GLOBAL RISK INSTITUTE

FINANCIAL INNOVATION SERIES

Machine Learning:
The Benefits & Pitfalls

Authors: John Hull

Professor, Rotman School of Management, University of Toronto

Niti Mishra

Research Associate, FinHub,

Alexey Rubtsov[1]

PhD, Senior Research Associate, Global Risk Institute

Machine learning (ML) is impacting financial services in many different ways. For example, it is automating routine lending decisions, it is proving useful in fraud detection and anti-money laundering, and it is making risk management and derivatives trading more efficient. This research report explains the tools commonly used in ML and some of the challenges it poses for financial institutions.

ML is a branch of artificial intelligence. The latter is fairly broad and concerned with examining all ways in which machines can imitate human intelligence. ML involves creating intelligence by learning from large volumes of data. For example, by providing it with data on many loans, some defaulting and some performing well, a machine can develop an algorithm for estimating the probability of default and make accept/reject decisions.

How do we assess the performance of an ML application such as the one just mentioned? It is a mistake to expect perfection. The key question is whether the algorithm performs the task as well or better than a human. It must be the case that ML passes this test for routine lending decisions. Otherwise, the use of ML in lending would have stalled.

There are many potential advantages of ML. It is very expensive to train a human to carry out a task such as processing loan applications competently. A machine once it has learned from data can do the job much faster. Its results are consistent. What is more, once an algorithm has been successfully developed, it can be transferred from one machine to another at low cost.

This is not to say that ML algorithms require no maintenance. It is important to re-train models periodically so that they reflect recent data. (ML models are no different from humans in this respect.) "Data drift" and "concept drift" are terms used to describe how the relationship between the inputs and outputs of a machine learning model change through time. An algorithm that works well when interest rates are 3% might need to be re-trained when they fall to 1%.

It is also important to ensure that the data used to train the algorithm are relevant to the decision at hand. ML algorithms work well for routine decisions, but not for special cases. For example, data collected on past loans are unlikely to be relevant for evaluating the difficulties experienced by borrowers during the COVID-19 pandemic. We must always be

[1] John Hull is the Maple Financial Professor of Derivatives and Risk Management at the Joseph L. Rotman School of Management, University of Toronto. He is also academic director of FinHub, Rotman's Financial Innovation Lab. Niti Mishra is a Research Associate at FinHub. Alexey Rubtsov is a Senior Research Associate at the Global Risk Institute.

prepared to let human judgement over-ride ML models when circumstances are different from those encountered in the past.

It is now recognized that model interpretability or model explainability is an important aspect of ML. It is important for line managers within a financial institution to have some understanding of an ML model they are using. There are a number of reasons for this. They need to have confidence in the model so that they are not always second guessing it. They need to ensure that the model is free from unacceptable biases such as those involving gender or race. They may also have to explain the model to regulators.

It is also important to understand the individual decisions or predictions made by an ML model. Consider again the use of ML for lending decisions. If a client of a bank asks why she has been refused for a loan, it is hardly acceptable for an employee to shrug shoulders and blame the algorithm.

Some algorithms correspond to the way humans process data and are easy to understand; others are black boxes. A great deal of progress has been made in recent years in developing a way of making black-box algorithms meaningful. Companies should insist that resources are devoted to interpretability before an ML implementation goes live.

ML is the new world of statistics. Whereas statistics has traditionally focused on hypotheses, confidence intervals, and significance tests, ML is concerned with reaching conclusions from large data sets. There are no hypotheses in ML. The data is entirely responsible for the conclusions that are reached. Like many fields ML has its own terminology, which can be confusing for someone with traditional statistics training. The variables input to a model are referred to as "features", their values as "labels", the outputs from the model as "targets", observations on variables as "instances." Other words that are part of data-science-speak include, "biases", "weights", "epochs", "regularization", and so on.

This report discusses unsupervised learning, supervised learning, reinforcement learning, and natural language processing.[2] The meaning of the terms "supervised learning" and "unsupervised learning" is somewhat obscure. In a supervised learning model, the data scientist specifies an objective and supervises the learning to ensure that the objective is met. The objective can be the forecast of something (e.g., the value of a house) or the classification of data (e.g., estimating which transactions are likely to be fraudulent and which are not). In unsupervised learning, no objective is specified by the data scientist. The algorithm looks for patterns in the data.

