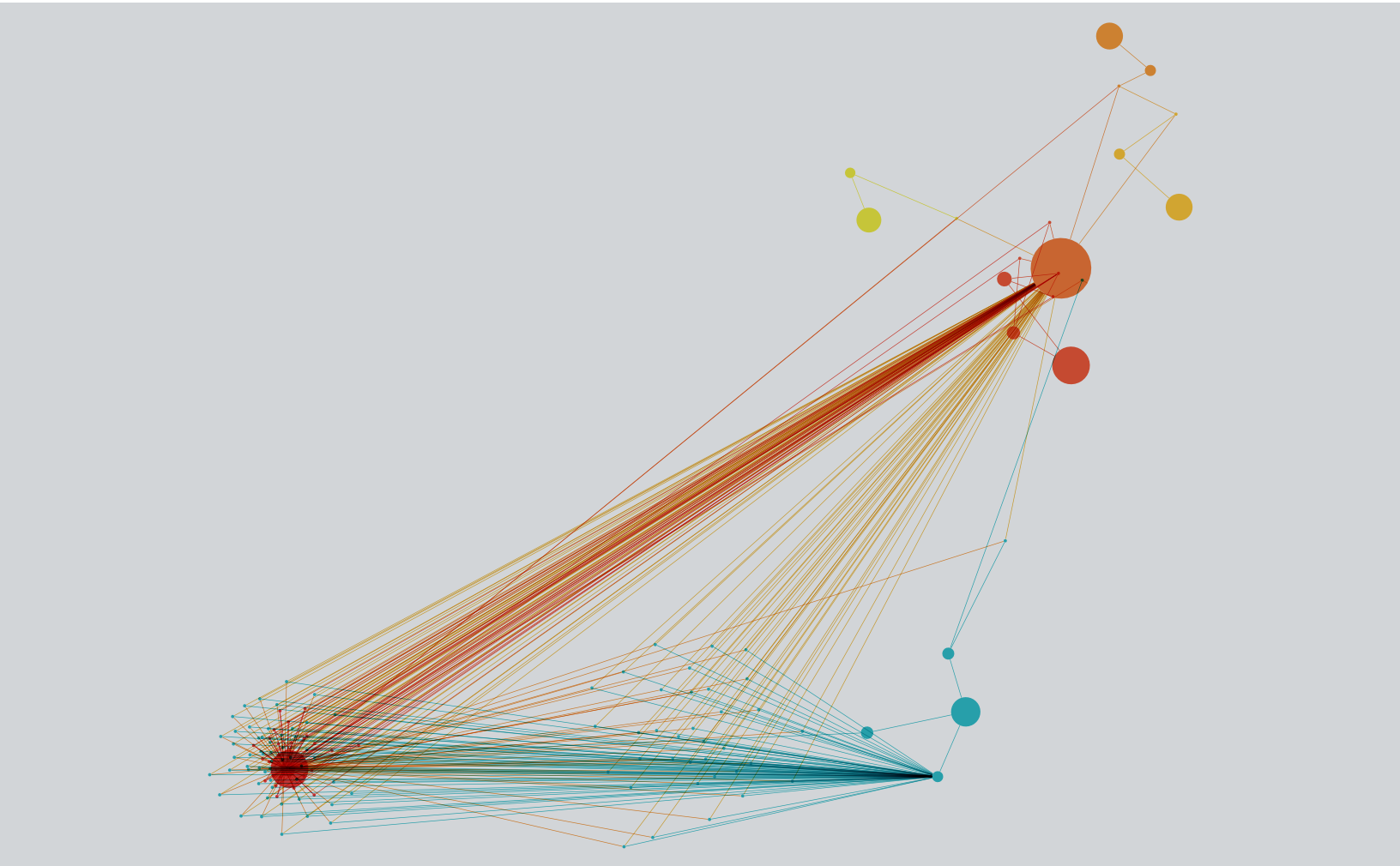
# Derisking machine learning and artificial intelligence

The added risk brought on by the complexity of machine-learning models can be mitigated by making well-targeted modifications to existing validation frameworks.

Bernhard Babel, Kevin Buehler, Adam Pivonka, Bryan Richardson, and Derek Waldron



FEBRUARY 2019 • RISK PRACTICE

Machine learning and artificial intelligence are set to transform the banking industry, using vast amounts of data to build models that improve decision making, tailor services, and improve risk management. According to the McKinsey Global Institute, this could generate value of more than $250 billion in the banking industry.[1]

But there is a downside, since machine-learning models amplify some elements of model risk. And although many banks, particularly those operating in jurisdictions with stringent regulatory requirements, have validation frameworks and practices in place to assess and mitigate the risks associated with traditional models, these are often insufficient to deal with the risks associated with machine-learning models.

