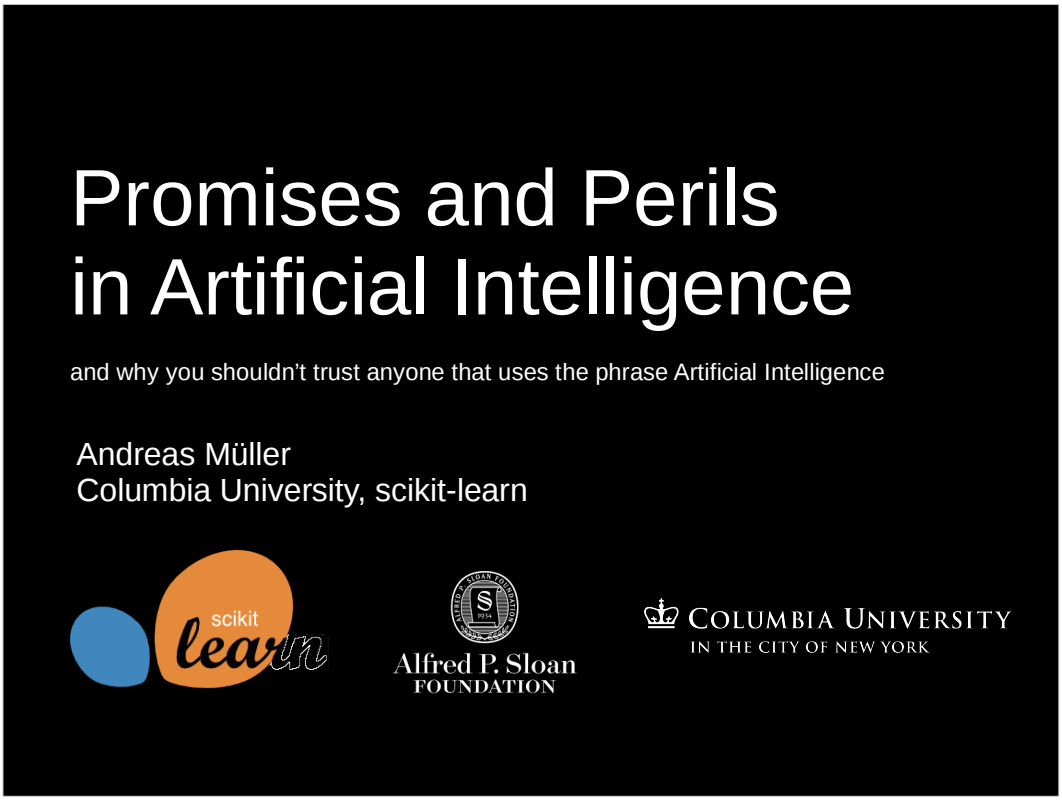


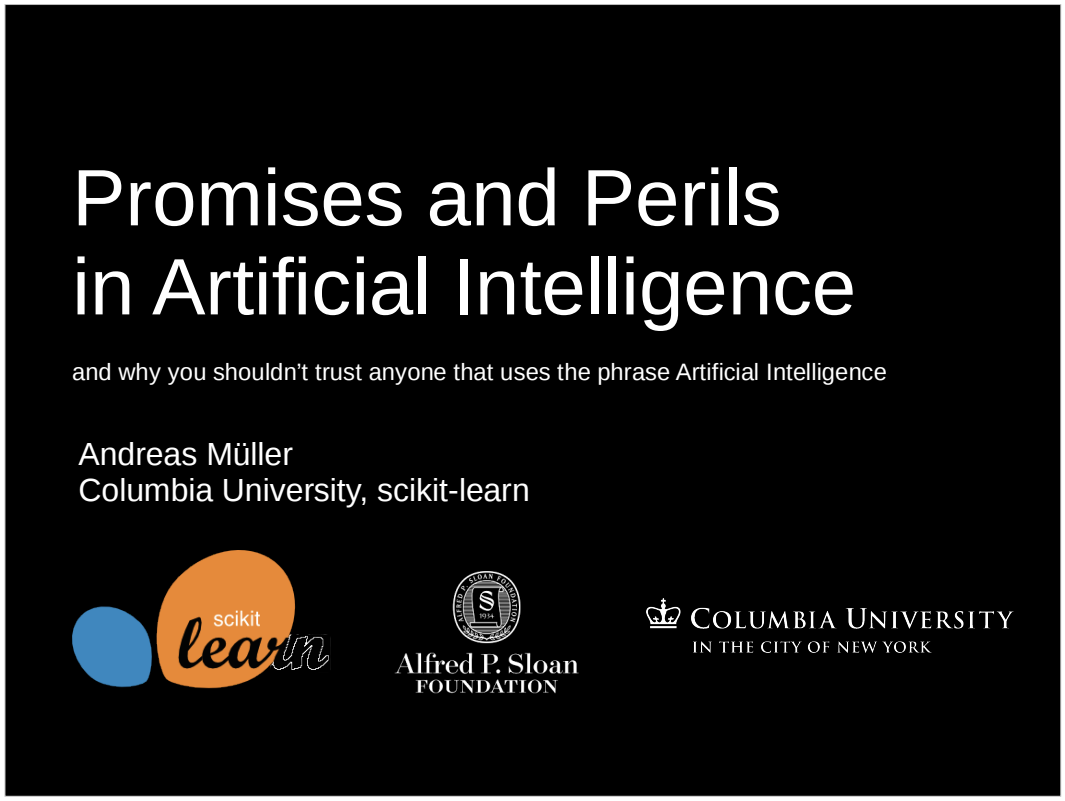
Thank you to the organizers for the invitation, and for the opportunity to give the closing keynote.

Let's see how far I can get into this talk before they'll regret giving me 40 minutes for dad jokes and ranting.

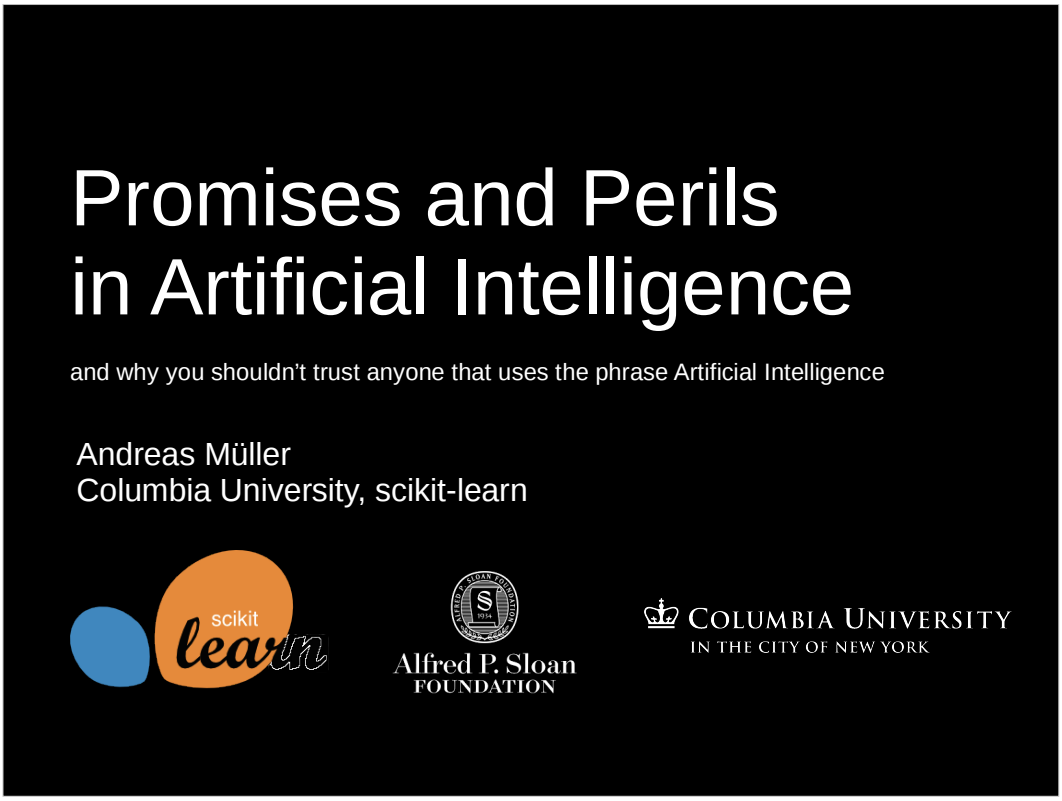
Though I feel putting me after dalek poetry was a bit mean, and I don't think I can live up to that.

Let me briefly say a little bit about myself.

I work in data science and machine learning, and I like to write software and solve data driven problems. I'm one of the core developers of scikit-learn and I teach data science at Columbia. Recently the Alfred P. Sloan foundation has given me money to work on scikit-learn while I sit in my office in Columbia, which is really amazing!

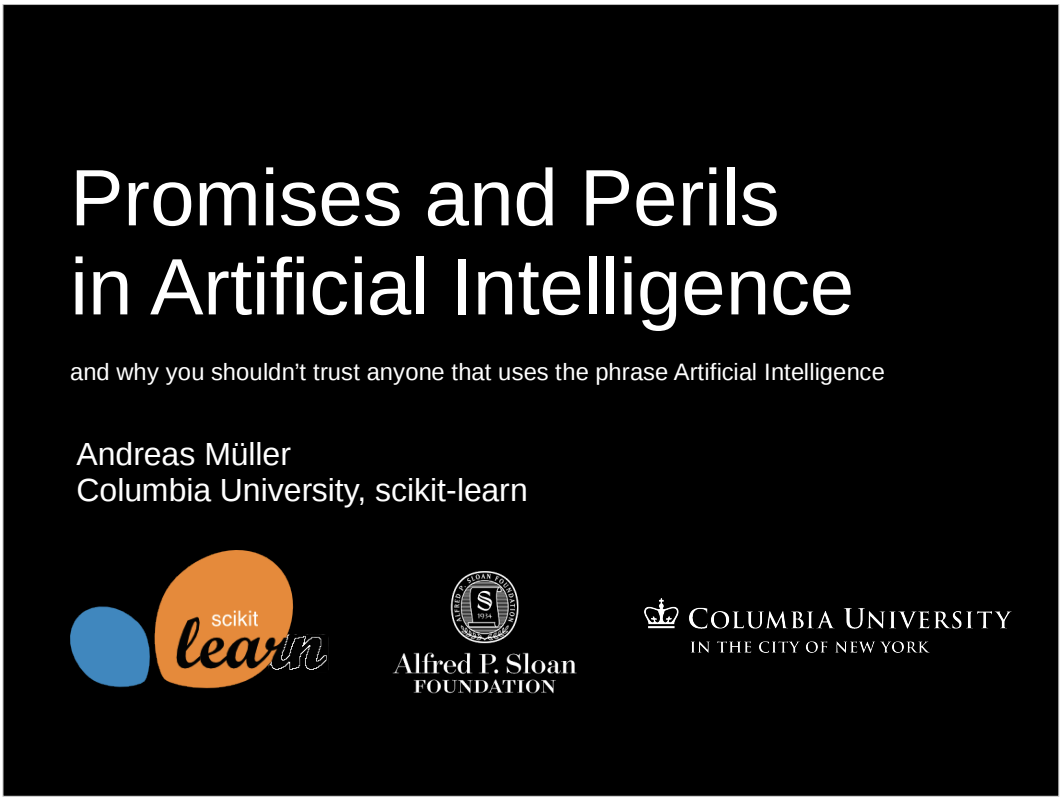


You probably heard about this big fear that AI and data science will take away people's jobs by automating them. And it does. I'm pretty sure a lot of jobs driving cars will be automated in the not-so-far future. But I think making other people's job obsolete seems a bit mean. So what I'm trying to do most of the time is to make my own job obsolete. I want data science to be so easy anyone can do it, so they don't need an expert data scientist like myself any more. And then I can go lie on a beach. Because that's what happens with you if your job becomes obsolete, right?



Clearly there's a lot of knowledge involved in doing data analysis, and some statistical knowledge for example is extremely important to create valid predictions and models, so there is a certain bar for using AI, but that bar should be as low as possible, and ideally only require the core concepts, and not some programming details.

This really relates to the first keynote today: my goal, and I think a shared goal of the scikit-learn project, is to create autonomy. I want scientists and domain experts to be able to built their own data driven systems, without having to convince a machine learning researcher to help them.



Most of my talk will be about where AI is right now and it's future potential and pitfalls. Also I want to give my own point of view of what's happening in data science and where there's hype and where there's actually cool stuff happening. In particular, I really don't like the word AI, so I'll stop saying that now, and say machine learning instead. Because that actually means something. I also want to talk about current and future challenges in applying machine learning and data analysis.

Everything in this talk is either about work that someone else has done, or that hasn't been done yet, so none of it is mine.

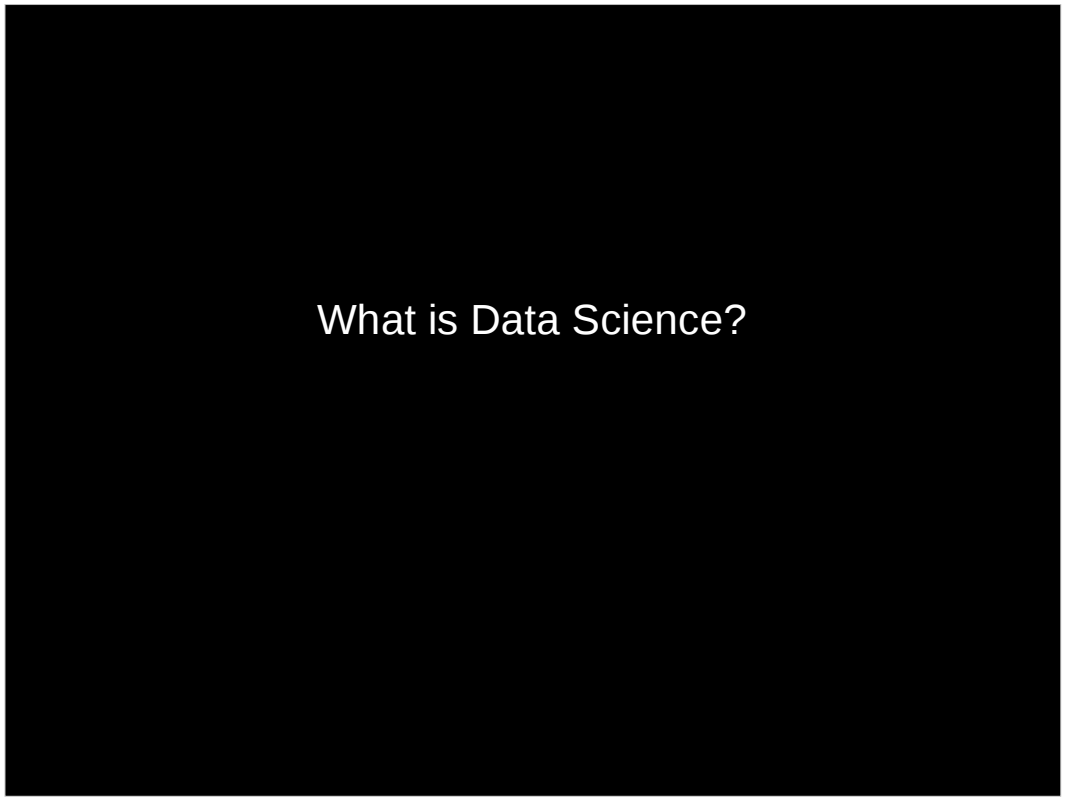
Because what I actually do most of the time is writing unit tests. And Nicholas, I have to tell you, I still punch the air every time they pass.

