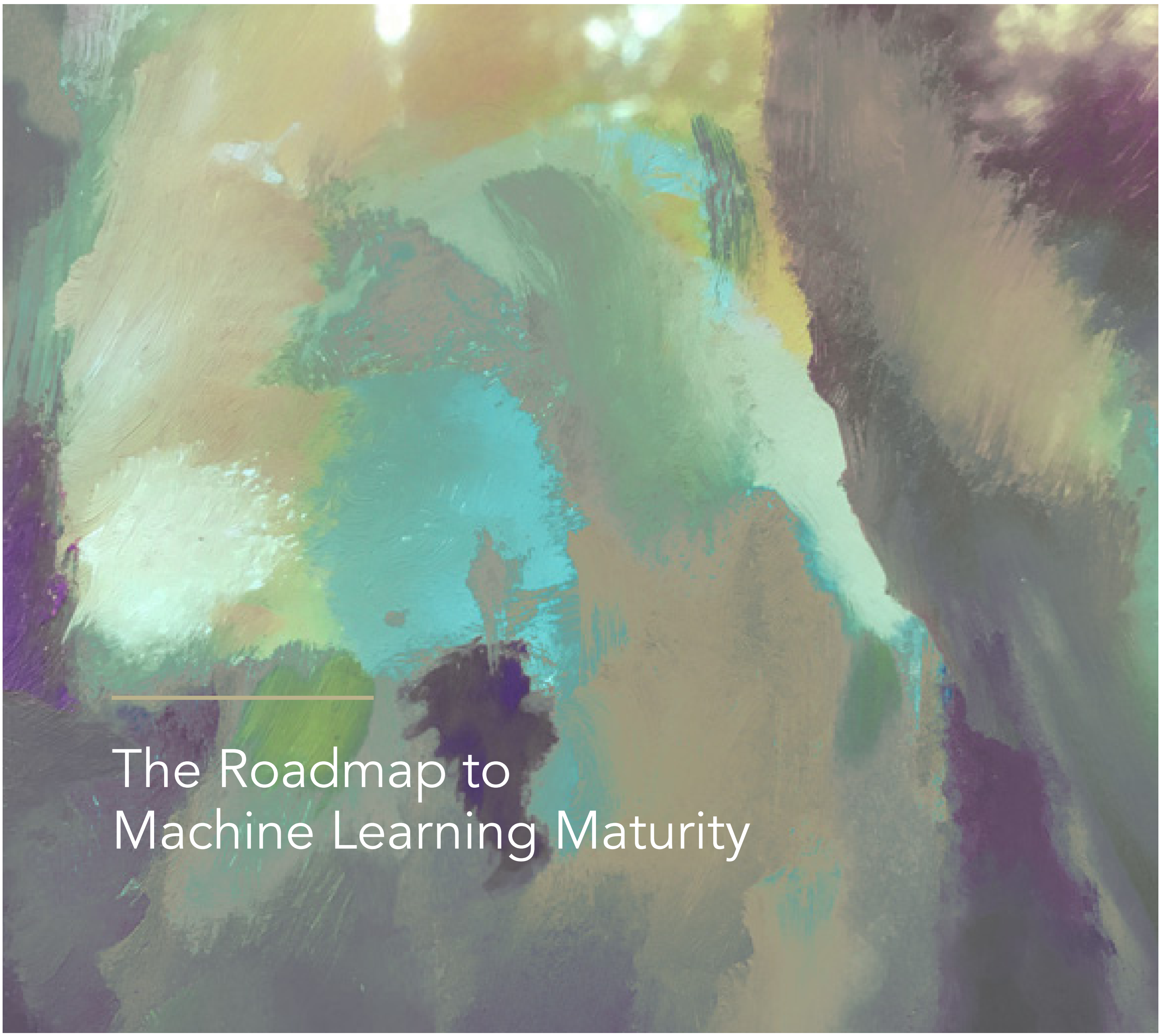
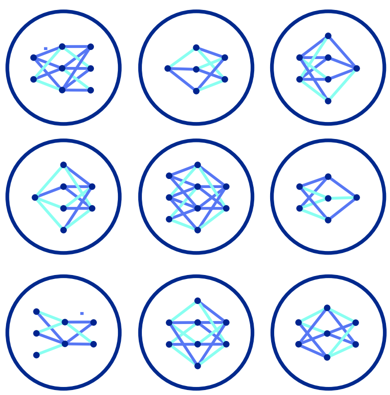
First Quarter 2019 | Algorithmia Research



The Roadmap to Machine Learning Maturity

Working with hundreds of companies performing machine learning (ML), Algorithmia has identified four distinct but interdependent stages of development necessary for any ML project. Divided into three levels of maturity, this framework can help you understand your level of ML sophistication, identify common pitfalls, and help you prioritize a path to success.