UNSUPERVISED LEARNING

Unsupervised learning can be a useful marketing tool. Consider a bank that has several million customers and 20 features describing each customer (age, average balance in account, average number of transactions per month, and so on). An unsupervised learning algorithm can group the customers into clusters. This can enable the bank to communicate with its customers in a way that reflects their likely needs. Some clusters of customers may require wealth management services; some may be in the market for a mortgage soon; and so on. It may even lead to the bank developing a new product that will appeal to a particular cluster of customers that had not previously been recognized.

"Algorithms can find clusters in data, but it is up to a human being to interpret them"

The most common unsupervised learning algorithm is the k-means algorithm. The "k" refers to the number of clusters. This must be specified in advance. Typically, a data scientist does not know the optimal number of clusters to specify. She therefore tries a number of different values of k and chooses what appears to be the best.[3] How the number of clusters was chosen is therefore a natural question to ask when the results from unsupervised learning are presented.

[2] For a more detailed discussion see Hull (2020).

[3] There are a number of statistical and other procedures that have been developed to help determine the optimal value of k.

"Features must be measured on similar scales for clustering and other ML applications"

In order to determine clusters, we need a measure of "closeness." How close is Customer A aged 39 with an average monthly balance of $2,800 to Customer B aged 36 with an average monthly balance of $2,500? An important point here is that the numerical value of the age difference (=3) is much smaller than the numerical value of the average balance difference of (=300). To cluster observations it is important that values of features are scaled so that differences are comparable. The most common approach is known as Z-score scaling and involves calculating the mean and standard deviation of the observations on each feature. The values observed for a feature are then adjusted by subtracting the feature's mean and dividing by the feature's standard deviation.

"Clustering requires a distance measure to be specified"

The k-means algorithm chooses k clusters so that the distances of observations from the cluster centers they belong to is as small as possible. How do we measure "distance"? There are a number of different possibilities, but the simplest approach is to use the Euclidean distance measure. If there are only two features, the Euclidean distance is the length of the line joining them when they are plotted in a two-dimensional chart. As we increase the number of features, it turns out that this distance measure can be extended in a straightforward way.[4]

"Feature selection is important for clustering and many other ML applications"

In addition to the values of the features that define the centers of clusters, it is important to know how tight clusters are. How distinct are customers in cluster X from those in cluster Y? This can be determined by calculating (a) the average distance of customers in cluster X from other customers in cluster X and (b) the average distance of customers in cluster X from customers in the closest neighboring cluster Y.[5] As this difference between these two measures becomes greater, we can have more confidence that the customers should be treated differently.

One of the most important factors in the success of unsupervised learning is feature selection.[6] Features that seem irrelevant in isolation may be important in combination with other features. On the other hand, some features that seem important can be disregarded if they are highly correlated with other features used in the analysis. Some experimentation may be necessary to identify the best set of features.

"It is important that the data used in ML be recent and relevant"

As with all ML applications, it is important to ensure that the data used is accurate and relevant to the task at hand. For example, it might not make much sense to use clusters of customers from 1990s to classify customers in 2020.

4 With n features the distance between two observations is the square root of the sum of the squares of the difference between feature values.

5 A measure based on (a) and (b) is known as the "silhouette score."

6 Indeed, feature selection is very important for ML in general.

KEY QUESTIONS:

- *How the number of clusters was chosen?*
- *How distinct are the elements in all (or particular) clusters?*
- *Are there any alternative ways to measure the distance between observations? How sensitive are the results to the choice of distance metric?*
- *Are there clusters that have only few observations so that they are not well represented by the data?*
- *Are there any extreme outliers and how were those dealt with?*
- *How were the features chosen?*
- *How were correlations between features handled?*
- *How often will the data need to be updated for this application?*

SUPERVISED LEARNING

In supervised learning there is a target to be predicted. The goal of a supervised learning algorithm is to learn the relationship between the features and the target from available data. One of the biggest challenges in this process is find the balance between:

1. A relationship that is too simplistic, i.e., it fails to capture key elements of the way the target depends on the features. (i.e., under-fitting the data)
2. A relationship that reflects idiosyncrasies in the data that is used for training that are not expected to be present in other data. (i.e., over-fitting the data)

Data scientists refer to this as the bias-variance trade-off. A model that is too simplistic tends to have high bias. It tends to consistently overestimate in some situations and underestimate in others. A model that is too complex has a high variance. It tends to capture random noise in the training data. It is the task of the data scientist to choose the ML model that strikes a proper balance between bias and variance.