I think the last three times we did a scikit-learn sprint, like we did last week, I refactored the unit tests, and I don't think anyone wants to hear about that.

I’m a machine learning enthusiast, and I think data driven computing and machine learning can help in countless applications. So I’m really hopeful for it making an impact in many areas. But there’s also a lot of hype out there, created by companies to sell their products, and by the media to sell their articles.

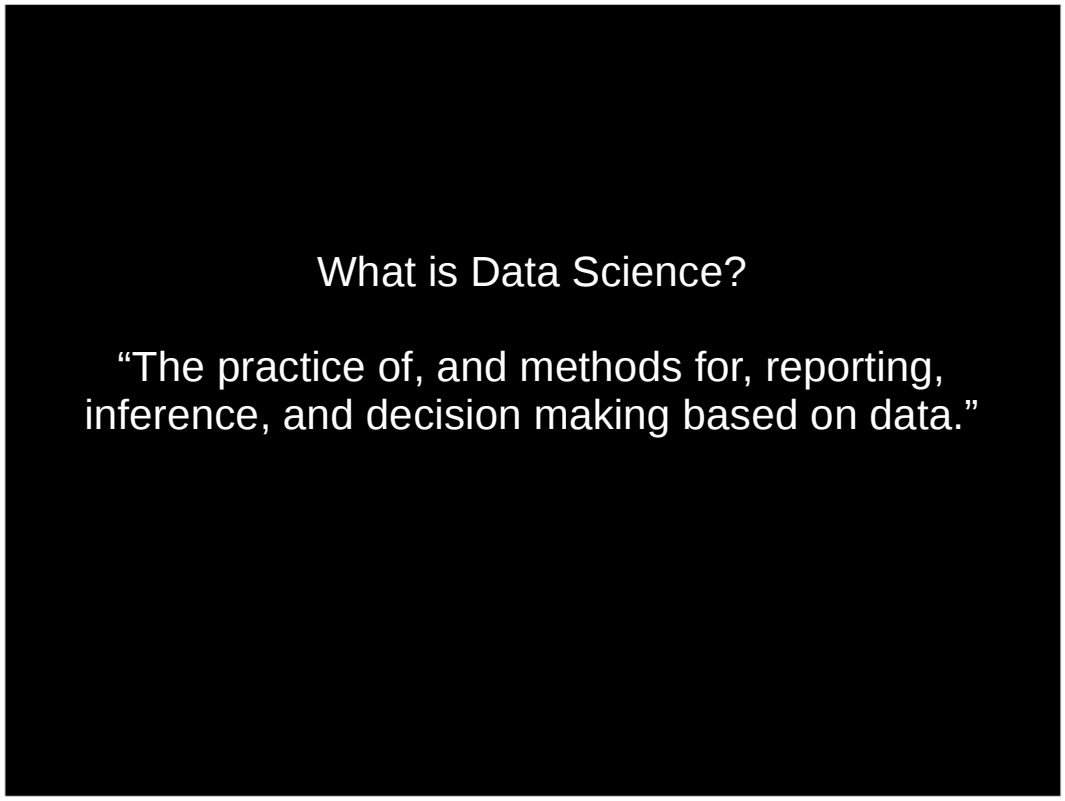
There has also been some fear-mongering about computers becoming our robotic overlords, but fortunately, that has diminished a bit – I think mostly because it doesn’t sell anything but maybe some bad pop science books.

So what I want to do first is bisect what data driven reasoning and machine learning is, and will be able to do in the future, and what is just buzzwords.



Before I go into any detail, I want to explain two terms, or at least give my personal working definition: Data Science and machine learning.

I don’t particular like the term data science, but I think it’s here to stay, so I’m gonna use it, and I want to explain briefly what I mean by it.

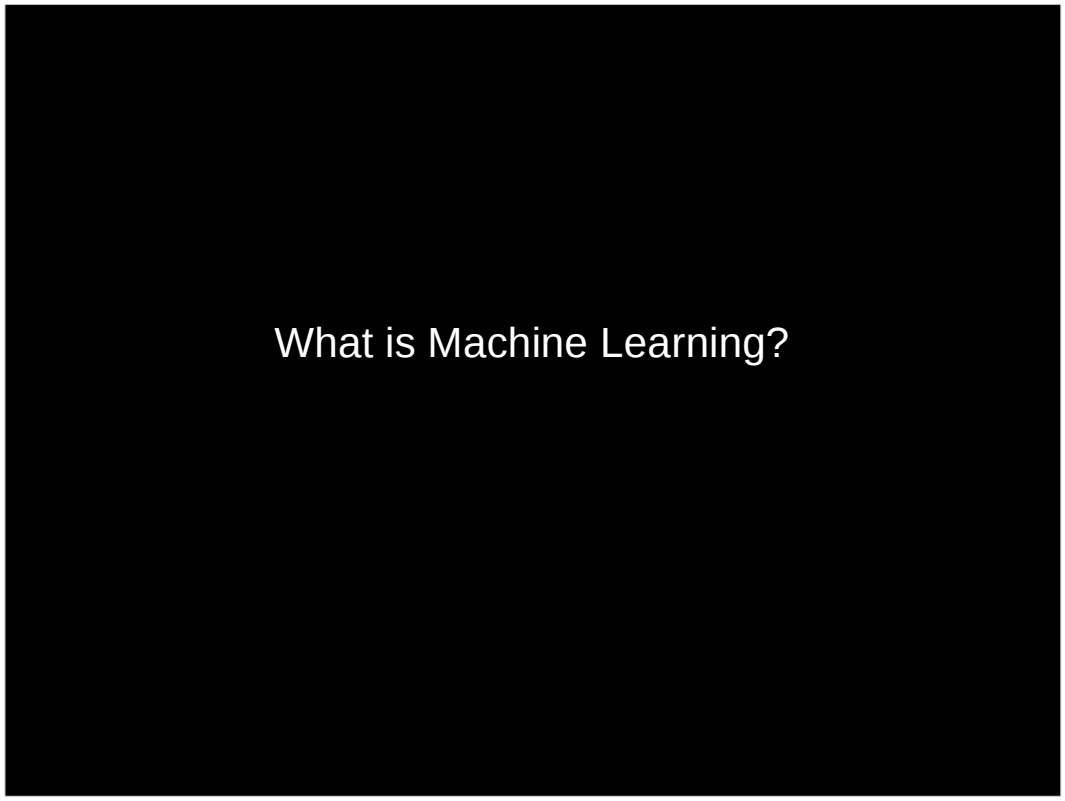


Usually data science is used as a very big umbrella term, and I like to phrase it pretty widely: <read>

Some people might argue that’s just all of empirical science. Or maybe that’s just statistics. And maybe that’s right. Feel free to entirely disagree with this definition, I don’t claim it’s universal, I just want to define what I mean when I say this.

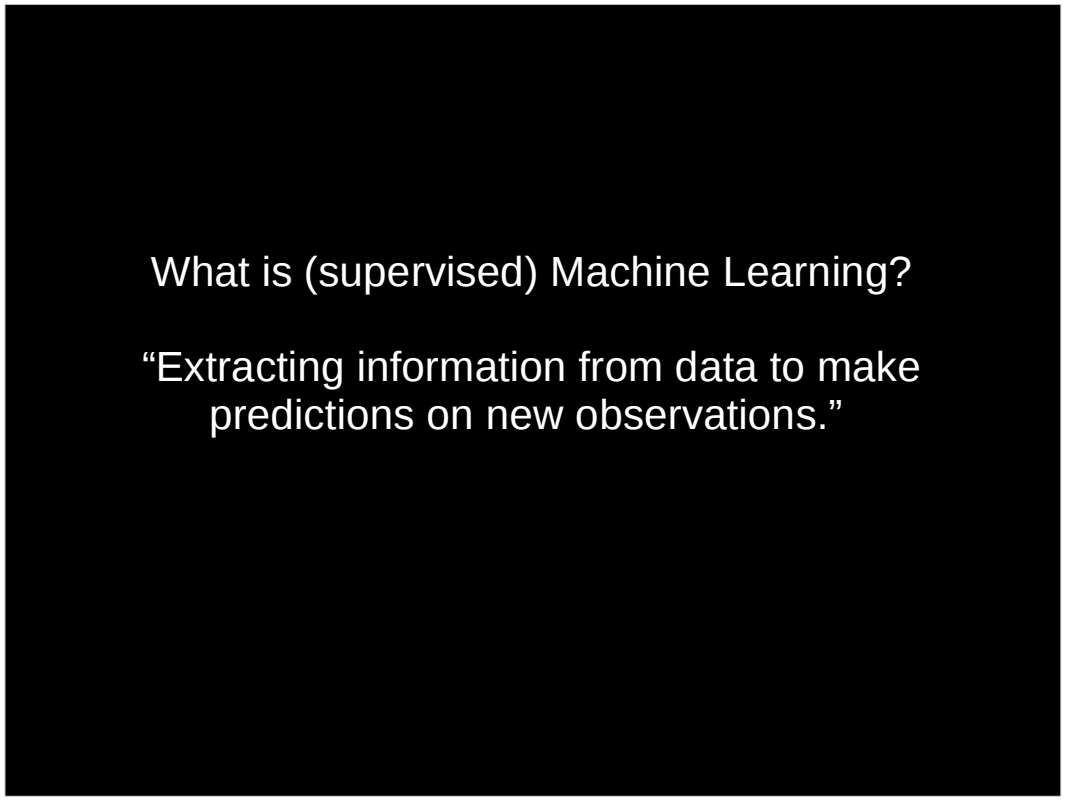
One important aspect here is that there are two sides: there is an applied side, which tries to gain understanding, and there’s a methods side, where people are working on improving the tools. This could be mathematical and statistical tools, or tools for data bases and distributed computing.

So this includes SQL queries and hypothesis testing and, most importantly, counting. Counting is probably the most commonly used and most important data science technique.

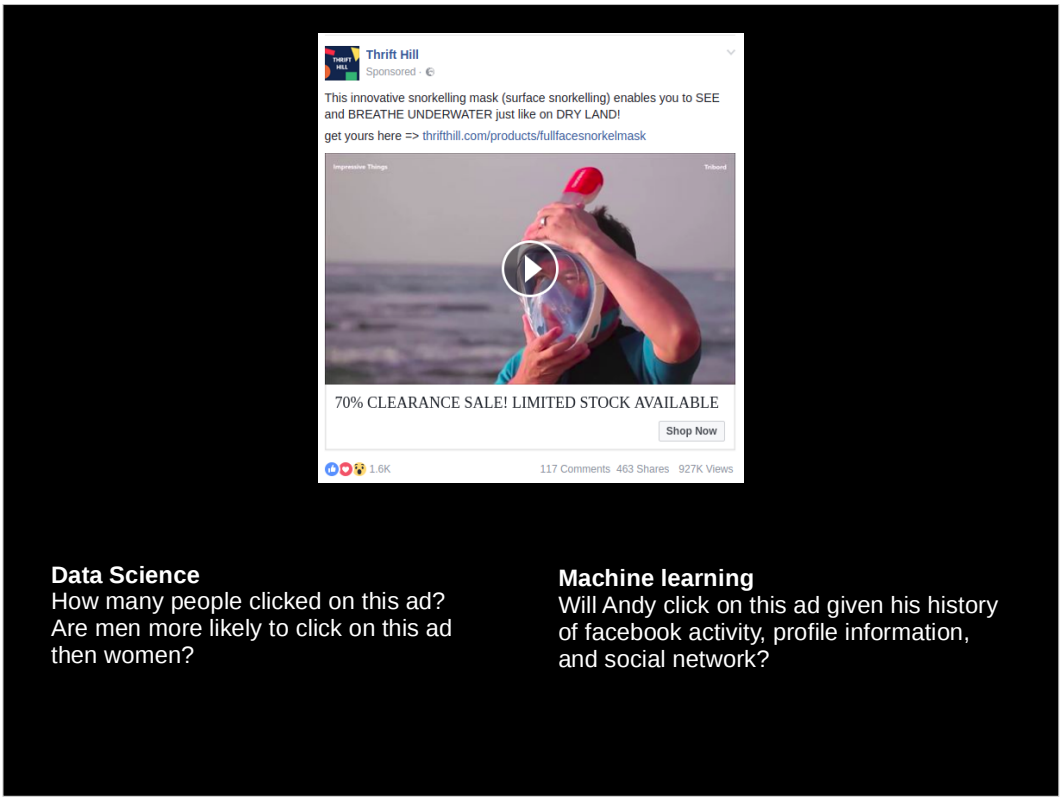


The other term I want to define is machine learning, which sometimes people call artificial intelligence.

Machine learning a much narrower field than data science, and the area I mostly work in. It’s also hard to define, and again I don’t aim for a general definition here, just a useful one.



To me, machine learning, or more specifically the subfield of supervised learning is about extracting information from data, to make predictions on new observations. So you have some data you collected, and you want to learn from that so that you can make future decisions automatically.



So here is a quick illustration of what I understand as data science, and what I understand as machine learning.

Facebook showed me this great ad for a snorkel that allows you to breath like a fish, or something.

