Roel Smits MSc
is a data scientist at KPMG.
smits.roel@kpmg.nl

Frank van Praat MA MSc RE
is a senior manager at KPMG NL.
vanpraat.frank@kpmg.nl

Trusting algorithms: governance by utilizing the power of peer reviews

Organizations that are able to build and deploy algorithms at scale are tapping into a power for insights and decision-making that potentially far exceeds human capability. However, incorrect algorithms or the inability to understand/explain how algorithms work, can be destructive when they produce inaccurate or biased results. It makes management therefore hesitant to hand over their decision-making to machines without knowing how they work. In this article, we explore how decision-makers can take on this responsibility via trusted analytics by laying out a high-level governance framework that reserves a special position for peer reviews.

Incorrect algorithms can be destructive when they produce inaccurate or biased results

INTRODUCTION

Over the past decade, we have seen an enormous growth in data and data usage for decision-making. This is likely to continue exponentially in the coming years, possibly resulting in 163 zettabytes (ZB) of data by 2025. That's ten times the amount of data produced in 2017 ([Paul19]). Obviously, organizations are looking for ways to leverage the huge amounts of data they have. Some organizations sell it, others build capabilities to analyze the data in order to enhance business processes, decision-making or to generate more revenue or gain more market share.

Regarding the latter, organizations tend to increasingly use advanced analytics techniques, such as machine learning, to analyze the data. Although it goes without saying that these techniques show real value and unmistakably perform better than traditional techniques, there is also a downside. Advanced analytics techniques are inherently more difficult to understand as they are more complex. The combination of complexity and huge amounts of data to work with, individual analyses (often referred to as 'algorithms') sometimes are perceived to operate as 'black boxes'. This is a problem for organizations that want to become more data-driven, as decision makers rely on these algorithms. Decision-makers have the responsibility to be able to trust them. They therefore need to balance the value coming from these advanced analytics techniques, with the need for trustworthiness to use them properly.

In this article, we explore how decision-makers can take on this responsibility via trusted analytics by laying out a high-level governance framework. Subsequently, we deep dive into one crucial aspect that should be part of it: peer reviews.

CHALLENGES IN TRUST

Organizations that are able to build and deploy algorithms at scale are tapping into a power for insights and decision-making that potentially far exceeds human capability. But incorrect algorithms can be destructive when they produce inaccurate or biased results. It makes decision-makers therefore hesitant to hand over decisions to machines without knowing how they work. Later in this article, we propose a high-level structure for decision-makers to take on the responsibility to trust the algorithms they want to rely on, but first we need to understand their challenges. Based on our experience, we listed some of the key questions that we receive when organizations aim to deploy algorithms at scale:

- How do you know if our algorithms are actually doing what they are supposed to do? Both now, as well as in the future?
- How do we know if our algorithms are actually in compliance with the laws and regulations that are applicable to our organization?
- How do we know if our algorithms are actually built in alignment with our own, and industry-wide standards and guidelines?
- How do we know if our algorithms are inclusive, fair and make use of appropriate data?
- How do we know if our algorithms are still valid when the world around us changes?
- How do we know if our algorithms are still valid when our organization makes a strategic change, e.g. optimizing on profit instead of turnover?
- How do we know if our algorithms can still be understood if key people that worked on it leave the organization?

Obviously, for trusting algorithms to achieve their objective and for decision-makers to assume responsibility and accountability of their results, it's essential to establish a framework (powered by methods and tools) to address these challenges and to facilitate responsible adoption and scale of algorithms. Yet, we also know there is a contradictory force that typically holds back the implementation of such a framework: the need for innovation. Because if we take a closer look how advanced analytics techniques are applied in practice, we notice that algorithms are often the result of cycles of trial and error driven by data scientists and other experts in search of valuable insights coming from data. It is a highly iterative process that benefits from a lot of freedom. If the only goal is to empower innovation, this approach is obviously very helpful. But as soon as the goal is to actually build algorithms that are ready for production, this same level of freedom will probably cause insufficient basis to do so. Because how can a decision-maker trust an algorithm that was developed by trial and error?

