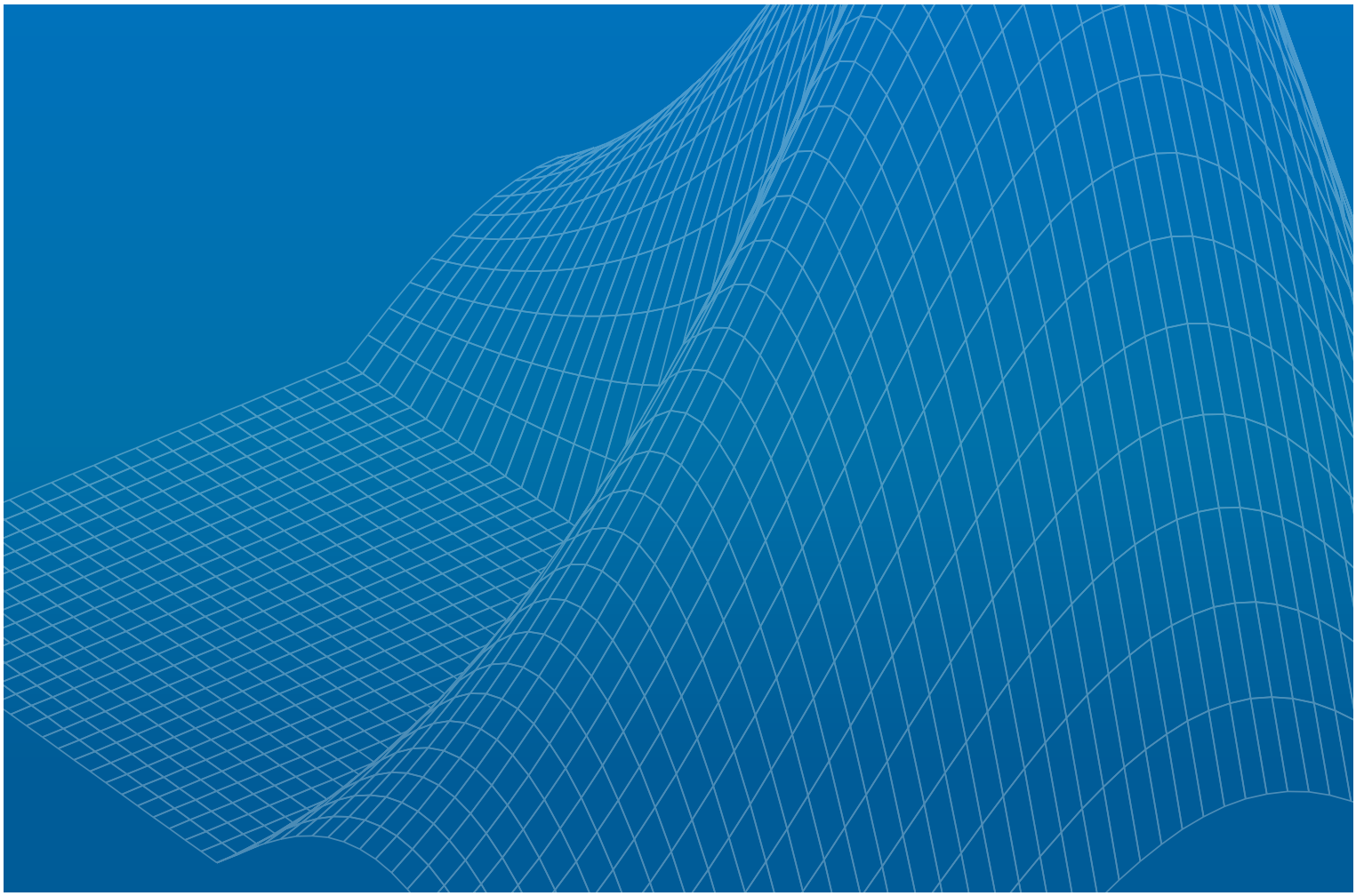


Financial Risk Management: Improving Model Governance with MATLAB

WHITE PAPER

MathWorks®

Introduction

Financial crises like the global financial crisis of 2008 can expose underlying faults in our financial systems and processes. The 2008 financial crisis exposed the failure of a prevalent laissez-faire economic and regulatory philosophy that financial institutions could and should manage their own risk with little outside interference.