"ML models must be tested on data that were not used while building the models"

Good ML practice requires the available data to be divided into three sets:

3. A training data set
4. A validation data set
5. A test data set

Typically, at least 60% of the available data is allocated to the training set with 10% to 20% being allocated to each of the validation set and the test set.

The training set is used to try out different models. The validation set is used to see how well the models generalize to new data. If a model over-fits the training set, the results for the validation set will be much worse than for the training set. The data scientist tries to choose a model where (a) the prediction errors are low (so that under-fitting is avoided) and (b) prediction errors for the validation set are similar to those for the training set (i.e., the model generalizes well). Once the best model has been chosen, the test set provides an estimate of its prediction error.[7]

We can distinguish between supervised learning models that are used to predict a variable that can take a continuum of values (e.g., price of a house) and models that are used for classification (e.g., distinguishing between good and fraudulent transaction). Supervised learning algorithms are very diverse

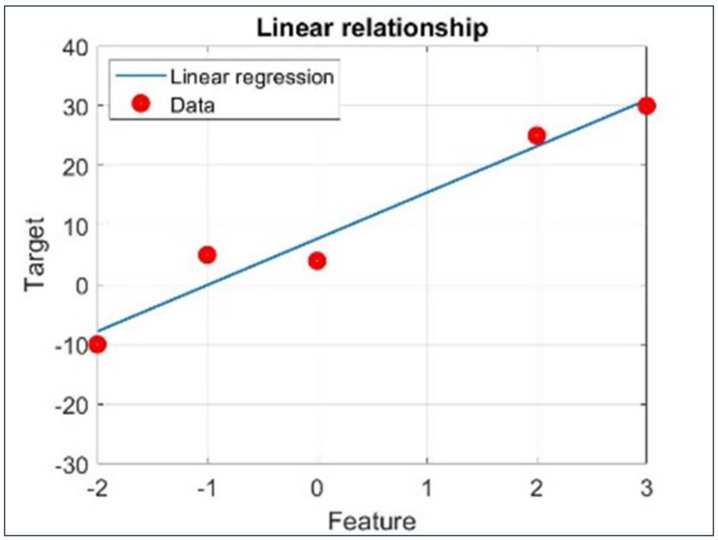
[7] Note that the test set is not involved in any way in choosing the model.

and vary from simple tools that have been known for more than a century (e.g., linear regression) to more sophisticated algorithms developed only in recent years (e.g., deep neural networks). It is these novel ML approaches that have recently demonstrated the power of ML.

"Feature selection is important. Using two or more highly correlated features can lead to a model that makes no sense"

Linear regression. As its name implies linear regression assumes a linear relationship between the target and the features (see Figure 1a). Tests for linearity should be made. If the true relationship is found to be highly non-linear (see Figure 1b), other approaches to predicting target values should be used.

(a)



b)

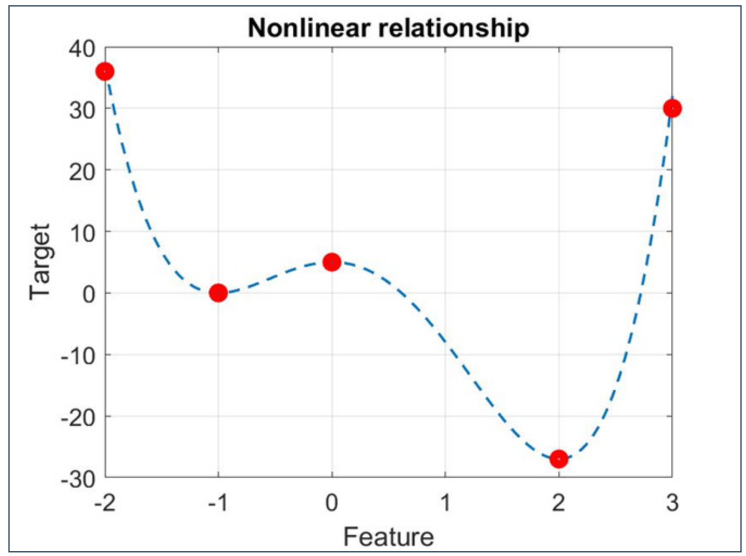


Figure 1: *Close-to-linear relationship between the feature and the target (a); nonlinear relationship between the feature and the target (b)*

When the number of features is large, some of them might be highly correlated and this, in turn, could result in overfitting. In some cases, highly correlated features result in a model that makes no sense. For example, in a linear model for house price prediction, the inclusion of number of bedrooms and living area (two strongly correlated features) could produce a best-fit model where house prices increase with living area, but decline with the number of bedrooms, or vice versa. A technique in machine learning known as regularization deals with this type of problem.[8]

[8] Suppose X and Y are (scaled) highly correlated features used to predict a target V. The best fit model prediction found by linear regression could be V = 100000X-98000Y when a more meaningful prediction that is almost as good is V=2000X. The latter would be found by regularization.

