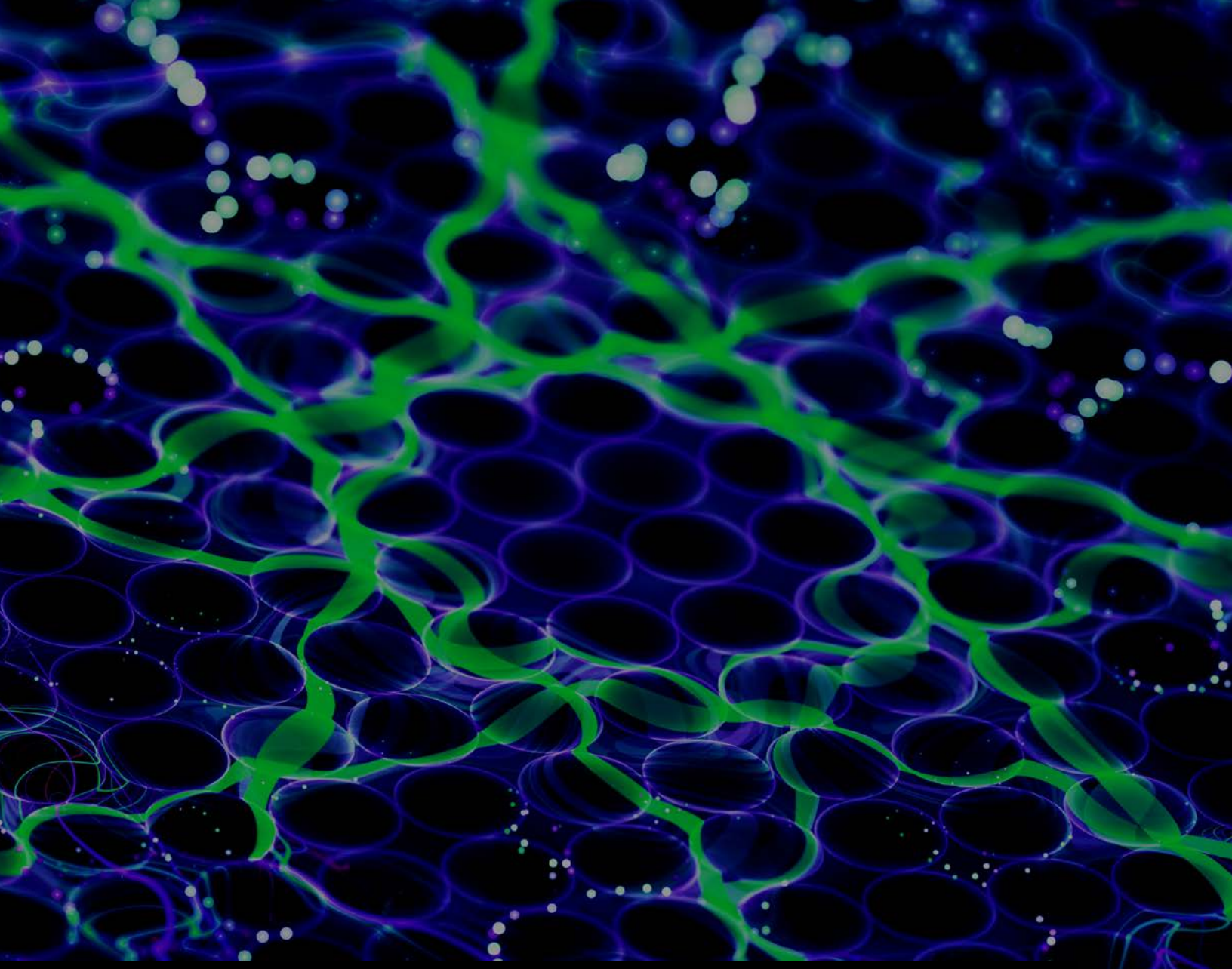


Generations of Machine Learning in Cybersecurity

Artificial intelligence is more than a feature; it must be in your DNA.

