A HUMAN'S GUIDE TO MACHINE LEARNING

INTRODUCTION

Many people start advanced analytics programs at their company and search for a methodology that will allow them to tackle their use cases as quickly and effectively as possible. In this eGuide, we'll discuss an approach that will help you understand, outline, and implement artificial intelligence solutions for business problems. This guide is designed to be a reference for the first few hours of internal discussions around a machine learning project.

For the purposes of this guide, we'll assume you're working in a group with a mixed background—that is, we'll assume the domain expertise, data science, and program management knowledge varies amongst the members of the team. One party in the conversation might have knowledge of the technology used, such as machine learning, databases, and business intelligence tools. Another party might have deep knowledge of the domain, but little mathematical or computer science background.

A Failure of Communication

In our 2023 Frictionless AI Global Survey Report, responses from 2,037 global professionals found that AI and data science project challenges are real and commonplace.

- 42% of organizations experienced an artificial intelligence (AI) project failure in the past two years
- 84% of organizations face limitations that slow down AI initiatives

Some of the key obstacles preventing successful deployment of AI in organizations are: lack of trust in models, siloed data, outdated legacy infrastructure, and lack of skills specific to AI within the organization.

Ensuring proper communication from the outset is key to building trust in models created, and is therefore a key component of a successful machine learning project.

Table of Contents

With this guide, you'll be able to create a clear plan that outlines the steps you need to take to build and implement a model prototype that can have a real impact on your business.

05 / Step 0: Responding to Common Objections
06 / Step I: Understanding the CRISP-DM
07 / Step II: Understanding the Business Case
09 / Step III: Defining the Label
13 / Step IV: Defining Performance and "Success"
17 / Step V: Discussing Profile Generation
20 / Wrapping Up

STEP 0 – RESPONDING TO COMMON OBJECTIONS

When embarking on a new machine learning project, objections often crop up from various parties. Many of these objections are unique to a given organization or work environment, but there are two common objections that are worth addressing here. Let's dive in.

"But what's in it for me?"
One of the most common roadblocks when it comes to developing and implementing a machine learning solution, especially from leadership, is the question: "What's in it for me?" People are often averse to change—this is especially true in business, where mistakes can potentially cost hundreds of millions of dollars. If a current solution is working, why risk trying something new?

This is a great example of why it's so critical to both outline the business problem and define what success looks like (which we'll talk about in detail in Step IV), at the beginning of a project. If you're having these kinds of conversations about machine learning, there's a good chance that people have already started to see there's room for improvement. You want to capitalize on that by showing the kinds of impact you believe the project can have and how its success will benefit the organization. If you don't articulate these things to the right stakeholders from the outset, the project can so often get bogged down in red tape, approvals, and back-and-forths that can stall—and potentially even kill—your project.