HIGH-LEVEL FRAMEWORK TO GOVERN ALGORITHMS

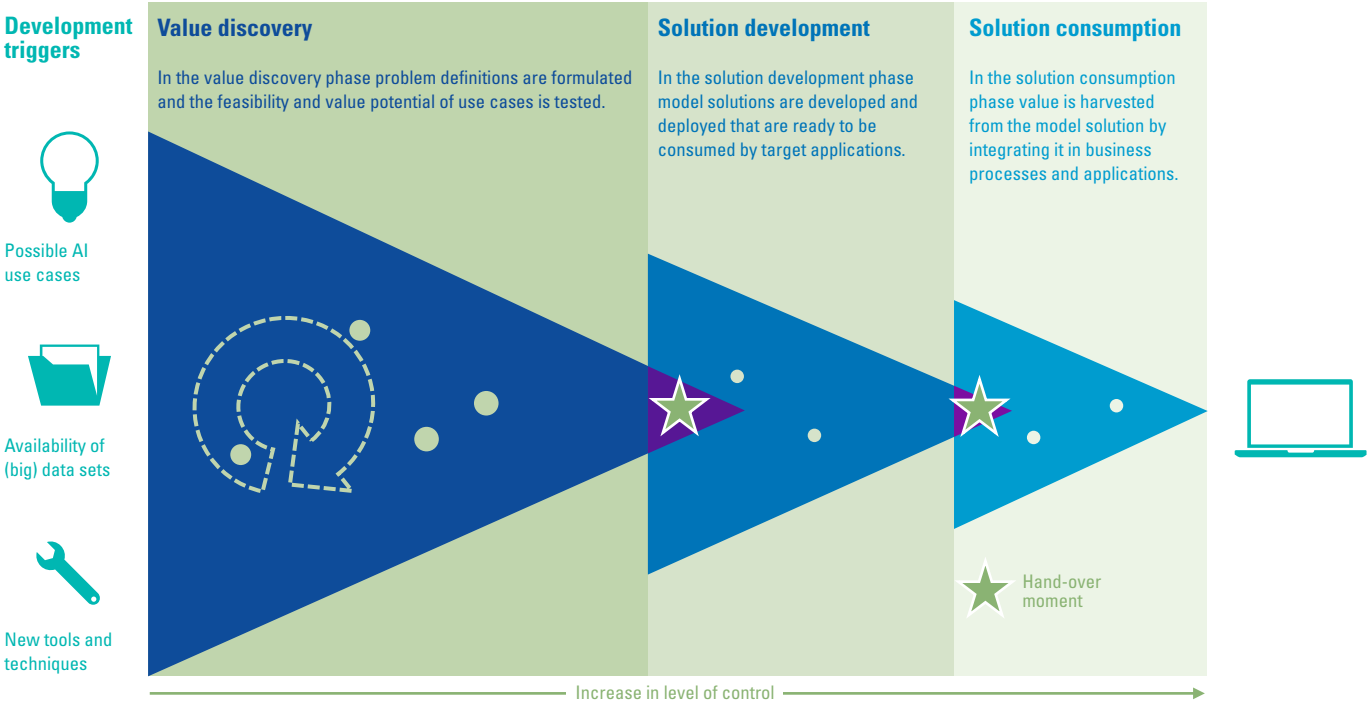
In the previous paragraph, we summarized the challenges of decision-makers and explained why in advanced analytics developments it is of upmost importance to carefully balance innovation and governance throughout a non-linear staged process: insufficient control leads to algorithms that cannot be trusted, while too much control will stifle innovation and therefore negatively impact the competitive power of organizations.

We believe the solution lies in a governance framework that uses a three-phased approach, in which each phase has its own level of control. In the first phase, the level of control is relatively low as this will help empower innovation. It will result in "minimum viable algorithms" that can be further developed in the solution development phase, which holds an increased level of control. Lastly, in the consumption phase, algorithms are actually deployed and monitored. If advanced analytics techniques are applied according to the lines of these three phases, the result should be inherently trustworthy algorithms. Per phase, the framework should consist of specific checks and balances that will help to govern the entire process, balancing the level of control in each phase. During each hand-over moment from one phase into the next, these checks and balances act as entry criteria for the next phase, which can be verified via internal peer reviewers.