CYLANCE

Summary

In this white paper, we aim to define generations of machine learning and to explain the maturity levels of artificial intelligence (AI) and machine learning (ML) that are being applied to cybersecurity today. In addition, the paper seeks to explain that while a great deal of progress has been made in the evolution of machine learning's application to cybersecurity challenges, there remains an immense amount of opportunity for innovation and advancement in the field, and we expect the sophistication of applications of machine learning to continue to evolve over time.

This white paper is organized into sections that provide the following information:

- An introduction which briefly summarizes the context of machine learning's application within cybersecurity, and the case for an official categorization of cybersecurity machine learning models into generations
- A review of key machine learning concepts and considerations when drawing distinctions between generations
- Definitions for five distinct cybersecurity machine learning generations
- The greater implications of this machine learning generational framework
- A brief conclusion

Introduction

The Defense Advanced Research Projects Agency (DARPA) has defined AI in three foundational ways, referring to these as the Three Waves of AI.

The first wave is **handcrafted knowledge**, which defines rules that humans use to carry out certain functions, and from which computers can learn to automatically apply these rules to create logical reasoning. However, within this first wave there is no learning applied to higher levels. One example of cybersecurity inside this first wave is the DARPA Cyber Grand Challenge.

The second wave is **statistical learning**. Often used in self-driving cars, smartphones, or facial recognition, this wave of AI uses machine learning to perform probabilistic decision making on what it should or should not do. In this second wave, the systems are good at learning, but their weakness lies in their ability to perform logical reasoning. In other words, the systems classify and predict data, but don't understand the context.

This is where the third wave, known as **contextual adaptation**, comes into play. In this wave, the systems themselves construct explanatory models for the real world itself. In the third wave, the systems should be able to describe exactly why the characterization occurred just as a human would.

Machine learning has been quickly adopted in cybersecurity for its potential to automate the detection and prevention of attacks, particularly for next-generation antivirus (NGAV) products. ML models in NGAV have fundamental advantages compared to traditional AV, including the higher likelihood of identifying novel, zero-day attacks and targeted malware, an increased difficulty of evasion, and continued efficacy during prolonged offline periods.

Most attempts to apply ML and AI to cybersecurity fall into DARPA's first wave, handcrafted knowledge, using human defined rules, and defined patterns. A scant few cybersecurity technologies can claim involvement, much less maturity, in DARPA's second wave, statistical learning.

The first wave ML models inevitably suffer from high false positive rates and can be easily bypassed. Since there are now several iterations of ML applications for AV, it is no longer sufficient to differentiate only between the current version or release of an AV, and the forthcoming one. Instead, the time has come to provide a high-level description of the evolving generations of ML both as it has been, and will be, applied to cybersecurity in the future.

In this paper, we explore the sub-categories of machine learning generations inside DARPA's second wave, statistical learning. We aim to explain the maturity levels of AI represented in applications within cybersecurity today, and how we expect them to evolve over time.

Concepts and Considerations

This section explains the terms and concepts employed in this document that assist in drawing distinctions between generations of ML models, and also provides commentary on why these concepts are relevant to security.

Runtime

Machine learning algorithms universally involve two fundamental steps:

- **Training**, when a model learns from a data set of known samples
- **Prediction**, when a trained model makes an educated guess about a new, unknown sample

The training step is the much more intense computational operation — modern deep neural networks can take months to train even on large clusters of high-performance cloud servers. Once a model has been trained, prediction is comparatively straightforward, although prediction often still requires significant memory and CPU usage. To train a classifier, samples from the input dataset must have associated labels (e.g. malicious or non-malicious).

Runtime is the environment where training or prediction could occur: local, e.g. on endpoint, or remote, e.g. in cloud. The runtime for each ML step informs how quickly a model can be updated with new samples, the impacts of decision

making, and dependence on resources such as CPU, memory, and IO. For supervised models, note that labels must be available during training, so training can only occur where labels are available. In practice, training is typically done in a cluster of distributed servers in the cloud. Prediction is more common in the cloud as well, but increasingly performed locally. Distributed training on local user or customer devices is an emerging technology. Although there are major possible benefits, including reduced IO and protection of sensitive data, there are many challenges such as heterogeneous resources, unreliable availability, and slower experimental iterations.