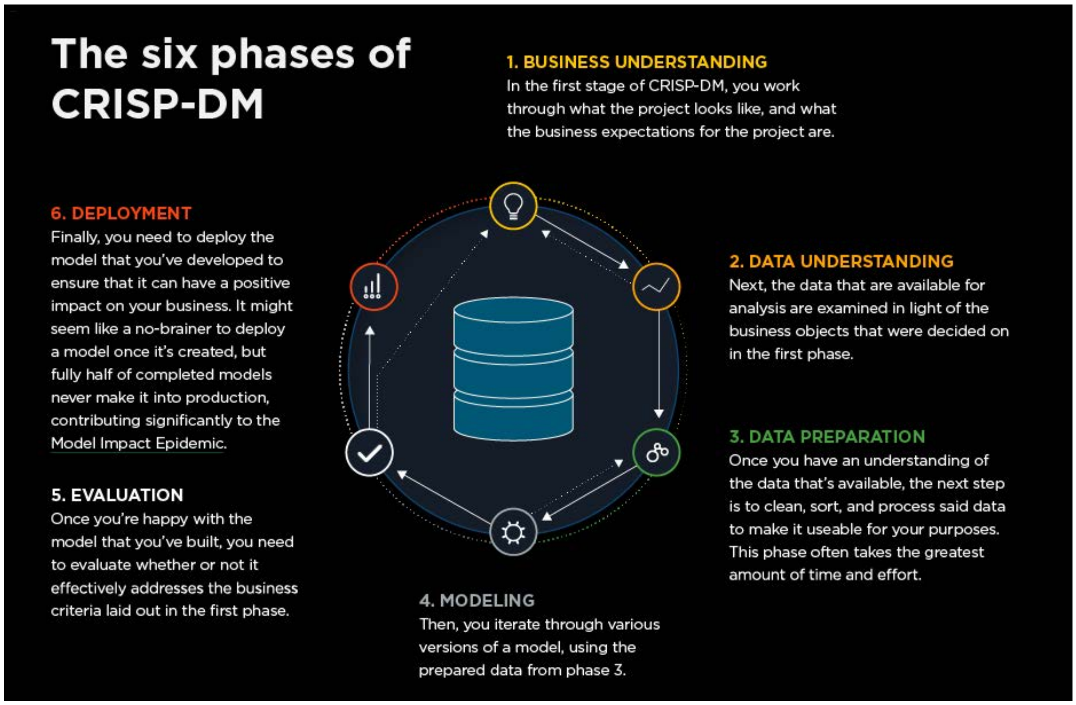
"But that's not my job."
The second common objection that crops up is resistance from those who would be doing the leg work. As stated above, one of the top obstacles preventing successful AI deployment is a lack of AI-specific skills within the organization. Data science projects often require people to work outside their comfort zones, especially during the early stages as they learn new skills and adapt to the new requirements placed on their infrastructure and job duties.

There are two steps you can take to mitigate objections from those doing the work. The first relates to the first objection: If you have clear buy-in from management about the goals you're working towards and what success looks like, it should be easier to convince teams and departments to support the project. After all, management is pushing to hit these goals.

The second step is to find a champion in the department you're working with to act as a liaison and project advocate. Their skills will probably be the most stretched, but if you can find someone who will willingly take on this task, they can provide an excellent point of contact between the different people working on the project and can serve as a project advocate among their peers.

STEP I – UNDERSTANDING CRISP-DM

First we need to talk about CRISP-DM, the cross-industry standard process for data mining. It's been around since 1996 and is a widely used analytics process. If you're getting started on a data science project, it's essential you understand CRISP-DM's basics and how they relate to the work you're doing. The CRISP-DM process consists of six phases:



Although there's a lot more involved in each of these phases, the summary above gives us enough to explore how Altair approaches these issues.

Let's look at how we approach the issues outlined in this process as we dive deep on the most critical components of implementing a successful machine learning project.
As mentioned above, the purpose of this guide is to focus on the early stages of a project,so the discussion here mostly elaborates on phases one through three of CRISP-DM, although we'll touch briefly on the other stages as well.

STEP II – UNDERSTANDING THE BUSINESS CASE

As discussed above, the first phase in CRISP-DM is developing a business understanding. In some cases, analysts don't take this step seriously enough. Data scientists often see the machine learning problem as the core problem, but the opposite is true—data scientists are tasked with solving business problems, not math problems. Every data scientist, but especially those new to business, need to understand that the reason we do analytics is to generate value for an organization. A model by itself has no value. Value is generated by putting models into context within the business processes of an organization to solve problems. To succeed, it's crucial to understand the business problem before moving on to the technology that's going to address that problem.

The business analyst often has the opposite problem. They understand the business and the problems they face, but don't understand machine learning. How can they know if machine learning can even solve the problems at hand? And if it is solvable, how can they assess the difficulty of developing and implementing a machine learning solution?

These understanding gaps can create a chicken-and-egg problem for the team, which is why it's so important to have a defined process in place to navigate the early stages of the project.

The Genesis of the Team

So how does a machine learning team come into existence in the first place? There's usually a specific trigger that creates a project team—a perceived problem or opportunity that the organization wants machine learning to address.

There's a two-fold effect that comes from this way of thinking about machine learning in business. On the one hand, if you wait for a problem or opportunity to present itself, you can miss a lot of use cases where machine learning could help. Because there was never a trigger to create a team, there's no one tasked with working on this challenge.