- Value discovery: in this phase, data scientists, engineers and developers search ('experiment') for interesting use cases for advanced analytics solutions and test these in a simulated environment.
- Solution development: in this phase, as soon as there are 'minimum viable algorithms', these will be further developed and made ready for production (agile development of algorithms).
- Solution consumption: in this phase, the actual algorithms are used in a real-world environment with (semi-)autonomous and continuous improvement cycles.
- Hand-over moments: during the hand-over moments, the entry criteria for each new phase should be met. A check that can be performed via internal peer reviews.

How can a decision-maker trust an algorithm that was developed by 'trial & error'?

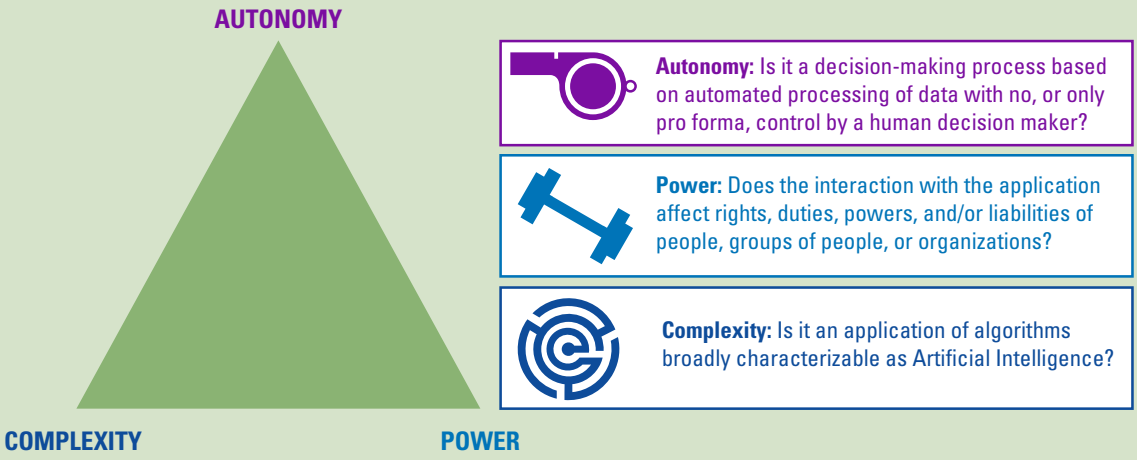
Figure 1. High-level governance framework.



Balancing impact and control

One of the checks and balances in the value discovery phase is to clearly define the purpose of the algorithm. This will help to assess for instance if an algorithm aligns with the principles (values and ethics) of the organization and if it complies to applicable laws and regulations. Furthermore, a clearly defined purpose should also be the starting point to assess the potential (negative) impact of the proposed algorithmic solution. We believe such an impact assessment is more or less crucial as it will lay the groundwork to enforce the appropriate level of control in the solution development and consumption phase. This is important because it would be a cost-worthy exercise to enforce maximum control over algorithms that have only a relatively low impact. An example of low-impact algorithm is an algorithm that is used in a 5-store building to optimally route lifts to appropriate floors. Though useful, the impact is relatively low. If you compare that to an algorithm that is used to detect tumors in MRI images, obviously it will score much higher on the impact ladder. But what is "impact"? We believe it emanates from an aggregation of three criteria: Autonomy, Power and Complexity (see Figure 2).

Figure 2. Algorithm impact assessment criteria.



By matching the level of control to the algorithm's impact, the cost of control ([Klou19]) can be managed as part of the high-level governance framework as well.

PEER REVIEWS

Now that we have introduced a high-level governance framework to govern advanced analytics developments, as stated in the introduction, peer reviews can play a very important role. In the remainder of this article, we will discuss how.