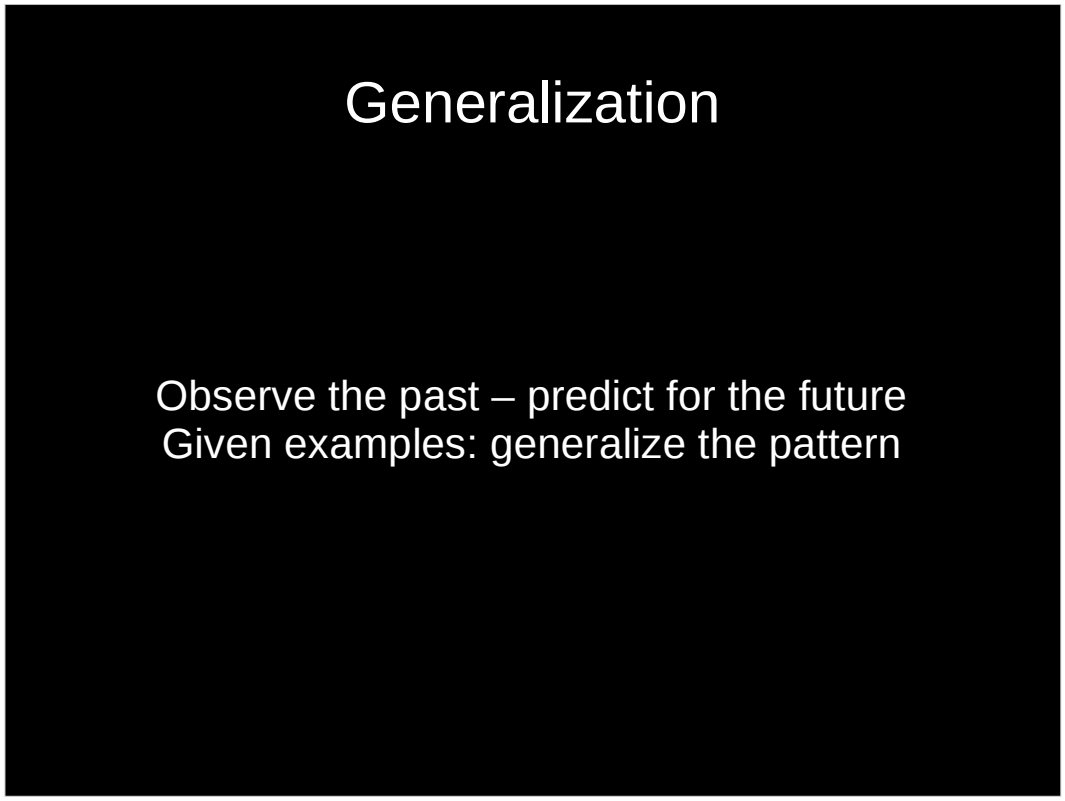
Data science questions associated with this are <read>

And these are interesting and important questions, though they can be answered just by counting.

A prediction or machine learning question would be <read>

Because facebook showed me this, I guess their algorithm said “yes”. Or at least that was the result of their considerations of auction pricing for placing this ad and click-through probability and whatever else goes into their decision making.

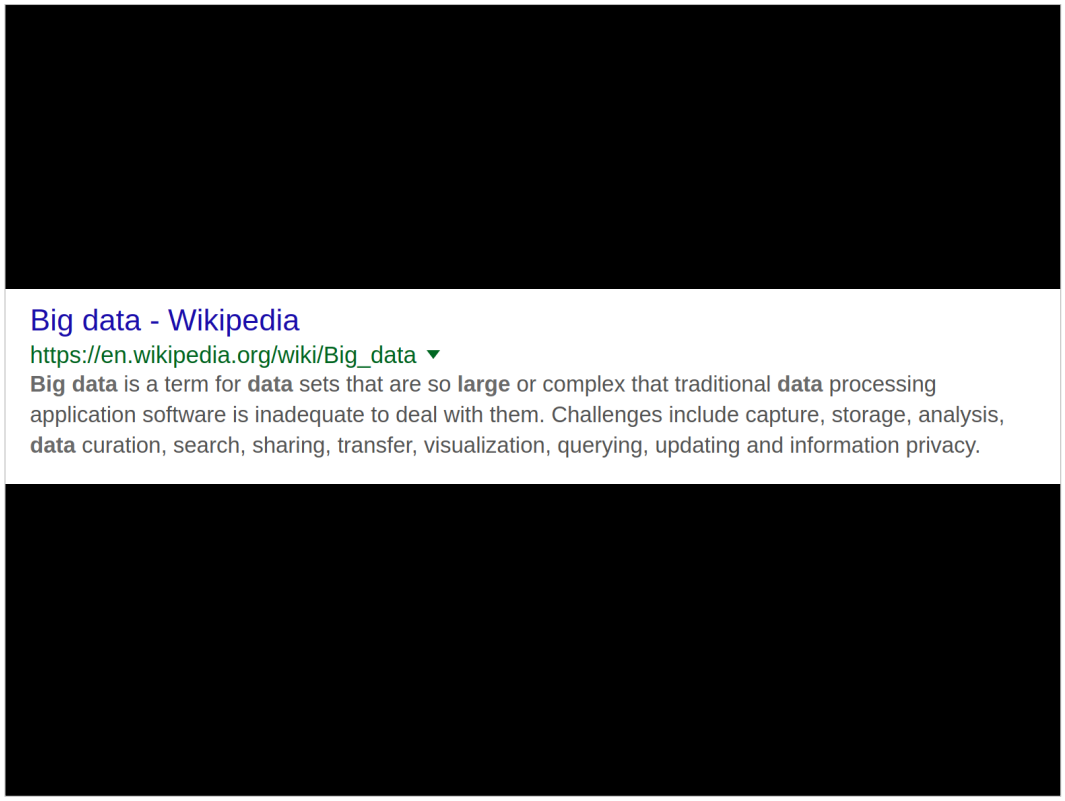
So just to put the fear of AI into perspective: facebook has probably one of the worlds best AI research teams and basically unlimited resources, and still they predict I’m gonna click on this? I think we don’t have to worry about skynet for a while.



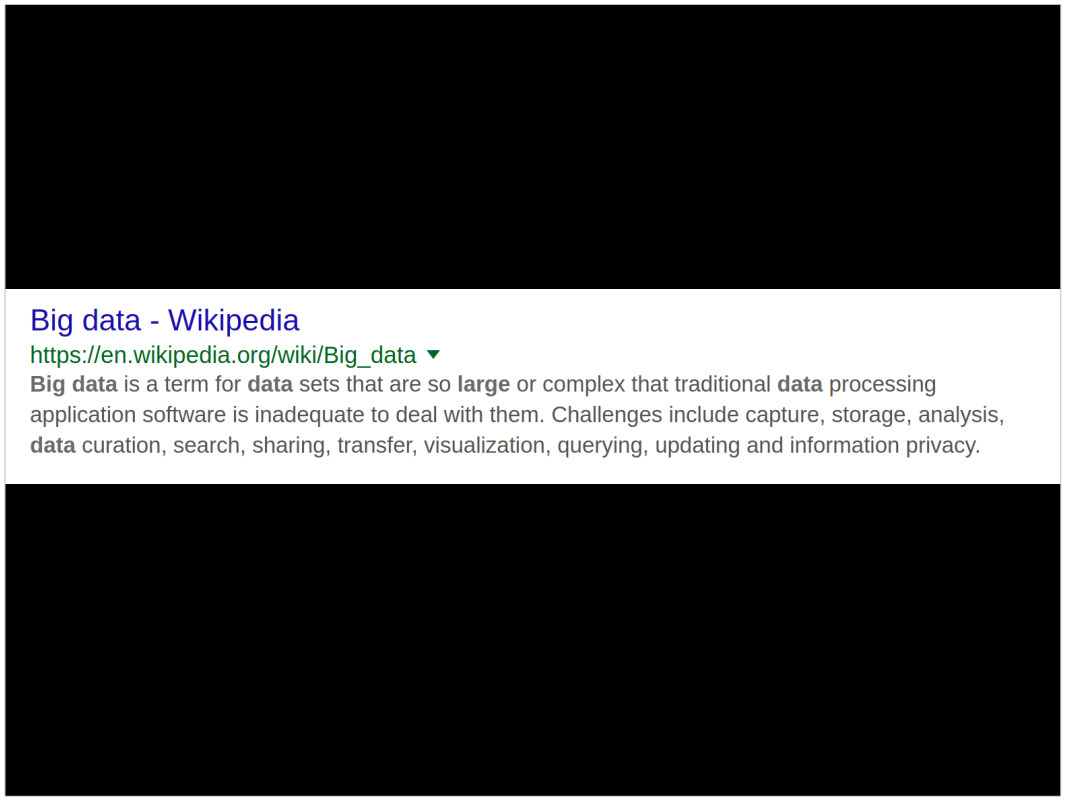
One of the core concepts in machine learning is generalization. In (supervised) machine learning, we collect a datasets of observations that have some inputs and some outputs that we're interested in. For the ad example, that could be a history of facebook users, together with whether, when they were shown the ad, they clicked on it or not. Then, the goal is to predict, for a user who we haven't shown the ad so far, whether they will click on it. So here we make use of past observations to try and predict what will happen in future events.

Another common examples is that we want to automate a manual process. For example we can collect patient data, together with a medical diagnosis from a doctor. Then we can try to learn from this data, and automate the diagnosis process, effectively replacing, or at least augmenting, the expert knowledge of the doctor. So that is ML in a nutshell.

There are some other fun words I like to discuss.



First this one: big data.
Wikipedia actually has a very useful definition <read>.
I have some problems with how this term is used.
First, this is often used to describe volume of data, but data analysis is not about the amount of data you have, but about information content and your ability to extract this information.
Second, this term is abused ad absurdum, and applied to basically any data.
There are people that have a lot of data. Let's say traditional computing and data processing software is a single machine with python. I boot up an cloud instance and get half a terrabyte of ram. I can do a whole lot with that. Most businesses have less data than that. Most scientists have less data then that.
Sure, if you're facebook, you might have "big data" and half a terrabyte doesn't cut it.
Or you're working on the sloan digital sky survey and get high resolution pictures of the whole sky. Sure, you need more than a workstation for that.

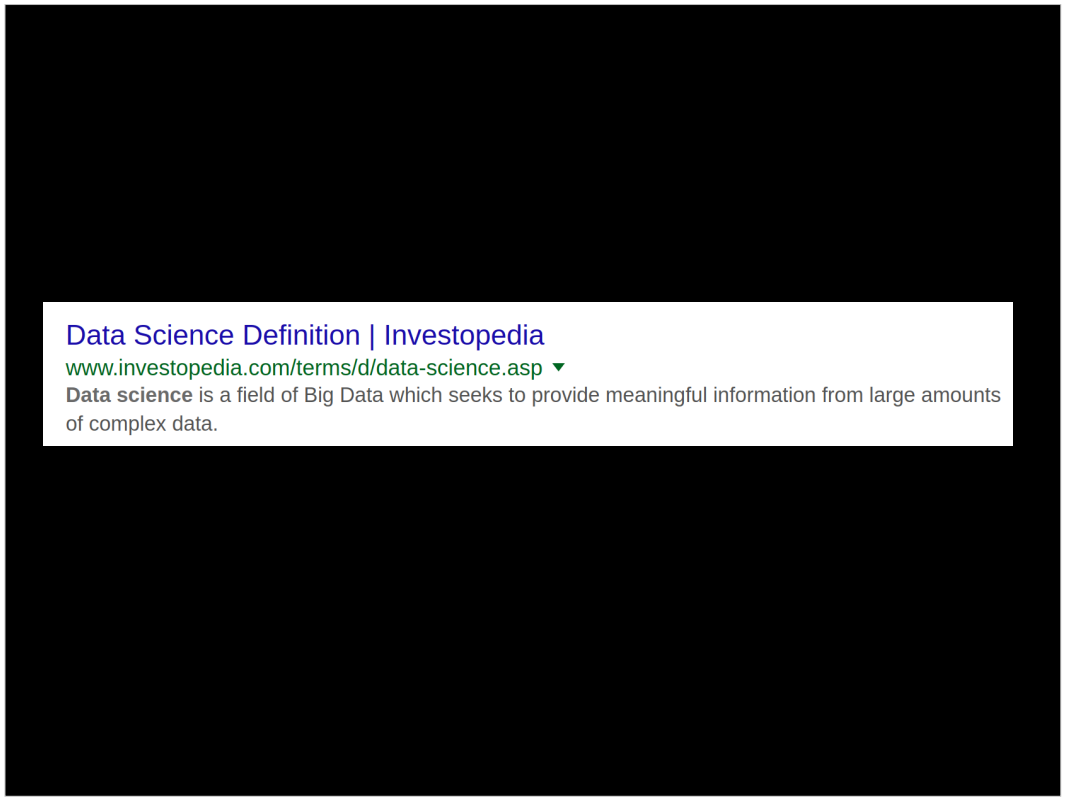


And I’m not saying these are not important problems, but these are not the problems most people have. I see people with 500mb of data setting up a hadoop cluster. Why?

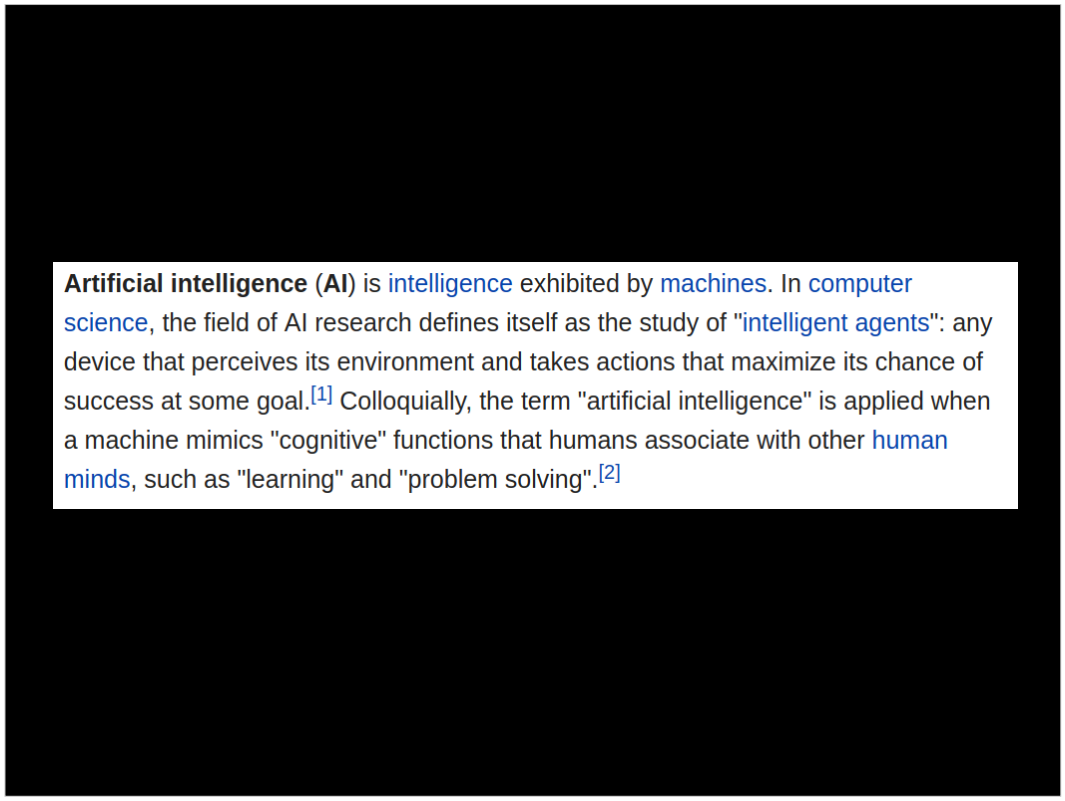
Some people have a table with 10.000 rows and 100 columns and they call it big data because excel crashes on them.

So a lot of people throw this term around to a point where it’s meaningless. Yes, there is big data, and yes, there are very interesting big data applications, but that’s not what most people do, and it’s not where we’ll see the biggest impact in my opinion.

So if someone uses the term big data, I’d be extra skeptical.