Conscious of the problem, many banks are proceeding cautiously, restricting the use of machine-learning models to low-risk applications, such as digital marketing. Their caution is understandable given the potential financial, reputational, and regulatory risks. Banks could, for example, find themselves in violation of antidiscrimination laws, and incur significant fines–a concern that pushed one bank to ban its HR department from using a machine-learning résumé screener. A better approach, however, and ultimately the only sustainable one if banks are to reap the full benefits of machine-learning models, is to enhance model-risk management.

Regulators have not issued specific instructions on how to do this. In the United States, they have stipulated that banks are responsible for ensuring that risks associated with machine-learning models are appropriately managed, while stating that existing regulatory guidelines, such as the Federal Reserve's "Guidance on Model Risk Management" (SR11-7), are broad enough to serve as a guide.[2]

Enhancing model-risk management to address the risks of machine-learning models will require policy decisions on what to include in a model inventory, as well as determining risk appetite, risk tiering, roles and responsibilities, and model life-cycle controls, not to mention the associated model-validation practices. The good news is that many banks will not need entirely new model-validation frameworks. Existing ones can be fitted for purpose with some well-targeted enhancements.

## New risks, new policy choices, new practices

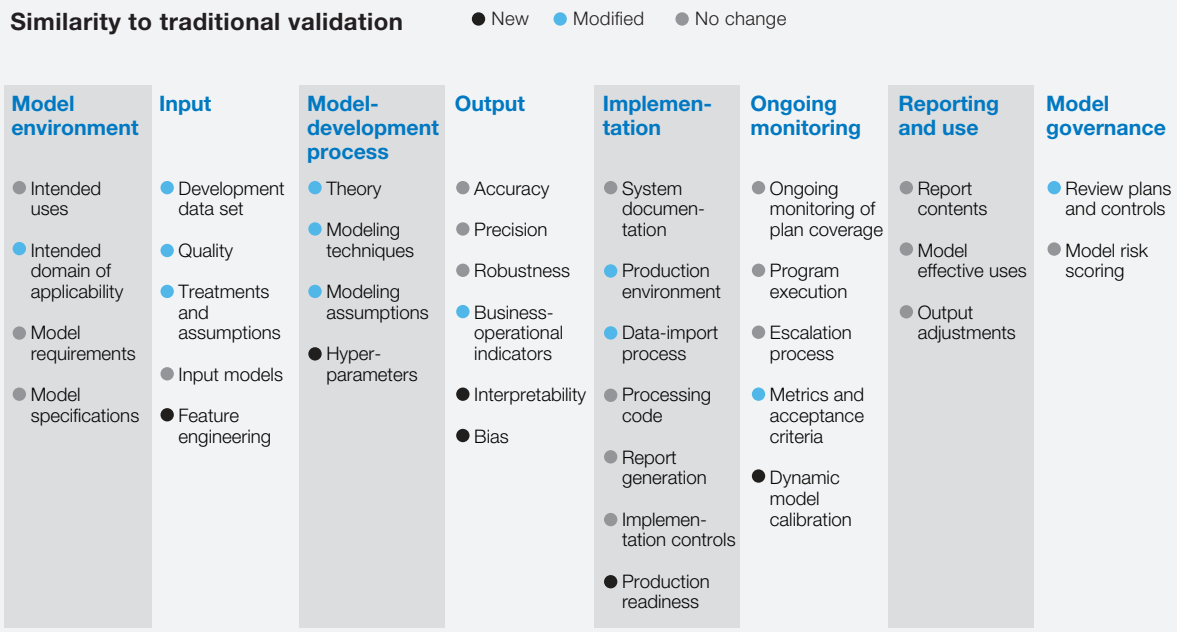
There is no shortage of news headlines revealing the unintended consequences of new machine-learning models. Algorithms that created a negative feedback loop were blamed for the "flash crash" of the British pound by 6 percent in 2016, for example, and it was reported that a self-driving car tragically failed to properly identify a pedestrian walking her bicycle across the street.

The cause of the risks that materialized in these machine-learning models is the same as the cause of the amplified risks that exist in all machine-learning models, whatever the industry and application: increased model complexity. Machine-learning models typically act on vastly larger data sets, including unstructured data such as natural language, images, and speech. The algorithms are typically far more complex than their statistical counterparts and often require design decisions to be made before the training process begins. And machine-learning models are built using new software packages and computing infrastructure that require more specialized skills.

The response to such complexity does not have to be overly complex, however. If properly understood, the risks associated with machine-learning models can be managed within banks' existing model-validation frameworks, as the exhibit below illustrates.