Logistic regression. For classification problems the most common tool is logistic regression. Consider the task of predicting whether a loan will default. Logistic regression provides a procedure for estimating the default/no-default probabilities for the loan. In essence, the algorithm tries to make the probability of default as high as possible for those loans in the training set that defaulted and as low as possible for those loans that did not default.

In business applications, classification typically requires a threshold to be specified. For example, we are likely to require the probability of default to be less than X% before accepting a loan. Determining the correct threshold is not easy and it is important that the data scientist presents the decision maker with alternatives. The decision maker can then weigh the cost of different types of mistakes. What is the cost of accepting a loan that defaults? What is the cost of turning down a loan that would have been OK?

The best ML algorithm to use in a particular situation is often found by trial and error

Decision trees. Whereas linear and logistic regression consider all features together, a decision tree considers them one by one. It identifies the most important feature first, then the next most important feature and so on. The importance of a feature is defined by its information content. This is a measure of the extent to which the feature can improve the estimate being made.

Figure 2 shows a tree constructed from loan data provided by Lending Club.[9] The feature that is most important is not surprisingly a potential borrower's credit score (FICO). The algorithm chooses a threshold for that (717.5). After that, it considers the borrower's income ($'000s), and then the debt-to-income percentage ratio (dti). As the tree indicates, the same feature can be considered more than once. The numbers at the end of the tree in the oval shape boxes are the probabilities that the loan will not default. Thus, the tree indicates that there is an 85.2% chance that an individual with a FICO score of 670, income of 100 and dti of 20 will repay a loan.

An extension of the decision tree idea is known as a "random forest". This involves building many decision trees similar to the one in Figure 2 and averaging the results in some way. In many situations, a random forest is found to give more accuracy than a single tree.

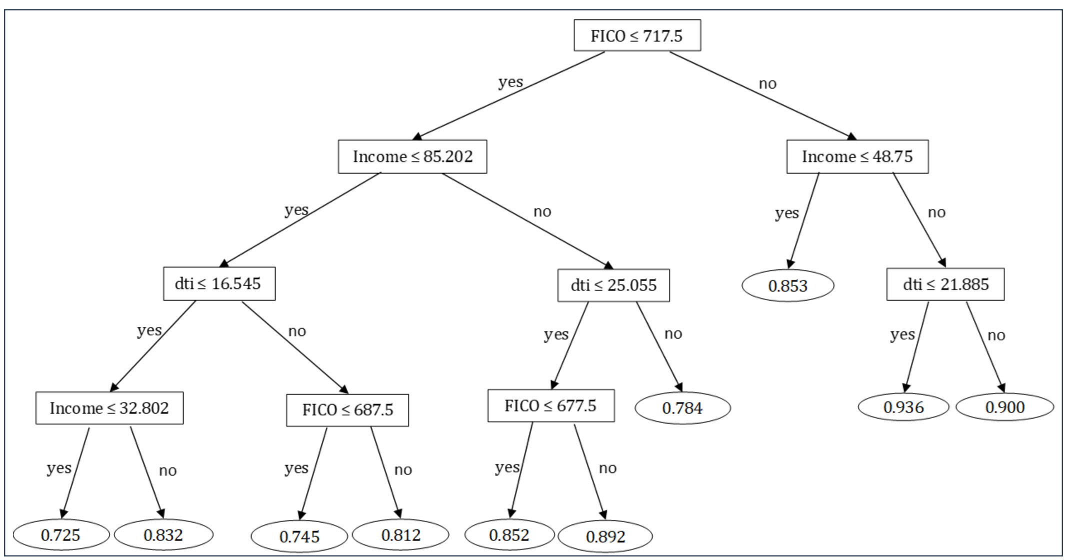


Figure 2: *Decision tree for determining the probability that a loan will be repaid*

9 This example is taken from Hull (2020).

Support Vector Machines (SVMs). The support vector machine algorithm takes a different approach to classification. Consider again the task of separating loans that will repay from those that will default. Suppose we use only two features: credit score and income. We could plot these on a two-dimensional chart and search for a line that makes fewest errors in separating good loans from bad loans (see Figure 3). This idea can be extended to the situation where there are more than two features.[10] Furthermore, there are techniques for finding a non-linear separating line.