This report is the product of Algorithmia Research as part of our efforts to help everyone deploy and manage machine learning models at scale.

More information can be found [here](#).

Table of Contents

Introduction

Machine learning will be the most impactful technological innovation of our time. It is already enabling cross-cultural communication, protecting natural resources, and discovering drugs and treatments that will revolutionize medicine. But how can your company get there and what do you need to do on the way?

To most organizations, introducing machine learning sounds like a quest to discover El Dorado. But ML pioneers are already reaping rewards, and their best practices can help map your journey.

We have had countless conversations with organizations of all sizes at various stages of their ML journeys, and we have developed a simple mapping system that you can use to:

- Locate your organization's current position
- Pinpoint your destination

- Chart priorities on the path to maturity
- Orient and align your stakeholders
- Navigate common pitfalls and roadblocks

Reader beware: the path to machine learning is long and in constant flux; embrace the possibility of changing course to get to your goal.

The Roadmap to Machine Learning Maturity

Machine learning is not a one-size-fits-all process and there is no standard operations manual, so companies have to chart their own paths.

In hundreds of interviews with organizations implementing machine learning, we found that regardless of the company's size or the scope of their projects, most businesses fall into a data strategy without first developing a specific plan for integrating ML. In short, they are figuring it out as they go.

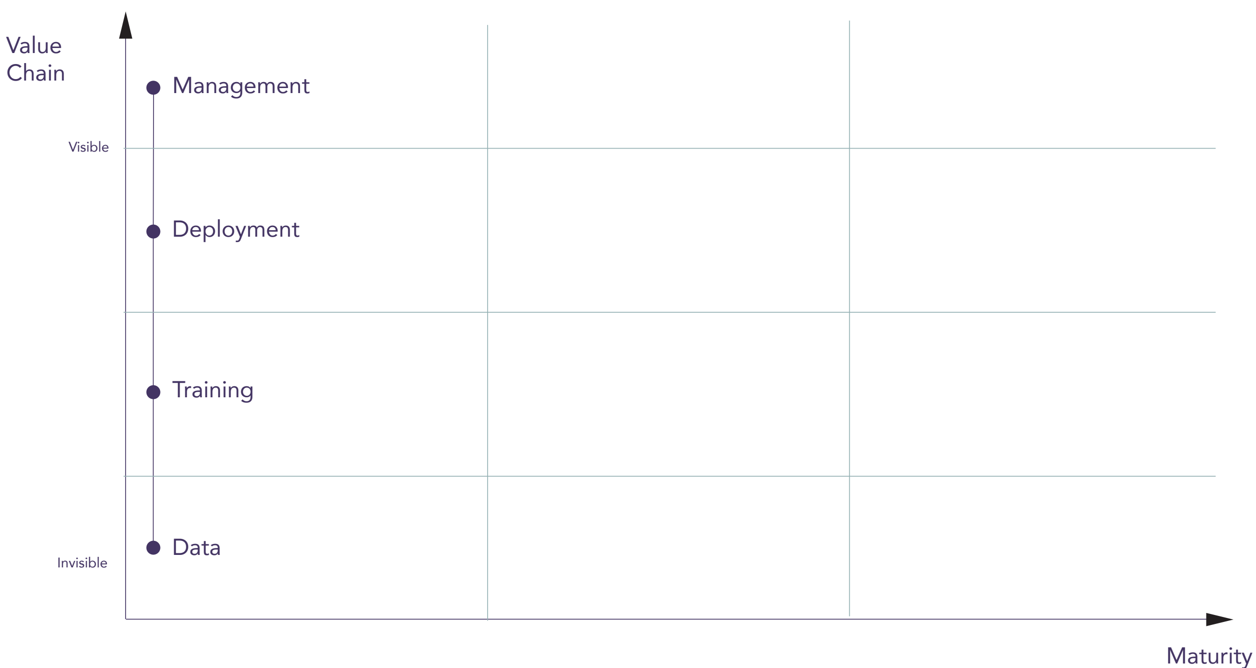
We have developed a roadmap that outlines the four stages to productionize machine learning. By finding your current location on the roadmap, you can chart a course, identify pitfalls, and plan for investment, hiring, and resource acquisition to enable your company to move to maturity.

Full disclosure: Algorithmia has built a machine learning operating system that allows you to deploy and manage machine learning models at any scale. It is in our best interest to help companies be successful through the entire ML journey because we believe that we can make the last stages much easier.

Roadmap Overview

You cannot fit every aspect of building an ML program into a 4x3 table, but we have found that these 12 fields provide a solid planning framework for planning for nearly any business.

