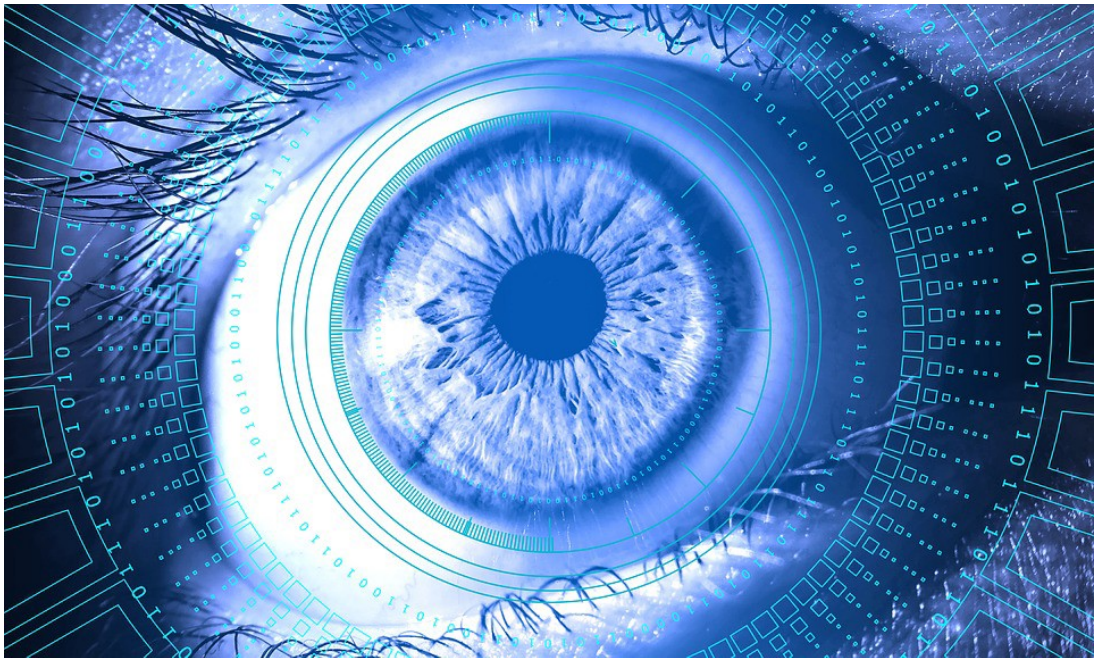
# Getting started with AI? Start here!

Everything you need to know to dive into your project

Cassie Kozyrkov Follow

Oct 19, 2018 · 18 min read



Many teams try to start an applied AI project by diving into algorithms and data before figuring out desired outputs and objectives. Unfortunately, that's like raising a puppy in a New York City apartment for a few years, then being surprised that it can't herd sheep for you.

> You can't expect to get anything useful by asking wizards to sprinkle machine learning magic on your business without some effort from you first.

Instead, the first step is for the *owner* — that's you! — to form a clear vision of what you want from your dog (or ML/AI system) and how you'll know you've trained it successfully.

My previous article discussed the **why**, now it's time to dive into **how** to do

this first step for ML/AI, with all its gory little sub-steps.

This reference guide is densely-packed and long, so feel free to stick to large fonts and headings for a two-minute crash course or head straight to the summary checklist version. Here's the table of contents:



**Cast of characters:** decision-maker, ethicist, ML/AI engineer, analyst, qualitative expert, economist, psychologist, reliability engineer, AI researcher, domain expert, UX specialist, statistician, AI control theorist.

Make sure the right person is tasked with the first step in machine learning and AI. More info here and here.

## Figure out who's in charge

The tasks we're about to tackle are the responsibility of the project's responsible adult. That's whoever calls the shots. If a PhD researcher is selected for this role, it should be because of that person's decision skills and deep understanding of your business. If you're about to put them in this role and then second-guess them later, you've chosen the wrong person. The entity we're calling *The Decision-Maker* (it could be a person or

committee) is the one that's supposed to get final say. Elect your benevolent dictator wisely.

> If the decision-maker is someone you're planning to second-guess later, you're doing it wrong.

If the decision-maker isn't well-versed in the art and science of decision-making, there's a fix: pair them up with a qualitative expert. But if the person in charge doesn't understand your business, you may as well just flush that cash now.

## Identify the use case

### Focus on the outputs

The key thing is that ML/AI is not magic and it doesn't solve every problem. It's a thing-labeler and it's up to you to figure out what you need labeled.