Features

The set of features, or feature space, specifies precisely what properties of each example are taken into consideration by a model. For portable executable (PE) files, the feature set could include basic statistics such as file size and entropy, as well as features based on parsed sections of the PE, for example, the names of each entry in the section table. We could include the base-2 logarithm of file size as another derived feature. Some features could be extracted conditionally based on other features; other features could represent combinations. The space of possible features is very large, considering that there are innumerable transformations that can be applied to the features.

The features are critical to any ML model because they determine what and how information is exposed. Besides the important question of what information to include, it also matters how to encode the information. The process of creating model-amenable features is called feature engineering. Some models are more sensitive than others to how features are encoded. Although it is often tempting to provide as many features as possible, there are disadvantages in using too many features: greater risking of overfitting, higher resource consumption, and possibly more vulnerability to adversarial attacks. The efficacy, interpretability, and robustness of the model all hinge on the features.

Data sets

The data used to train and evaluate the model fundamentally and hugely impacts its performance. If the data used to train the model are not representative of the real world, then the model will fail to do well in the field. Labels for each sample, such as benign or malicious, are necessary for training classifiers. The labels need to be vetted carefully, since any mislabeling, also known as label noise, can bias the model. As more data is gathered, the labeled datasets must be continuously monitored to ensure consistency and ongoing hygiene. In practice, the data may come from a wide variety of sources. Each source must be evaluated for the degree of trust and reliability so that downstream uses can take these factors into account.

A common problem which is present for many security applications is how to handle unbalanced data, w hich occurs when one label (benign) is much more common than others (malicious). Unbalanced labeled data can be mitigated by

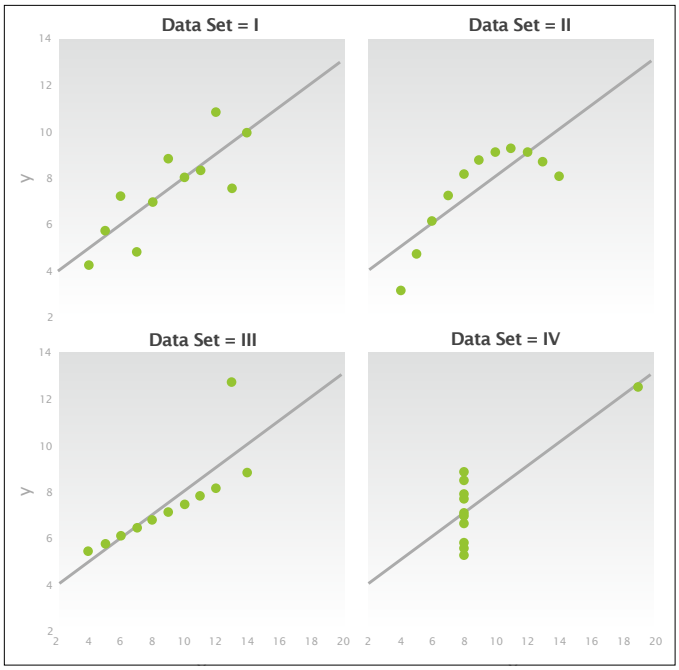


Figure 1 — Four very different datasets, shown by points, which result in the same fitted predictive model, represented by each line. This famous set of datasets is known as Anscombe's quartet.

various modelling strategies, but ideally, there are many representative samples for each label. The feature set and dataset are closely related, since many features will be generated using the training set. The dataset also impacts crucial feature pre-processing, such as normalization, or weighting schemes, such as term frequency-inverse document frequency (TF-IDF).

For a sophisticated model, it's necessary to have a very large dataset. However, it is easy to fall into the trap of assuming that a sufficiently large dataset will lead to better performance. While, in general, larger datasets enable training of more sophisticated models, a huge dataset does not guarantee performance. A good dataset should have a wide variety and should fairly represent the samples that a model might see when deployed. The desired variety can be represented quantitatively as rough balance in feature values among labeled examples.

Human Interaction

Models are often thought of as black boxes, but they need not be. Models which can support modes of interaction with people have several advantages. They can receive expert feedback more readily, which can be useful for improving both labels and features, and allowing the model to improve in otherwise difficult ways. Human confidence and trust in the model can be made more quickly when there is some way of understanding how the model decisions are made.