On the other hand, a lot of use cases aren't solvable by such a team, even if one is created, because it turns out the problem they're looking at isn't a machine learning problem.

Although this second scenario can kill the project, that isn't always a bad thing. Data scientists often think they have all the answers to solve business problems, but this isn't always the case. It's okay if a particular use case is a better fit for traditional business intelligence methods than for machine learning. You don't need to use a high-precision laser scalpel to open a box. Often, "traditional" methods like business intelligence or Six Sigma should be tried before exploring machine learning.

Your team's first session should focus on mapping the problem and use case and understanding it in enough detail to provide clarity on what the correct solution looks like. To that end, you need to come up with answers to the following questions:

- What is the problem/opportunity/challenge?
- Why is it important?
- Who in the organization cares?
- What happens if we solve the problem badly? Who suffers? How can we know if this has happened?
- How has this problem tradtionally been solved?

It's important that the data scientists in the room hold themselves back during this initial discussion. The other people will often already have an idea of how to solve the problem at hand, and in this part of the process, we don't want to do data science – we want to do business. If you're too quick to jump to a solution at this stage, you're in danger of solving the problem sub-optimally by forcing a machine learning solution where it might not be appropriate. You could also potentially miss the business problem and create something that doesn't address the challenge you're focusing on.

We're tasked with solving business problems, not math problems.

STEP III – DEFINING THE LABEL

Once all parties clearly understand the business problem and agree that machine learning is the right way to tackle it, it's then the data scientist's task to map the problem to a data science method. Ideally, you want to transform the problem to a supervised learning problem.

"Supervised" in this case means that you know ahead of time the labels you're trying to predict—for example, in a categorization problem, it would mean that you already know the categories you want your algorithm to sort the data into. With an unsupervised problem, rather than defining the categories ahead of time, you let the algorithm decide what categories are present in the data. Even if it takes a lot of effort to make the problem into a supervised one, we still recommend doing it. Unsupervised use cases are much harder to optimize, since they don't provide a qualitative measure to evaluate and tune your model with.

A good example of this difficulty is customer segmentation. It may sound intuitive to treat this as an unsupervised clustering problem. In fact, historically, most marketing departments have treated segmentation problems as grouping problems. However, when generating a clustering, you'll encounter the problem of measuring which clustering is better—that is, which categories are the right ones? The one with age as an attribute or without? The one with Euclidian distance or Manhattan distance? It's very hard—nearly impossible—to answer these questions based on a performance value. The groupings usually need to be interpreted by humans, which can lead to ambiguity.

An alternative option is to turn this into a supervised problem by treating it as a targeting problem: you want to predict whether someone will purchase an item or not. This is often also called "segmentation by one." It's easy to calculate a performance measure for this and optimize on it.

It's imperative to define a performance measure that fits your business needs. Ideally, this performance measure is something that has a direct business impact. We'll discuss this further in Step IV.

The Power of Math

While mapping the problem to a machine learning solution, data scientists should be careful with their language. Some people may not be fans of using math to explain machine learning. If you're used to speaking and writing in equations, math is a helpful language. If not, it's confusing. Think of the following quote from Stephen Hawking when using math: "Someone told me that each equation I included in my book would halve the sales. I did put in one equation, Einstein's famous equation, E=mc^2. I hope that this will not scare off half of my potential readers."

Data scientists need to be aware that they must be able to sell their methods to stakeholders. And if you want to sell your methods, confusing people isn't a good strategy. Similarly, this isn't about "fancy" methods, but about the concepts. You can explain all the concepts of machine learning by using a decision tree algorithm; there's no need to start with something more complicated like neural networks.

Always remember the "KISS" principle: Keep It Simple, Stupid.

At this point in the conversation, every participant should be aware of two things: the use case being considered, and the concept of supervised learning. So, let's move on to the real problem: What are you predicting?

Selling Your Methods

Having a plan in place in the early stages ensures you're communicating the value of your project from your first steps, and not scrambling later to get buy-in.

As resources are committed to deploy a model to production and timelines are put in place, it's critical that you're able to clearly articulate the impact your solution is having. Will it be via an interactive dashboard? A weekly report? A part of a website?