Thing-labeling doesn't only mean classification — "Is this photo of a cat or not?" — that's not thinking big enough. By label I mean output. It could be a category, a number, a sentence, waveform, a group ID, a single action, a joystick movement, a sequence of actions, Y/N on whether something is an anomaly… so many possibilities!

> Don't hire that PhD guru before you've confirmed you need them. Focus on the outputs first.

If you read my recent article about how algorithms work, you will have noticed that the article took for granted that it's worth labelling bottles of wine as liked or disliked by Cassie. Who signed off on wasting everyone's time with that stupid use case?! How does it help the business? Should that classifier even exist? Supposing it can be made to work, is it worth building?

*That's* the kind of job you're doing right now, my friend. It's the first job.

> Just because you can do something, doesn't mean it's a good use of anyone's time.

Imagine your ML/AI system is operational and ask yourself if you're happy you sunk company resources into making it. No? Keep brainstorming. Better to discover no one needs your application *before* several PhDs waste their lives on it.

This task can be hard because there are so many options, so settle into a comfy couch and meditate. Try my drunk island exercise if you need a bit of help brainstorming.

### Now is not the time for inputs

Some of you decision folk are gorgeously fluent with data. You can talk inputs and outputs all at once… and you understand the difference. My advice may surprise you: resist temptation! Just don't talk about inputs yet. Seriously. I know you can, but don't. Here are two reasons of many.

***Reason 1: Missed opportunities***

This is the obvious one. Some of your stakeholders are not as fluent as you and they get confused easily. In the early days, you might be pitching your idea to them in the hopes of securing resources for your project and you really don't want them to misunderstand why your system is worth having. Don't confuse them! Stay focused. Tell them what it makes, not how it makes it.

> Ask yourself, "Is this the end or the means?" If it's

> the means, don't talk about it for now.

The trouble with many fluent folk is that they think everyone shares their fluency. Some utterly brilliant people in tech have surprised me by not having this skill, so now I know not to take it for granted.

To some people, data is data. It's all the same. (Dear reader, if you're not sure whether you're fluent in this, really force yourself to slow down. Keep asking yourself, "Is this the end or the means?" Make sure you focus on the ends for now.) Stakeholders might just not be able to follow the thread of your argument, which means your pitch will fall flat and you'll miss a chance to make the world more awesome with AI.

Some folks have trouble figuring out which variable is the input and which one is the output... it all looks like one big confusing lump to them and they need your focus to appreciate why the outputs are worth having.

***Reason 2: Tacit agreement***

As an engineer who has been around engineers for a long time, I've noticed that our kind loves latching onto details. Big picture be damned, it's so much fun to kick the stuffing out of every nitty gritty, especially when someone is wrong about something! We love us some technical correctness.

I, for one, have to be on my guard against getting caught up in the technical correctness of minutiae. Joke: Technically correct is the best kind of correct. Tautologically correct is a kind of correct.

So here's the tragicomedy: when you've spent the last 6 hours arguing with your buddies about whether or not variable *x* (raw, standardized, or normalized?) is a good input with suitable logging for predicting output *y*, you've, ahem, normalized the idea that *y* is worth pursuing. You stop questioning the point of working on *y* in the first place and end up building things that don't need to be built.

### This is about teeming multitudes

Let's imagine that automating the Y/N labelling of wine bottles is the use case you're going for. (If you prefer tea to wine, here's the alcohol-free version of this text.)

Make sure you're not thinking about labelling just one or two bottles. ML/AI makes sense for automating *many* repeated decisions. It's not for one-offs.

> ML/AI is not for one-offs, so make sure your business needs an impressive number of items labeled.

You're imagining labelling at least a few thousand of them? And when this thing is live, you're sure you can't just look the answers up instead of predicting them? Good. Let's continue.


Grab a pen, write down the labels you'll accept (binary Y/N in this exercise is easy to write down, but you could have chosen to make them more exotic if you needed to get creative). Write down how you'd know if the answer is right for one of them. Write down what mistakes would look like. Expect mistakes in machine learning! If you're expecting flawlessness, it's best to back away quietly before disappointment crushes your soul.

