It's not who has the best algorithm who wins, it's who has the most data.

\- Andrew Ng

Managing Machine Learning Projects with Google Cloud

Offered by Coursera.

Previously this course was known as **Machine Learning for Business Professionals**.

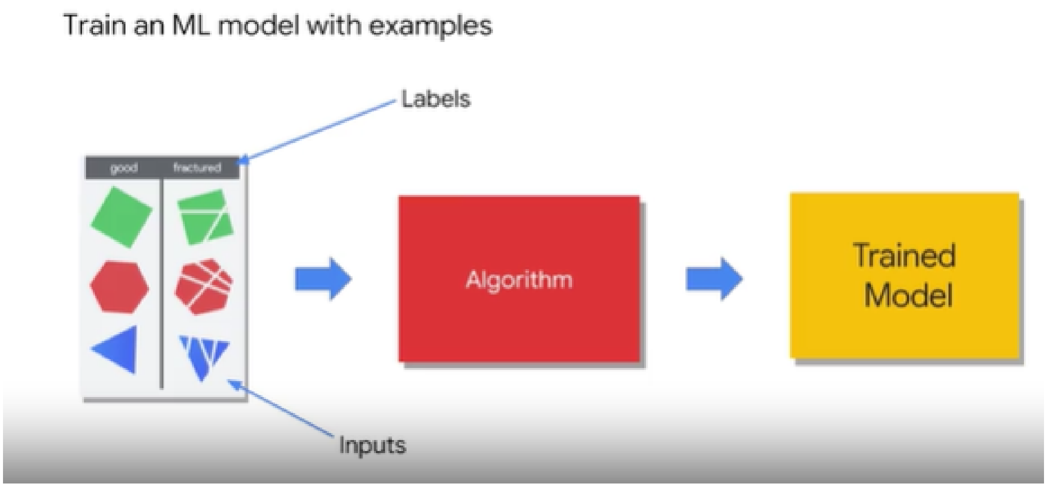
Syllabus Week 1: **What is ML?** 1 - 11 | Week 2: **Employing ML** 12 - 26 | Week 3: **Discovering ML Use Cases** 27 - 37 | Week 4: **How to be successful at ML** 38 - 47

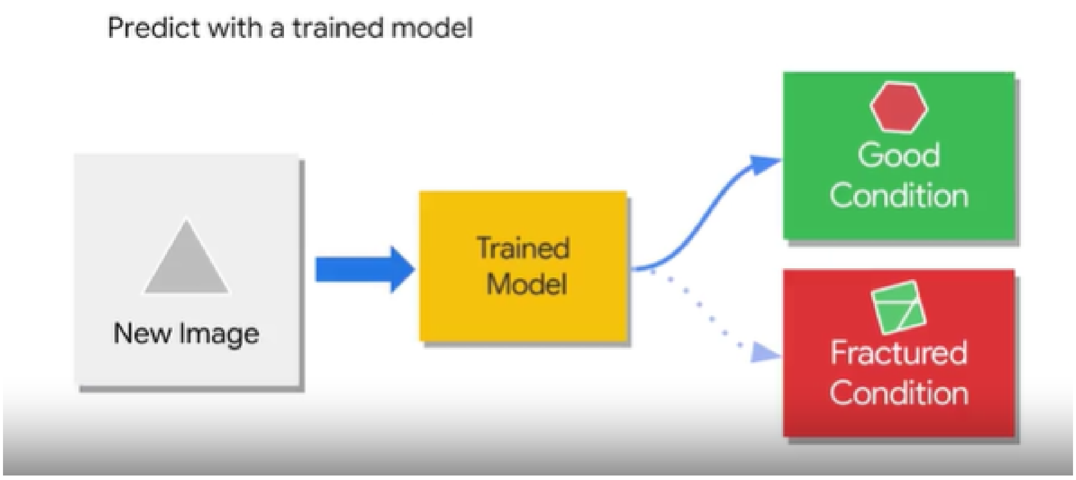
Teachers Carolyn Ujcic, Lak Lakshmanan, Valentine (Val) Fontana

The formatting guidelines of this note is given on the end page.

1. Def: **ML (Machine Learning) is a way to get predictive insights from data to make repeated decisions.**
2. Backward looking vs forward looking data:

- Most analytics in your business is probably backward-looking, where you look at historical data to calculate metrics or identify trends.
- Predictive analytics are forward looking.
- A business analyst's reviews a report and sees that the demand is increasing for a specific product in a specific region. The analysts then suggests a new price for that product in that region to increase profit. Now, the business analyst is making a predictive insight, but is that scalable? Can that business analysts make such a decision for every product and every region? Can they dynamically adjust the price every second based on how many people want that item at that very instant? **In order to make decisions around predictive insights repeatable, you need machine learning.**



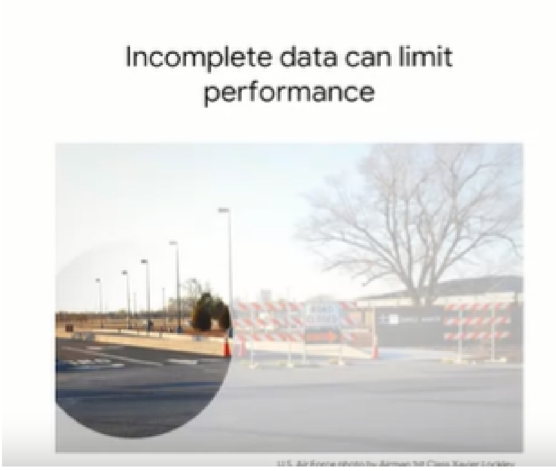


standard algorithms - these algorithms exist independently of the use case.