Just to illustrate, look at this.
Researching this talk I looked for other definitions of data science online, and I found this one from investopedia – this was the second hit, I think.
It defines data science as a subfield of big data.
So it's not science unless it's petabyte scale?
Or maybe they don't know what big is?
I don't know.
But ok, please don't mention big data to me, you can see, it's not good for my heart.



The other term that I really don't like, as least as it is used now, is Artificial Intelligence.

So here's a good definition, again from wikipedia <read>.

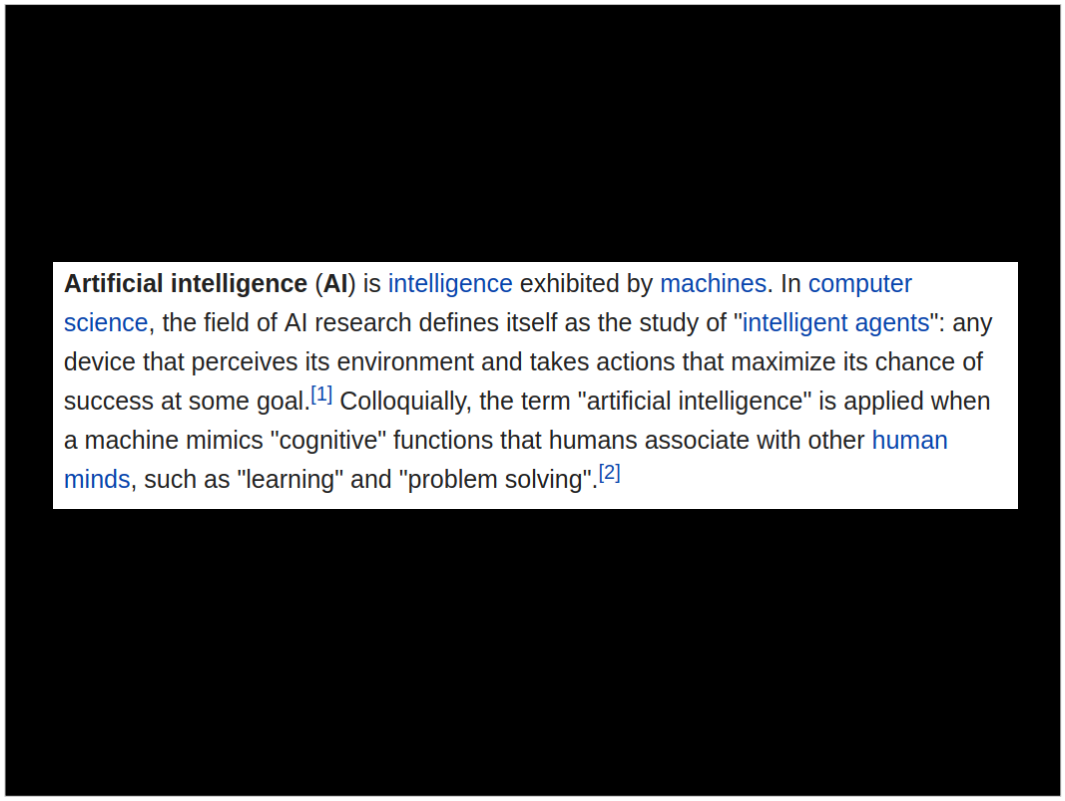
The problem with this is that it is so general. It's at least as general as my definition of data science.

Any program with an "if" clause can be claimed to behave intelligently, because it reacts to input and solves a problem.

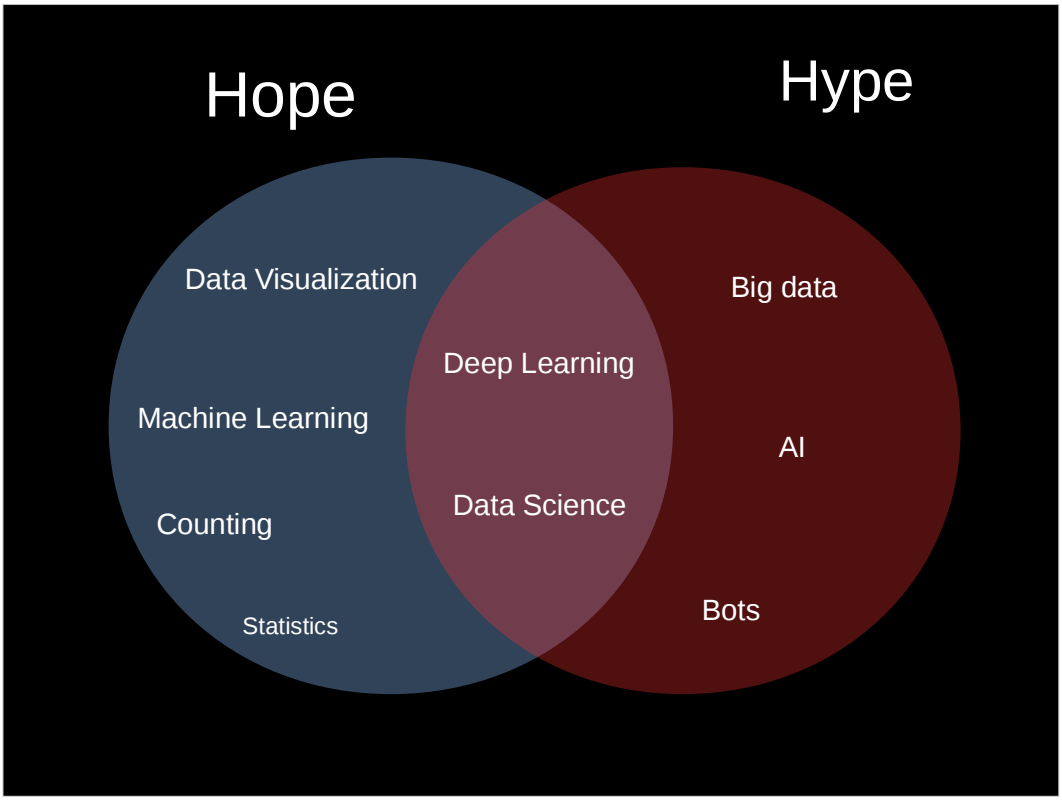
Right now, often, when people say artificial intelligence, they mean machine learning. But sometimes, they mean they wrote an if clause.

Artificial intelligence is anything that anyone perceives as intelligent, which really can be anything.

It's also misused, in that it is often related to giving the impression of human-like agents, even if the agent is just an avatar image that does a text search.



There are companies with AI in their title that do much less learning that what the face detection in your phone does. The face detection in your phone or on facebook or where ever is actually some quite interesting machine learning right there. But it's so common place now, that probably few of you think of this as intelligent or AI.



Here's a quick summary of what I think are hopeful technologies that will bring us forward, and things that are full of hype.

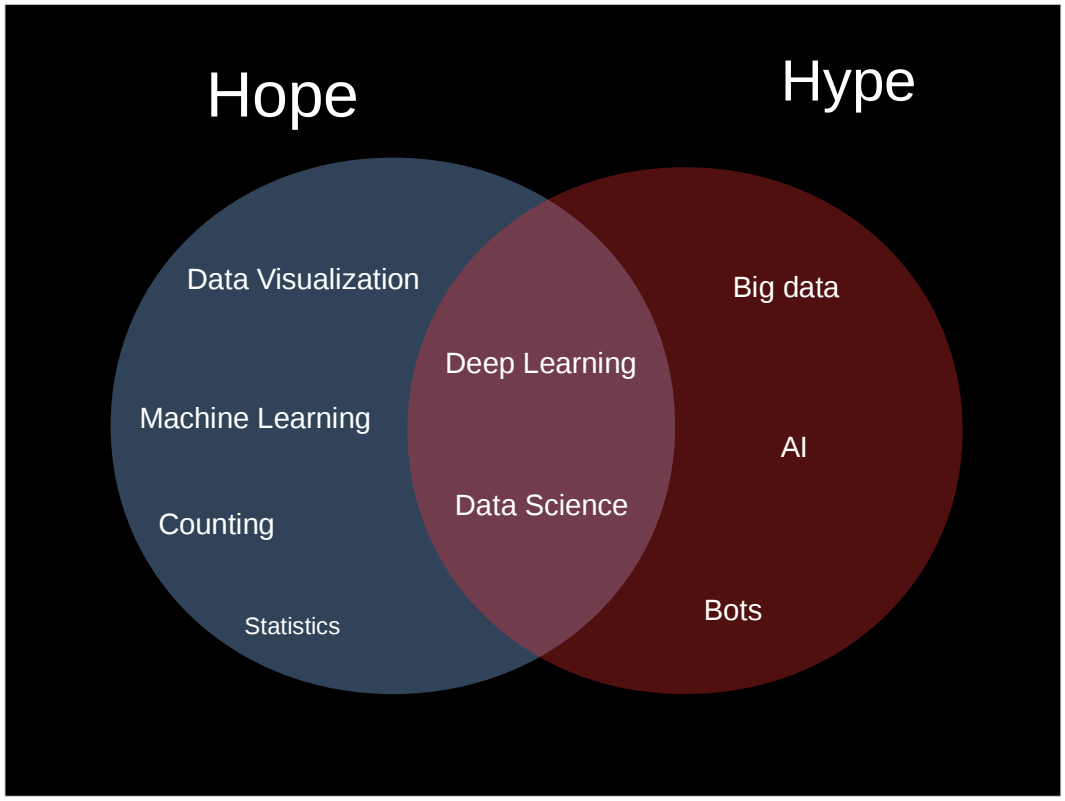
Data science is basically any data analysis or any data analysis tools, so much of that is useful. But it's also nothing new. It's a new name for lots of techniques that existed for a long time, but it's probably a useful umbrella under which there are many important technologies.

My favorite technology is counting, because its very easy and effective and often easy to interpret.

But other core technologies are statistics, data visualization and machine learning. Maybe ML got some hype, but not a lot. I didn't actually talk about the difference between ML and Statistics, and they are actually quite similar tools. Usually I say statistics is about inference, and machine learning is about prediction, but the boundaries are blurry.

What gets a lot of hype is deep learning, and for a good reason. I'll talk about an application in a second, but deep learning is a set of machine learning algorithms that have been incredibly valuable, in particular in computer vision, audio and speech. It also helps to make your selfies look like they were painted by van gogh or miro.


DEEPART.io



That doesn't mean they are a cure-all, though, and applying them to any new domain usually requires a lot of work. Also, they have same fundamental limitations as any other machine learning algorithms, and I'll talk about that in a bit.

And then there are my favorite, or maybe least favorite buzz words. Big data, which either means peta-byte data, and then nearly no-one needs it, or mean just any data, and then is pretty empty, and probably shouldn't be called big, if I can process it on my laptop.

Also, AI, which, as I mentioned, can mean anything or everything. You can say "strong AI" which means human level AI, and that's something you can aim for in long-term research, but if a company says "we use AI", that means really nothing.

Because the hype side looked empty I also put bots in there, mostly because people are incredibly excited about anything that looks or behaves slightly like a human, but I feel trying to imitate humans on a such a shallow level is totally useless. We can't create anything that even remotely thinks like a human, and I think pretending otherwise is not helpful.

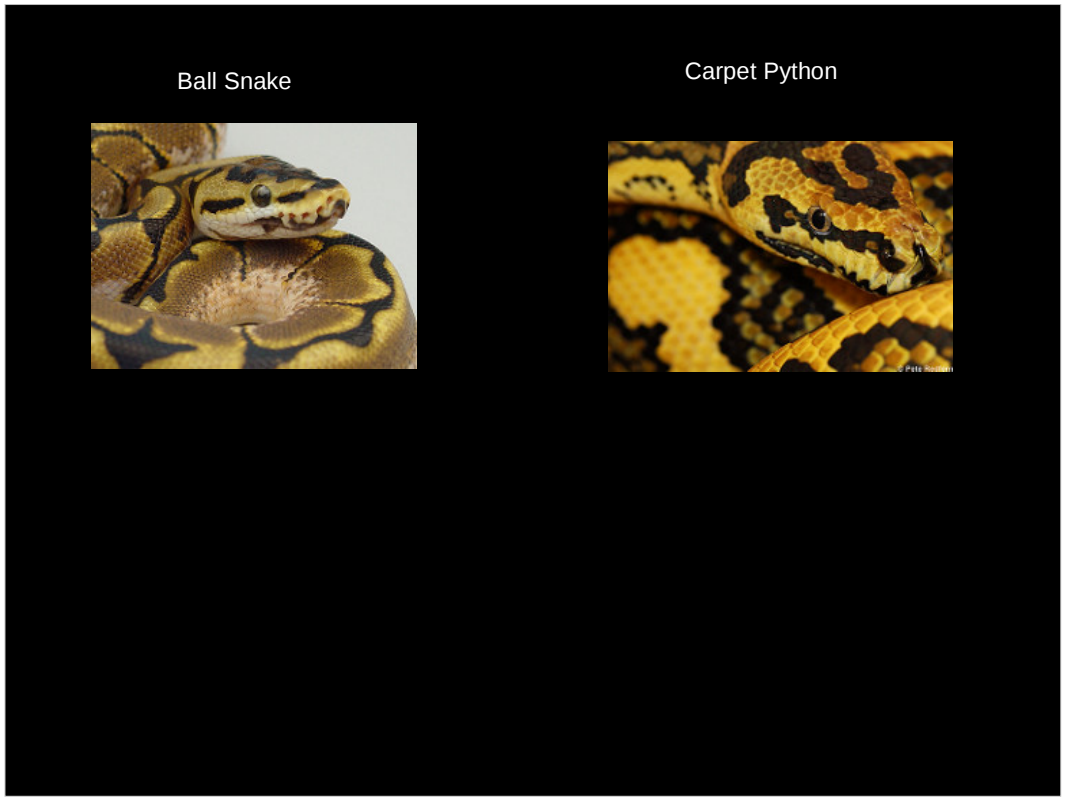
But let me talk a bit more about the actually useful stuff.

After all this ranting, I want to talk about about what machine learning can do today, because it is really amazing.