The Real Problem: Defining Our Label

Here again, you'll face an issue of different cultures. Data scientists coming from an academic environment are used to having a clearly defined label—that is, what's being predicted. After all, most university assignments and coding competitions will tell you what to predict. In business, it isn't always that simple. You need to remind yourself that defining the label is like formulating the question you want to ask the data. Once that's clear, you'll need to be sure that you satisfy three vital requirements with the label you choose.

Requirement 1: The label needs to match business needs

In Step I, you should have discussed why the project you're doing is important to the business. Any prediction problem you're working on needs to be aligned with that goal. You may have encountered proof-of-concept models where small subproblems or entirely different problems are worked on, rather than ones aligned with business value. Rather than proof of concept, try to think of the first prototype as proof of value, because the important thing at this stage is to generate evidence demonstrating the project can create business value. Before you move ahead with planning, you need to make sure your label is directly connected to the business needs.

Requirement 2: The label needs to exist

The second requirement is that the label actually exists. A "magic wand" attitude towards machine learning leads to unsatisfiable expectations. You might run into situations where a label is impossible to measure in real life. For example, imagine a factory has a vat of chemicals that rarely boils over but, when it does, it costs a lot of money. Perhaps it has happened three times in the last decade. With such little data, it would be exceptionally difficult to train an accurate model. And, even if you were to build a system to predict this event, you wouldn't be able to assess the model's quality because the event (and its corresponding data) is so rare.

A “magic wand” attitude towards machine learning leads to unsatisfiable expectations.

Note that the requirement that the label exists doesn’t exclude use cases where human judgement is the only way to measure the label. A classic example of this is sentiment analysis, where humans rate whether a comment is positive or negative as training data. This might cause difficulties in acquiring training data in the first place, but it’s not impossible.

Requirement 3: The label needs to be actionable
The best machine learning algorithm doesn’t help if the insights you derive from it aren’t actionable. You need to be able to answer the question: If I could predict this, what business action would I take?

A good example of this problem is churn prevention modeling. Typically, you’d predict whether or not a customer is still a customer in x months. But even if you can do this, it doesn’t mean you’re able to prevent churn. The business value of the churn model isn’t generated from predicting churn but from preventing it. What interaction do you trigger if you predict a customer is about to churn? Can you prevent it x months ahead? If so, how? You need a clear idea about what actions you can take to address the problems a model identifies.

The un-requirement: A label without noise

We've now talked about the three main requirements for a label you're trying to predict. But there's one thing you don't need when generating predictions – a perfect label. Sometimes people believe that an incorrect label could make it hard to interpret and implement results. But it's important to remember that no measurement is perfect. Obviously different labels have different qualities. Labels based on human judgement, like the sentiment of a text, are more subjective than the measurement of a voltmeter. But the voltmeter still has uncertainty in its measurements.

It's important to be aware that the uncertainty inherent in the label you're trying to predict sets an upper bound on the best accuracy of your model. You can't build a model that predicts the voltage better than your voltmeter, but that's the only limitation—don't be afraid to predict labels that aren't of the best possible quality, just be careful when you do.

STEP IV – DEFINING PERFORMANCE AND "SUCCESS"

A common mistake at this stage of a project is to take your data scientists, add some data, send them off for a few weeks, and expect this recipe will magically create a deployable model. Unfortunately, this is more likely a recipe for disaster. Only by knowing what you want to achieve can you know if you've solved the problem at hand. You need to ask yourself: How do I measure this algorithm's quality?

On Data Science Quality Measures

If you ask a data scientist about quality measures, they'll usually opt for mathematical measures like RMSE for regression, or AUC or AUPRC for classification. There are good statistical reasons for this from the data scientist's perspective, but, as discussed above, you need to remember that you're solving a business problem – not a stats assignment.

So how do you measure the quality of what you're building in a business-oriented way? You need to ensure the quality measures you choose are both appropriate for the problem at hand and understandable to other team members.

Regression tasks

Let's look first at regression problems. Assume the true value of your label is five. If you predict three, you'll take a different business action than if you predict seven. Thus, underestimating and overestimating are of potentially different severities in terms of the business problem at hand.