Now many see the danger of fragile financial service networks bringing systemic risk to the real economy. Politicians, economists, financial managers, and the general public all came to appreciate the overweight significance of financial services factors on the micro- and macroeconomy, and subsequently sought to enforce change. Financial organizations face increasing scrutiny and regulation, including regulations such as **Basel III** for reserve capital allocation, the European Union's **Solvency II Directive** to enhance consumer protection in insurance, and the **International Financial Reporting Standard (IFRS) 9** for financial instrument accounting.

One topical regulation impacting compliance officers at systemically important global banks is the Basel-derived Fundamental Review of the Trading Book (FRTB). This impacts institutions active in Europe, stressing tail "extreme" risks at the desk level, aggregating them up to build a firm-wide view of market risks. But banks are not ready for them, and standalone implementations are complex and expensive, (according to ***one recent report***, costing banks $200 million).

However, regulations tend to repeat, differ, or be inconsistent across geographies and functions, and are rarely harmonized in a way that actually makes them easy to apply. Contrarily, supervised institutions are best placed to understand the inconsistencies, inefficiencies, and above all the risk of risk regulation.

With increasing scrutiny and regulation facing financial organizations, risk professionals are moving toward greater accountability, with formal risk assessment and management processes. Technology problems have plagued financial organizations, with legacy systems, technology "silos," vendor-supplied black boxes, and worse, some technology that adds risk to risk management.

This white paper describes how risk-aware institutions are using ***MATLAB®*** to reduce model risk and improve model governance. It also describes the opportunity to service multiple regulations from a single risk management stack.

Challenges Facing Risk-Aware Institutions

Errors Traced to Spreadsheets

The global financial crises of 2008 exposed numerous weak points in finance organizations worldwide. Many were traced to the humble Excel® spreadsheet, commonly used in finance to calculate, model, and share financial estimates and results. ***Preventable errors traced to spreadsheets*** have exposed the enormous risks created by sloppy processes. Spreadsheet miscalculations from a 2010 paper by Harvard economists Carmen Reinhart and Kenneth Rogoff, when corrected, appeared ***to reverse the authors' widely quoted conclusions*** about government debt and GDP growth. At J.P. Morgan, ***spreadsheet errors contributed*** to the nearly $6 billion "London Whale" losses.

Technical Debt

Compounding the risk of procedural errors and technical mistakes, firms face increasing complexity in maintaining and running their financial systems, not just in the present but in the future. IT departments use a good word for this: technical debt. Organizations struggle to manage their tech debt "interest" as demands on the system increase, leading to an entropic cycle of increasing maintenance and reducing functional improvement.

Legal Risk from Open Source Code

Risks also arise from poorly sourced executable code. Many open source software programs are used to construct financial algorithms. It is important to examine the source and validity of any open source code you employ. Also note that when financial engineering teams distribute applications containing open source code using the GPL agreement, those applications themselves can become open to the public domain, exposing a firm's intellectual property. The advantages of using open source code must be weighed against potential exposure to legal risks and costs.