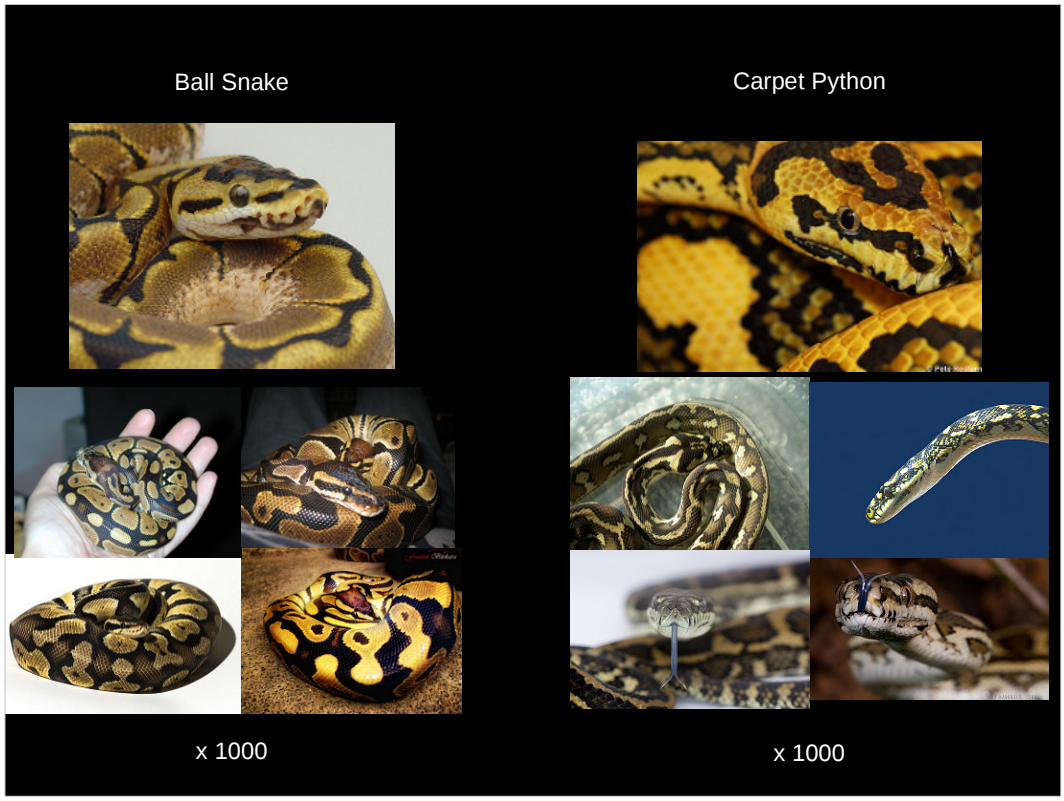
One area where there has been huge progress in recent years is computer vision, the area of my phd.

I want to pick an example task from computer vision, which is classification, and walk through what computers can do right now, and what they can’t.

Classification is one of the most easy to understand tasks, and also one of the most useful ones.



So lets say, I have a picture of a snake, and I want to know whether the little guy in the picture is a ball snake or a carpet python. So as I said earlier, to build a machine learning model, we need to collect a bunch of data for which we know the right answer. So I collect a bunch of images for which I know they contain a ball snake, and others for which I know they contain a carpet python.

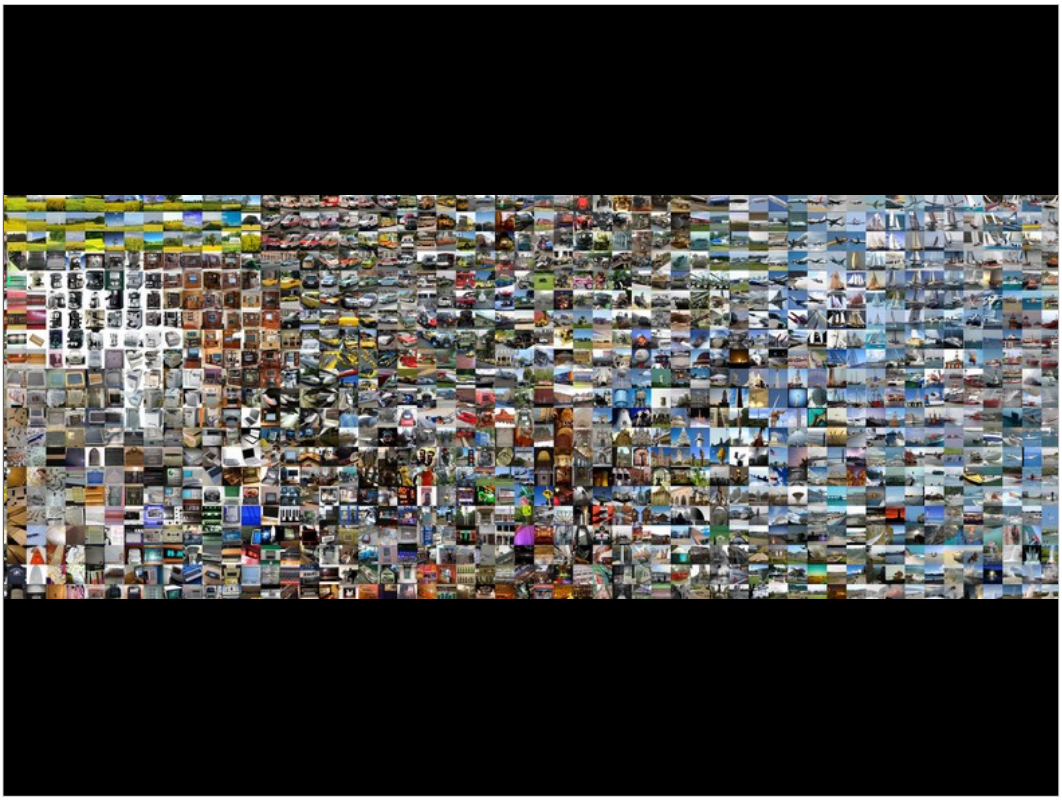


So here are 4 each, but more realistically, you'd have a couple of thousands or tens of thousands usually.

If you collected this data, you can probably build a pretty good classifier to distinguish these two kinds of pythons. And I think that's pretty impressive.

Thousands might seem like a lot of images, and there's a good reason for that: each new machine learning model basically learns from scratch. Any time we solve a new task, we basically start with a newborn baby, that known nothing about the world, and that doesn't even really know how it's own eyes work.

If we start from literally zero knowledge, we need much more data then when I would want to try to teach **you** to distinguish these to snakes – I'm pretty sure you already have some idea what a snake is, and how perspective and color work, for example.



One of the big advances in computer vision in recent years was that people build models that were generic enough about learning to “see” so that you could start learning new concepts much more easily, say using hundreds instead of tens of thousands of images. That was a pretty big break-through, enabled by deep learning. Models were trained to recognize many different classes, and that created a generic model to start from.

Now you can use one of these ready-made models off-the-shelf, for example by just using pip install keras, and you can learn new visual concepts relatively easily.

Still, this is not how humans learn. You still need hundreds of examples, and these models don’t know anything about scale or rotation or perspective or any of the things that are important for understanding the world.

It’s like taking a newborn baby, and flipping hundreds of photographs in front of its face and telling it what everything is, instead of the baby interacting with the world.

People built similar generic models for audio and text processing, that allow you to learn more easily from less data, by allowing you to not start from scratch each time. But these are all still very rudimentary, and are no-where near what humans can learn.

However, machine learning algorithms have a big advantage over people: they are very patient, and relatively fast. So they can look at a lot of data.

Maybe you can learn to distinguish these snakes from 10 pictures, and a computer needs 10 thousand.

But if you show the computer 10 million (and these were the right pictures) the computer could become much better than any human. But only at doing this exact single repetitive task of labeling snakes, and only these two kinds of pythons.

It would still not know that all these snakes actually live in a 3d world, or that they can move or literally anything else. But it could really perfect this binary decision.

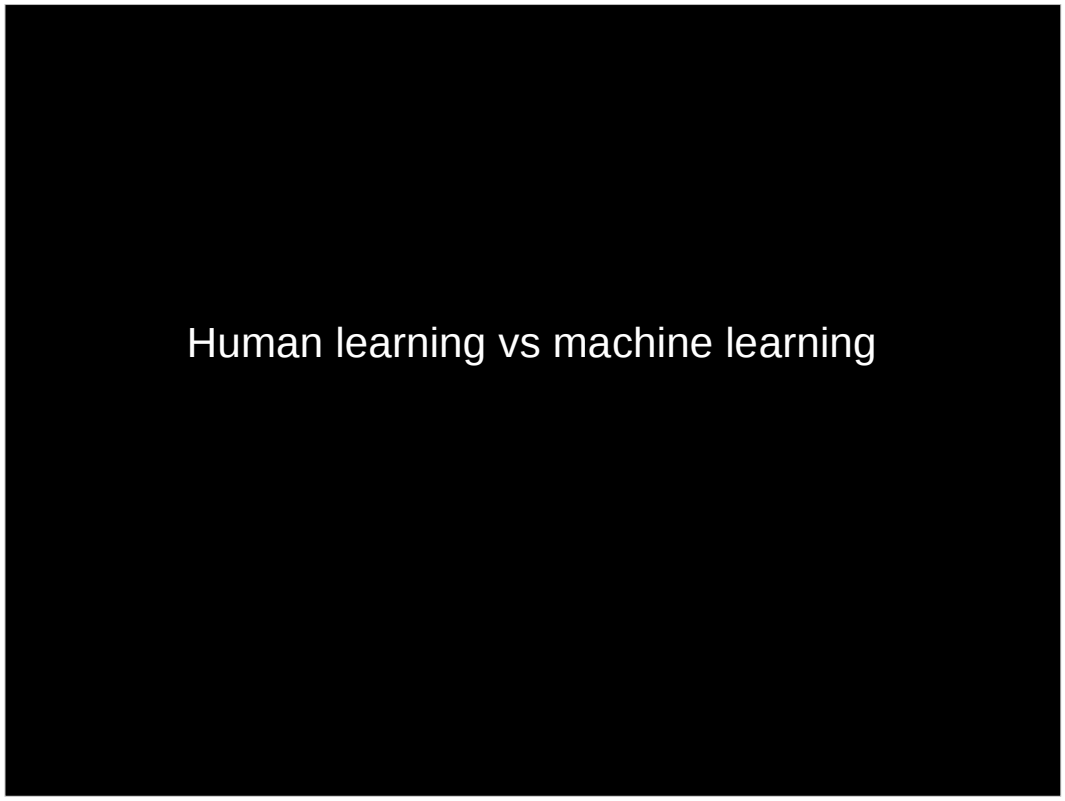
One example that I like to give here is medical imaging. Finding cancer in medical images is often very hard, and requires expert training, and doctors are expensive, so having computers assist with this kind of diagnosis seems like a good idea.

On the other hand any doctor will only see so many tumors in her lifetime. She might have a vast knowledge of how the human body and the cancer works, but if the computer can look at maybe 1000 times more images, it might become better in the end.

So if you have enough data, you can probably do pretty well today if your task is to answer a yes-no question, like sick or healthy. Or if you want to distinguish two kinds of snakes.

Actually, researchers can even train systems that can distinguish 10,000 or more different visual object categories reasonable! That’s pretty impressive, I think.

For classification tasks, if you can gather enough data, you can now train a super-human algorithm.

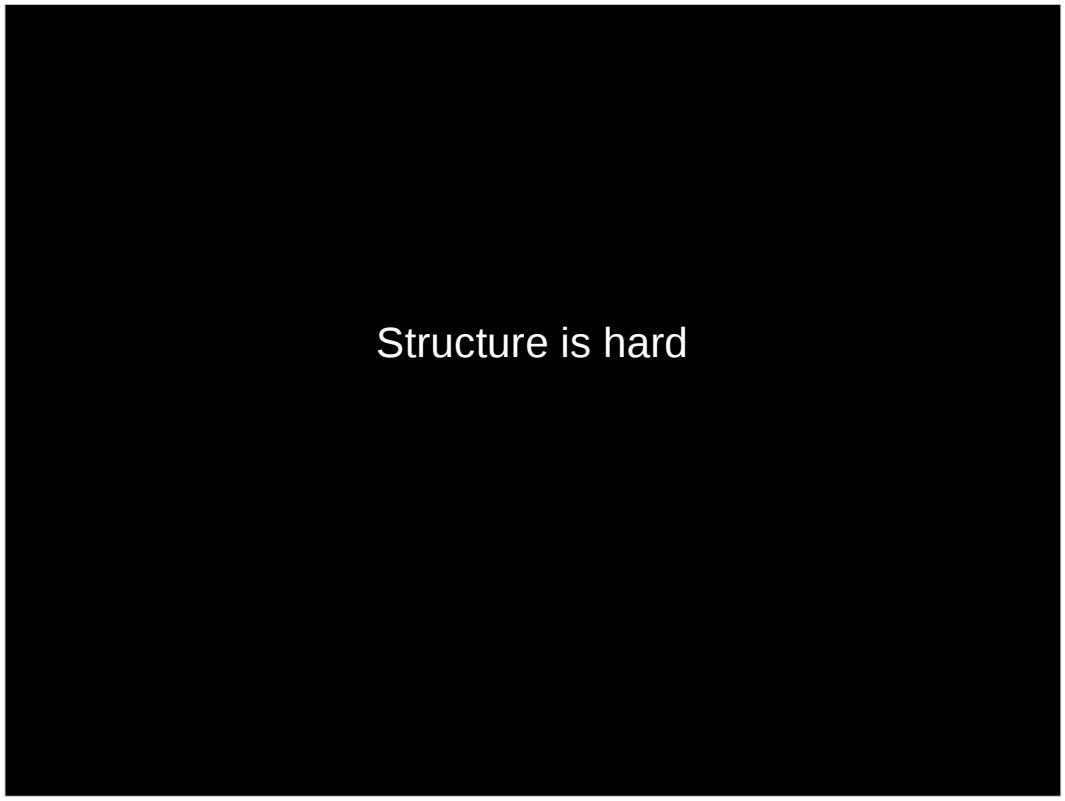


So to summarize, machines generally need to start from scratch when learning any new task, though there is some progress in vision, text and audio to improve this. For the record, these areas are particularly interesting because humans are really incredibly good at it. There is large parts of our brains dedicated to vision and audio, so it takes more for computers to catch up.

Still, the algorithms need a lot more data than a human would, and what they learn usually applies only to the very narrow task they were trained to do.

This is partially caused by the inability of most algorithms to interact with the world, which many researchers think is a requirement for building rich “mental” models of what is perceived.