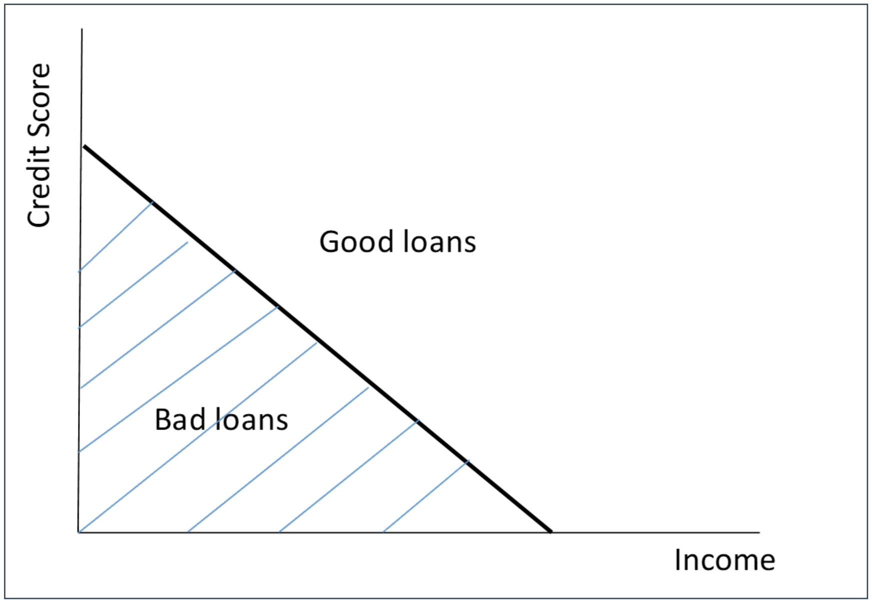


Figure 3: *Loans are divided into two categories (good and bad). Loans predicted to default are those in the shaded area (low credit score combined with low income)*

"Neural Networks are used to capture complex relationships between features and targets"

[10] With n features, a hyperplane in n-1 dimensions separates good and bad loans.

Neural Networks. When a large amount of data is available it is not necessary to assume a linear relationship between the features and the target. Artificial Neural Networks (ANNs) are powerful algorithms that will find a non-linear relationship, however complex that is embedded in the data.[11] The target can be the prediction of the value of a variable or the classification of data.

The structure of a neural network is shown in Figure 4. The target is not determined directly from the features. There are a number of intermediate "neurons" where values are calculated. The values at the first set of neurons (the v_{1j}) are calculated directly from the features. The values at the second set of neurons (the v_{2j}) are calculated from the v_{1j} and so on. Splitting the relationship between the target and the features into a number of stages in this way allows any relationship to be found. The columns of neurons are referred to as hidden layers.

"ANNs have many parameters and are liable to overfit the training data"

Earlier we mentioned the issue of overfitting and the use of a validation data set. This is very important in ANNs. An ANN has a large number of parameters and can easily produce a model that reflects idiosyncrasies in the training set. It is important to ensure that a model is not over-fitting by testing whether it generalizes to the validation set.[12]

Once an ANN has been constructed, it provides a very fast way of determining targets from features. As a result, ANNs are sometimes used to speed up complex computations. Consider, for example, an exotic derivative that has traditionally been valued using Monte Carlo simulation or another computationally slow numerical procedure. An ANN can be used to replicate the pricing of the derivative given by the numerical procedure. This can improve computation time by a factor or one thousand, or even one million. As a result, the efficiency with which scenario analysis and other risk management tools can be employed is improved.

[11] The relationship must be a continuous function (i.e., no jumps).

[12] Regularization, which we mentioned in connection with linear regression, is one of the tools that can be used to simplify the model so that it does not over-fit.