By way of example, consider predictive packaging. The idea here is that you forecast the number of purchases of a given item. This would allow you to package items ahead of time. Overestimating causes too many items to be pre-packaged, which will then just lay around. On the other hand, underestimating the demand will result in shipment delays. As you can see, although the two predictions (three and seven) are mathematically the same distance from the true label (five), the business impacts – and thus costs – associated with the two predictions are different.

However, typical statistical performance measures for regression tasks—such as RMSE or R—assume that errors are symmetrical, and should thus be avoided if possible, given their lack of alignment with business concerns.

Classifications tasks

In the case of classification tasks, the problem is similar to what we saw with regression above—namely, false positives and false negatives will have a different impact on your business.

Medical tests are a perfect example. A test that falsely predicts that you have a given disease (a false positive) will trigger more extensive, more expensive tests. These tests will then correctly determine that you don't have the disease in question.

The other error type is when the test results indicate that you don't have the disease, even though you actually do (a false negative). Here, the incorrect result will prevent proper treatment, potentially causing serious harm and even death.

Common data science measures of classification accuracy like F1 score and AUC assume that false positives and false negatives are of equal severity. These measures are thus misleading in a business environment and should be avoided.

The Solution: Business-Aligned Performance

Because of these issues, you need to identify a performance measure that's more closely aligned with the business problem you identified in Step I. A proven way to do this is a value-based performance measure.

Consider a predictive maintenance scenario where the cost of replacing a part before it fails is much different than not replacing a part in time and having it fail (like the medical test issue discussed above). In this case, you want your models to account for the costs of these different scenarios.

Data scientists usually look at measurements like accuracy when evaluating model performance, but sometimes it's easier to look at evaluation from a more business-oriented angle. For example, say that your department can only handle 100 requests per day. Naturally, you want to optimize the number of correct predictions in that batch of 100 so you can have the most impact. By optimizing your model for these 100 most impactful cases, you'll ensure the performance metric you're using is directly tied to business concerns.

Model Costs and Values

Altair has been pioneering the value-based approach to building models by providing ways to take costs and benefits into account during model building. This helps identify the best model for the use case, based not only on the model's statistical and mathematical accuracy, but also on the impact the model's predictions will have on your bottom line.

For example, consider a model to predict churn. If you're planning to offer discounts to those customers who are predicted to churn, you not only want to know what effects those discounts might have on churn rates, but also how they'll affect your revenue if customers take you up on the offer. It might be that identifying churn and then offering steep discounts to get retention is less cost effective than simply letting some of the customers churn – but you won't know this unless you're working in a value-based approach.

A Reprise on Unsupervised Problems

Now that you understand some of the issues with performance measures as they relate to business problems, you're in a better position to understand the statement we made earlier that you should opt for supervised problems whenever possible.

It's much harder to find performance measures for unsupervised problems like clustering or topic modeling that also align with business concerns. For example, if you want to use clustering for customer segmentation, you'll struggle to assign a business-oriented performance measure to the analysis. Usual measures like Davies–Bouldin index have the same flaw as RMSE or AUC— they don't correlate with business value.

Let's take the example of topic modeling for Amazon reviews. If you search for six topics, you'll find a topic that's about hot beverages like coffee and tea. If you increase the number of topics to twelve, the beverages topic will be split into two different topics: one about coffee and the other about tea. There's no way the algorithm can assess whether it's better for the business to separate these two or not.

That said, this shouldn't prevent you from doing such an analysis with a human in the loop if it's likely to have a bigger impact. But it should be clear from the start that the difficulty of measuring performance in an unsupervised problem is much greater when compared to an equivalent supervised problem.

When Is Our Project Successful?

The next problem to tackle is answering the question: When can I call the project a success? When you're iterating on a machine learning project, you usually don't know how good you could make your model if you invested unlimited resources. When do you stop trying to get better results? That is, when do you know a model is good enough?

We're big fans of moving to deployment as soon as the model generates decent value. We've often seen models that could save hundreds of thousands of dollars per year that aren't being deployed because the data science team was confident that, given more time, they could get even better results.