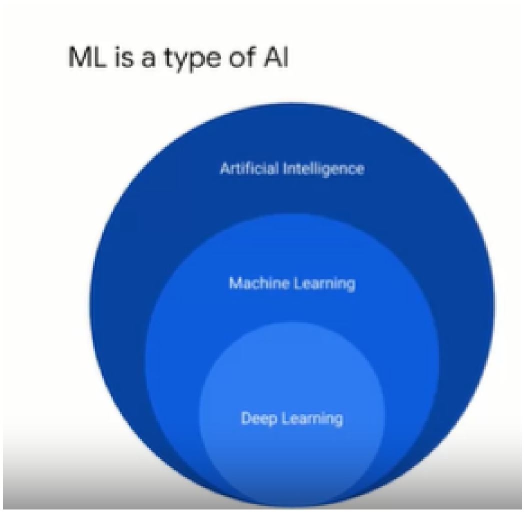
3. ML models aren't programmed with traditional logic, like if, this, then, that. Instead, you program them by giving them examples and letting them come up with the

logic. Models will be as good as data.

4. Qualities of good data: broad coverage, clean & complete. The more wrong data (dirty data) you have, the more correct data you will need to provide to counterbalance.



5. Difference b/w ML and AI



▼ What are expert programs?

AI programs designed to use if-then rules to emulate human inference and decision-making. They work well when you already know the logic. To create them, you first collect all of the domain of knowledge like a table of diseases and their symptoms. Then you convert this knowledge into if-then rules. The problem with these experts systems is that there aren't that many use cases where you can fully capture all the logic. In medicine, textbooks capture a lot of information for a doctor to make a diagnosis, but they don't capture everything. However, one area where it can be successful is in process control. In Japan, their trains have excellent on-time performance. Researchers in Japan have created expert systems that will adjust train schedules when there's a disruption in the same way the best human operators would. Another challenge with these systems is that it is expensive to code up that logic in the form of programs. Many times the logic is just a heuristic, and you spend years maintaining and fixing problems in the heuristic, as you discover the cases where it doesn't work. Why is ML a better tool than expert systems in many cases? Well, let's say you have a record of customer browsing activity on your shopping site, but you can't describe the logic for determining whether a customer will buy. For example, your logs might indicate that some small portion of people who looked at sunscreen ultimately bought sandals, and none of the people who looked at towels did. With ML, the model can learn based on customers' activity, how likely they are to buy sandals. Many business problems like this one are a better fit for ML because it is hard to articulate the logic. But we do have examples of what actually happened historically.

6. Traditional BI dashboards and data analytics - historical data, (vs) predictive analytics - unknown data.

> ML works for making repeated decisions, not occasional ones. As you think about applying ML in your business, I want you to remember this part of the definition. It is often where you will get the most business value. That's because many of the benefits of ML, including scalability and efficiency, are only realized when you use it repeatedly.

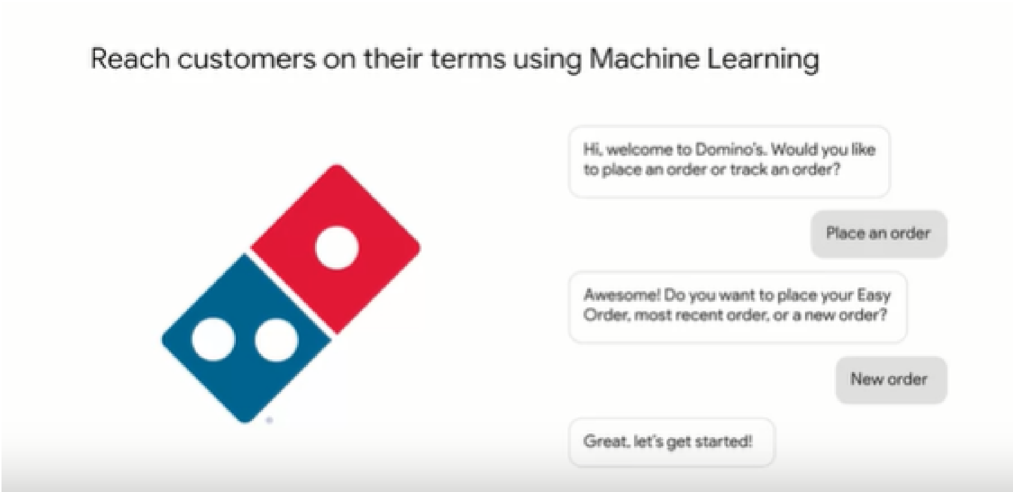
Key ingredients for a recipe for ML: data, standard algorithms, predictive insights, and repeated decisions.

Where can we use ML? Quality control of a Space Shuttle? [less data] What about an annual sales forecast? [not frequent, also many statistical tools present] What about trying to model the likelihood that a truck might break down? [infrequent] But what if I told you that each day, mechanics decide whether they should inspect vehicles for issues to find if the truck is likely to break down? Then it becomes a frequent decision? We can use ML here.