Here is the map:



The Maturity Axis

On the x axis is Maturity, which measures the sophistication of a machine learning initiative across three variables:

> **People**
>
> Who are the likely stakeholders? Who is (or will be) working on your ML projects and in what capacity? Do you have executive buy-in?
>
> **Tools**
>
> How sophisticated are your tools? Are they holding you back? Are you locked in by proprietary technology? Are there other tools you need?
>
> **Operations**
>
> Are your operations ready to scale and pivot if need be? Will costs spiral as processes scale?

The Four Stages of Machine Learning

On the y axis are the discrete stages every machine learning project requires. The stages near the bottom are more foundational, enabling the later stages. As you move up the axis, each stage brings you closer to delivering and scaling impact, with the payoff appearing between Deployment and Management.

1. Data

It's the first word and primary focus of data science. To begin any machine learning project, you need a repository of comprehensive, up-to-date, and relevant information. That means cleaning and structuring data, building ETL processes to move data from place to place, securing the system, and committing to long-term data maintenance.

2. Training

Model training is the heart of machine learning; it’s where you discover patterns and connections in your data.

3. Deployment

Deploying models into production gets your machine learning off laptops and makes it available to applications. Whether they are generating reports, feeding applications, or working together with other algorithms, your models start to generate value in the Deployment stage.

4. Management

Machine learning does not happen just once or in a vacuum. Every aspect of the system will continuously change and grow, and you need to maintain and tune it constantly to ensure peak performance at optimal cost.

Locate Your Organization on the Roadmap

To locate your company on the roadmap, evaluate your People, Tools, and Operations for each stage. For each of the stages, we will give examples of how low, medium, and high levels of ML maturity might look.

Data

Data is often referred to as the "new oil" because of its value in enterprise. Because it powers machine learning and everything that enables, you'll need to make sure your company has data that is accurate, easily accessible, and ready for data scientists to start working with it.

Low Maturity

"I have that information…somewhere." Sound familiar? At this level, data assets are often scattered throughout silos such as email, spreadsheets, and purpose-built databases that make achieving a holistic view of the data impossible. In many cases, companies do not know what data they own, so deriving insights from large-scale data operations is daunting.

Low-Maturity Data Characteristics

People	Tools	Operations
• Individual data owners • Department-level administrators • Considering hiring a data architect	• Personal and shared file storage • SaaS-based applications	• Limited IT oversight • No defined ETL processes • Minimal data cleanliness standards

Medium Maturity

More mature organizations have recognized the problem and tried to move all their corporate data into modern, searchable data stores. Structured data makes cross-repository projects possible, but there are still inconsistencies among repositories, and there is no top-level view of all the assets. At this level, the number of silos is greatly reduced but their managers may not yet coordinate effectively.

Medium-Maturity Data Characteristics

People	Tools	Operations
• Department-level administrators • Database administrators (DBAs) • IT	• Corporate shared file storage • SaaS-based applications • Corporate database	• Emerging data governance • ETL • Strong IT policy management

High Maturity

The most mature organizations view data as their most critical asset and a core priority from the executive level down. They have assembled or are building large data lakes and targeted data warehouses, and they have a backlog of analytics and ML projects they plan to run.

High-Maturity Data Characteristics

People	Tools	Operations
• Chief Data Officer • Data architects • DBAs • IT	• Data warehouses • Data lakes	• Data-first strategy • Strict, comprehensive data governance • Measurable KPIs for their data teams

Of all the stages in the roadmap, Data is generally the most mature. Many organizations have taken on big data projects in recent years, creating structure and governance in the process. At the same time, nimble startups have built their businesses around data from the beginning. With that said, if your business is behind the curve, organizing data for future projects is a fantastic first ML initiative.

> *"Only when data is accessible, you can explore and transform it. This includes the infamous data cleaning, an under-rated side of data science."*
> Monica Rogati

Training

Until we apply trained models, data just adds to the storage bill. The Training stage is where businesses define and build toward outcomes. Those outcomes and the tools that support them change as companies mature.

Low Maturity

Data scientists in early-stage machine learning projects are often fresh graduates or have moved internally from adjacent professions. In many cases, they were analysts working on trend reporting, and they frequently bring those skills and goals with them to ML. End-to-end workbenches are common first tool choices, and the outputs are often amped-up versions of old analytics destinations, like business intelligence dashboards or email reports.

Low-Maturity Training Characteristics

People	Tools	Operations
• New data scientists (recent graduates, retrained data analysts)	• Data science workbenches (Dataiku, DataRobot, Domino Data Science Platform, H20, SageMaker, etc.) • BI dashboards (PowerBI, Tableau) • Email reports • Office applications	• Ad-hoc projects • Advanced trend reporting • Predictive analytics

Medium Maturity

As organizations become more sophisticated, they focus on scaling insights, looking for additional validation and nuance. At this point, companies with medium maturity have hired experienced data scientists or retrained their internal staff, and diversified business goals have forced them to seek out more flexible frameworks. A common and unfortunate side effect of increased diversity is a lack of standardization, which can derail training efforts and reduce operational efficiency.