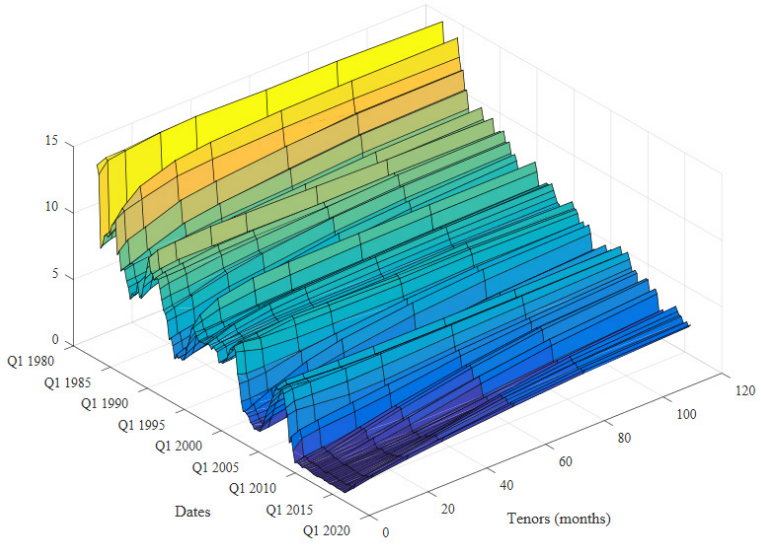
Rising Complexity

We suggest the complexity of modern financial systems requires processes that are more robust, less prone to error, and also incorporate validation and testing. Good model governance is key to ensuring accuracy, review, and transparency of the models. It is also vital to achieving software quality within an organization and facilitating model re-use across an organization.

Otherwise, organizations can waste money and increase risk, such as when they create and maintain multiple, siloed scenario generators across teams and projects–for example, to manage stress tests, IFRS scenarios, and balance sheet asset-liability management (ALM) scenarios.

Example: Responding to Stress Testing Cycles

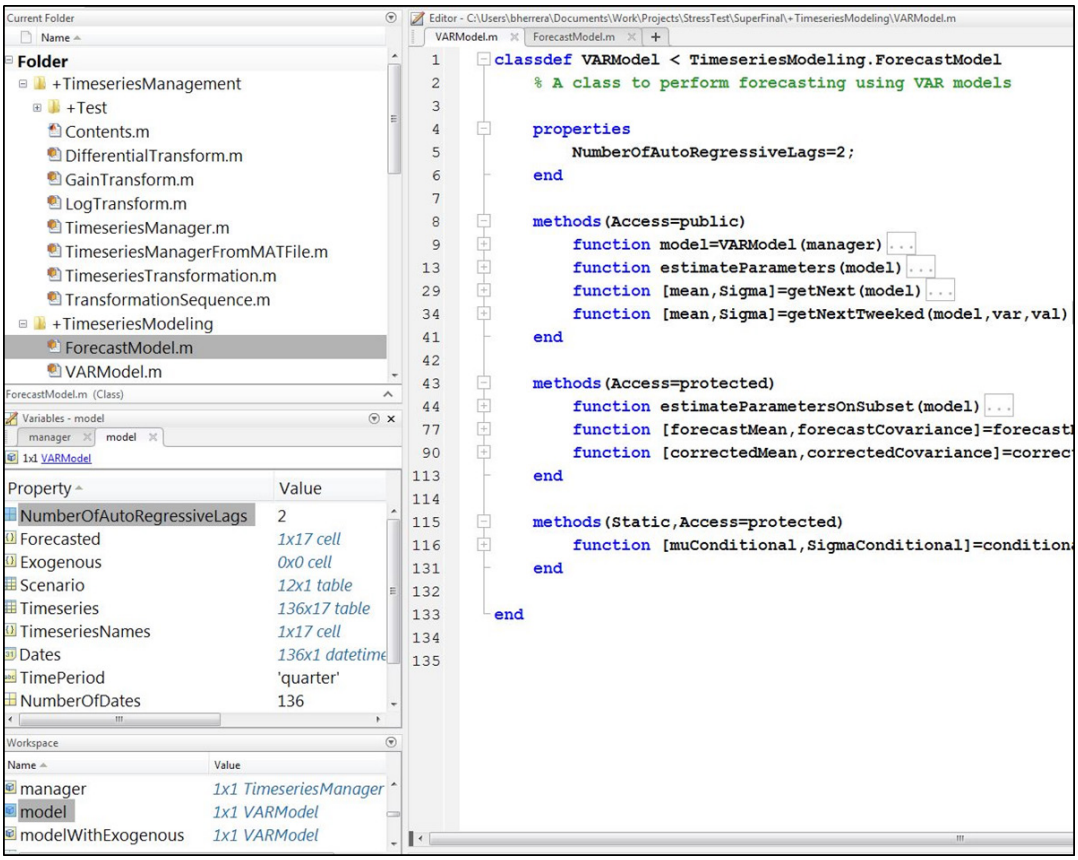
Central banks, regulatory bodies, and financial organizations work to analyze risk based on agreed stress testing frameworks, whether CCAR, EBA, or other standards. These stress tests analyze the effects of shocks on the economy and how they affect the capital resilience and liquidity coverage of banks.