7. Normal life eg: Do you get less spam in your inbox, or have you used a voice assistant to call your mom? Or search within Google Photos to find the best picture of your dog, or you might have watched that YouTube video that was recommended to you on the YouTube homepage. Think about your business processes. Where do you have major pain points? How might ML help?

    - Aucnet built a real-time car image recognition system powered by Google Clouds ML technologies which reduced the time it took to list a car for auction from 20 minutes to only two to three minutes.

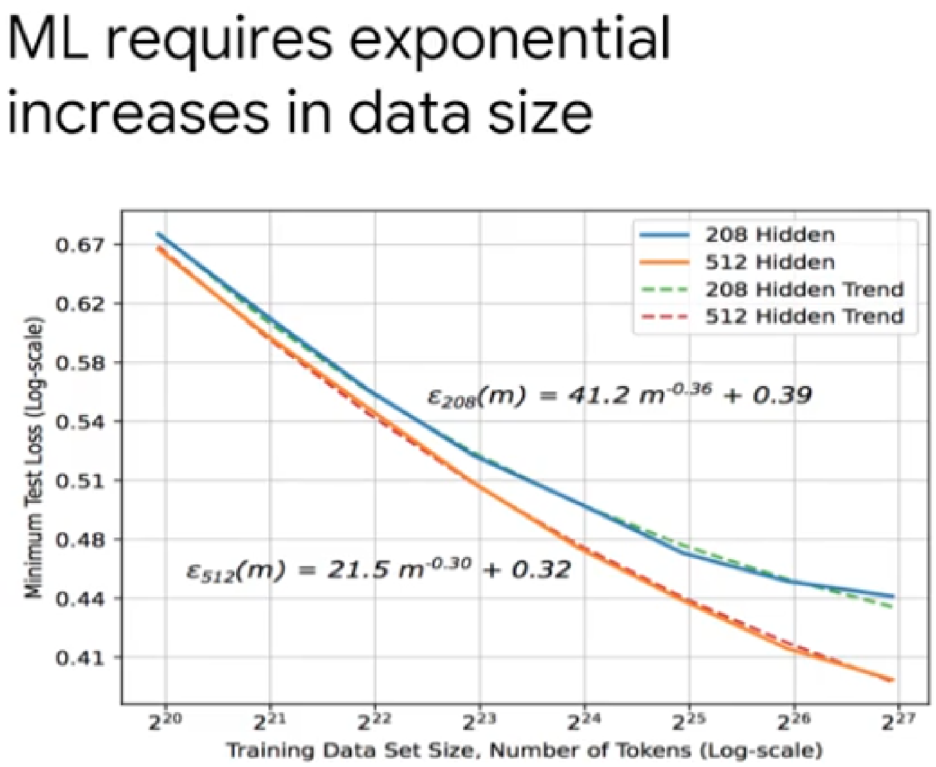
- Using Google's Dialogflow chat bot technology, Domino's was able to improve the ordering experience even more for their customers. The bot can now handle more difficult or complex orders from customers.



8. Lab: Creating a Pizza bot with Diagflow and Cloud Image Classifier with Auto ML (Vision API).

   The ***Vision API*** is an API that uses machine learning and other Google services to extract information from images. It can do a variety of things, whether it's detecting the presence of certain classes within the image. Detecting and extracting text, determining if the image is safe to serve, or detecting the presence of corporate logos. Unlike the Vision API, ***AutoML*** Vision can be trained to use whatever classes you need for your business use case. You provide the labelled datasets.

9. Machine learning has been around since the 1970s, when it was used in fields such as astronomy. So why the popularity now? PART 1:



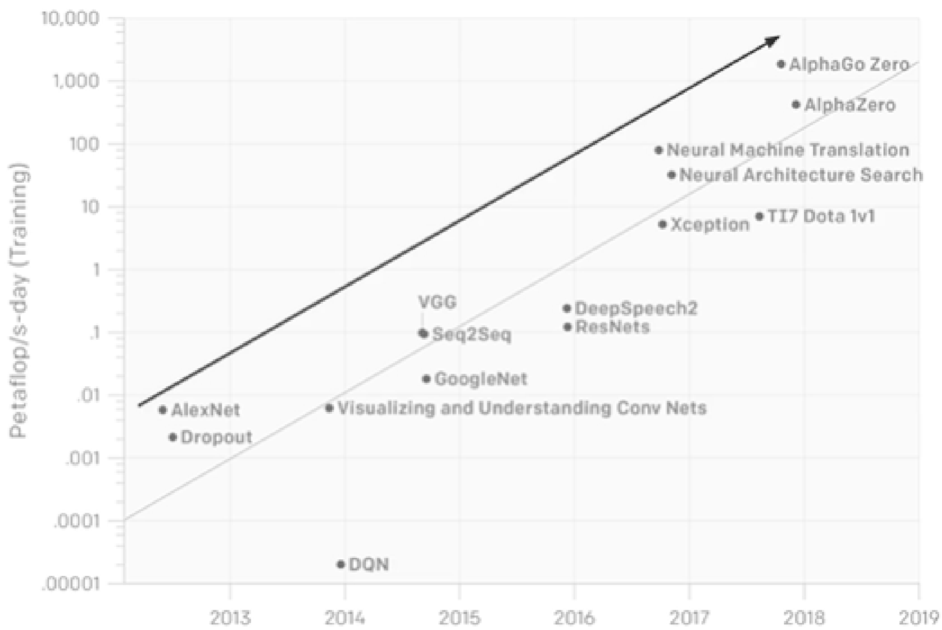
What made it ready was the convergence of 3 factors:

- Increasing availability of data
- Increasing maturity and sophistication of ML algorithms
- Increasing power and availability of computing hardware and software

The cost of sending data over the Internet in the US dropped from $1,200 per megabit per second in 1998 to $75 in 2005, that's a 94% decrease. During the same period, the cost of storage also fell significantly. Not just for disk storage, which is the sort of storage necessary for retrieving archived information, but also for RAM memory, which is the sort of storage necessary for big and fast applications. In 2005, the cost of 1 gigabyte of disk storage was 0.2% of what it was 10 years earlier, and the cost of RAM was less than 0.1% of what it was 10 years earlier. Then in 2007 the iPhone launched, bringing the desktop web to phones. Users started consuming and generating data constantly.

10. PART 2: There are also changes related to algorithms. There have been lots of ML algorithms before. ***Linear regression, decision trees, genetic algorithms, support vector machines,*** just to name a few. One in particular has revolutionized machine learning recently, neural networks. But for complicated mathematical reasons, those deep layers of neurons didn't work in computing. So people used ***shallow neural networks***, which weren't very powerful. But then came advances in what we call ***deep neural networks.*** Many advances were made by Geoff Hinton at the University

of Toronto. He and his team of students got deep learning to finally work, with the help of many other researchers. These deep neural networks are examples of ***deep learning,*** which propelled ML forward even further. One key advance was a special type of layer in a neural network called a ***convolutional neural network***, or ***CNN.*** Distributed computing allowed large computing jobs to be split into manageable pieces. But distributed computing is subject to diminishing returns because there are network, speed, and reliability issues when computers have to talk to each other. GPUs, which are built as graphics cards for rendering and games, turn out to be really good at accelerating machine learning computations. Andrew Ng demonstrated in 2006 that GPUs function as ML accelerators. Between AlexNet, which started off the deep learning revolution and Neural Architecture Search, which powers Google's AutoML, the computational cost of training grew 100,000 times greater. That's because when you have more layers in a deep learning model, you need more data and your compute needs become more expensive.



Compute performance has hit a plateau (disobeying ***Moore's Law*** because in the past three years, growth has slowed dramatically as manufacturers run up against fundamental limits). One solution is to limit the power consumption of a chip. You can do that by building what are called application-specific chips, or ASICs. Google designed new types of hardware designed specifically for ML. The tensor processing unit, or ***TPU***, is specifically optimized for ML. And it has more memory and a faster processor. Google has been working on the TPU for several years. And has made it available to other businesses.

Because of all of this technology, new hardware, distributed computing, new algorithms, it became feasible for more businesses to use ML, but it still wasn't easy. Part of the problem was that few ML software frameworks were designed for industry scale usage. If companies wanted to do ML, they would need the expertise to create their own frameworks. And that was a steep cost, especially because ML talent is still relatively rare. But over time, more and more ML frameworks are becoming available and are making it possible to use ML at high levels of abstraction. One such framework is TensorFlow. But even with frameworks like TensorFlow, implementing a custom model requires technical expertise. That's where cloud technology comes in for businesses that want to run ML models they already have in the cloud, but for whom the economics of owning the infrastructure doesn't make sense. Additionally, TPUs scale linearly, so you can train faster by running on more cores, which is another benefit of the economics of the cloud.

Reasons to do ML in the cloud

- IaaS allows businesses to rent instead of buy
- TPUs allow for linear scaling
- PaaS lets businesses think at a higher level of abstraction
- SaaS lets businesses concentrate on delivering value to their users

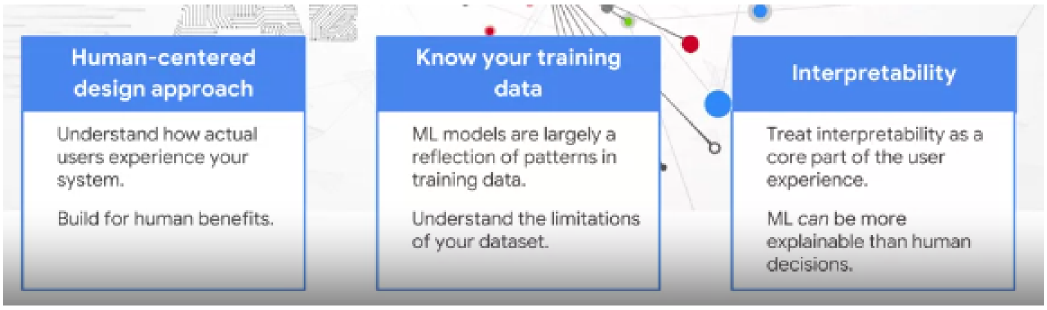
Reasons you may have to do ML without the cloud