But how do you define "decent value"? This highlights why it's so vital to define your success criteria at the beginning of a project. If you have a clear threshold to make a decision, you can use this step of your analysis to identify the first performance milestone that defines your minimum viable product. If you know what your threshold is, and you hit it, you can pause and deploy, or deploy the first version in parallel while you continue to refine your model.

On Baseline Models

Now that you have a performance measure to assess the quality of your model, you want to have some kind of baseline for comparison. What should you use for you baseline? It can vary depending on your particular use case, but there are basically two kinds of baseline models: currently deployed solutions and naïve solutions.

Where possible, we want to compare the model we're building against whatever current solution is in place. For example, are you are using linear regression and Excel to predict maintenance? If so, that's great—because you can then ask what the performance of that current system is. This is critical because it allows you to put your results into perspective and to justify the time and money spent developing your new analysis.

However, what do you do if the problem you're trying to tackle doesn't have a current solution? After all, many of the triggers to create a team are new problems that traditional solutions can't solve. It's also possible that some current solution exists, but isn't quantifiable — for example, decisions are made on the fly by humans and are not recorded.

In this case, you'll want to compare yourself to naïve or simple models. The naïve or default model could be the majority class in a classification problem, the average in a regression problem, or the demand yesterday in a demand forecasting scenario. Basically, what would you do to quickly get a baseline of the current situation if you didn't have access to the model you're building? This will let you compare your solution to a baseline to demonstrate the value of what you're building.

On Validation

Validation is perhaps the most important part of any data science project. While the data scientist should be extremely careful about validation, ensuring they use best practices like ensuring that training and test sets are distinct from one another and using cross-validation, we won't address these issues in depth here, as the details tend to be complex for non-data scientists.

STEP V – DISCUSSING PROFILE GENERATION

After discussing the business problem at hand, what you want to predict, and what success looks like, you're ready to move on to discussing the final step—figuring out what kind of data you can use to make your predictions.

The General Idea

The rise of AI in business has caused problems for data scientists in terms of how those who aren't data scientists view their work, which usually goes something like: Throw data into a deep learning model and the problem will be solved.

If only it were so easy! Unfortunately, this is far from accurate. Before data scientists can start the process of building models to solve problems, they need high-quality data that's been prepared for the task at hand. There are good reasons why there are whole schools of teaching just around data preparation.

So how do you align your team and set realistic expectations? We call this step profile generation. You do it by putting the problem in context by calling machine learning pattern recognition. This makes it sound a bit old school, but it highlights an important issue here. You're detecting a pattern in your data to predict your label.

To do this, you first need to create a one-line-per-customer (or machine, or asset, or etc.) representation. This one-line representation is what we call a profile. The art of data science is to build a profile that's both complete and dense. Complete means that the data has all the information possible that may help the algorithm make its predictions. Dense means you've reached completeness with a low number of individual attributes. Large numbers of attributes can not only lead to longer runtimes, but they can also prevent the algorithm from identifying patterns in the data. The difficulty of getting the balance between completeness and denseness right makes the data preparation step some of the heaviest lifting in data science.

As mentioned above, whole schools of thought exist around how best to prep data. In what follows, we'll provide you with a high-level overview and some important details.

Source systems and availability of data

Where does your data come from? This is a very important question, because you need to make sure that you can access the data you need for your project. Some systems and infrastructures lock users out, preventing data access. This needs to be checked early in the planning of your project, and processes put in place to grant you access to the data you need to build your models.

The other question about data that needs to be addressed is: At what point is the data available? In data science, you usually work with data in two different ways. In the first, you build your models in a batch session, meaning that the calculation of models is either completely offline, or is run periodically every x days or months and then redeployed. In the second, the application of the model can be in (or close to) real time. You thus need to make sure that all data is available in both scenarios, or at the very least establish which way the data will be available during deployment, as that may influence how the model is built, as well as the infrastructure that supports it.

You may have run into situations where you learned after finishing a model that a given attribute would not be available at runtime. It might be because the measurement takes 24 hours to create, which means you won't be able to use it on the fly. This issue can also arise because of IT infrastructure, where you might be working with a system that only updates every six hours. In this stage of your project, you want to make sure that you're aware of what kind of data is available in both cases.