Medium-Maturity Training Characteristics

People	Tools	Operations
• Data scientists	• Data science workbenches (Dataiku, Domino Data Science Platform, H20, SageMaker) • Multiple deep learning frameworks (TensorFlow, PyTorch, CNTK)	• Increasing strategic importance of ML • Accelerating existing initiatives • Multiple concurrent projects • Emphasis on increasing scale

High Maturity

The organizations with the highest ML maturity are seeking to glean new insights from their data and are modeling incredibly diverse landscapes of theoretical possibilities. Very few companies have reached this point. Of those that have, most tend to be in financial services, pharmaceuticals, healthcare, and technology. Toolsets at this stage tend to vary widely by use case, but deep learning frameworks are the most popular.

High-Maturity Training Characteristics

People	Tools	Operations
• Many experienced data scientists • Subject matter experts	• Multiple deep learning frameworks (TensorFlow, PyTorch, CNTK)	• ML as essential business component • Multiple concurrent projects • Massive scale

> *"...different data science teams are going to independently adopt tools and frameworks that are better suited for their jobs and experience."*
>
> Jesus Rodriguez

Deployment

It is surprising how little time companies spend on the output of their machine learning projects. Deployment is how models generate results, but it is generally an afterthought until it becomes too painful to ignore.

Low Maturity

In the earliest stages, data scientists run models locally and share results manually, so deployments are essentially nonexistent. That is good news for the data scientist, who can focus primarily on training models, but when demand for those models grows, sharing their output becomes a burden.

At that point, data scientists begin to throw models over the wall, leaving developers and DevOps engineers to translate those models into a format they can incorporate into applications and pipelines. Since machine learning is outside most app developers' comfort zone, it can take days or weeks to make each new model work. Because of the overhead, many developers eventually scrap the projects altogether.

Low-Maturity Deployment Characteristics

People	Tools	Operations
• Data scientists • App developers • DevOps engineers	• DevOps / workflow tools • Data workbench APIs	• Manual handoffs • One-off integrations

Medium Maturity

In the second stage of maturity, businesses begin to automate processes, scripting the actions DevOps previously performed manually. Initially, this speeds up subsequent deployments but it runs the risk of instantiating bad operational choices as code. Every less-than-optimal decision made by non-experts becomes the foundation of yet more instability.

Since different frameworks and languages introduce new twists, these workflows are fragile and will ultimately break down as ML projects begin to scale—just when you need them most. Businesses in this stage frequently find that creating and maintaining a growing collection of scripts is not worth the time, and many eventually retreat to manual processes.

Medium-Maturity Deployment Characteristics

People	Tools	Operations
• Data scientists • Developers • DevOps engineers	• Scripting languages / frameworks • DevOps / workflow tools	• Configuration-specific automation • Script management

High Maturity

The most mature businesses leapfrog automation fragility by offloading jobs to a deployment platform. With integrations and translations handled by the platform, data scientists can deploy models from any framework into a pre-built pipeline and developers can consume an API, built on and optimized for whatever cloud-based or on-premises infrastructure the business requires. Deployment platforms also provide a single point of contact for integration with CI/CD systems and other tools. This approach takes a company out of the business of building infrastructure and allows it to focus resources on generating value.

High-Maturity Deployment Characteristics

People	Tools	Operations
• Data scientists • Developers • DevOps engineers	• Deployment automation platform • Role-specific tools (IDEs, deep learning frameworks, orchestration)	• One-button deployment • Automatic model translation • CI/CD Pipeline integration • Infrastructure-agnostic tooling

> *"Big companies should avoid building their own machine learning infrastructure. Almost every tech company I talk to is building their own custom machine learning stack and has a team that's way too excited about doing this."*
>
> Lukas Biewald

Management

Model deployment is the launch event, but not the end of the journey. Machine learning management is an ongoing process of review, refinement, and redeployment. Without proper management, ML programs and the models that support them can break, become irrelevant, or—if successful—spike operational costs to the point where they become revenue-negative.

Low Maturity

Most companies beginning the machine learning journey completely ignore model management. Individual data scientists often run their own projects on an ad-hoc basis, so model oversight and governance is not a priority.

Additionally, many of the workbench solutions popular at this stage tend to focus heavily on model creation and training, so sophisticated management tools simply are not available.