Yield curve historical data and a simulated future projection.

Organizations struggle to muster their resources to address four to six month scenario-to-report timelines. Calculating and aggregating risk valuations across different asset classes and departments, often applying different methodologies in each, can result in densely populated data spreadsheets (or similar) passing from person to person, department to department.

A more robust process uses MATLAB to build an integrated and efficient stress testing platform. MATLAB offers capabilities invaluable to stress testing frameworks, including formal methods to encapsulate scenario or default types, model types and projections, calibration approaches, and impairment or default calculations. In addition, these capabilities let you automatically report output with comprehensive documentation, and reproduce stress test scenarios and output in case of specific challenge by the supervisor.



Supplying a scenario set to a suite of macroeconomic models, with reproducible data in case of supervisor query.

Solutions Call for Complex Systems, Managed Smartly

Risk-aware institutions can and must manage complexity in risk better–not by patching together legacy systems with more spreadsheets and tools of unknown origin. Smart risk managers work in a harmonized technology platform that manages, models, and reports risk in the organization. This platform should be able to deal with requests from regulators and supervisors.

This software goal is realizable. Nonfinancial organizations such as the aerospace or automotive industry have built automated safety-first model-to-production processes, with validation and verification built-in.

Financial services can and should strive to do the same, as the industry strives for good model governance and reduced model risk. It is quite possible for risk, projection, and valuation models to be built, customized, and improved, rapidly, consistently and in coordination. Risk-aware institutions can implement these systems while minimizing technical debt, applying good development processes that in turn foster continuous system improvement. It is quite possible too for those models to be made available to whoever needs them, whether ardent researcher, FATCA-liable executive, or prospective customer.

However, this requires cultural change as well as technical improvement. Financial institutions must seek to reduce complexity where complexity adds nothing, both in communication and in model development. Financial institutions should aspire to a single system with reduced operational, model and legal risks, servicing multiple disconnected supervisory regimes, in turn improving productivity through risk-aware development.

Using MATLAB for Risk Management

To minimize risk, while embracing the complexity of modern financial systems, risk-aware organizations increasingly use MATLAB. Put simply, MATLAB is the easiest and most productive environment for risk modelers, as it is for engineers and scientists. In the case of risk workflows specifically, it is also the most transparent and most assured platform.

With MATLAB, risk analysts and managers work with developers, integrators, stakeholders, and chief risk officers to blend, scale, and customize research.

At its best, when using MATLAB a single risk model stack can service multiple compliance regimes, and also multiple front and middle office functions. When use is more ad hoc, MATLAB can facilitate and automate market, credit, economic capital, and systemic risk modeling.

Banks, asset managers, supervisors, and insurers use MATLAB in:

- Capital requirements regimes such as Basel
- Stress test infrastructures such as ***CCAR and DFAST in the US; SSM and EBA in the EU***
- Accounting regulations such as IFRS 9 and trade reconciliation requirements such as MiFID
- Market risk, credit risk, operational risk, and compliance and fraud monitoring

MATLAB facilitates risk-aware development using industry best practices. Mathematicians, quants, data scientists, and others use MATLAB to perform risk calculations that are faster than spreadsheets. They create models more quickly than in C++, with greater transparency and customization than black box products, and with greater quality and consistency than open source applications.

MATLAB makes it easy to incorporate new and updated models directly into professional risk applications. It is easier for IT teams to maintain and troubleshoot components, without adding to technical debt. Teams can ensure quality and accuracy by incorporating unit and performance tests automatically.

Why MATLAB?