Exhibit

**Existing validation frameworks can address machine-learning-model risk with some well-targeted enhancements.**




Highlighted in the exhibit are the modifications made to the validation framework and practices employed by Risk Dynamics, McKinsey's model-validation arm. This framework, which is fully consistent with SR11-7 regulations and has been used to validate thousands of traditional models in many different fields of banking, examines eight risk-management dimensions covering a total of 25 risk elements. By modifying 12 of the elements and adding only six new ones, institutions can ensure that the specific risks associated with machine learning are addressed.

## The six new elements

The six new elements—interpretability, bias, feature engineering, hyperparameters, production readiness, and dynamic model calibration—represent the most substantive changes to the framework.

### Interpretability

Machine-learning models have a reputation of being "black boxes." Depending on the model's architecture, the results it generates can be hard to understand or explain. One bank worked for months on a machine-learning product-recommendation engine designed to help relationship managers cross-sell. But because the managers could not explain the rationale behind the model's recommendations, they disregarded them. They did not trust the model, which in this situation meant wasted effort and perhaps wasted opportunity. In other situations, *acting upon* (rather than ignoring) a model's less-than-transparent recommendations could have serious adverse consequences.

The degree of interpretability required is a policy decision for banks to make based on their risk

appetite. They may choose to hold all machine-learning models to the same high standard of interpretability or to differentiate according to the model's risk. In the United States, models that determine whether to grant credit to applicants are covered by fair-lending laws. The models therefore must be able to produce clear reason codes for a refusal. On the other hand, banks might well decide that a machine-learning model's recommendations to place a product advertisement on the mobile app of a given customer poses so little risk to the bank that understanding the model's reasons for doing so is not important.

Validators need also to ensure that models comply with the chosen policy. Fortunately, despite the black-box reputation of machine-learning models, significant progress has been made in recent years to help ensure their results are interpretable. A range of approaches can be used, based on the model class:

- Linear and monotonic models (for example, linear-regression models): linear coefficients help reveal the dependence of a result on the output.

- Nonlinear and monotonic models, (for example, gradient-boosting models with monotonic constraint): restricting inputs so they have either a rising or falling relationship globally with the dependent variable simplifies the attribution of inputs to a prediction.

- Nonlinear and nonmonotonic (for example, unconstrained deep-learning models): methodologies such as local interpretable model-agnostic explanations or Shapley values help ensure local interpretability.

## Bias

A model can be influenced by four main types of bias: sample, measurement, and algorithm bias, and bias against groups or classes of people. The latter two types, algorithmic bias and bias against people, can be amplified in machine-learning models.

For example, the random-forest algorithm tends to favor inputs with more distinct values, a bias that elevates the risk of poor decisions. One bank developed a random-forest model to assess potential money-laundering activity and found that the model favored fields with a large number of categorical values, such as occupation, when fields with fewer categories, such as country, were better able to predict the risk of money laundering.

To address algorithmic bias, model-validation processes should be updated to ensure appropriate algorithms are selected in any given context. In some cases, such as random-forest feature selection, there are technical solutions. Another approach is to develop "challenger" models, using alternative algorithms to benchmark performance.

To address bias against groups or classes of people, banks must first decide what constitutes fairness. Four definitions are commonly used, though which to choose may depend on the model's use:

- *Demographic blindness:* decisions are made using a limited set of features that are highly uncorrelated with protected classes, that is, groups of people protected by laws or policies.

- *Demographic parity:* outcomes are proportionally equal for all protected classes.

- *Equal opportunity:* true-positive rates are equal for each protected class.

- *Equal odds:* true-positive and false-positive rates are equal for each protected class.

Validators then need to ascertain whether developers have taken the necessary steps to ensure fairness. Models can be tested for fairness and, if necessary, corrected at each stage of the model-development process, from the design phase through to performance monitoring.

### Feature engineering

Feature engineering is often much more complex in the development of machine-learning models than in traditional models. There are three reasons why. First, machine-learning models can incorporate a significantly larger number of inputs. Second, unstructured data sources such as natural language require feature engineering as a preprocessing step before the training process can begin. Third, increasing numbers of commercial machine-learning packages now offer so-called AutoML, which generates large numbers of complex features to test many transformations of the data. Models produced using these features run the risk of being unnecessarily complex, contributing to overfitting. For example, one institution built a model using an AutoML platform and found that specific sequences of letters in a product application were predictive of fraud. This was a completely spurious result caused by the algorithm's maximizing the model's out-of-sample performance.

In feature engineering, banks have to make a policy decision to mitigate risk. They have to determine the level of support required to establish the conceptual soundness of each feature. The policy may vary according to the model's application. For example, a highly regulated credit-decision model might require that every individual feature in the model be assessed. For lower-risk models, banks might choose to review the feature-engineering process only: for example, the processes for data transformation and feature exclusion.