Low-Maturity Management Characteristics

People	Tools	Operations
• Data scientists	• End-to-end workbenches (Dataiku, DataRobot, Domino Data Science Platform, H20, SageMaker)	• None

Medium Maturity

As efforts scale, data scientists begin to integrate additional frameworks and broader infrastructure, with more stakeholders and greater exposure to governance. For example, a model written in R might call a model written in Scala, with each running on different cloud-based infrastructure. The output is then consumed by a variety of different applications, each with its own audit and chargeback processes. At the same time, these businesses need to constantly reevaluate and update their portfolios, test and integrate new algorithms, maintain or sunset support for deprecated model versions, and tune their infrastructure for the most cost-effective performance.

As businesses mature into this sort of scale, they usually rely on their infrastructure provider's built-in toolsets. These offer rudimentary levels of control, such as assigning a model to a GPU or CPU, or setting throttles on performance and cost, but they lack sophisticated versioning and lock users into a one-vendor system.

Medium-Maturity Management Characteristics

People	Tools	Operations
• Data scientists • Developers • DevOps engineers	• Platform-specific infrastructure management • Algorithm marketplaces	• Ad-hoc management

High Maturity

As with deployment, businesses managing the most mature machine learning programs rely on a platform for flexibility, visibility, and control. Like an operating system, this platform provides a consistent interface across a wide range of infrastructure and applications, giving data scientists, operations specialists, and app developers a single console to integrate, manage, and optimize their own processes without running into barriers. With it, data scientists can test and replace models from multiple frameworks, operations can tune and move services among infrastructure providers, and developers can browse and consume solutions without worrying about language or framework compatibility.

High-Maturity Management Characteristics

People	Tools	Operations
• Data scientists • Developers • DevOps engineers	• Deployment automation platform • Role-specific tools	• Ongoing, iterative review and adjustment • Regression testing

> *"Before you know it, there are 10 different versions of your original model all of which need to be trained, deployed and monitored."*
> *Jesus Rodriguez*

The wealthiest, most forward-thinking companies in the world are all in on machine learning. They have already identified the stages in the roadmap and injected tremendous resources into each. They are able to mature along each stage at full speed. For those companies, the benefits of a leadership position in ML are worth the cost and risk of moving first.

The rest of us have finite budget and staff, and need to be a bit more strategic with our investments. But strategy is in short supply under competitive pressure. Companies know data is critical, so they invest furiously in data infrastructure. Next, they spend a huge amount of time and money to hire data scientists and ML engineers, dreaming of profitable new insights from models they will build and deploy. Since they lacked clear goals at the start, they were not able to verify that their data was, in fact, what they needed in order to train their models. As a result, the newly hired data scientists spend their time learning to do data engineering.

Despite much time, effort, and money, those companies have still not started training models—or built a system to get those models into production where they can actually generate ROI. And if those programs start to stall, companies risk losing their talent to competitors with more structured ML project goals.

This is where the roadmap comes in. If you know where you are trying to go and know what is likely coming next, you can plan to invest appropriately across the entire ML lifecycle.

Pinpoint Your Destination

So what do you target first? As you increase maturity across each stage of the roadmap, you will see a different payoff.

Shoring up gaps in one stage can accelerate the benefits of another. If you have models generating a lot of good insights that are going unnoticed, automated deployment could encourage developers to increase consumption. If your data is strong and structured, investing in data scientists and model training could pay off with an impressive proof of concept.

While working toward the end goal of maturity across the board, you'll notice outputs that suggest exciting opportunities or eliminate painful blockers. If something on the roadmap jumps out, it is probably a good near-term goal.

A word of caution: Before you begin any work, ensure that your projects will see the light of day. That means establishing baseline capability across all four stages, from data collection to model management. If any stages of the roadmap are at zero maturity, start there.

You certainly do not need to delay advanced model training or data warehousing until you have a sophisticated deployment system, but you do need something in all four stages to begin, and getting there should take priority.

What is your destination?

Value Chain

Visible

Management	Low performance, unpredictable costs	Model versioning, security, and labor-intensive pipelining; coarse cost control and performance management; vendor lock-in	Optimized performance and cost management across infrastructures; model evaluation and comparison, governance, predictive scaling
Deployment	Share results manually	Integrate at high effort with specific endpoints	Scalable deployment infrastructure to multiple endpoints, optimized compute
Training	"Analytics plus" — one step above Excel	Advanced models with high effort and non-optimized compute	Advanced, pipelined models continuously improved
Data	High-effort, ad-hoc queries	Project-specific data cleaned and tagged	Organization-wide data ready to use at scale