### You might not be ready for machine learning

Still struggling to find a use case? Consider pausing ML/AI in favor of analytics for a while. The goal of analytics is to generate inspiration for the decision-maker. Once you're inspired, you can come back to ML/AI and start over. Analytics (a.k.a. data-mining) is a great idea for every project, whereas ML/AI is only for projects where the goal is to use data to automate thing-labeling. Although the underlying math is often the same, the processes are very different. Data-mining is all about maximizing the speed of discovery, while ML/AI is all about performance in automation. In data-mining, there's only one mistake your team can make, whereas ML/AI has an impressive list of potential kablooies. Sign up for those headaches only when you have a use case that's worth the bother.

### Who is it for? Think about your users!

Who is your dazzling invention for? Who benefits? This is a great time to consult with a UX (user experience) specialist and map out your application's intended users.

> Ideating new technology often starts with the what, but it's important to cover the who before you proceed to the how.

Something I've learned by spending time with UX designers is that my

knee-jerk description of who my users are… is usually rather naïve. Have I thought about usability for indirect beneficiaries, for society-as-a-whole, for other businesses, for machine systems using the output as input, for engineers engaged in debugging, and so on? To avoid shoddy UX design, please take the time to think about all your myriad user categories before you proceed. User doesn't just mean customer or end user in the most obvious sense.

### Is it ethical to proceed?

What if your idea isn't uniformly beneficial to everyone? While mapping out your ideal use cases, consider those who might be harmed by the existence of your system. I don't just mean competitors in your industry. Can any humans be harmed by your application? This is especially important if your technology scales to millions or billions.

> Think about the humans your creation impacts! Who benefits and who might be harmed?

If you care about developing technology in an ethical and responsible manner, it's your duty to think about *all* the people your creation could affect. Thinking about them *after* you've built it is irresponsible. The time for it is *now*! An ethicist can help you shoulder some of the burden. Bringing them on board as an advisor is a great idea if your project has the potential to strongly affect human wellbeing. Along with your UX specialist, their participation in the project will help you ensure that groups affected by your creation are given a voice.

## Do some reality checks

Once you can clearly articulate what labels you're after, it's time for a quick reality check: do you have data about this business problem?

No access to data means no point in proceeding. You might be able to get what you need online, though — there's a rising trend in making data available for free (for example here).

It still has to be relevant, though. You know you can't use *my* Marmite consumption patterns (define: big data) to predict *your* blood sugar levels. Clearly useless data doesn't count. You don't have to analyze the data yet (that's later in the project) but you should check that you will actually have something to analyze when the time comes. No data means no ML/AI.

> No access to relevant data or no computers to crunch it? There's no nice way to say this...

This is your dream of ML/AI if you have no data.

You'll also need to verify that you have the computing power to process your data. (Have you heard of my employer? They have plenty and they like to share. Just sayin'.)

### Reality checklist

Make sure you can answer yes to all of these. Each one of these is likely to get its own guide later, stay tuned. This is just a quick overview of questions for identifying a nonstarter.

- ***Appropriate task:*** Are you automating many decisions/labels? Where you can't just look the answer up perfectly each time?
- ***Reasonable expectations:*** Do you understand that your system might be excellent, but it will not be flawless? Can you live with the occasional mistake?
- ***Possible in production:*** Regardless of where those decisions/labels come from, will you be able to serve them in production? Can you muster the engineering resources to do it at the scale you're anticipating? You'll be looking into this question in more detail when you sit down with engineers, but for now the glimmer of a sanity check will suffice.
- ***Data to learn from:*** Do potentially useful inputs exist? Can you gain access to them? (It's okay if the data don't exist yet but you have a plan to get them soon.)

- ***Enough examples:*** When you're sharing a beverage with your statistician or machine learning engineer buddy and you casually mention the number of examples available plus the kind of output you're aiming at, is their expression free of cringes? (I'll teach you how to develop this intuition yourself in a future article.)
- ***Computers:*** Do you have access to enough processing power to handle your dataset size? (Cloud technologies make this is an automatic yes for anyone who's open to considering using them.)
- ***Team:*** Are you confident you can assemble a team with the necessary skills?
- ***Ground truth:*** Unless you're after unsupervised learning, do you have access to outputs? If not, can you pay humans to make them for you by performing the task?
- ***Logging sanity:*** It's possible to tell which input goes with which output, right?
- ***Logging quality:*** Do you trust that the data actually is what its purveyors claim it is? (To learn from examples, you need good examples to learn from.)

### Assemble your team

Once you've cleared the reality checklist, it's time to start recruiting, which you can do in parallel with wading through the rest of this guide. My advice on the roles you're looking for is here.

## Craft a performance metric wisely

### Own the tradeoffs

The next bit can get a bit tricky if you're new to it. You're in charge of deciding how much each kind of outcome is worth. Is the disgusting bottle of wine that got a Y twice as bad as the delicious bottle we missed out on?