Having methods for exploring the model can also help to validate the underlying data. Figure 1 shows Anscombe's quartet, in which four very different input datasets yield precisely the same linear regression model. Based on the summary statistics and model parameters, the four

cases are practically indistinguishable. When plotted, it is immediately clear that only the upper-left quadrant model is fit appropriately to its dataset. The other models, with more dimensions and parameters, are much more difficult to explore and understand. However, without some type of human validation, it is likely that qualitative model or data issues could go quietly unnoticed, and lead to poor efficacy or vulnerabilities.

Supporting modes of human interaction is also important in cases where the model fails. If the model is a black box, it can be difficult to identify the cause of systematic modelling errors. Tools for inspecting and understanding the model enable troubleshooting and diagnostics. Such tools need to be carefully controlled and may not be integrated into the end product, since they leak intellectual property and could potentially expose vulnerabilities to adversaries.

Goodness of Fit

Some models better represent the real world better than others. When a model is oversimplified, it has poor efficacy but generalizes well to new data. These models are called "underfit", in the sense that there is more information available to the model which it is not fully taking into account. Conversely, a model can memorize, or "overfit". When overfitting, the model learns too much about the specific samples on which it was trained, but does not transfer its representation well to new samples in the real world.

In Figure 2, the dashed line represents the decision boundary of an overfit classifier for green vs. gray points. The green line represents an appropriate decision boundary. Although it does not classify perfectly on the shown points, its performance will be better for new points.

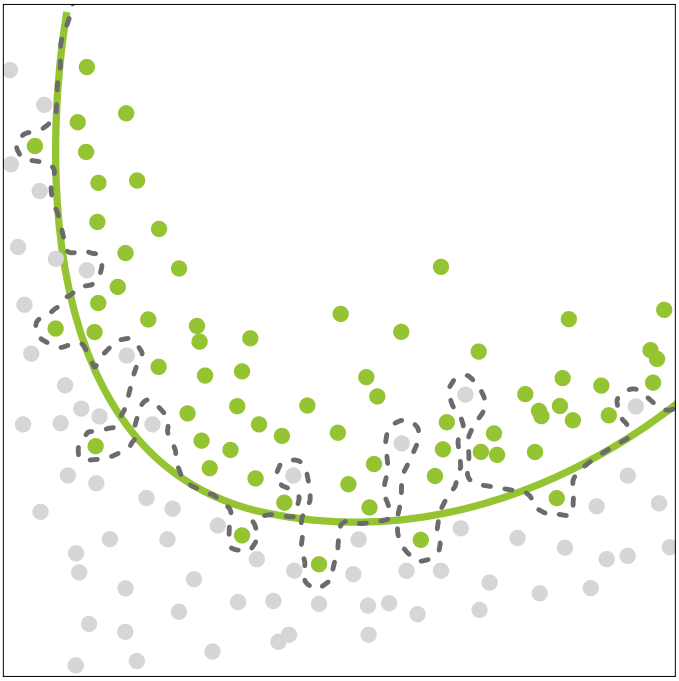


Figure 2 — Data points from two classes, each class indicated by its color. The two lines show alternative decision boundaries from hypothetical classifiers.

A well-fit model will maintain its validation performance after deployment. Concept drift is a related concept which occurs when there are nonstationary changes in the data over time, e.g. the set of PE files on endpoints changes from year to year. As the population of sample PE files change, the model should be prepared to adapt to the changes in the population it targets.

How Generations Are Defined

Cybersecurity machine learning generations are distinguished from one another according to five primary factors, which reflect the intersection of data science and cybersecurity platforms.

- **Runtime:** Where does the ML training and prediction occur (e.g. in the cloud, or locally on the endpoint)?
- **Features:** How many features are generated? How are they pre-processed and evaluated?
- **Datasets:** How is trust handled in the process of data curation? How are labels generated, sourced, and validated?
- **Human Interaction:** How do people understand the model decisions and provide feedback? How are models overseen and monitored?
- **Goodness of Fit:** How well does the model reflect the datasets? How often does it need to be updated?