Data types

The complexity of the modeling process, as well as the predictive power of a model, depends a lot on the type of data you're using. In our experience, the raw data is often in a form that looks something like this:

Date	Id	Action	Value

An example of this might be sensor analytics, where you have the following information:

Date	SensorId	Channel	Action

Or with customer analytics, the table might more look like this:

Date	CustomerId	Website	ClickedItem

These data sets are time series by nature. However, this data often exists in data warehouses where you can get access to the aggregations of this data, but not the full data itself. On the one hand, it's great to have aggregated data to use for quickly building an initial prototype, because it takes less work than collecting live data, and the aggregations do often contain meaningful information. On the other hand, all types of aggregations remove information compared to what is present in the raw data.

The raw data is never aggregated in such a way that the aggregated data is as appropriate for the task of machine learning as the raw data. Thus, to get the best results possible in your project, it's important to get access to the underlying data and use that for model training and evaluation.

First Ideas on Data Preparation

When talking about availability, a data scientist can get a lot of information about the data. As we know, data is always dirty. It's part of the nature of data to be dirty. But to be able to clean the data, we need to understand what cleaning means.

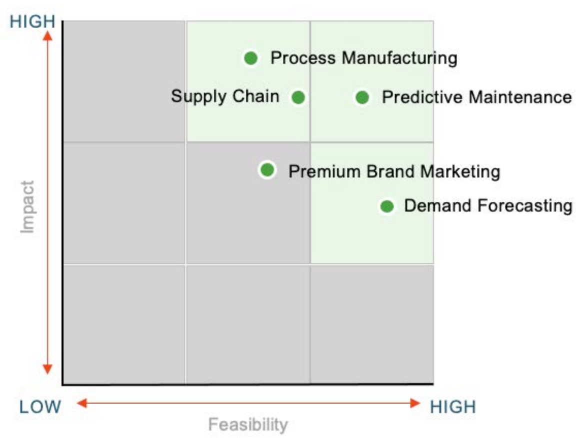
There are often obvious ways to clean up the dirty data once you talk to the subject matter experts, and data scientists should be willing to engage with those in the know to gather information and get ideas about data cleaning. The process of cleaning and prepping the data is obviously not doable in a short period of time, but you want to have a clear idea early on of how much time you'll need for this process so that you can plan accordingly.

WRAPPING UP

If you follow this guide, you'll be able to assess the value and feasibility of a machine learning project correctly right from the get-go, including getting early buy-in on critical aspects of the process. This will result in fewer projects being killed in the early stages that could have been successful, while also helping prevent work on projects that won't yield value, regardless of the reason—whether it's because they're too challenging, because the necessary data doesn't exist, or because there are too many political hurdles to clear to get the model into production.

Based on the aforementioned Frictionless AI survey, 52% of organizations are looking to scale their data and AI strategies. This demonstrates the clear need for more planning to prioritize use cases and identify areas for immediate impact. In fact, when you're getting started with a project, we usually recommend spending time up-front to map out not just one use case, but as many as possible using the process described in this document, rather than going with the problem > trigger > team model. The result of such a process is an Impact-Feasibility map which can be used to prioritize use cases and lobby internally.

There's a German saying to keep in mind when assessing the viability of machine learning for your potential use case: Better a painful end than pain without end. Don't be afraid to not use machine learning if there's a better option available!



Altair offers an AI assessment as part of our Center of Excellence methodology. Together, we map the highest priority use cases based on feasibility and value, and then help tackle those first. We also upskill the business line teams so they can address data analytics problems themselves, and train everyone – from business analysts to data scientists – to use the right tools for the problem.

If you're interested in learning more about data science and how it can impact your business, you can also take free courses at the Altair RapidMiner Academy—we suggest starting with the Machine Learning Applications and Use Cases Professional course.
We wish you the best of luck in your machine learning endeavors!

Altair is a global leader in computational science and artificial intelligence (AI) that provides software and cloud solutions in simulation, high-performance computing (HPC), data analytics, and AI. Altair enables organizations across all industries to compete more effectively and drive smarter decisions in an increasingly connected world – all while creating a greener, more sustainable future.

To learn more, please visit www.altair.com/data-analytics