3.4823 times? Up to you!

Struggling? Grab someone who likes numbers and have them help you brainstorm. Qualitative experts are specially trained for this, but your standard calculator-slinger will do in a pinch. If you want the best helper, utter the formal jargon for it (*eliciting indifference curves*) out loud in a pentagram to summon an economist.

> Economists make surprisingly useful advisors for AI projects.

Now that you've figured out how to trade off various results on **one** single output, it's time to think about how you'd like to score a few thousand of them at once. Using the average is an average choice, but you needn't be average. Again, decision-maker is boss here. The right way to score it depends on what's right for your business.

### (Optional) Expert mode is heavy on simulation

Tricky, complex projects benefit tremendously from *simulation*. That's where an analyst specializing in generating fake but plausible data can help you see the potential consequences of the choices you're making in this article.

> A dress rehearsal helps opening night go smoothly.

Simulation gives you a dress rehearsal, which helps you get a bunch of kinks ironed out before you have to start your project for-realsies. Like analytics, it takes some of the burden off the decision-maker's mandate to *meditate hard and think of everything*.

### Make your metric

There are many different ways to make a metric. In our wine example, you could go for the really simple one: **accuracy** a.k.a. *"Don't be wrong."* Every mistake is equally bad (0), every correct response is equally good (1), then you take an average (which you've been itching to take — someone needs to put a central limit on how appealing people find averages).

Hang on, maybe instead you have FOMO when it comes to good wines and you're quite happy to have duds along the way? Well, that's a different metric called **recall**. Or maybe you don't want to have your money wasted; you're in a tight budget. If you're a starving graduate student (I've been there) and you need to be sure that when the system says Yummy you won't be wasting your money, but you're okay with missing out on beautiful bottles. That's an entirely different metric called **precision**. We'll slow down and deep-dive on metrics in another post. For now, nevermind what they're called, just make one up that reflects what's right for your business.

Need help? Has your economist wandered off already? No problem! Perhaps you have a friend who loves designing games? Games geeks have unknowingly been training for this their entire lives! If you don't hang with that crowd, you can call your qualitative expert instead, since it's their job to help decision-makers clarify their thoughts on this kind of thing.

### Ask for an expert review

In applications where human wellbeing is substantially on the line, seek a consultation with a panel of experts to verify that it isn't possible to get a high score on your metric in some perverse and harmful way.

Which experts? Have you heard the one where a decision-maker, an ethicist, an AI control theorist, a statistician, a user experience researcher, a behavioral economist, a domain expert, and a reliability engineer walk into a bar…?

Sure, this can be overkill on benign business applications, so your

qualitative expert's introductory-level familiarity with each of these disciplines has you covered and your game designer friend's instincts are also tapping into a similar vein.

### Hello, business performance metric!

When you're done, you'll hear an angelic chorus. You've created your business performance metric!

This is not the same thing as a loss function (we'll talk about those later). When it comes to metrics, the possibilities are endless and it is up to the decision-maker to figure out what's actually important. In case you're feeling anxious about tackling this, I'm brewing a few more articles to help you master metric development.

### (Jargon trigger warning) Something your AI experts should know

Here's a nuanced thing you can skip until we dig into it in a later post, but if you know what a loss function is then you'll realize we'll have *two* metrics in play. If that means nothing to you, let's not worry about it for now, your job here is only to ensure your ML/AI experts read the next paragraph. Many of them missed this lesson in school. ***Warning:*** many decision-makers will find that it reads like a jargon-gibberish nightmare — my apologies! — so just forward it along.

> The loss function is for optimization, not for

testing.

"In applied ML/AI, the loss function is for optimization, not for statistical testing. Statistical testing should ask, *"Does it perform well enough to build/launch?"* where *perform* should be defined by the business problem and its owner. You're not supposed to alter the business problem statement to suit your convex optimization ambitions. For expediency, you're free to optimize using a standard loss function that moves in the same direction as the function your leader's imagination just spawned (perform correlation checks* analytically or with simulation), but please test with *their* function.** Have you been using loss for all your evaluations? Don't worry, it's a common mistake which may have something to do with software defaults, college course format, and decision-maker absenteeism in AI."

** If no standard loss function correlates decently with the performance metric, please alert your decision-maker now that what they're asking for is very difficult and probably requires investing in optimization researchers.