These factors enable us to separate cybersecurity technologies into five distinct generations of ML, each defined by its progression in each category. Typically, a productized model takes two to three years to advance from one generation to the next, and the majority of technologies that integrate machine learning will become trapped in the first or second generations. Only a few have entered the third generation, and that evolution was hard won after many lessons learned in the field. Graduating to the fourth and fifth generations will require substantially more research and development. The requirements of the domain applications in cybersecurity are quickly catching up to the state of the art in ML research, particularly in the areas of adversarial learning, active learning, federated learning, and model interpretability.

The following table summarizes the characteristics of the generations according to the achievement within the factors previously described.

Generation	Runtime *Where do training and prediction occur?*	Features *Characteristics, elements*	Data sets *Sizes and label provenance*	Interactivity *Human interpretability*	Goodness of Fit *How well the model suits the real world*
First	▪ Cloud training ▪ Cloud prediction	▪ Over 1,000 features	▪ Over 1M data examples ▪ Human labeled	▪ Human understands decisions	▪ Underfit, high false positive rate
Second	▪ First generation ▪ Local prediction	▪ Over 100,000 features	▪ Over 100M data examples ▪ Human labeled, some heuristic labels	▪ Model struggles to explain decisions	▪ Overfit, misleading false positive rate
Third	▪ Second generation ▪ Cloud enhanced models	▪ 1 to 3M features	▪ Over 1B data examples ▪ Largely heuristic labeled	▪ Model provides understandable explanations	▪ Fit appropriately, accuracy metrics generalize
Fourth	▪ Third generation ▪ Local training	▪ Over 3M features	Online learning	▪ Model explains strategy, receives high-level feedback	▪ Model fits current inputs as well as future inputs
Fifth	▪ Fourth generation ▪ Unsupervised local training	▪ Unlimited with semi-supervised discovery	Active learning	▪ Human input optional; interpretable insights	▪ Model identifies and adapts to concept drift

The Greater Implications of Each Generation of Machine Learning

The table above lays out the distinguishing features of each generation. Each generation builds on the last one. The dataset size and number of features grows substantially in each generation. Below, we focus on the qualitative differences from each generation to the next,

- **First-Generation Machine Learning:** Application of off-the-shelf ML toolkits such as scikit-learn, using standard models. All good/bad labels are provided by human analysis, meaning the feature set is small, fixed, and picked by a human. These models cannot be deployed to endpoints. They typically result in high levels of false positives and will suffer from very limited efficacy. They are also easy to bypass.
- **Second-Generation Machine Learning:** Most labels are still applied manually, but at this stage, heuristics are used to supplement human labels. This application allows for local model predictions, but still requires cloud-based training. The local model is a clone of the cloud model. Interpretation is provided by human descriptor methods, which are post-hoc and not truly connected to the model's decision procedure. Models in this stage are typically overfit to training data. Although models in this class have some predictive power, they still need periodic updating to avoid suffering from concept drift.
- **Third-Generation Machine Learning:** The cloud model is more advanced, and complements and protects the local model. Decisions are explained by the model in a way that reflects its decision process. Models are evaluated and designed to be hardened against attacks. Concept drift is mitigated by great generalizability.
- **Fourth-Generation Machine Learning:** Models learn from local data, without needing to upload observations. Features are designed by strategic interactions between humans and models. New features and models are constantly evaluated by ongoing experiments. Humans can provide feedback to correct and guide the model. Most are robust to well-known ML attacks.
- **Fifth-Generation Machine Learning:** Supervision becomes optional. Models learn in a distributed, semi-supervised environment. Human analysis is guided by model-provided insights. Models can be monitored and audited for tampering, and support deception capabilities for detecting ML attacks.