The scientific community uses peer reviews already for decades as a quality control system to help decide if an article should be published or not ([Beno07]). A scientific peer review consists of multiple stages in which scientists, independent from the authors, are reviewing the work done by their peers. From our experience we have learned that parts of such a control can be very helpful in an algorithm context as well as an extra pair of eyes in the development cycle. We are convinced that it helps increase the level of trust in business-critical algorithms before they are deployed into a live environment (e.g. the 'latest' phase of our high-level algorithm governance framework).

We will present an overview of topics that we consider most important when performing an external peer review. We will elaborate on how they are positioned in the algorithm development cycle and provide some guidance of relevant aspects to consider when performing a review. Subsequently, we will disclose our most important lessons learned. From there, we will conclude the article by describing how we think the presented peer review topics can help overcome the key challenges as described in the introduction and how organizations can leverage internal peer reviews as part of the high-level governance framework.

Peer review topics

The process of an external peer review is visualized in Figure 3. It basically consists of three stages that cover a specific number of topics. Each stage feeds information into the following.

The basis of a peer review, and therefore the first step, is to get a basic understanding of the algorithm by reading as much relevant documentation as possible. This is combined with getting an understanding of the ways of working of the development team. The next stage is the core of the peer review because it puts focus on the performance and quality of the algorithm itself. This stage has overlap with software quality reviews because topics like "production pipeline", "tests", "code quality" and "platform" could be found in a software quality review as well. In the last stage all findings are summarized, aligned and reported to the developers and management team. In the following sections, we will define how a review topic relates to the development cycle of algorithms, and we disclose their practical implications when performing peer reviews.

Way of working

The (agile) way of working for algorithm development team highly impacts its hygiene. It determines if tasks and responsibilities are shared and if single points of failures are prevented. A proper way of working stimulates innovation and consistency.

Implications in practice

- During a peer review, we look at processes, development roles and maintenance tasks. Think of different permissions in the version control system of the code base, branching strategies, or obligatory data scientist rotations. For example, the latter rotations will enable brainstorming discussions and therefore opens up room for innovative ideas while it prevents single points of failure.
- Another topic that we assess as part of 'way of working' is how the team is managing 'service tasks' (e.g. operational activities such as running periodic reports to monitor the algorithm's performance). Ideally, this responsibility is shared across the team, as it will increase the level of ownership and knowledge by the team members.

Logic and models

In algorithm development, logic and models determine if the output of an algorithm is accurate, fit for its purpose and therefore trustworthy to support business decision making. When we perform a peer review, we investigate the mathematical and statistical correctness of, and intention behind, the logic and models. We assess, for example, whether all assumptions of a statistical model are satisfied. If possible, we also try to suggest improvement points to optimize the algorithm's performance.

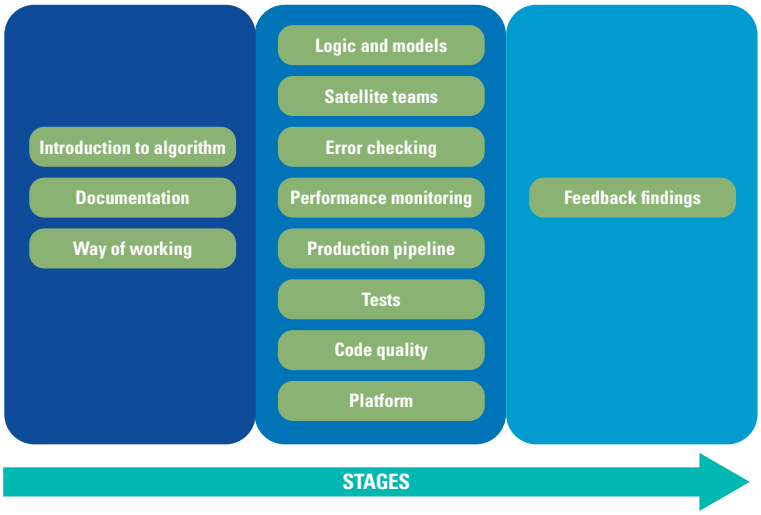


Figure 3. Peer review stages and topics.