Seven Reasons to Use MATLAB for Risk-Aware Development

Financial organizations rely on MATLAB for developing transparent, documented, and replicable risk and stress testing models in days, not years. MATLAB offers agility in the midst of rapidly changing regulatory and business environments.

The MATLAB environment reduces risk, increases productivity, and accelerates development in seven key ways:

1. MATLAB speaks math.
2. MATLAB is designed for financial engineering and data analytics.
3. MATLAB toolboxes just work.
4. MATLAB has apps.
5. MATLAB integrates workflows.
6. MATLAB is fast.
7. MATLAB is trusted.

1. MATLAB Speaks Math

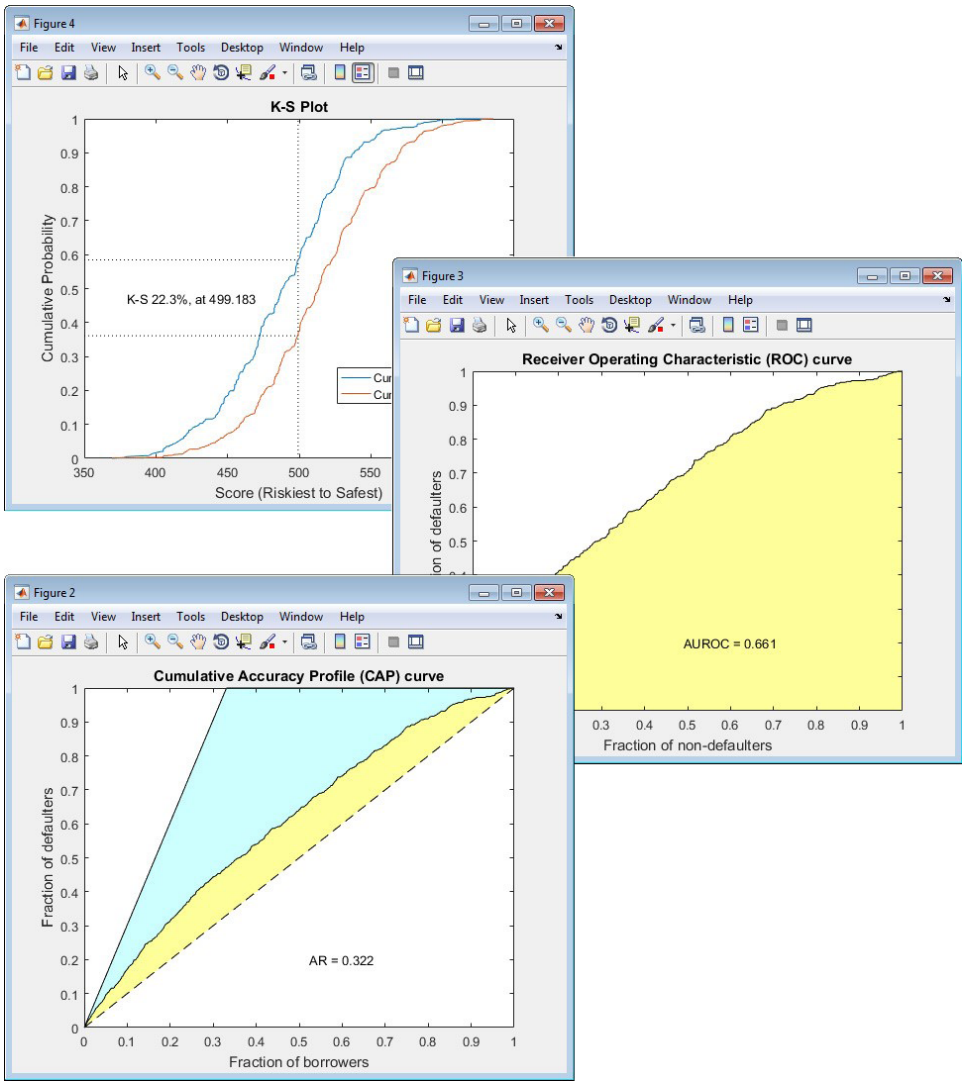
MATLAB fosters productivity. Financial engineers, data scientists, and economists need a programming environment that lets them express financial mathematics directly. Mathematical equations and models look very familiar in MATLAB code–unlike other programming languages.