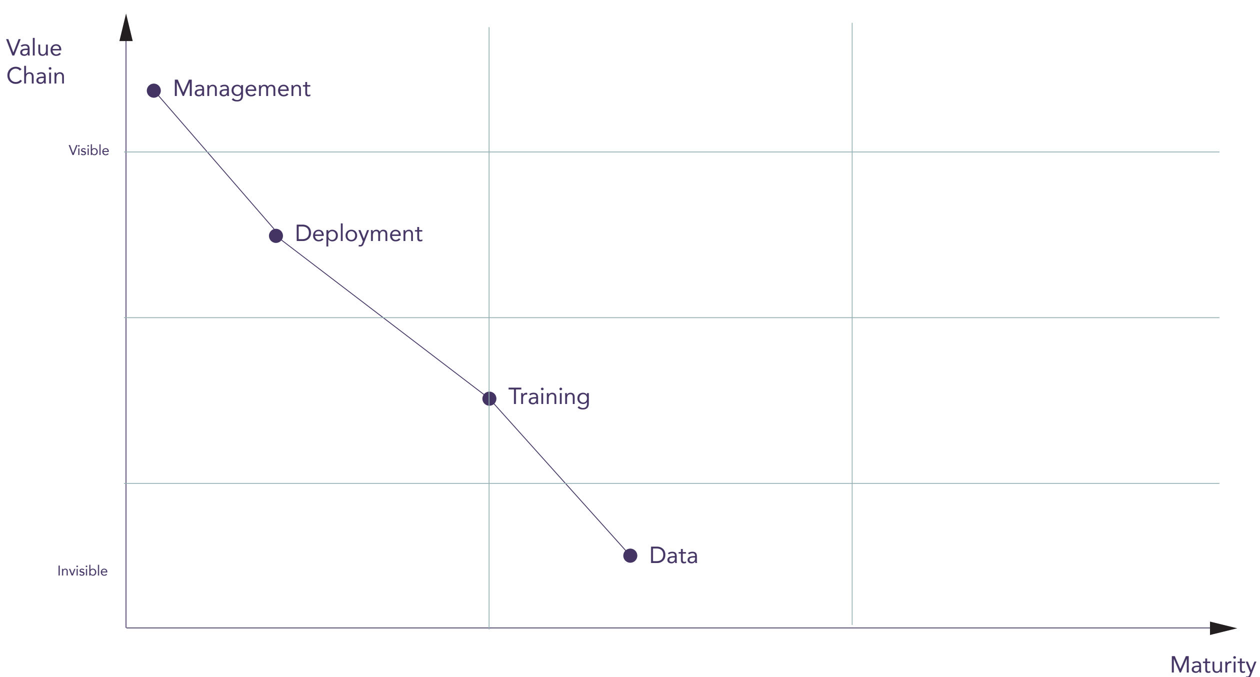
Invisible

Maturity

Case Study

We interviewed a medium-sized enterprise that has acquired several companies and their data over the past 15 years.
In late 2015, the company began building a data warehouse to combine normalized customer and transaction data. The first heavy-lifting phase of the project was completed in early 2017, with revisions ongoing.

The Case Study Company's Current Maturity Level



At roughly the same time, data scientists in different business units began creating experimental projects. Initial projects used TensorFlow, but subsequent projects incorporated tools the scientists had used elsewhere—DataRobot, PyTorch, CNTK, and possibly others.

Many of these projects were ultimately scrapped, but the customer service and outside sales departments integrated several models into their first-party applications, which are currently running entirely on Amazon Web Services (AWS). Data scientists handed off each model independently and application developers translated each model manually, at substantial cost.

The company's CTO identified three key problems:

Wasted productivity: Developers were essentially rewriting models every time they were revised, with data scientists by their side answering questions. This delayed delivery schedules for new application features and slowed the pace of model updates. Sales requested that Development focus on "need-to-haves," such as performance improvements and connectors to other applications, rather than "nice-to-haves," such as ML integration.

Versioning: Different developers were calling different versions of models; knowing which models to support or call was becoming an issue.

Cost management: Models were burning expensive GPUs and running unchecked, spiking the company's AWS bill, with no clear ROI for their efforts or internal accounting to apportion costs.

Despite the overhead, the company's ML programs already seemed to be increasing conversion retention rates and shortening the sales cycle, but the CTO had no budget to add headcount.

The organization is currently running a proof of concept with Algorithmia to automate its deployment infrastructure and eliminate the need for model translation. As a secondary goal, the company hopes to gain a better understanding of its costs, target resources more effectively, and make business units accountable for their own compute costs, pushing the decision about performance goals to the business owners, rather than having IT act as a cost center for their entire company.

By using Algorithmia's AI Layer to leapfrog from low to high maturity in the Deployment and Management stages, the organization will be able to spend resources hiring and training new data scientists to increase their capabilities in the Training stage.

