective problems, better than Monte Carlo techniques.

Derivative pricing and modeling are more challenging as prices and payoffs are contingent on underlying assets. Nonparametric option pricing approaches have been shown to yield better results than the traditional Black-Scholes model. Informer, an efficient transformer-based model, is a top performer for long sequence time series forecasting.

Managing Compliance

Credit or counterparty risk is the potential of a borrower failing to meet their obligations, and is the single largest risk faced by banks. Market risk is the risk of losses due to market volatility. Liquidity risk is the inability to meet cash-flow requirements particularly under stress events. Basel accords stipulate that financial institutions hold capital in proportion to the exposure of their assets to provide adequate provision for risks. Under the Basel framework, eligible banks may follow a standardized approach, where external credit agency ratings are used, or they can follow an internal rating-based approach (IRB), where banks have to calculate many parameters to determine the minimum capital requirements to cover potential future losses. Given that banks cannot invest this capital for more attractive returns, they are turning to machine learning to help estimate the capital requirements accurately to keep it at a minimum while meeting compliance guidelines.

Functional Task	Description	Applicable ML Tasks	Key Algorithms	Data Sources
Risk classification	Classification of loans into risk categories based on the inherent estimates risks of the loans	Classification	Neural network classification	Loans and repayments dataset
Optimize risk weighted average	Forecasting average credit loss for optimizing capital requirements in compliance with Basel regulations	Regression	Neural network regression	Non-performing loans dataset
Optimize derivative margins	Simulation of credit, funding and capital valuation adjustments (XVAs) when entering derivative transactions	Regression	Neural network regression, Monte Carlo methods	Non-performing loans dataset
Model backtesting	Validating and backtesting risk models	Time series anomaly detection	Neural networks	
Regulatory stress testing	Modelling of the link between macro-economic developments and banking variables to determine the impact of extreme scenarios on a bank	Regression	Adaptive lasso, MARS	
Macroeconomic forecasting	Forecasting macroeconomic metrics to determine risk exposure of assets in a portfolio	Time series forecasting		

Table 6: ML applications for managing compliance

Non-performing loans (NLP) are long-standing exposures without borrower paying installments. Dataset of historical NPL transactions can be used to model risks. Banks are keen to segregate transactions into well-calibrated risk classes, as poor calibration is penalized by the regulatory authorities. Loan characteristics form the features to train a classification algorithm that assigns a risk rating to each of the loans in the portfolio.

The capital requirement is made up of two components – expected losses (EL) and unexpected losses (UL). EL is factored on Probability of Default (PD), Loss Given Default (LGD) and Exposure at Default (EAD). Unexpected loss is determined based on a factor called Risk-Weighted Assets (RWA). XVAs (where X can be credit, debt, funding, margin or capital) are valuation adjustments particularly for derivatives (future, options). All these factors can be predicted accurately using ML regression techniques. These factors are determined for each rating class. Maximum Recovery Period (MRP) is another important element, as it is inversely proportional to LGD.

Risk datasets are typically imbalanced which can result in reduced classification performance. Techniques such as smote should be applied to synthetically generate datapoints and balance the datasets across prediction classes. Data quality can be suspect, with high possibility of missing borrower information due to GDPR restrictions. A rule-based decision is recommended to evaluate impute/delete feature options (as incorrect imputation can worsen data quality whereas deletion will reduce available information).

Backtesting is testing a model on historical data. In the context of financial risk, backtesting involves taking a portfolio back in time and comparing the predicted results with those achieved. Regulators require banks to backtest their models frequently as part of model risk management. However, traditional approaches to backtesting do not show anything about future performance. They are also susceptible to the seven deadly sins of data analysis. Machine learning based approaches to evaluating backtests are much better at explaining non-linear factors and generalize better to out-of-sample scenarios.

Simulation-based approaches are applied to stress test for liquidity risks. Time-series forecasting approaches, similar to those described in Managing Portfolios section above, are applied for predicting vital macroeconomic metrics, which are indicators of future market risks.

In Deep MVA, a whitepaper by IHS Markit, the authors propose a deep neural network-based approach to simulate derivative pricing models which is 10,000x faster than traditional approaches.

Managing Resilience

Operational resilience is ability of a bank to identify potential threats and implement mitigatory interventions to prevent disruption to critical operations. Corporate reports include many sub-categories including fraud, anti-money laundering and financial crime, cyber security and many others.

Functional Task	Description	Applicable ML Tasks	Key Algorithms	Data Sources
Fraud detection	Detecting fraudulent payments transactions	Anomaly detection	Bagging	
Money laundering patterns	Detecting patterns of money laundering focused clusters based on clustering or outlier detection	Clustering, anomaly detection	K-means, GMM, Isolation Forests	Transactions
Suspicious transactions	Flagging transactions suspected of money laundering or terror financing	Classification	Bayesian networks, neural networks, decision trees	Transactions
Intrusion detection	Detecting network intrusion	Anomaly detection, classification		

Table 7: ML applications for managing resilience

Frauds, such as in credit card or online payments, are crimes perpetuated by scamming or through unauthorized use of another user's account. Fraud detection must be real-time (or near real-time) to prevent a crime, which implies fast inferencing. This use case also demands high precision, as too many false positives render the model unusable (legitimate transactions that get flagged as suspicious cause needless alarm to customers and impact fraud analyst productivity). In minimize false alerts, models typically attach a score to the risk of suspicion, which a rule-based model can use to selectively flag transactions that are above a certain prescribed threshold. Supervised models are applied where labeled data is available, through the dataset will be highly imbalanced. As an alternative, unsupervised anomaly detection techniques are applied to detect unusual transactions as potential frauds. Model explainability is another critical concern, and approaches such as Shapley additive explanations (SHAP) and local interpretable model-agnostic explanations (LIME) are applied to provide importance values for contributing features. In money laundering, funds are routed towards terror financing or other illegal activities, typically under the guise of normal transaction to avoid detection. Clustering algorithms applied on networks are used to identify groups of people working together on money laundering activities. Decision tree-based models are applied to predict risk levels based on customer risk factors to create a set of rules that can be applied to detect suspicious transactions.

Intrusion detection is detecting potentially malicious network activity, a key part of cybersecurity. Like fraud detection, intrusion detection is based on detecting unusual network activity patterns by applying anomaly detection models.

Recently GAN-based anomaly detection models have shown advances in imbalanced problems.

Conclusion

This whitepaper covered the importance of machine learning for the financial sector in five areas: managing customers, trades, portfolios, compliance, and resilience. We believe that Financial Services firms will gain immensely by applying these ML techniques to the use cases outlined above.

About the author

Amitabh Manu, *Delivery Manager, Leading Innovation for Financial Services*

- Leading innovation for financial services vertical. Exploring technology frontiers for readying ourselves for the future. Developing intelligent solutions to solve challenging client problems

References

1. https://www.swift.com/about-us/discover-swift/fin-traffic-figures
2. https://paperswithcode.com/sota for state-of-the-art machine learning models
3. https://machinelearningmastery.com/no-free-lunch-theorem-for-machine-learning/
4. https://medium.com/@azzamsantosa/efficiency-welfare-and-financial-intermedation-454a51e93681
5. FinRL: A Deep Reinforcement Learning Library for Automated Stock Trading in Quantitative Finance
6. The Seven Deadly Sins of Quantitative Data Analysis
7. Deep MVA: Deep Learning for Margin Valuation Adjustment of Callable Products
8. https://www.mdpi.com/2227-9091/7/1/29/htm

For more information, contact askus@infosys.com

Infosys®
Navigate your next