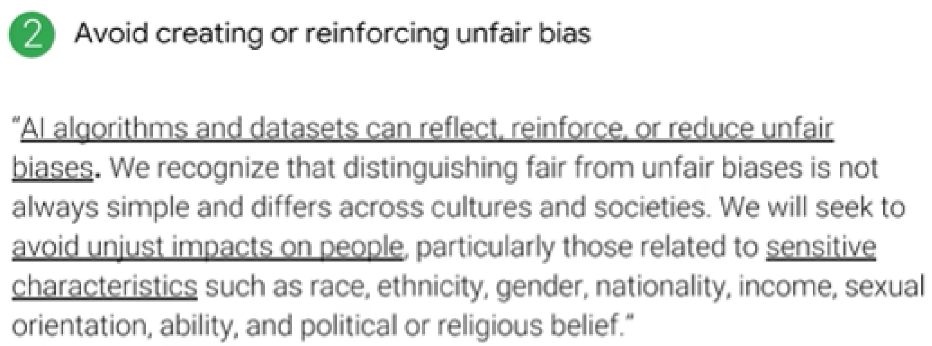
- Data locality requirements
- Need to make predictions offline

A common paradigm is to run a fast but not great model locally. But use the cloud when a more accurate answer is needed. Thus, the cloud is recommended for most ML use cases.

11. Using AI Responsibly - Google Principles

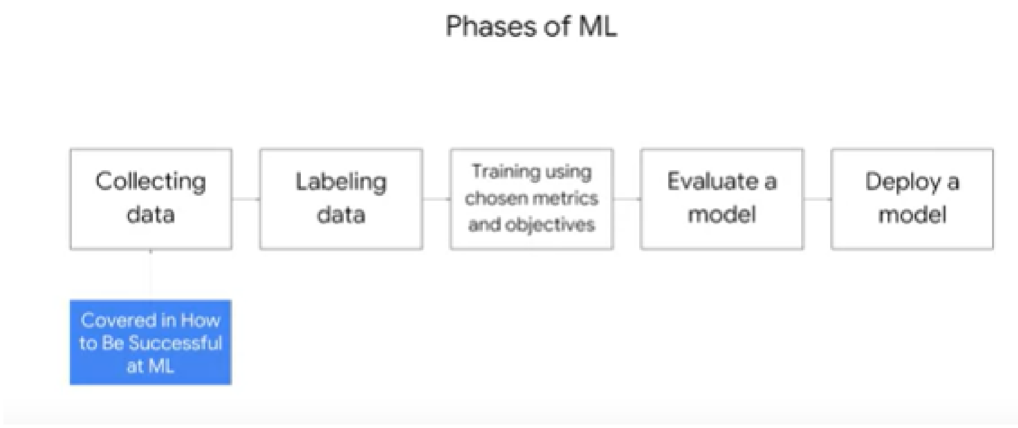
Some ways to put these principles into action:





[2 from Week 2]

12. Machine learning involves learning from examples. An example consists of labels and features. A label is just the true answer for an input. If you're trying to predict how much a customer in a bank will deposit in a given year, the amount that was actually deposited by that customer in that bank in a particular year is the label for the example. When we train a model, we compare the prediction for that example against the label for that example and use that information to update the model. [Phases of ML 13 - 19]



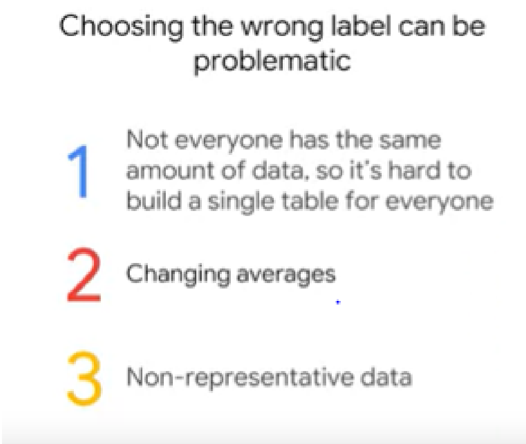
13. Labels can be either numbers or they can be categories. The amount that somebody deposits in a year, for example, that's a number. Whether that amount is high or it's low, that's a category.

Regression vs. classification

Label Data Type	ML Problem Type	Example
Categorical	Classification	High vs. Low
Numerical	Regression	Dollars

The features are the inputs to the model or a distinctive attribute. There are other forms of machine learning that don't require a label, but they're less mature. In the case of images, the pixels of the image might be enough as features.

14. Choosing the wrong label can cause problems.



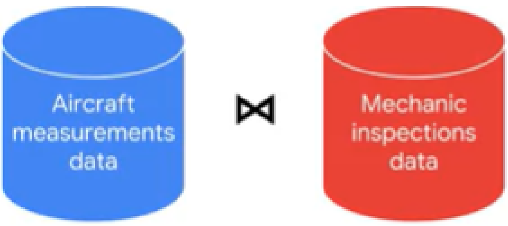
Aucknet

- feature - historical data (pictures) on condition of Tyre
- label - H.D data on price sold for
- regression problem

Youtube

- feature - country, how many of the people who watched this current video went to watch the next video, aggregate history of all of it to one billion plus users
- label - next video (H.D) because we identify videos by a video ID, the video ID is a label.
- classification problem [see: Week 2> ML Terminology in Context]

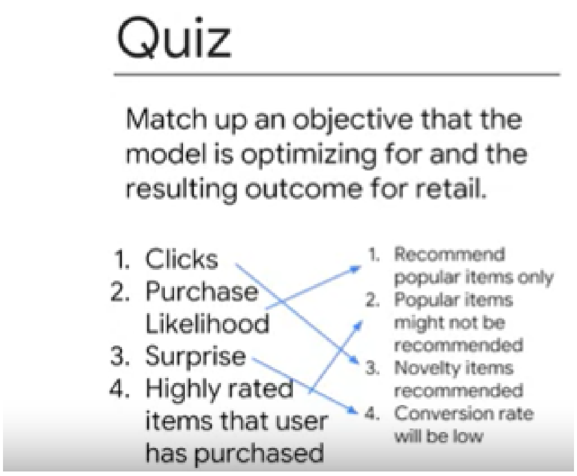
15. Acquiring labelled data - need to have all required data in one place.