> *"[MATLAB] is not as complex to code as C++. It is not as restrictive in terms of applications as Excel.... For communicating the idea—selling the idea—MATLAB has helped a lot.... I can create visuals very quickly."*
>
> *—Attilio Meucci*

2. MATLAB Is Designed for Financial Engineering and Data Analytics

Everything about MATLAB is designed specifically for engineers and scientists. It has dedicated functionality for computation, data analytics, and risk management.

> *"Risk calculations that would take an hour with a spreadsheet-based system are completed in a few minutes with MATLAB."*
>
> *—Athanasios Bolmatis, Fulcrum Asset Management*



Validating credit scorecard models with Risk Management Toolbox.

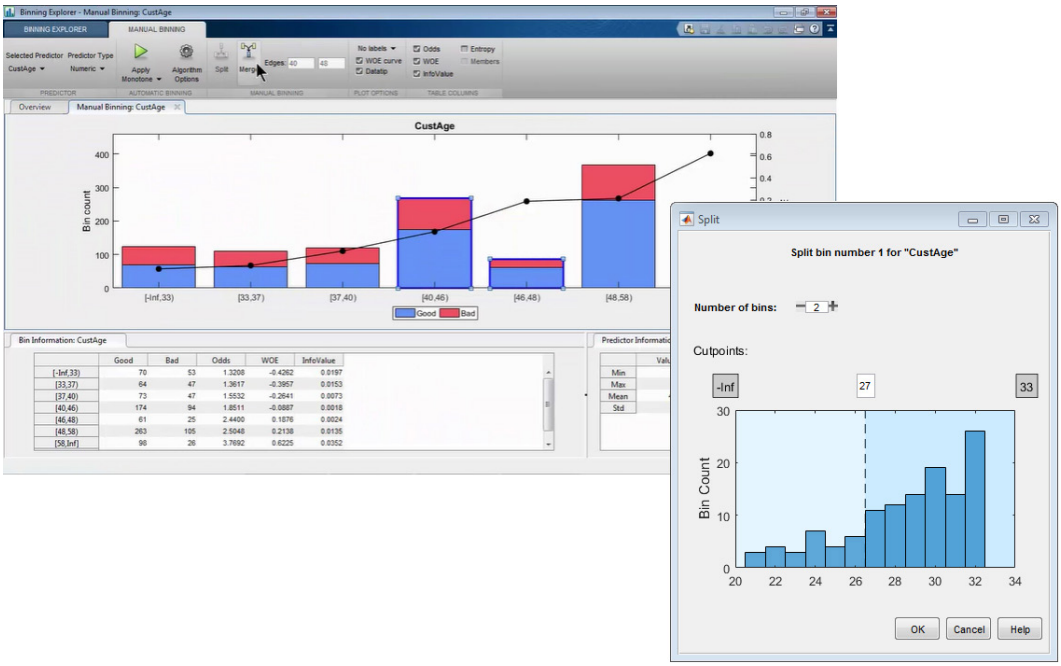
3. MATLAB Toolboxes Just Work

MATLAB toolboxes offer professionally developed, rigorously tested, field-hardened, and fully documented functionality for a range of financial and risk management applications. Risk management teams rely on toolboxes including:

- *Financial Toolbox™*
- *Econometrics Toolbox™*
- *Datafeed Toolbox™*
- *Database Toolbox™*
- *Spreadsheet Link™ (for Microsoft Excel®)*
- *Financial Instruments Toolbox™*
- *Trading Toolbox™*
- *Risk Management Toolbox™*

4. MATLAB Has Apps

MATLAB apps are interactive applications that combine direct access to large collections of algorithms with immediate visual feedback. Prebuilt functionality gives you a host of algorithms for common financial engineering challenges, machine learning, and data analytics.