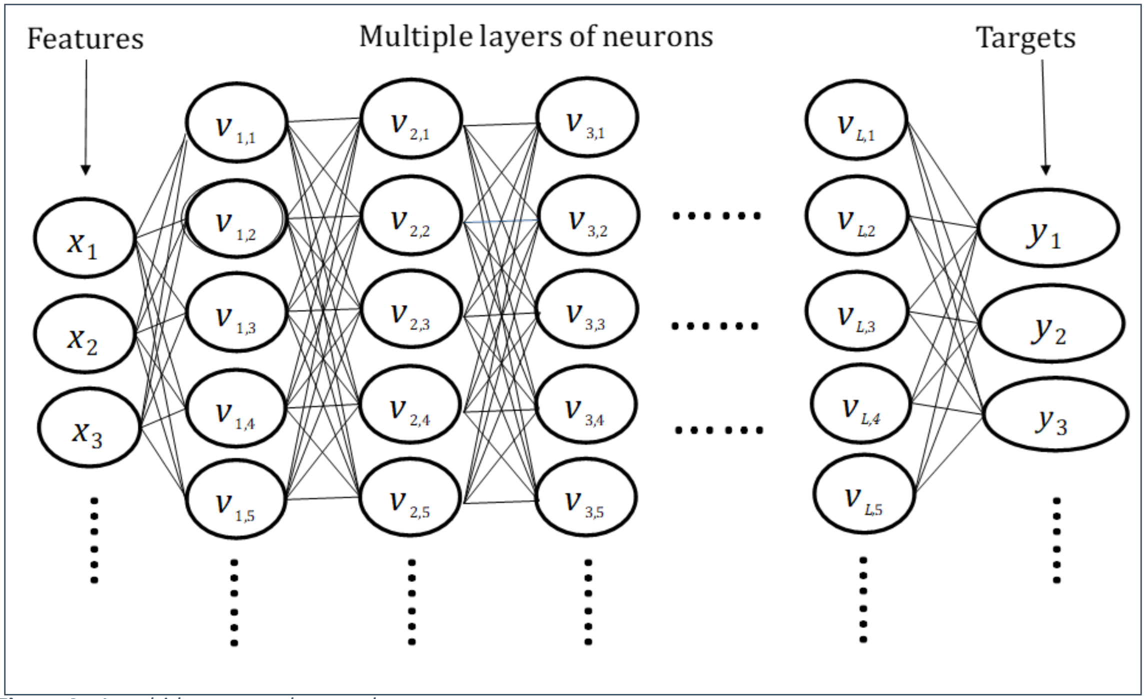


Figure 4: *A multi-layer neural network*

ANNs are black boxes. Interpretability is a challenge

Lack of interpretability of ANNs makes it difficult to identify undesirable behavior of ANNs and impedes model validation. For example, interpretability of the linear regression could allow us to identify the problem in the fitted model for house price prediction: house prices increase with living area, but decline with the number of bedrooms, or vice versa (see the discussion of Linear Regression). In this respect, it would be also useful to estimate the impact of each feature on the prediction made by an ANN. Although there are some approaches that attempt to do this, it remains an important direction of future research.[13]

KEY QUESTIONS:

- *How was the available data split between training set, validation set, and test set?*
- *What alternative ML models were tested in regards with bias-variance trade-off?*
- *How was the best set of features selected?*
- *For what range of feature values, the model can be applied?*
- *What is the advantage of the chosen model over others that might be used?*
- *Is interpretability an issue for the model?*

[13] See, for example, "A Benchmark for Interpretability Methods in Deep Neural Networks". Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, Been Kim. 33rd Conference on Neural Information Processing Systems (NeurIPS 2019).

REINFORCEMENT LEARNING

Reinforcement Learning (RL) is concerned with discovering actions that maximize a certain objective function by trial-and-error interactions with the environment. Consider hedging a portfolio of derivatives as an example[14]. The objective is to maximize expected risk-adjusted returns from portfolio of hedge instruments and derivatives. To achieve the objective, an RL algorithm applies different trading strategies to available data and strategies that yield higher cash flows receive higher rewards. After many trials the algorithm determines the best strategy based on average risk-adjusted reward. In this respect, the following terminology is used in RL:

- *States. These represent "the market" described by current and past prices of hedge instruments, costs, past trading decisions, and anything else that is relevant for determining risk management strategies.*
- *Policies that consist of actions. At each point in time and in each state, we make our trading decision (action) which is a part of our more general hedging strategy (policy).*
- *Rewards. This could be cash-flows from implementing a certain policy (net of costs).*
- *Objective. In this example we would want to maximize the sum of expected future risk-adjusted rewards.*

"Reinforcement Learning algorithms are data-hungry"