AutoML offers labeling service.

16. Thus summarizing,

17. Let's say you're running a retail website, where you're selling lots of different items. surprise: items that user might not have otherwise bought, clicks: novelty.



There is a difference between popular in the sense that things are rated highly and popular (4) in the sense that people buy them a lot (2). The hardest part of a recommendation system is not the algorithm because the algorithm is standard. It's not the data because you probably have purchase transactions data already. **In this case of a recommendation system, one of the hardest parts is getting the objective right.** In reality, the objective that most retailers pick will be a weighted combination of all these factors. The retailers will typically tweak the relative weights as the company strategy changes. There is no right answer, it's all a trade-off. So discuss what the right objective is with your business stakeholders.

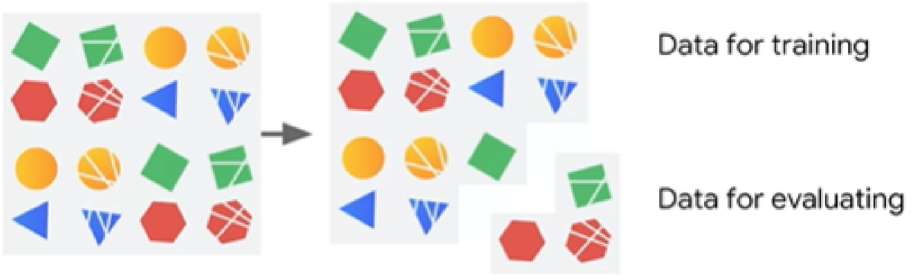
18. When we train a model, we compare the prediction of that model against the label and use that information to update the model, and that part is basically just applying the standard machine learning algorithm to your data. Although it's possible to train a machine learning model only one time, t**he best-performing machine learning models are trained continuously** by taking a trained model from earlier and retraining it with new data. That's because the world is always changing and only recent data will have a record of these newer changes. If you've built a workflow where humans label the data, then you've done half the work already. The next step is connecting their output to the model. Not only that, you set yourself up for success by keeping humans in the loop. Many people assume that this isn't unnecessary, but **for great machine learning, it absolutely is necessary to have humans in the loop.**

Here are some standard algorithms to choose from for ML

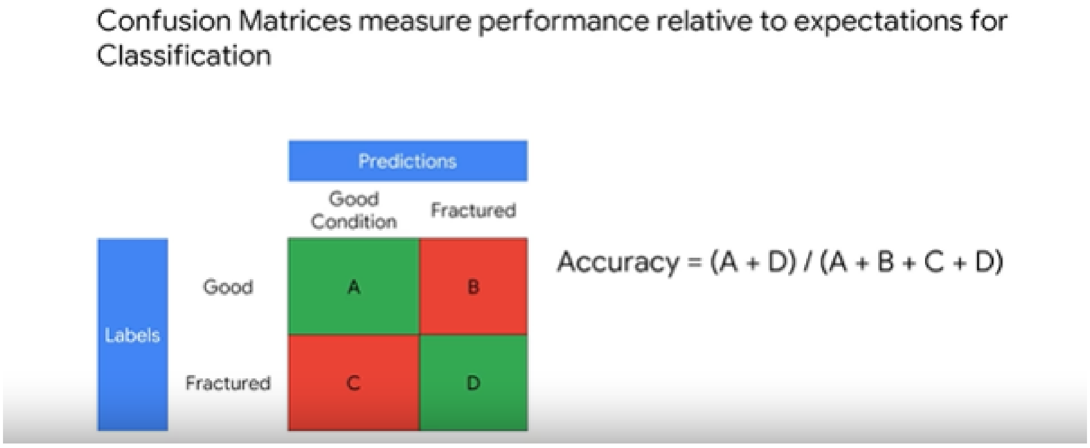
ML task	Standard algorithm
Image classification	ResNet (originally by Microsoft Research, an implementation open-sourced by Google)
Text classification	FastText (open-sourced by Facebook Research)
Text summarization	Transformer (open-sourced by Google)

For business problems, mostly you will probably use standard algorithms. People create new algorithms all the time and they compare them with previously known algorithms, but the improvements are very, very minor.