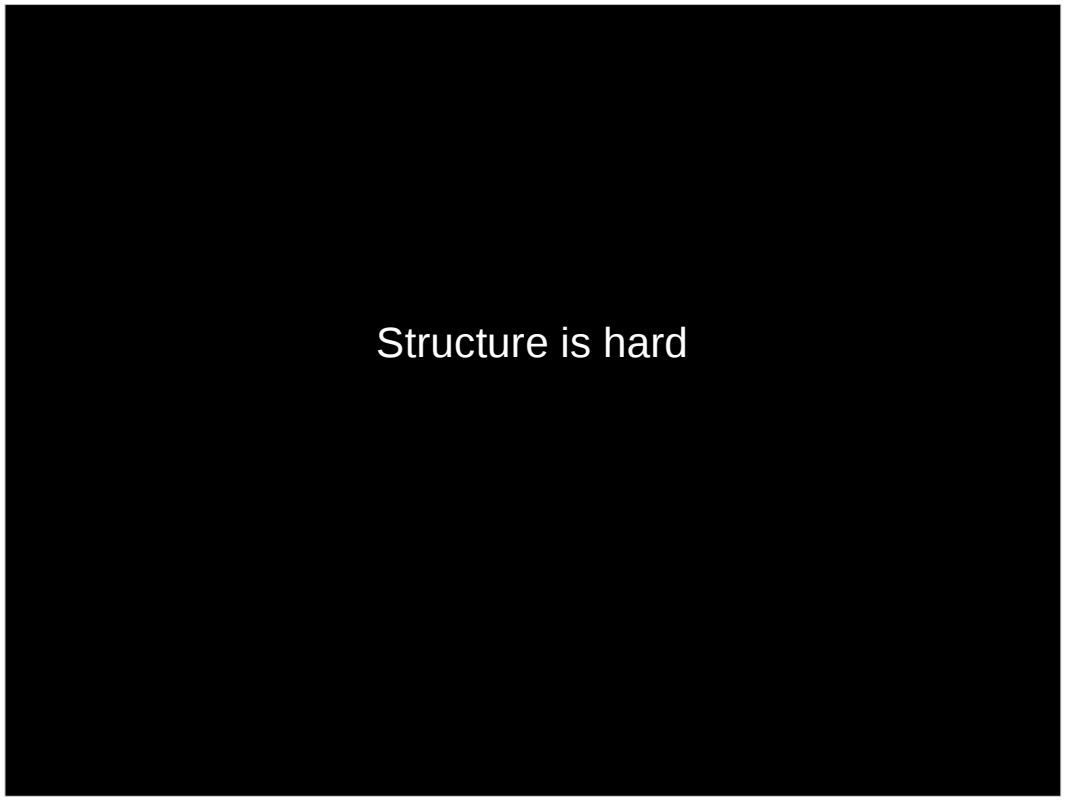
However, because computers can potentially ingest much more data than any single person, algorithms can become better than humans at many, maybe even most narrowly defined task – given the right data.



But 10,000 different possible answers is not that many, if you think about real-life questions.

Producing a whole sentence has many more possibilities, and so can be much more tricky. There are many tasks where machine learning is used to generate text or other structured output. Think for example about google translate. Thats is a pretty impressive system, but it’s no-where near a human expert yet.

Similarly, there are systems that try to answer general questions, or that try to summarize text, and even with millions of examples, they are no-where close to human level.



There are some areas where generating text is easy, like sports and business. There are automatically written business reports or newspaper articles on sports games, and they are often indistinguishable from a human writer. But that’s because they are really, really boring. Player x scored for team Y in minute Z. Team A won by B points.

Or: “The quaterly earnings went up by X, the stock went up by Y, production went up by Z”.

You don’t even need machine learning for that, you can just use text replacement. But people also call that AI.

Knowledge is hard

I already mentioned that most algorithms only learn about very restricted tasks. That often leads to very strange mistakes, and means machine learning can’t solve some kinds of tasks. We often use an amazing amount of knowledge of the world to understand our surroundings, without even noticing it.

The snake classifier hasn’t learned that there is a 3d world. So if I show it a picture of a hand holding a photo of a ball snake, it will tell me it’s a picture of a ball snake. It has not learned the concept of what a photo is or how a hand could be holding one.

This is not part of the task, and unless you explicitly train the algorithm to detect hands, it will not figure out that something is wrong.

Knowledge is hard

Generally, that prevents algorithms from solving any tasks that involves inferences or knowledge about real world objects. Often objects are defined by their function, not their appearance and as long as algorithms don’t have any concept of how the world works, they can not resolve functions or context. I’m not aware of any system that really successfully integrates learning with a knowledge system in any wider application.

Here is an example that people like to use. Everybody of you probably got what's happening in this picture.

A guy is weighing himself, obama puts his foot on the scale, and everybody around laughs.

No machine learning algorithm can understand this. If they are good, they can count the people and give you their gender and facial expression and tell you they are in suits, or maybe even give you a skeleton of where all their limbs are. But to really understand the picture, you need to know how a scale works, and you need to know that the person on the scale doesn't see the foot.

No machine learning algorithm can do this yet, because it requires world knowledge. And no matter how many pictures of scales you show the algorithm, it will never be able to learn how it works.

In principle, if you give an algorithm all of wikipedia, maybe this would be possible, but there is no algorithm that can do this yet.

Classifying with many examples	Works in the real world, sometimes superhuman
Generalizing to new concepts using few examples	“works” in papers
Producing text or complex objects	“works” in papers
Reasoning with world-knowledge	no-one knows

So let me summarize a little bit where we are today. If we have a narrow, repetitive task, and we have a lot of data, there’s a good chance a machine can become better than a human. I think that’s amazing.

And since a while ago, narrow repetitive task not only includes chess but also go. It does not include driving a car yet, but maybe soon.

Creating any complex output, like language, is still very challenging, and only works well in narrow fields.

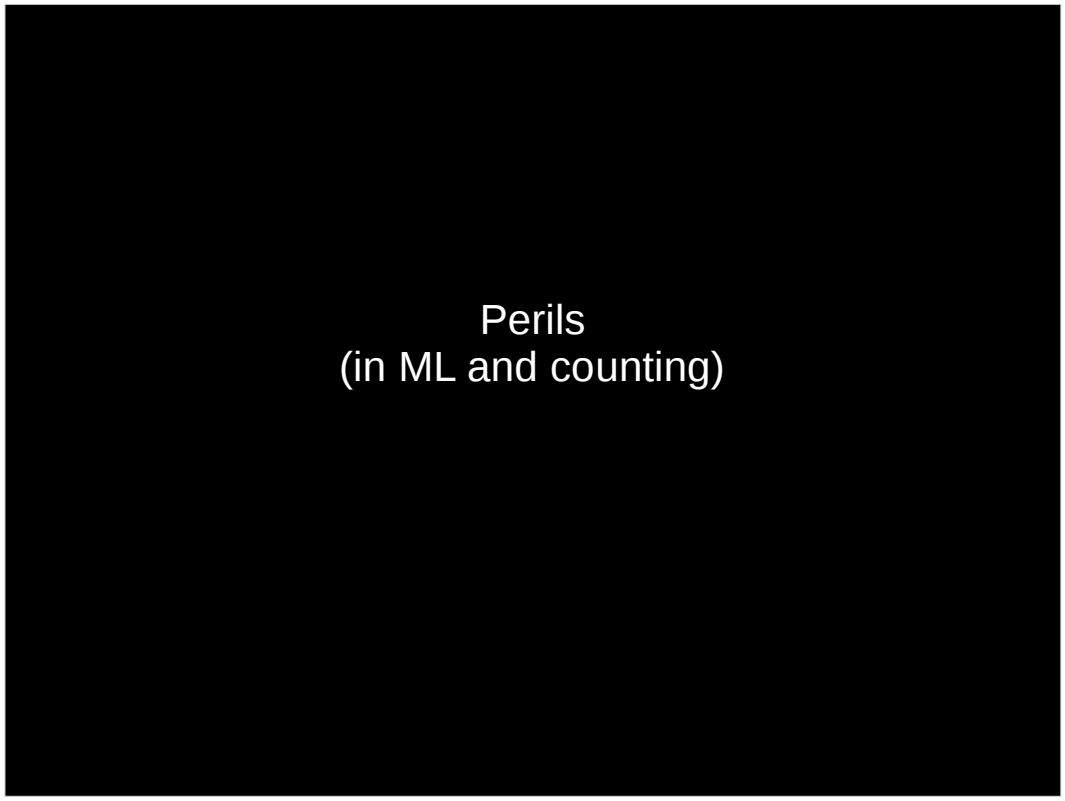
And finally, anything that requires knowledge of the world, or reasoning about function or behavior of things in the world is basically unsolved, and I speculate that it will be for a while.

This is simply because of the very narrow way task and data are defined right now, and the inability of algorithms to interact with the world.

It’s very hard to find out that gravity exists, if you can not hold and drop something.

I thought about talking about applications of ML, but I decided not to. I think you saw a few during this conference. And it’s just impossible to pick them out, because they are everywhere. If you look at any reasonably complex website, there will be many ML algorithms at work. If you look at any major business, online or traditional, they will have ML and Statistics in multiple places of their business logic. If you look into any data driven science, you will see people using ML and Statistics.

So instead of telling you all the places where it works, I’m gonna keep talking about problems and limitations, because that’s the kind of guy I am.

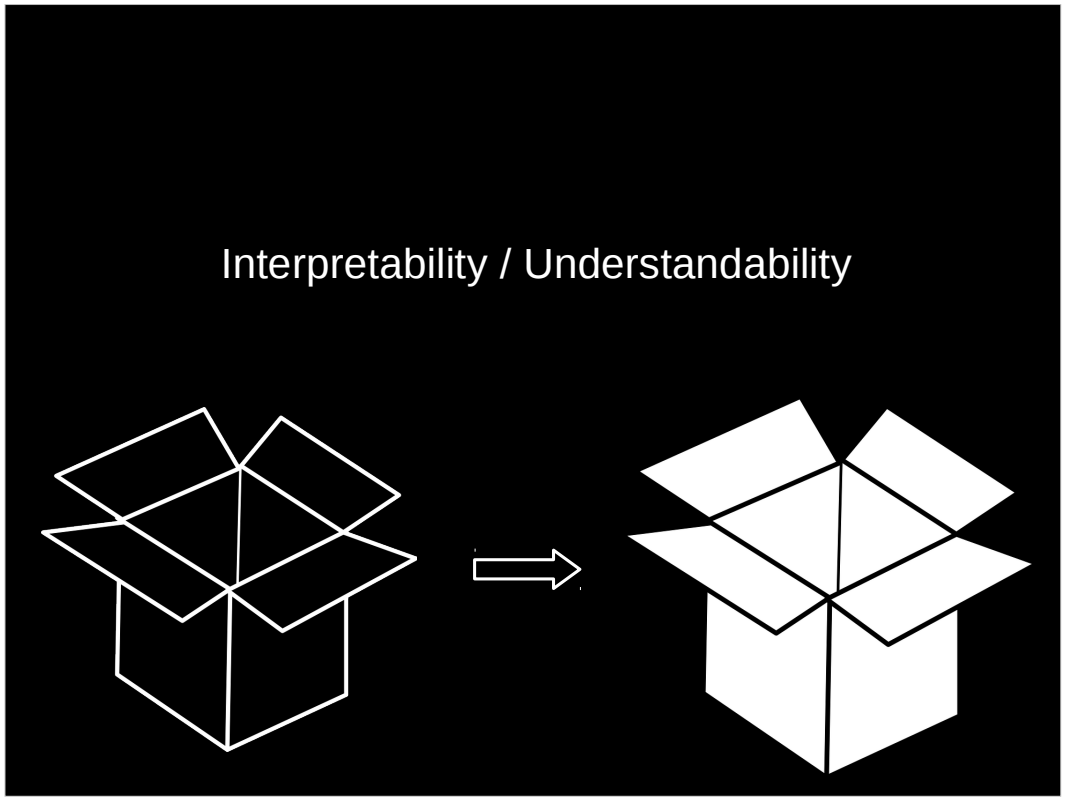


So while ML and statistics are very helpful in many applications applications, there are still some issues that come up in these tasks.

These are shared by most current machine learning methods and applications, and most of them will need consideration for the foreseeable future.

Maybe these are not so much perils, but ongoing challenges that we need to take into account.

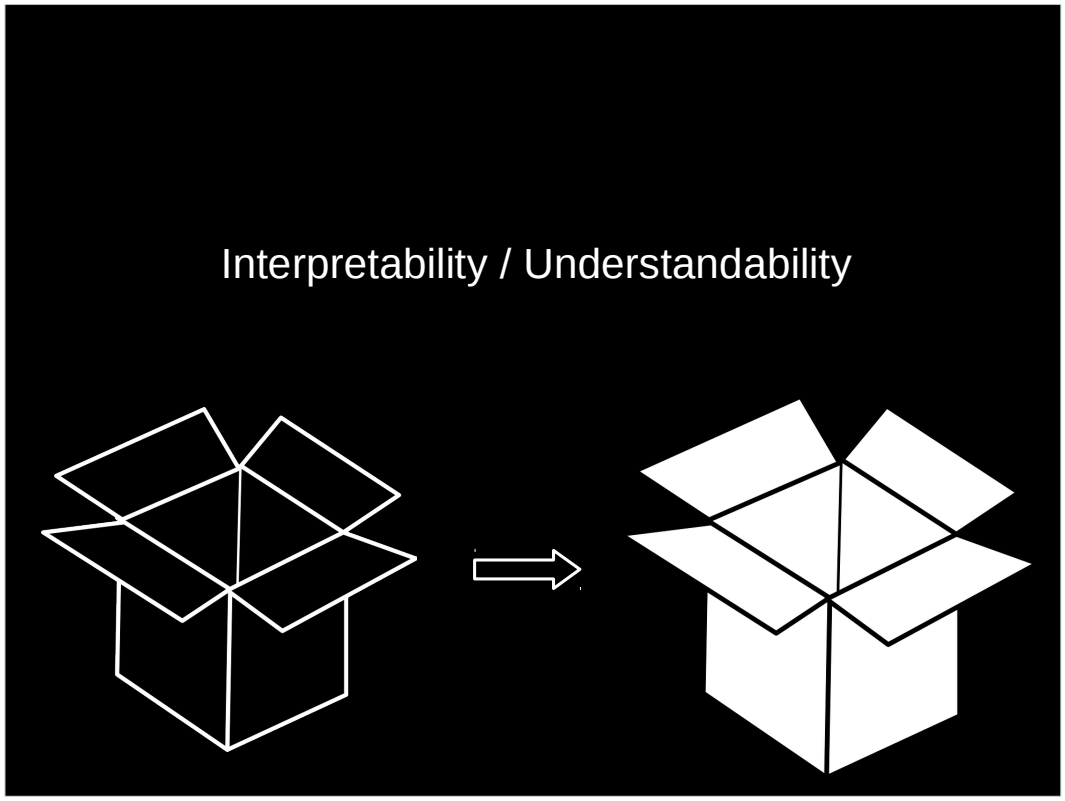
And they apply whether you are using ML, traditional statistics, or even just counting.



One of the biggest issues that people have with machine learning is understanding how the algorithm works, and why a particular decision was made.

The problem is, the more powerful an algorithm the more complex it is, and the less open to interpretation. Any algorithm that is powerful enough to categorize natural images will be very hard to understand and introspect.

For some applications it’s ok to have a black box that provides answers that are usually correct. For other applications it’s crucial that each decision can be introspected. If facebook suggests to tag the wrong friend in my selfie, that might not be so bad. But if tesla misses a pedestrian in front of my car, we might want to understand why, and make sure to fix the problem.