Validators should then ensure that features and/or the feature-engineering process are consistent with the chosen policy. If each feature is to be tested, three considerations are generally needed: the mathematical transformation of model inputs, the decision criteria for feature selection, and the business rationale. For instance, a bank might decide that there is a good business case for using debt-to-income ratios as a feature in a credit model but not frequency of ATM usage, as this might penalize customers for using an advertised service.

### Hyperparameters

Many of the parameters of machine-learning models, such as the depth of trees in a random-forest model or the number of layers in a deep neural network, must be defined before the training process can begin. In other words, their values are not derived from the available data. Rules of thumb,

An institution built a model using an AutoML platform and found that specific sequences of letters in a product application were predictive of fraud—a spurious result.

parameters used to solve other problems, or even trial and error are common substitutes. Decisions regarding these kinds of parameters, known as hyperparameters, are often more complex than analogous decisions in statistical modeling. Not surprisingly, a model's performance and its stability can be sensitive to the hyperparameters selected. For example, banks are increasingly using binary classifiers such as support-vector machines in combination with natural-language processing to help identify potential conduct issues in complaints. The performance of these models and the ability to generalize can be very sensitive to the selected kernel function.

Validators should ensure that hyperparameters are chosen as soundly as possible. For some quantitative inputs, as opposed to qualitative inputs, a search algorithm can be used to map the parameter space and identify optimal ranges. In other cases, the best approach to selecting hyperparameters is to combine expert judgment and, where possible, the latest industry practices.

### Production readiness

Traditional models are often coded as rules in production systems. Machine-learning models, however, are algorithmic, and therefore require more computation. This requirement is commonly overlooked in the model-development process. Developers build complex predictive models only to discover that the bank's production systems cannot support them. One US bank spent considerable resources building a deep learning–based model to predict transaction fraud, only to discover it did not meet required latency standards.

Validators already assess a range of model risks associated with implementation. However, for machine learning, they will need to expand the scope of this assessment. They will need to estimate the volume of data that will flow through the model, assessing the production-system architecture (for example, graphics-processing units for deep learning), and the runtime required.

### Dynamic model calibration

Some classes of machine-learning models modify their parameters dynamically to reflect emerging patterns in the data. This replaces the traditional approach of periodic manual review and model refresh. Examples include reinforcement-learning algorithms or Bayesian methods. The risk is that without sufficient controls, an overemphasis on short-term patterns in the data could harm the model's performance over time.

Banks therefore need to decide when to allow dynamic recalibration. They might conclude that with the right controls in place, it is suitable for some applications, such as algorithmic trading. For others, such as credit decisions, they might require clear proof that dynamic recalibration outperforms static models.

With the policy set, validators can evaluate whether dynamic recalibration is appropriate given the intended use of the model, develop a monitoring plan, and ensure that appropriate controls are in place to identify and mitigate risks that might emerge. These might include thresholds that catch material shifts in a model's health, such as out-of-sample performance measures, and guardrails such as exposure limits or other, predefined values that trigger a manual review.

Banks will need to proceed gradually. The first step is to make sure model inventories include all machine learning–based models in use. You may be surprised to learn how many there are. One bank's model risk-management function was certain the organization was not yet using machine-learning models, until it discovered that its recently established innovation function had been busy developing machine-learning models for fraud and cybersecurity.

From here, validation policies and practices can be modified to address machine-learning-model risks, though initially for a restricted number of model classes. This helps build experience while testing and refining the new policies and practices. Considerable time will be needed to monitor a model's performance and finely tune the new practices. But over time banks will be able to apply them to the full range of approved machine-learning models, helping companies mitigate risk and gain the confidence to start harnessing the full power of machine learning. ■

[1] For the purposes of this article machine learning is broadly defined to include algorithms that learn from data without being explicitly programmed, including, for example, random forests, boosted decision trees, support-vector machines, deep learning, and reinforcement learning. The definition includes both supervised and unsupervised algorithms. For a full primer on the applications of artificial intelligence, we refer the reader to https://www.mckinsey.com/business-functions/mckinsey-analytics/our-insights/an-executives-guide-to-ai.

[2] Lael Brainard, *What are we learning about artificial intelligence in financial services?*, Fintech and the New Financial Landscape, Philadelphia, PA, November 13, 2018, federalreserve.gov.

**Bernhard Babel** is a partner in McKinsey's Cologne office; **Kevin Buehler** is a senior partner in the New York office, where **Adam Pivonka** is an associate partner and **Derek Waldron** is a partner; **Bryan Richardson** is a senior expert in the Vancouver office.

The authors wish to thank Roger Burkhardt, Pankaj Kumar, Ryan Mills, Marc Taymans, Didier Vila, and Sung-jin Yoo for their contributions to this article.