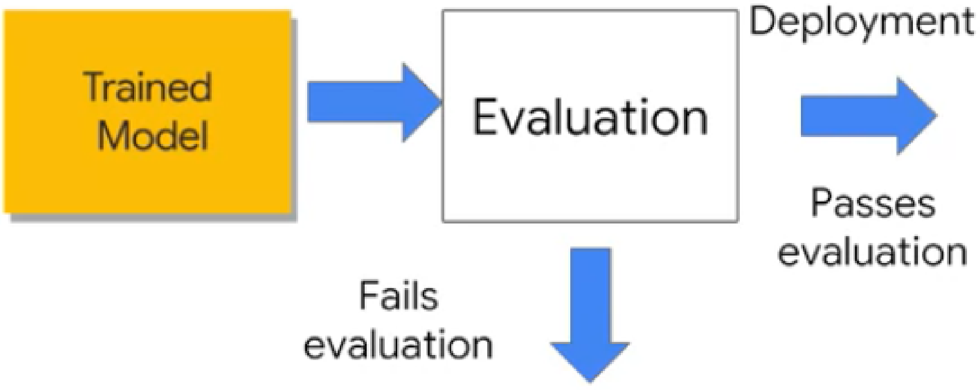
19. How to get an idea on how good the model is?

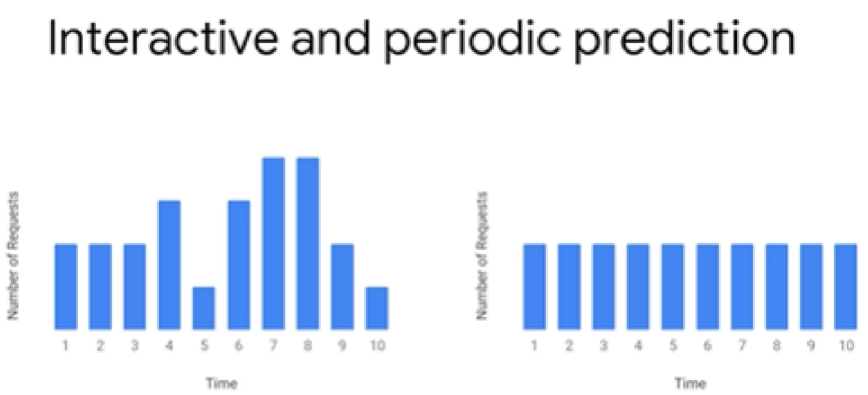


We don't show the model all of the data that we have (why? original training data can't evaluate - model memorize), instead we split that data into two parts. Maybe we show the model 80 percent of the data (***training data***), and have it learn from that, and then we try out the model on the remaining 20 percent (***test data***). Create a ***confusion matrix.*** Accuracy can be computed simply by finding the fraction of times that the model is correct.



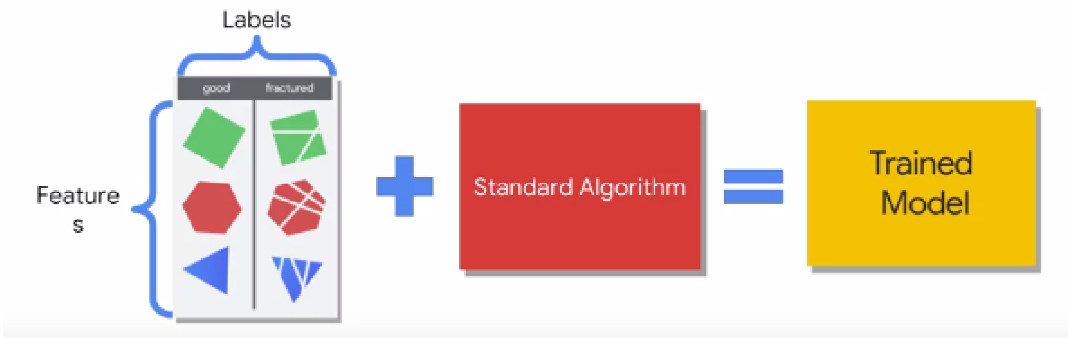
Usually accuracy is enough, but in some cases, you will be creating a machine learning model for rare occurrence. For example, you may be building a model to identify credit card fraud. The vast majority of transactions are not fraudulent. So, even if the model says, nothing is ever fraudulent, it'll only get a very small fraction wrong. The accuracy will be something like 99.999 percent. In such cases, we use other metrics computed from the confusion matrix. Matrix that emphasize the performance on the rare stuff.