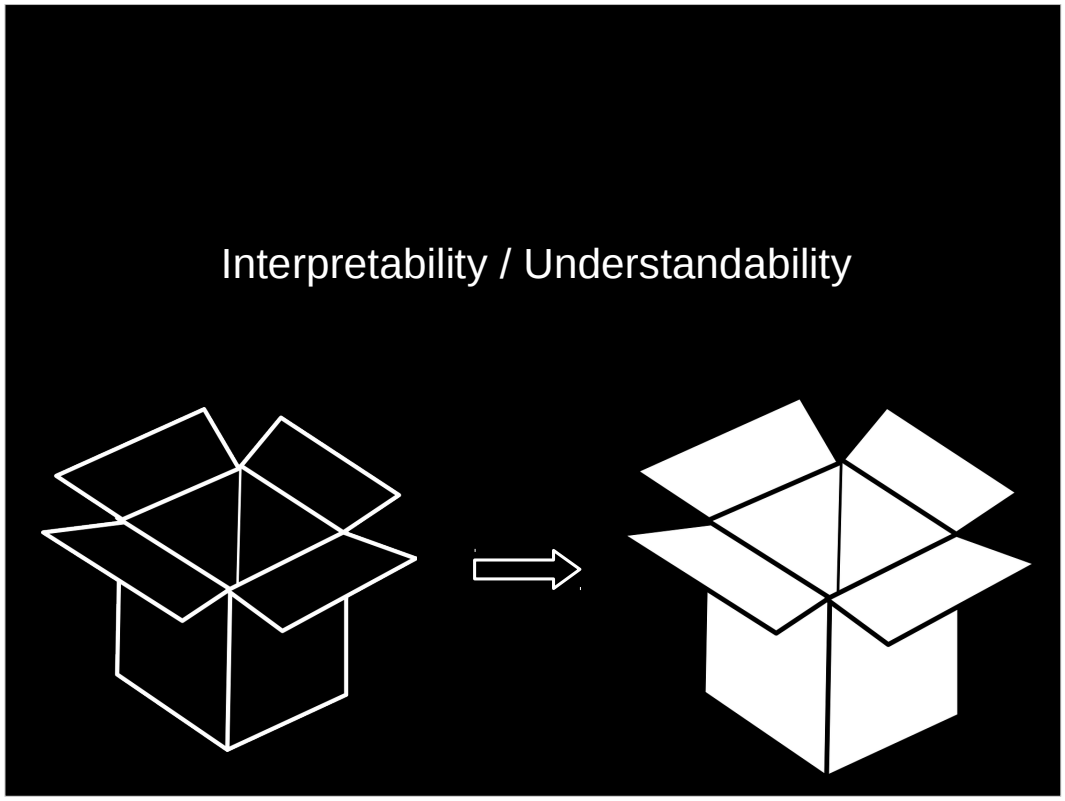


In many situations we might need to explain a decision – or ideally we should. For example if a loan is not granted, or maybe even if a bail decision is made.

Both the financial system and the criminal justice system in the US make wide use of machine learning, but often without the ability to understand why a particular decision was made.

On the other hand, regulation in some areas of finance require very strict explanations of the models, which means that only very simple models can be used, resulting in overly simple models, for example for credit scoring.

Apart from the actual application, there is an important point for interpretable results for the engineers and researchers developing a system: there is currently no real debugger for ML, and if a system doesn’t work, it’s really hard to find out why.



So as I said, the most powerful models are really hard to understand and hard to interpret. But actually, most machine learning models are.

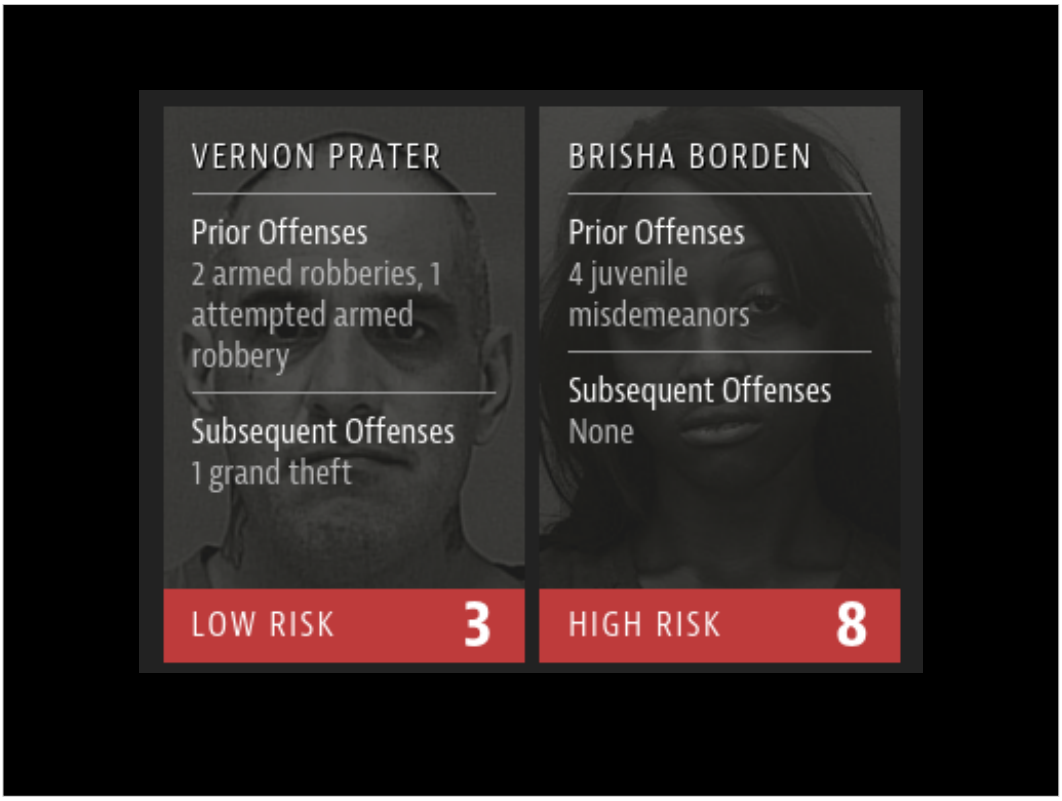
There are two main questions: given the data, what knowledge has the model learned, and why?

And, for a particular example, why did this model make this decision?

You can always write down a formula for why a model made a decision, but for most models, the result can’t be understood by even the most skilled scientists.

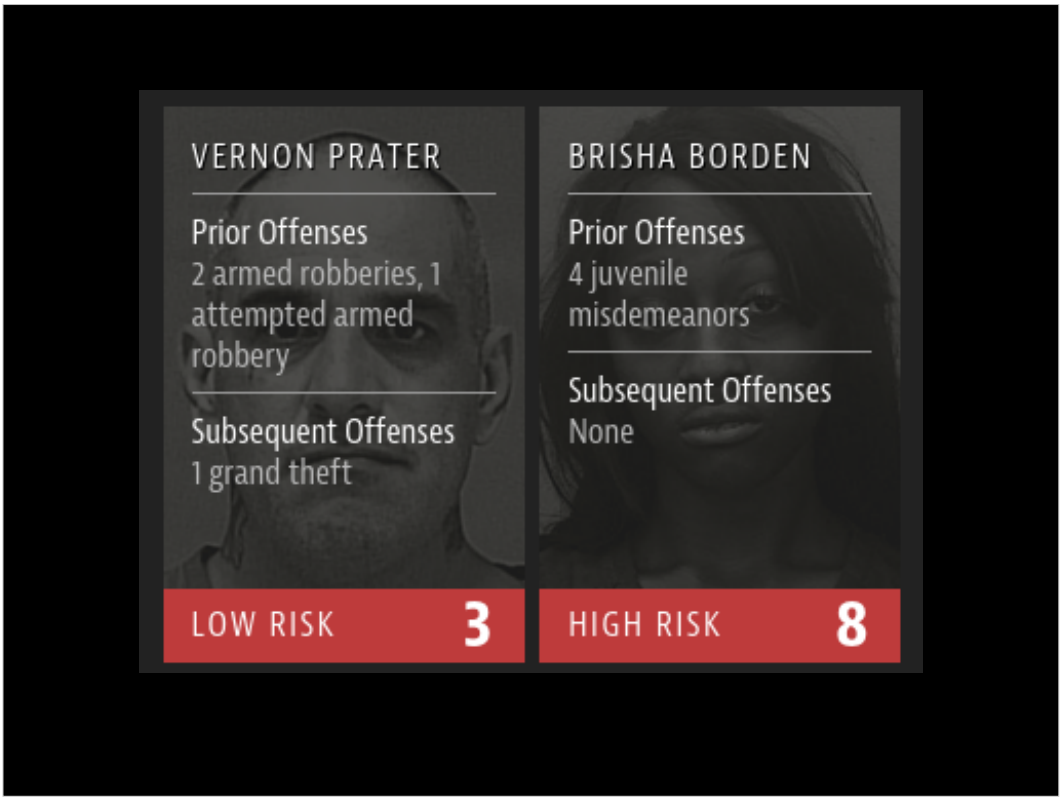
Finding out what the model has learned can be even harder, and understanding how the data influenced what the model has learned is usually near impossible.

People are working on it, but right now, most machine learning is black-box.



Another issue that's closely related to interpretability is bias. Again, if we are making life changing decisions for people based on these algorithms, we better make sure they are not biased. We all have our more or less conscious biases, and ideally we'd like machines to be more objective then us. But often machines learn from past decisions made by humans, and so they inherit the societal biases.

There was this example, featured a while a go in propublica, on bias in the US justice system in assessing the risk of someone to offend again. There were clearly racially biased decisions, but arguably the worst thing here was is not the bias, but the fact that these decisions are made by a company with proprietary algorithm whose workings are not available to the public, or even to the justice system.



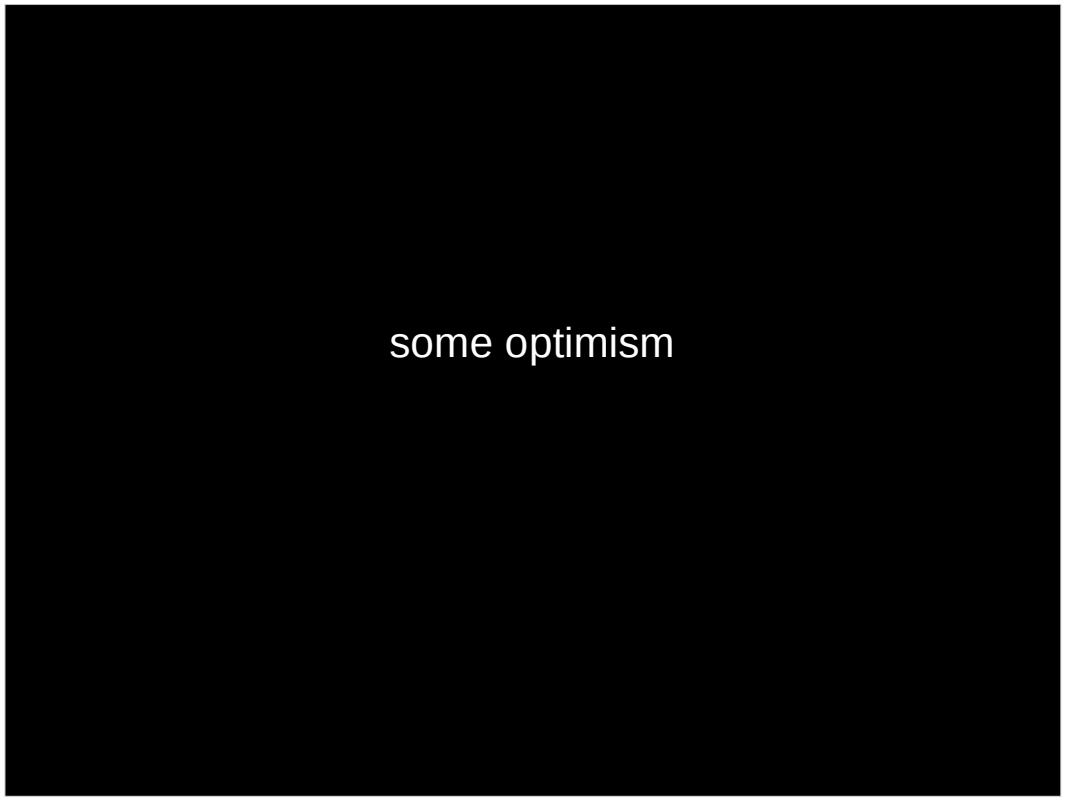
Actually, if you look in their specifications, their definition of fairness is that the probability of recidivism, so offending again, given their score should be independent of race, which is the same as saying that their score is maximally racially discriminative….

Bias in machine learning systems is very tricky, but there's actually a whole annual conference on fairness and transparency in machine learning, called fatml, and you should check out their papers and talks, and probably attend the next one, too.

The great thing with algorithms is that in many cases, if we do it right, we can actually make sure that no bias is present, which we can't really do for people. But only if we agree on what bias means.

By the way, even completely transparent systems, that just count, can easily be biased. You only need a single "if", no fancy machine learning.

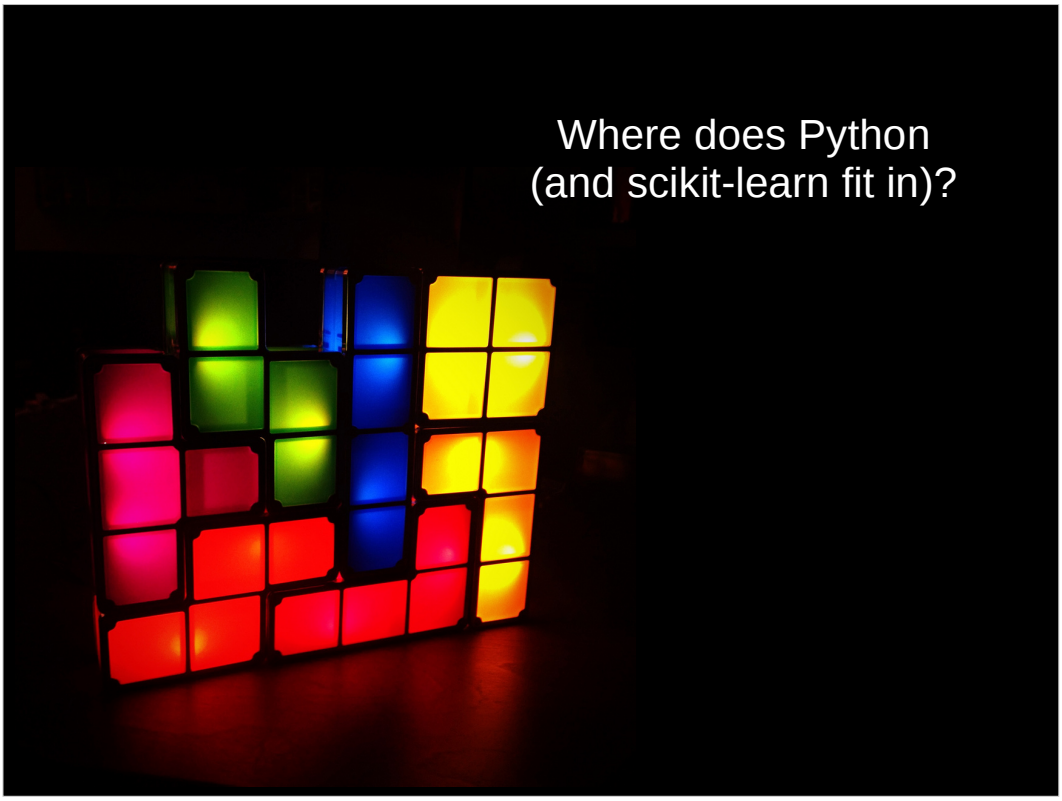
- Another important issue which has many aspects is privacy. By aggregating and learning from data, companies can learn incredible amounts about us, which is concerning to some. On the other hand, privacy concerns limit the application of data science and machine learning in medical applications where they could save lives. So we need to strike a balance between protecting individual privacy, and allowing to learn from aggregate data. There are actually many interesting approaches, like differential privacy and learning from encrypted data that allow the creation of useful models without sharing personal information directly.
- Outside of the medical domain, the issues mostly seems to be settled, though, with users happily giving away all their information. Great for data science, but it’s unclear when that is in the personal interest of the users.