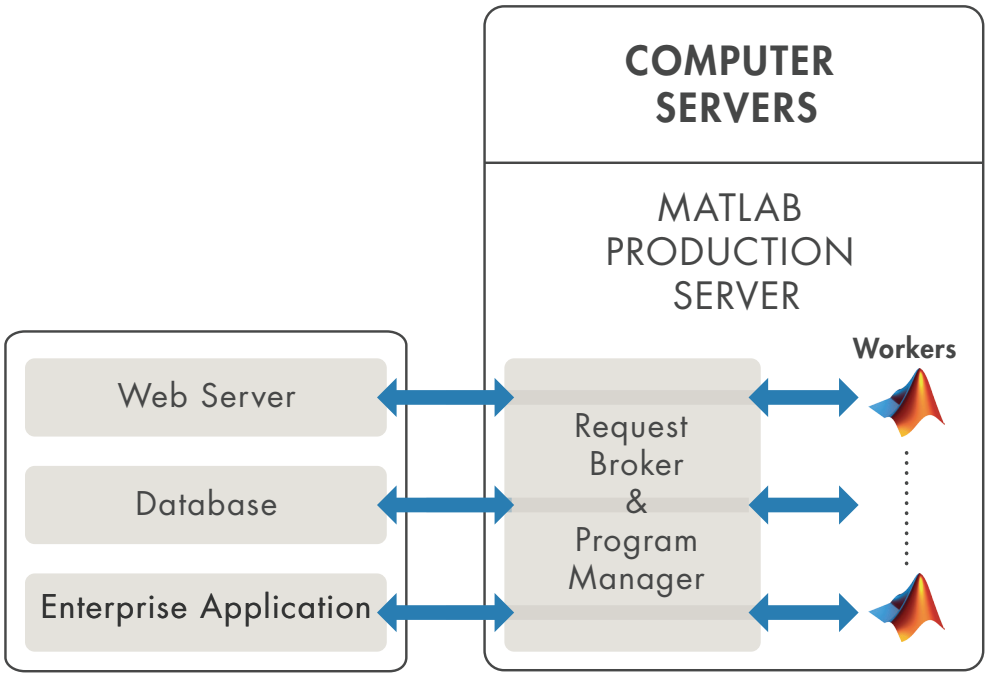


Interactive application in Risk Management Toolbox.

5. MATLAB Integrates Workflows

MATLAB helps automate the entire path from research to production, including deployment to production applications, cloud, desktop, Excel, and in some cases hardware devices.

With MATLAB, risk professionals and integrators can share a risk analytic or model from a single risk analytics stack through extensible interfaces that offer connections to reporting applications, batch processing environments, business intelligence tools, and risk and reconciliation systems. It can do this through connections to C++, Java®, .NET, Python®, SQL, and web services.

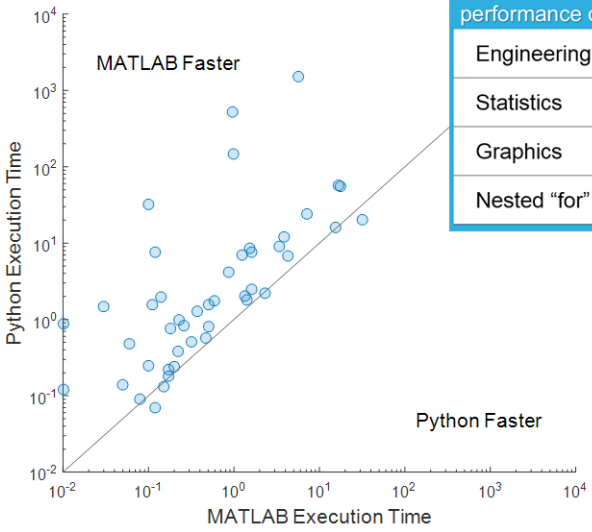


Connecting MATLAB applications to existing systems.

6. MATLAB Is Fast

MATLAB does the hard work of making your code fast. Math operations are distributed across your computer's cores, library calls are heavily optimized, and all code is just-in-time compiled.