The Case Study Company’s ML Maturity Roadmap

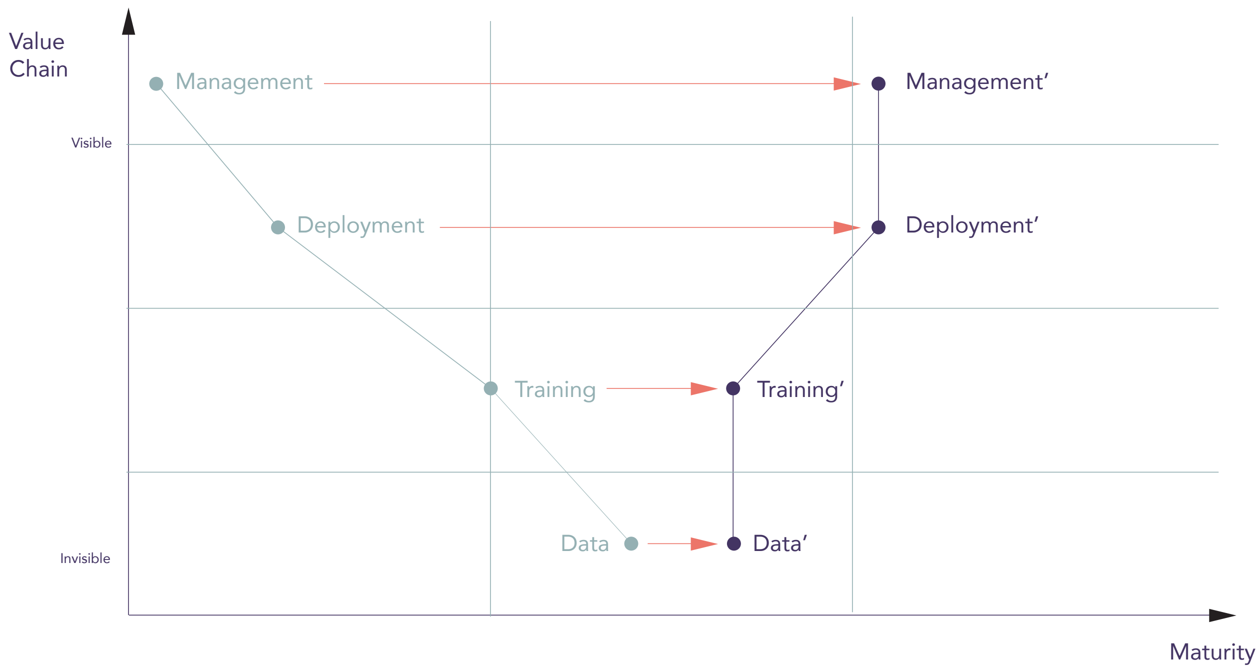


Chart Your Path to Maturity

Now that you have found where your company sits on the roadmap and picked your targets, we can chart a path to get there.

A note about DIY solutions: There is no shortage of content from engineers at the biggest tech companies explaining how they built their own machine learning infrastructure. It is worthwhile information, but consider that they likely have far more resources, more engineers, and fewer legacy tools than you will. Even if you take the same path, it is unlikely that you will reach their same level of maturity, and you will need to maintain any apps and integrations you build to sync with the rest of the industry ecosystem over time.

Front-Load Quick Wins

All machine learning projects are a process, and the more complicated the project, the longer it will take to see results. To build political capital and speed your team's learning, focus first on discrete tasks that can show results in a matter of weeks. Look for opportunities that leverage existing staff and tools, and that solve a current, distinct problem. After a few completed projects, you will have a better idea of the tools and skills you will need and the challenges you will face, allowing you to scope a larger campaign with more confidence.

Ensure You Can Deploy

Models running on laptops are great for enhancing presentations and reports, but they cannot realize the true value of machine learning. To see a substantial payoff, you need to deploy your models into production at scale.

There are two ways to do this. You can either:

> Have a DevOps team build and manage in-house infrastructure. This is what the biggest tech companies have done. It will take years of effort from a large team of top-level engineers and require constant, dedicated maintenance.

or:

> Deploy your suite of ML models in minutes with Algorithmia's AI Layer. We may biased here, but we highly encourage you to test a model on the AI Layer before you attempt to build your own deployment system. We will even make engineers available to help.

Skip the Line

You are not the only business on this journey. Take advantage of third-party services that help you leapfrog directly into maturity, whether that means consuming trained models from third-party marketplaces or adopting a deployment and management platform that removes your integration pains. These are easy wins that free your data scientists to build your proprietary tools while scaling the value of your existing ML program right away.

Choose the Right Tools for Success

Any project or process that takes off will have to scale dramatically, and retooling to make that happen may not be an option. Examine where you plan to be in a best-case scenario 18-24 months out, and focus resources on automation that can support that theoretical load. Since technology changes quickly, the best path to success may be a third-party solution, whether that is a hosted data provider, a set of external models, or a deployment and management platform.

Increase Flexibility

If your infrastructure depends on any one product or person, that is a problem. Look for single points of failure and build the capacity to grow in different directions as circumstances dictate.