** Dissenting AI experts, read this. Although it's not about functions, the general ideas about working with decision-makers apply.

## Set testing criteria

### Define your population of interest

Talking about your system "working" makes no sense until you specify which instances you're intending for it to work on. All US summer inputs? All inputs globally? (These are different!)

Before you proceed, you'll need to define your statistical population of interest, the broad collection of instances your system needs to demonstrate good performance on before you give it the green light. I've got a two-part guide for you in case you need a refresher on that.

### Commit to crushing it!

Now that you have your performance metric and population handy, you have one more job to do before you can go put your feet up. Since getting here can take months, your poor feet must be pretty sore.

Here's the last task: decide on the minimum performance you're willing to sign off on, because I'm about to make you promise that you're not going to let this system take over labeling stuff for you unless it's good enough.

Setting a bar for testing the system is a responsibility the decision-maker should take very seriously. (Image: Maria Fernanda Murillo winning a high jump competition, credit: Diego Sinisterra)

What does "good enough" mean? How high should you set the bar? Up to you, but you *must* commit now.

> Setting criteria up front is part of how you keep yourself (and me) safe from horrible machine learning and AI.

This criterion is not a guiding star. You can also have a performance level that you tell your team to reach for (purrrrfection?), but that's not what you will test against. Test against the bare minimum.

### You biased species you

Why are you coming up with this cutoff *now*, before you've even assembled a team for this project? Turns out that as members of the human species, we suffer from a few lovely cognitive biases (don't say *sunk cost* or *endowment effect* or *confirmation bias* unless you want that economist to appear out of nowhere again, this time with a psychologist in tow) which boil down to this: when humans invest time and effort into something, we fall in love with what we have made… even if it is a pile of poisonous rubbish. Then we find ourselves bargaining: *"Awww but the performance is not so bad. I'm kind of proud of 12% accuracy. Maybe we could launch it anyway? Why don't I test against 10%? See? It passes. It's statistically significantly good enough."*

If you want to wallow in this sad topic with me for another 6 min, who am I

to stop you?

> We humans fall in love with what we have poured effort into... even if it is a pile of poisonous rubbish.

While we are still sober(!) and before we've poured much of ourselves into it, we're going to take a cold, hard look at the business problem and say, *"If it does not meet **this minimum** requirement, I* promise *that it DIES."*

### Better than human?

I hope you have an inkling now as to why I'm forced to suppress a chuckle when people ask me, *"Is AI better than humans at stuff?"*

> Better than human? Tautologies are tautological.

If its maker required it to be better than human and did the testing properly, then if it wasn't better than human… they crushed it. If it exists, then yes. Unless they didn't require it to be better than human, in which case, it probably isn't. Why are people asking *me*? Ask those decision-makers how they set up their criteria and testing. (Speaking of populations, *which* humans is it supposed to be better than? That one volunteer?)

Also, let's not obsess so much about whether machines are better than us at stuff. My computer has always been better than me… at performing multiplication. This bothers me not at all. My bucket is better than me at holding water. What's the point of a tool if it doesn't reduce your effort or increase what you can achieve?

Instead, focus on whether it's good enough to be useful.

### Don't be too demanding

Always requiring better-than-human performance might make you miss out on profit. It's a bit like saying that you'll only hire an Olympic gold medalist to lay bricks for you. Maybe an Olympian is stronger than your average Joe, but having such a strict hiring bar might leave you with no workers at all. Lower your standards to where they make good business sense, but no lower. Setting the testing bar at your bare minimum is *incentive-compatible,* as economists might say. (If you're fresh out of B-school and want to nerd out on mechanism design, we're essentially setting up a BDM auction for our hypothesis testing procedure.)

> Don't miss out on a profitable solution by testing against an excessively high bar.

**Sign up for Get Better Tech Emails via HackerNoon.com**

By HackerNoon.com

how hackers start their afternoons. the real shit is on hackernoon.com. Take a look.

Your email

Get this newsletter

By signing up, you will create a Medium account if you don't already have one. Review our Privacy Policy for more information about our privacy practices.

Learn more.

Medium is an ope
million readers co
dynamic thinking
undiscovered voi
of any topic and b
surface. Learn mo

Medi

That's step one of ML/AI done! Step two involves data and hardware (and engineers!) so you
brush up on some vocabulary in anticipation of forthcoming attractions.

If you found any of the ideas in this guide worthwhile, please tell them to whoever in your net
likely to find themselves in the decision role. Let's build a skilled and responsible new crop of
bright AI future!