So I talked about all these challenges and limitations of current ML and data sciences, because I want people to have a realistic idea of what's happening.

But actually, I'm really optimistic – I just like to complain a lot. I think ML can already do really amazing things, and if you're not using ML in your company or research, you're probably missing out.

And while there is still open problems with the existing tools, in terms of interpretability, fairness, privacy and of course usability, that only means that we don't have go lie on the beach just yet. Instead we can keep on working on the existing tools, and write more python code, which I actually prefer.

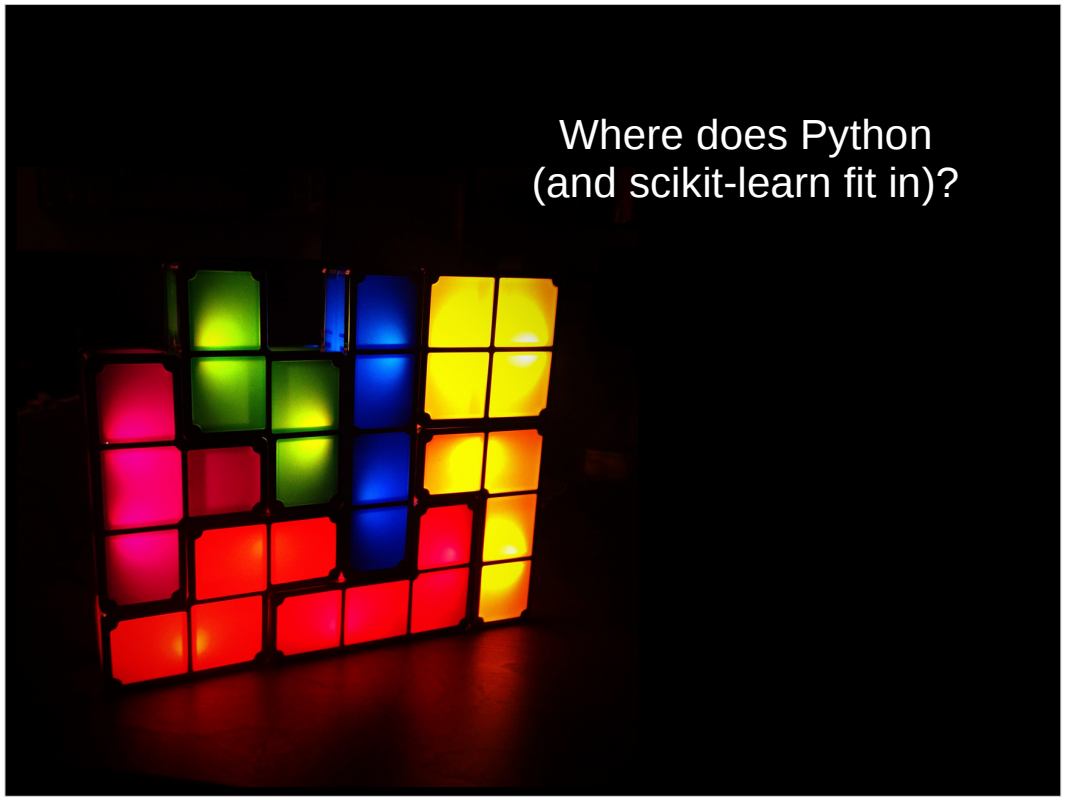


So far, I talked about data science and machine learning in general, and haven't really talked about tools. I left that mostly to Gaël in the session before this.

I think Python is playing an incredibly important role in this area right now, in particular because it provides free, accessible and easy-to-use tools, with numpy, scipy, matplotlib, pandas and scikit-learn at the core.

There is also incredible work being done in keras, a library for deep learning, which allows putting together systems for learning from image, text and audio more simply than ever before, but also providing extremely powerful tools.

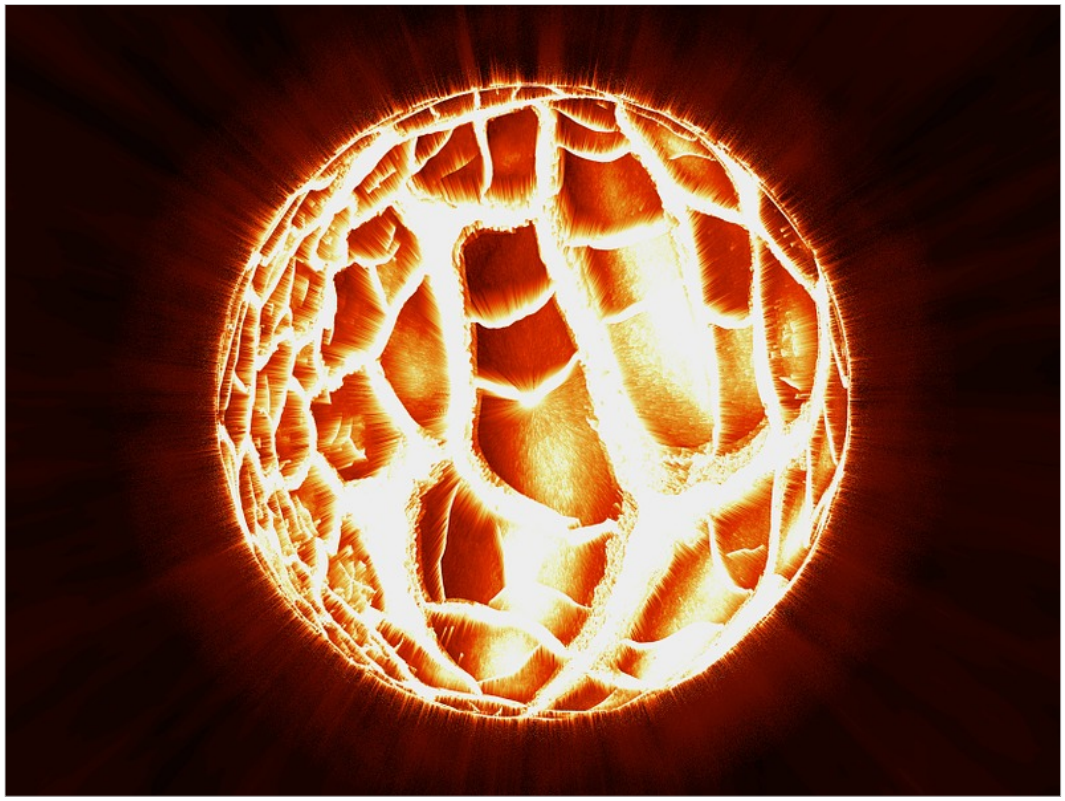
These core libraries are all focused on in-ram computations on a single machine. There are other tools coming up for distributed computing, like dask, and the spark interface pyspark.



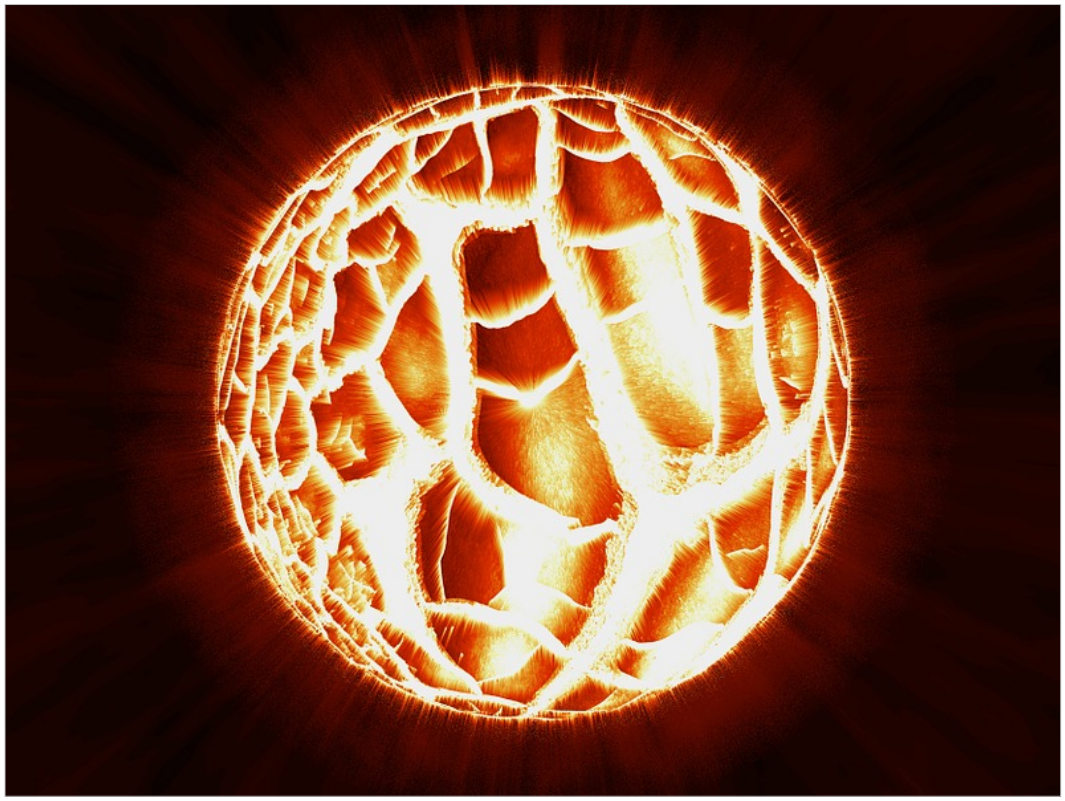
But as I alluded to earlier, I think a lot of very important data analysis is not “big data”, but instead only a couple of gigabyte or even megabyte.

And this is the use-case I’m most excited about, because we have the right abstractions and tools in place, and it is possible to solve so many interesting applications.

It would be nice if scaling out would be a bit more smooth, and you wouldn’t have to replace your whole stack once your data becomes bigger, so I think this is an important problem to work on. But I feel there is a lot more we can do for the “simple” case, and we shouldn’t all follow the distributed hype.



- I mentioned some projects that I think are core, but obviously there are many more data science and machine learning packages out there, with many specialized solutions. But I think another important contribution of these core libraries is that they provide a set of reliable algorithms and provide a shared basis from which we can all work.
- I remember working in C++ a couple of years ago, and you needed to pick you array library, and then write all your IO from scratch and then pick a GUI framework to create a plot and so on.
- The pydata community has mostly settled on interfaces and data structures, and we have algorithms that people can rely on.
- This removes a lot of friction for any new project, or for any new collaboration, or for reproducing an existing analysis or system. I think this common foundation is one of the most valuable things the PyData community has.

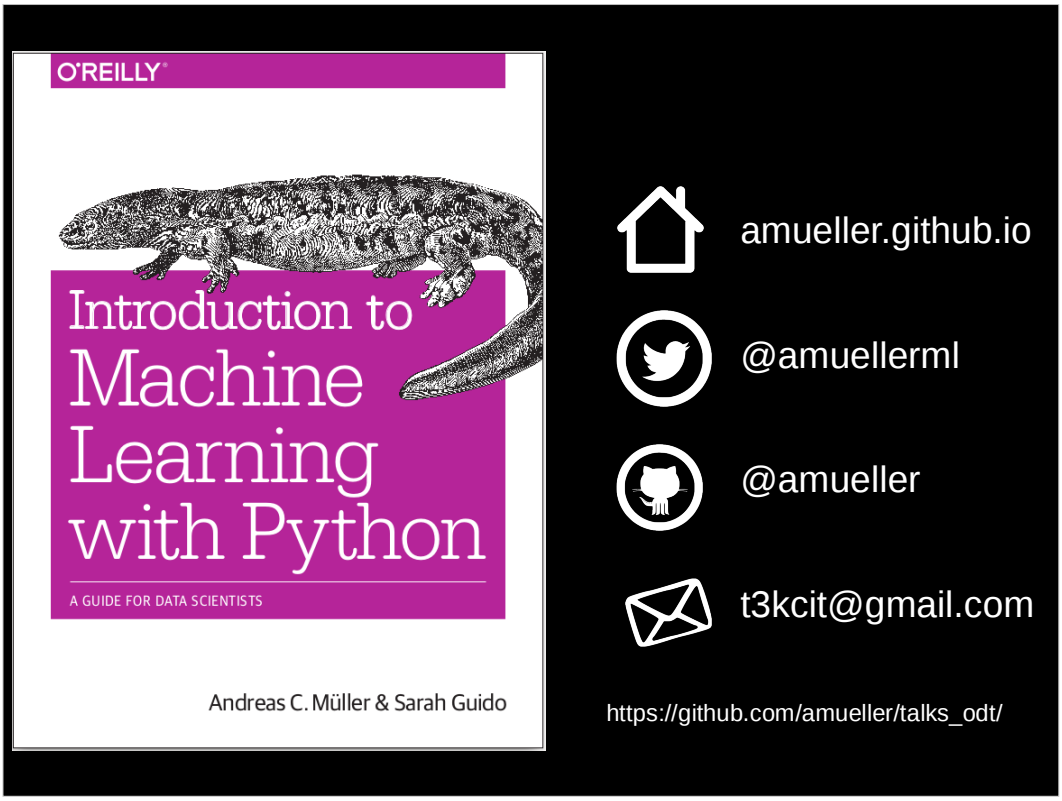


To keep this core alive, there is a lot of work to do, though, and we must constantly curate and iterate.

There is still some movement in the data structures, discussions about pandas1.0 and 2.0, matplotlib recently changed all its styles , and scikit-learn is constantly growing, and it takes a lot of effort to keep this all in sync. We also want to keep the quality bar high, as we have been doing so far, while still allowing for the ecosystem to grow organically.

But I keep being impressed by this community, and by the energy and enthusiasm, and I think we are on the right track to make sure that the ecosystem keeps thriving.

So that’s all I had for today. There’s only one more thing I wanted to mention.



Buy my book. It's about machine learning. It's written for programmers, and should be a pretty easy read for anyone that knows a bit of numpy. I avoided adding to much math, or really any math at all. If you want math, there are many awesome books to check out. This one is more about coding and how to get started with ml in python…. Which, you know, is why that's the title…

It can also serve as a guide to how to use scikit-learn effectively.

I hope my rant today was a bit informative, or at least entertaining. And a big shout out to the organizers for putting together such a great conference, and thanks again for having me!