It is also straightforward to speed up applications with minimal code changes, using clusters, GPUs, and cloud.



MATLAB performance over Python	Average	Best
Engineering	3.2x	64x
Statistics	2.7x	52x
Graphics	31x	540x
Nested "for" loops	64x	64x

MATLAB performance metrics.

7. MATLAB Is Trusted

Engineers and scientists trust MATLAB to send a spacecraft to Pluto, to match transplant patients with organ donors, and to develop financial risk models for bank examiners. This trust is built on impeccable numerics stemming from the strong roots of MATLAB in the numerical analysis research community. A team of MathWorks engineers continuously verifies quality by running millions of tests on the MATLAB code base every day.

MATLAB is also trusted because it offers capabilities for professional software development, important to maintain scalability and transparency and to make systems future-proof. This is useful when building any robust framework, risk, finance, or otherwise. MATLAB includes object-oriented programming, unit testing, source control system integration, and refactoring capabilities among other capabilities found in programming languages such as C++ or Java.

> *"Object-oriented programming in MATLAB enabled us to write less error-prone code, define reusable interfaces, and make rapid updates. As a result, we can give our investors better insight into how we manage our funds and how we look at markets."*
>
> *—Ariel Fischer, Trient Asset Management*

Developing and Implementing Your Risk Management Systems with MATLAB

Let's examine these platform strengths directly in risk management and risk model governance.

Regulators and CROs rightly emphasize the importance of building risk services on trusted platforms that fit into existing technical architectures. The U.S. Federal Reserve issued ***SR 11-7 Guidance on Model Risk Management*** in 2011, warning of the dangers of faulty model development processes and stressing the importance of model validation and model risk assessment.

If we consider the relevance of MATLAB to the earlier stress testing example, it illustrates key themes that directly support this federal guidance document:

Business Challenges	**MATLAB Philosophy**
Many Regulators	Single stack: easily repurposed
Short Seasonal Cycles	Rapid development with formal methods
Multiple Teams	Common and bespoke interfaces to stack
Transparency	Readable, documented code
Reproducibility	Encapsulation and objects, documented code and scripts
Supervisor Requests	Clear process, trusted numerics by industry
Technical Challenges	**MATLAB Capabilities**
Data Aggregation	Database, tables, reading from multiple formats and feeds
Scenarios and Models	Proven prebuilt tools, build-your-own; objects
Model Management	Object oriented programming
Validation and Verification	Debugging, unit testing, code review, and many opportunities for model review, comparison, and validation
Aggregation	Statistical aggregation, database
Reporting and Access	Report generator, interactive analysis in spreadsheets, apps, web, databases
Excel	**Managed Excel Add-ins**

MATLAB is trusted in heavily regulated and risk-aware industries from aerospace and automotive to semiconductors and medical devices for building robust, scalable, and secure products and services, many "safety-critical." Likewise, risk management, through the vehicle of model governance, is a key strength of MATLAB, reducing risk but also increasing business opportunities. Similar to how automotive manufacturers improved quality while increasing feature development, the risk management industry is increasingly having an impact on business opportunities, through better, faster pretrade risk assessment and price optimization, for example.

Conclusion

With increasing scrutiny and regulation facing financial organizations, risk professionals are moving toward greater accountability, with formal risk assessment and management processes. They seek to reduce model development times and develop and implement applications that are transparent, consistent, and scalable.

Financial organizations need a single system with reduced operational, model, and legal risks, servicing multiple disconnected supervisory regimes, while improving productivity through risk-aware development.

Risk managers increasingly rely on MATLAB to build and deploy their risk models and applications, integrate them into existing IT environments, and leverage them for building new models and applications.

Resources

- *Risk Management Toolbox*
- *Risk Management with MATLAB Overview*
- *Using MATLAB for Macroeconomic Stress Testing*
- *Munich Re Trading Creates a Risk Analytics Platform with MATLAB: Project Overview*
- *Munich Re Trading Creates a Risk Analytics Platform with MATLAB: Demonstration*

93075v00 02/17