Conclusion

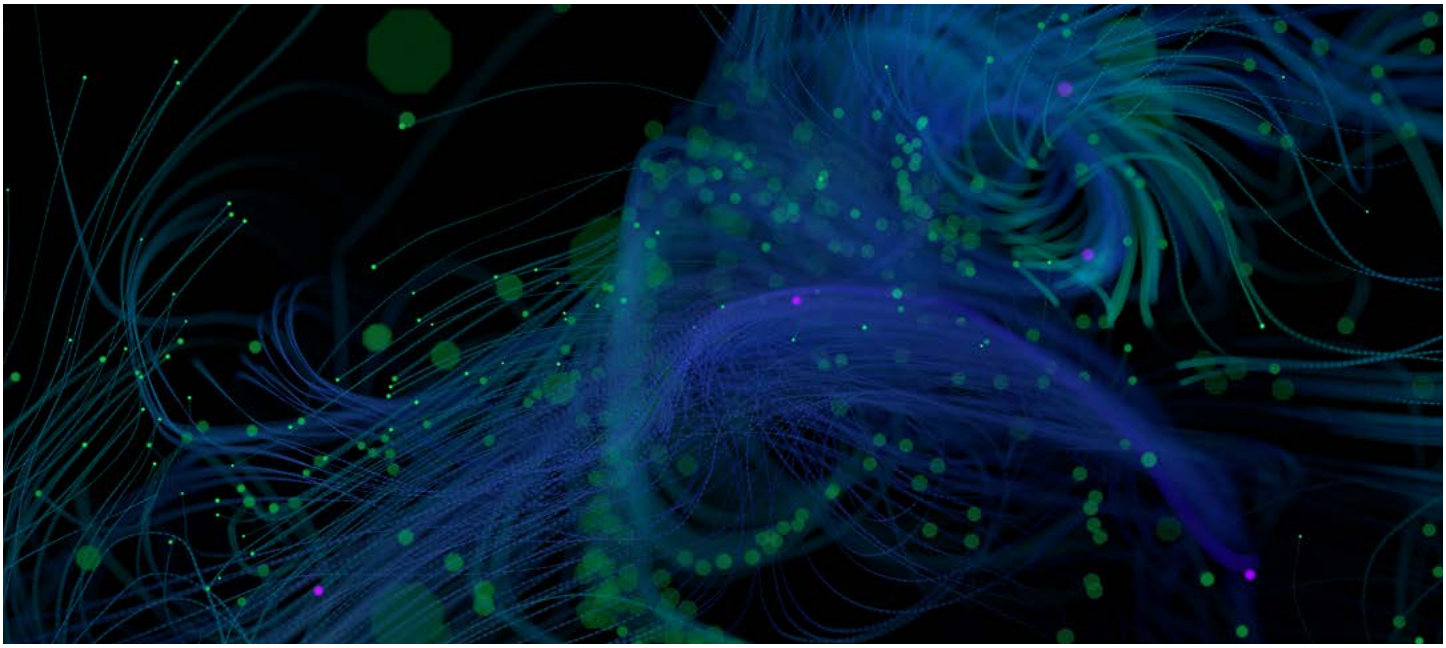
The application of machine learning to cybersecurity is still a very recent development in the industry (since circa 2013). As such, we are just beginning our journey to apply ML in an attempt to solve very challenging problems in the journey towards cyberprotection. As each ML advancement is made, future generations will only get better and better at providing all five areas of maturity (Runtime, Features, Datasets, Human Interaction, Goodness of Fit).

Hype around ML in cybersecurity has been driven in large part by two areas of application, which are respectively outside and inside of cybersecurity:

- State-of-the-art ML engineering by Google, Amazon, Facebook, and others, primarily targeting mass market applications, e.g. image and video, natural language, recommender systems, and self-driving cars
- Very simple, off-the-shelf ML applied to classical problems in cybersecurity

While ML has demonstrated some degree of applicability in a wide variety of domains, the adaptations to cybersecurity are still relatively young. The importance of cybersecurity merits novel research aimed at open problems in cybersecurity, and not just training a simple model on a cyber dataset.

Each generation represents a qualitative improvement over previous generations. Maturity has a direct impact on the value provided by ML because the changes are not just marginal improvements in efficacy, but rather represent leaps in the fundamental abilities of ML to detect and prevent attacks. The ML approach has quickly proven to have value, but the full defensive potential will only be developed by the more advanced generations of ML.



+1-844-CYLANCE
sales@cylance.com
www.cylance.com
18201 Von Karman Avenue, Suite 700, Irvine, CA 92612

20170807-1525