RL algorithms require large amount of data and this presents a challenge in most financial applications. For instance, 20 years of daily observations amounts to about 5,000 data points which is considered a small sample for RL. One of the ways to overcome the data scarcity problem is to generate synthetic data. In this respect it becomes critical to use data generators that preserve most important features of the real data (high-order moments, tail correlations, etc.). Recently, ANNs have been applied to generate market scenarios with fairly complex dependence structures.[15]

"There could be many alternative ways to specify rewards and the objective"

Defining rewards and the objective for an RL algorithm requires domain-specific expertise. For example, risk of a portfolio can be measured in a variety of ways such as Value-at-Risk, Conditional Value-at-Risk, standard deviation of returns, etc. The objective that defines a proper risk-return trade-off should generally depend on internal and regulatory constraints. In this regard we note that the results of RL could be sensitive to different specifications of the rewards and the objective.

In many applications, RL algorithms are used in conjunction with other ML tools such as ANNs. This, in turn, implies that risks inherent to ANNs should also be addressed for such applications.

KEY QUESTIONS:

- *How were the training data generated?*
- *Are there any simplifying assumptions made in the model (e.g., no transaction costs, no liquidity constraints, no market impact)?*
- *How were the rewards and the objective determined?*
- *Were any other ML models used in conjunction with RL?*

[14] See "Deep Hedging: Hedging Derivatives Under Generic Market Frictions Using Reinforcement Learning", H. Bühler, L. Gonon, J. Teichmann, B. Wood, B. Mohan, J. Kochems. Swiss Finance Institute Research Paper No. 19-80 (2020)

[15] See "The Market Generator", A. Kondratyev, C. Schwarz. SSRN (2020)

NATURAL LANGUAGE PROCESSING

Natural language processing (NLP) is an important application of ML because much of the data generated in the world is in the form of written or spoken words. There are many challenges in NLP. Words can have several meanings depending on the context. Therefore, it is much more difficult for a machine to interpret words than numbers. A language's nuances such as metaphors, sarcasm and irony are tricky for a machine to recognize. The problem of applying ML to language is compounded by the fact that different languages have different grammatical rules, different structures, and different characters.

In spite of this, a great deal of progress has been made and we can expect to see exciting developments in NLP in the future. Simple applications of NLP involve forming a vocabulary of relevant words and characterizing a document by the number of times each word appears. Once that has been done, the clustering and predictive tools we have mentioned can be used. More sophisticated approaches known as "word embeddings" enable a machine to identify words that have similar meanings by looking at other words they tend to be used with.

In what follows we consider a number of applications that are potentially relevant to the finance sector.

CHATBOTS

Speech recognition is an important application of NLP and one where a great deal of progress has been made in the last few years.[16] Tools such as Apple's Siri, Amazon's Alexa, and Microsoft's Cortana can recognize human speech to provide information and perform a variety of simple tasks.

Major banks have introduced some level of customer service automation through chatbots. For example, Bank of America's Erica is a mobile virtual assistant that accepts voice and text commands and combines predictive analytics with NLP to help customers with their basic queries such as check account balances and money transfers.[17] TD Bank Group announced its plans to integrate Kasisto NLP technology, "Clari", into its mobile app to provide customers with real-time support and spending insights.[18] Abe AI is a virtual financial assistant that works with a number of platforms to provide more convenient banking.[19]

"ML-driven chatbots inherit risks of the underlying ML models"

Application of ML techniques to chatbots could pose significant risks. For instance, in 2016 Microsoft's chatbot Tay learned undesirable behavior and offended millions of Twitter users by posting racist, misogynistic comments. Chatbots that might give wrong answers, especially to delicate questions, could damage the reputation of a financial institution. In this respect, taking advantage of ML tools makes chatbots prone to ML-specific risks such as biases, lack of interpretability of some ML tools, etc.

LANGUAGE TRANSLATION

Training a machine to translate from one language to another is challenging. The early attempts involved programming the machine to follow rules inputted by humans. In November 2016, Google announced "Google Neural Machine Translation" which is a highly successful ML application where the machine learns its own rules from a large volume of translated text.

"Some applications require development of highly specific ML-based translators"

Applying translators that work well in general could yield wrong results for some applications. For instance, PanAgora Asset Management had to develop a specific NLP methodology to analyse Chinese equities based on discussions

[16] For example, programs for converting speech to text work quite well.

[17] https://promo.bankofamerica.com/erica/

[18] https://kasisto.com/kai-consumer-banking/

[19] http://www.abe.ai/products/virtual-financial-assistant/

at online investor forums.[20] Apart from language-specific issues, the main challenge was to teach the machine to understand cyber speak that Chinese traders use to avoid government censors. For example, the word "rubbish" was replaced by a phonetically similar expression "spicy chicken".