As all production data for algorithm training purposes has incorporated some level of bias, we also need to verify how the algorithm ensures that it will not get biased without flagging or alerting the users of its output, basically a baseline test.

Satellite teams

The various teams that contribute to the core data science team that build algorithms are also commonly referred to as 'satellite teams'. As part of a peer review, we assess the collaboration between the teams that work together on an algorithm. We focus on the teams that are either involved in data preparation, or teams that have to use the outcomes of a model. We evaluate these teams as part of our review process, to provide a good insight in the end-to-end lifecycle of data-driven decision-making.

Implications in practice

We consider the following teams as satellite teams in data-driven decision-making, amongst others: data-engineering teams or analytics-platform teams, teams that provide the input data, teams that use the algorithm output, and teams that are responsible for error checking and monitoring.

Error checking

In algorithm development, an indispensable factor to increase the overall performance of individual algorithms is performing root cause analysis on incorrectly labelled outputs (such as false positives/negatives or completely inaccurate outcomes). We call this error checking. During a peer review, we evaluate the error checking in place so that we can make sure that the errors made in the past will be prevented in the future.

Implications in practice

Questions to ask when performing a review:
- What process is in place to review errors?
- Do you use real-time alerts (continuous monitoring) or do you periodically review logs?
- Do you make use of tooling to automatically detect errors?
- Do you have a dedicated team working on root cause analysis of errors?
- Do you have a way to prioritize the errors for further investigations?

Performance monitoring
Performance monitoring provides insights into how well an algorithm is performing in terms of for example accuracy, precision, recall or others. The performance of an algorithm should be taken into account when decisions are based on the outcomes, as the performance will provide details on the uncertainty of these outcomes. Monitoring on a continuous basis is even better, as it provides insights into the overall stability of an algorithm. For instance, a downward trend of the overall performance might indicate that an algorithm has to be further aligned to certain (external) changes in the sector or market it operates in.

Implications in practice

We look at:
- How the precision or recall of an algorithm is monitored over time.
- How fine-grained is the monitoring. For example, in cases of deteriorating performance, are teams able to get sufficient detail from the monitoring dashboards to be able to assess the root cause?
- How are monitoring teams able to drill down to Key Performance Indicators (KPI's) and summary statistics of small enough subgroups of the data sample an algorithm is created from.

Production pipeline
The performance and stability of a production pipeline determine the stability and consistency of algorithms over time. In addition, a well-structured pipeline makes the development or update cycles of algorithms shorter.

Implications in practice

Questions to ask when performing a review:
- Have you implemented a job scheduler? And how is it monitored?
- How much time does it take for each part of the production pipeline to run?
- Have you put an alert system in place to notify teams if a part of the pipeline fails?
- Have you assigned responsibilities for follow up in case of failure?

Software tests and code quality (combined)
Typically, software tests and code quality aren't directly associated with algorithm development. Both aspects are actually very important to consider. Software tests will help ensure that algorithms are actually doing what they are supposed to do. Good code quality makes updating and maintaining algorithms a lot easier in comparison to algorithms that are built on spaghetti code.

Implications in practice

Questions to ask when performing a review:
- As part of algorithm development, have you performed sanity checks, unit tests, integration/regression tests and A/B tests?
- How have you structured the algorithm code? Is it for example modular, or based on one long script?

Platform
From a technological perspective, the basis for algorithm development lies at the tools and resources data analysts have to work with. These tools and resources are typically provided by platforms. Well known platforms are for example Amazon Web Services (AWS), Google Cloud or Microsoft Azure. These platforms typically work with all sorts of open-source frameworks such as Pytorch or Tensorflow.

Implications in practice

Questions to ask when performing a review:
- Which frameworks, industry standard packages and software libraries do you use?
- How do you make sure these frameworks, packages and libraries and are up to date?
- How do you ensure that the platform you are using is stable and future-proof? Will the platform be able to handle the potential growth of data and users, for example?