Orient and Align Goals and Stakeholders

Machine learning efforts are going to require buy-in from many different parties. DevOps, IT, product teams, data scientists, ML engineers, and leadership will all need to be involved. The following are best practices for posturing a company toward a ML initiative.

Align with Business Goals

Are you trying to win a Nobel Prize for data science? (If so, we're hiring!) If not, do not build something because it sounds cool, or just to test your capabilities. Everything you build, including a proof of concept, should align with shared business goals and answer a question your company is already asking. You may only get one chance to prove ML is worth it.

Accelerate Existing Programs

If possible, tie your project to an existing initiative that has hit some snags. You will already have a defined team, and clearing blockers will garner support for your processes. Great candidates for this are internally focused classification and integration projects that sap IT productivity from other activities.

Identify the Team

To move fast, you will need to define your entire team up front. This will probably include DevOps, IT, product management, data scientists, and ML engineers, and every team should have at least one executive sponsor who can champion the program to leadership.

Consider Stakeholder Needs and Hesitations

Define the impact of your program, and think of ways machine learning can enhance and extend the roles of everyone at the company. You will have much better luck getting buy-in, and you will build a better product.

Seek Executive Buy-in

Even a small project will require significant input from all team members at first. Be sure to make management understand that commitment. If possible, tie the project to an internal hackathon or innovation effort to offset the impact.

Navigate Common Pitfalls

While scoping your project portfolio, keep these nine rules in mind to avoid momentum killers.

1. Focus on outcomes, not on the process

Processes and tools are necessary details, but not the end goal. Do not focus too much on the path at the expense of defining the end results. Business value should drive every aspect of your ML efforts.

2. Don't try to be perfect

Promising—or even pursuing—academic perfection is a sure way to stall your plans. In some cases, even 60 percent accuracy might produce the desired ROI.

3. Consider the environment

ML does not happen in a vacuum. Whatever systems you introduce need to integrate with your existing enterprise. That means all of it, since successful ML programs will ultimately branch out to every node of your organization.

4. Don't reinvent the wheel

Data scientists are rare, expensive, and a limiting factor in the scope of your machine learning efforts. Luckily, you do not need to build everything yourself. Most of what you need has probably already been built. You can use models and data sets from third parties so you can focus on building only the pieces that are truly critical to your business and unavailable elsewhere.

It is also worth mentioning that there are some services and data sets that never make sense to create in-house. For speech and face recognition or other models requiring massive amounts of sophistication and data, an available third-party service is almost always the better choice.

5. Avoid waterfall payoffs

It is easy to get excited by long-term potential and make promises that never materialize. Keep your ML projects agile and iterative, with incremental goals you know you can hit. Short sprints and achievable goals will help you:

- Limit scope creep as stakeholders attach more goals to your project
- Avoid analysis paralysis from too many interested parties
- Adapt to shifts in business goals or underlying technology
- Identify and recover from bad assumptions and failed experiments

6. Benchmark against the status quo

Do not build solutions for problems yet to arise. Deprioritize fixes for processes that are already good enough. You do not need to fundamentally transform your business every time you build a model but you should have room for long-term efficiency or accuracy gains. If the target is too close, you should probably aim elsewhere.

7. Say no to lock-in

Every stage of maturity offers an opportunity to trade comfort for flexibility. Avoid the trap. Do not be slavishly attached to any single database, framework, language, or infrastructure provider. They are all just tools on your journey, and you should always select the best tools for any job. We have created Algorithmia on the assumption that training frameworks, data sources, and compute best practices are going to change multiple times with every project, so we built the AI Layer to sit between models, hardware, and applications to allow maximum flexibility.

8. Don't confuse tools with solutions

Following on the previous point, tools are often a starting point or introduction to machine learning, particularly some of the popular end-to-end workbenches. Many of these tools offer integrated workflows that can get you started quickly.

In the short term, you should absolutely use familiar and available tools to produce results, but once you have committed to a machine learning journey, look toward your long-term goals. Constraining your aspirations to the capability of the tools on hand will rob you of your creativity and hamstring a project before it begins.

9. Audit honestly, revise constantly

Audit the performance of your models in a production environment, and do the same for your production processes. At every sprint, take a moment to reconsider your strategy, infrastructure, and ROI projections, and keep stakeholders informed.

About Algorithmia

Algorithmia helps organizations extend their human potential with its AI Layer, a machine learning operating system.

The AI Layer empowers organizations to:

- **Deploy models** from a variety of frameworks, languages, platforms, and tools.
- **Connect popular data sources**, orchestration engines, and step functions.
- **Scale model inference** on multiple infrastructure providers.
- **Manage the machine learning lifecycle** with tools to iterate, audit, secure, and govern.

To learn more about how Algorithmia can help your company accelerate its ML journey, visit our website.