SENTIMENT ANALYSIS

An important and very common application of NLP is sentiment analysis. It aims to analyze the sentiment or the tone of text that is gathered from surveys and social media and classify it as positive, negative, neutral, litigious, etc. This can be done in real time and provide important inputs to a company's decision making. How well has a new product been received by the market? What is the response to a new advertisement? Is damage control necessary as a result of recent comments by one of the company's executives or recent well-publicized actions taken by the company?

"Lexical approach: Meaning of words could change depending on the context"

There are two main approaches to sentiment analysis — lexical and algorithmic. Lexical approaches involve a panel of domain experts labelling a dictionary of words with their related semantic polarity and strength. However, labeling of words could be challenging. For example, in one of the most notable sentiment dictionaries used in finance, Loughran-McDonald dictionary, the words "antitrust" and "concede" have negative meaning, whereas they could have neutral or even positive meaning in non-financial contexts.

"Algorithmic approach: texts used to train NLP algorithms could be biased"

Algorithmic methods use machine learning-based sentiment classifiers that are trained on huge text data. To avoid biased results, NLP algorithms should be checked for biases. For example, Baker et al. (2015) find similar movements in their Economic Policy Uncertainty index based on right-leaning and left-leaning newspapers, suggesting that political slant does not seriously distort the index.[21]

In the financial industry, news and commentaries have a direct impact on stock prices. NLP models scan through these data sources and apply sentiment analysis to extract key insights, determine public perception or track market reactions to significant events. AlphaSense is an AI-based company that has created a sentiment analysis tool that analyzes earnings calls, transcripts, brokerage research and news publications to discover changes and trends in the financial market.[22] The system then provides a summary of the most relevant information for search queries that financial analysts then use to make predictions on the stock market. This drastically reduces the turnaround time for analysts as timely insights inform their recommendations. Sentifi, a Switzerland based company claims to offer similar software that can mine insights from news, social media, financial influencers and blogs using NLP techniques and provides insights regarding relevant topics of discussion in the market or events that might impact their investments.[23]

[20] See "Teaching Machines to Understand Chinese Cyber-Slang", Chen, M.,J. Lee,and G. Mussalli. PanAgora Asset Management. (February 2019)

[21] Measuring Economic Policy Uncertainty. Baker, S., Bloom, N., Davis, S.: Quarterly Journal of Economics 131(4), 1593-1636

[22] https://www.alpha-sense.com/financials

[23] https://sentifi.com/page/products/

ORGANIZATION OF INFORMATION

The techniques used by search engines, such as Google, can be employed by companies to organize and retrieve documents. For example, JP Morgan Chase automated its process of reviewing legal documents by building a COiN (Contract Intelligence). COiN runs on the bank's private cloud network and is able to extract many relevant attributes from annual commercial credit agreements in a matter of seconds. Similarly, the consulting company, Sigmoidal, claims its NLP application can automate the task of mining for information on market developments from news sites and social media.[24]

Understanding the relationship between documents is another application of NLP. For example, 10k forms that have mandatory risk factors sections are scanned to examine the distribution of topics overlapping between companies to gain insights on companies that share common underlying risks. Kensho, an AI company, has created Kensho New Economy indices based on regulatory filings and other public information.[25] The indices help identify the leading companies in each industry and the entire ecosystems supporting those companies. It was acquired by S&P Global in March 2018 for roughly $550 million. Other financial institutions use vendors such as Dataminr that provide real-time AI-platform that detects earliest signs of high impact and emerging risks from publicly available data.[26]

Other applications of NLP can be used to summarize reports and we may not be too far away from applications that can learn to write the reports!

KEY QUESTIONS:

- *How were ML-related risks addressed (e.g., biases, data scarcity, overfitting)?*
- *What approaches were taken to address domain-specific issues?*
- *Were simpler solutions considered (e.g., keyword search)?*
- *Are there any security threats (when NLP solutions are provided by a third party)?*

REFERENCES

Hull, John C. "Machine Learning in Business: An Introduction to the World of Data Science" 2nd Edition, 2020. Available from Amazon.
See www-2.rotman.utoronto.ca/~hull for details.

24 https://sigmoidal.io/product/

25 https://indices.kensho.com/

26 https://www.dataminr.com/corporate-risk