Experiences: challenges and lessons learned

During our peer reviews on specific algorithms, or their development cycles, we have come across some lessons learned that are very relevant to consider when internal or external peer review processes are implemented as part of a governance framework. We have listed three of them that we consider as the most relevant:

- **Documentation makes not only the peer reviewer's job easier, but it also will help ensure that the algorithm can be verified by someone else other than the developers.** We often notice that documentation of algorithms is only partly available, or not up to date. We believe this is because the return on investment is not high enough and proper documentation slows down the development cycle in general. However, we know from our experience that documentation makes a peer review's job much easier. Interviews usually give an ambiguous picture, for example because details are often not correctly remembered by the team members, making it difficult for a reviewer to get a comprehensive view on parts of the algorithm which are not properly documented. The traditional auditor's statement of "tell me, show me, prove me" also applies to peer reviews.
- **Organizational culture influences analytics.** From our peer reviews, we have learned that the culture in an organization greatly impacts algorithm developments. In a culture where mistakes are costly, or even a matter of people's safety, algorithms and software are usually properly documented, tested and formal procedures are in place to manage updates of production code and/or pipelines. In a fail-fast-learn-fast culture, the opposite is often true. In those cases, alternative procedures are required to compensate for the increased risk of failure that is caused by for example a general lack of testing (e.g. better monitoring).
- **Tailoring the reviewer's communication style enables constructive dialogue.** A final experience is that findings of a peer review should be carefully aligned and reported in accordance with the reviewee's needs. For example, an open team discussion to align and report findings from the peer review will enable a constructive discussion and room for the reviewees to disclose their concerns. On the other hand, a more traditional approach of reporting can help align findings amongst larger groups and enable management to enforce change.

CONCLUSION

Data science maturity is increasing rapidly. The growing industry is borrowing heavily from good practices in academia, where, especially in domains like high-energy physics, data science has already been running in a production-like setting for decades ([Klou18]). Peer reviews have proven indispensable in these domains because they:

- ensure that algorithms are fit for their purpose;
- ensure to identify and remove mistakes and flaws;
- ensure the algorithms do not solely reflect the opinion and work of only one person.

As we have shown, the peer review method follows a staged approach to examine a wide array of topics, critical to the quality of algorithms in question. If we link this to the need for decision-makers to trust algorithms and their outcomes, we believe that all topics are highly relevant to be integrated as part of a high-level governance framework. The topics "Logic and models", "Error checking", "Performance monitoring", and "Software tests and code quality" need to get specific attention because we believe these topics should be integrated as part of the internal peer reviews during the hand-over moments as well. In this way, a high-level framework that utilizes the power of peer reviews will help decision-makers take a good step forward in taking on the responsibility of trusting the algorithms that they rely on.

References

[Beno07] Benos, D.J. et al. (2007). The ups and downs of peer review. *Advances in Physiology Education*, Vol. 31, No. 2. Retrieved from: https://doi.org/10.1152/advan.00104.2006.

[Klou18] Klous, S. & Wielaard, N. (2018). *Building trust in a smart society*. Infinite Ideas Limited.

[Klou19] Klous, S. & Praat, F. van (2019). Algoritmes temmen zonder overspannen verwachtingen. Een nieuwe uitdaging op de bestuurstafel. *Jaarboek Corporate Governance 2019 – 2020*, p. 79-89.

[Paul19] Paulsen, J. (2019). Enormous Growth in Data is Coming – How to Prepare for It, and Prosper From It. Seagate Blog. Retrieved from: https://blog.seagate.com/business/enormous-growth-in-data-is-coming-how-to-prepare-for-it-and-prosper-from-it/.

About the authors

Roel Smits MSc is a data scientist at KPMG NL. Roel has a background in applied physics, with a specialization in solid state physics. Roel has experience in proof-of-concepts and reviewing existing products containing advanced analyses using statistical and artificial intelligence methods. For example, predicting railroad switch failures and finding causes of blocked valves in district heating networks.

Frank van Praat MA MSc RE is a senior manager at KPMG and leads KPMG's Trusted Analytics team. He has extensive knowledge about Emerging Technology Risk Management, Advanced analytics governance, and the human side of AI.