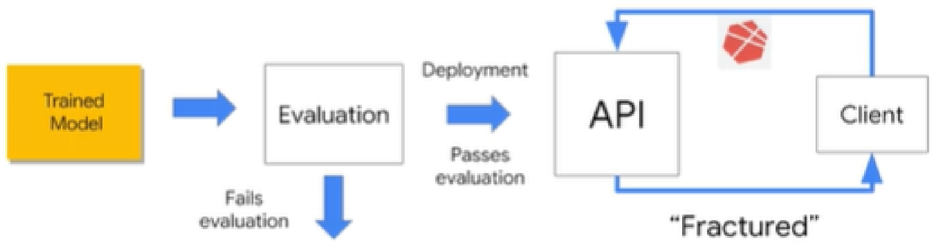


20. Make different models [combination - diff features, complexity (neural n/w layers)] and select what works best.

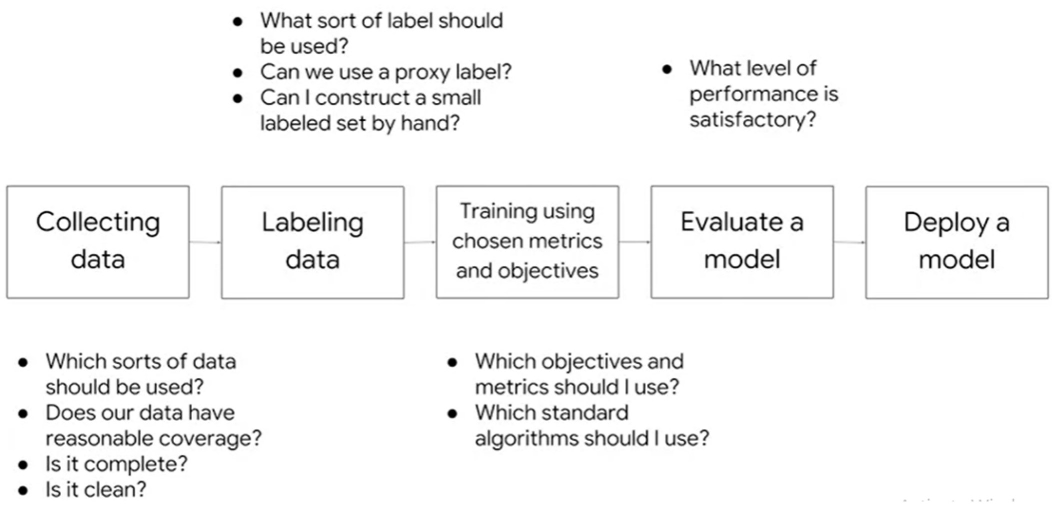
21. The first stage of training looks like this.



The process of evaluation & deployment looks like this.



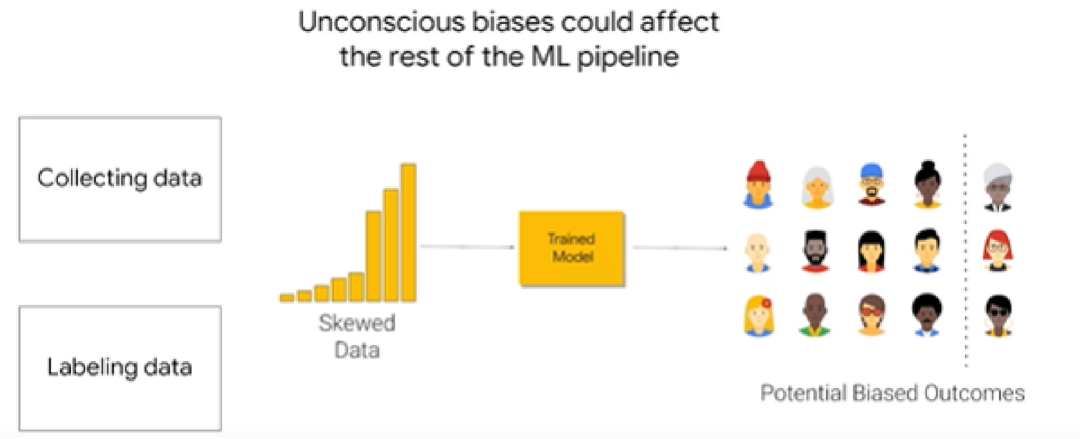
The more complete process looks like this.



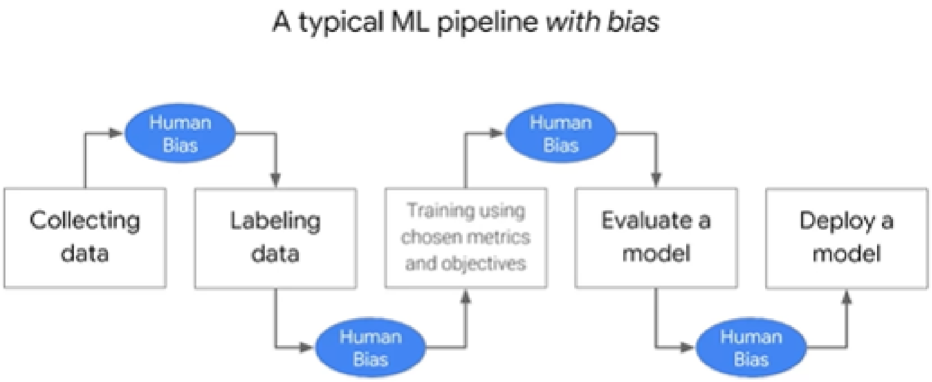
What's common to all of these questions? They're decided by you, the creator of the model. Machine learning is not automatic, it is highly driven by the decisions that we make in terms of how we train the machine learning model. The human impact doesn't stop with you. Maybe the model is selecting which customers are higher priority. So who's calls you're going to answer first? Maybe the model is selecting which products you're going to recommend to a specific customer. Maybe the model is selecting which transactions are likely to be fraudulent, and that you are going to examine further. Maybe the model is selecting which trees are being cut illegally. The end customers of your business therefore see some kind of an effect because of the predictions of the model. They're provided with some specific quality of service, or they get some kind of a product, or the wait in the long run queue. They see one product or they see another, they can go through their credit card purchase, or the transaction gets denied. Your end customers see the impact of the decisions that you make. So the decisions that you make as you do machine learning end up having an impact on the real world because of the way that the predictions of the machine learning model are going to get used.

22. More about bias in data:

Human biases that exist in data because data found in the world has existing biases with regard to properties like gender, race, and sexual orientation. Selection bias happens because the subjects that get into our samples represent a privileged type of user. Confirmation bias refers to only looking for data which confirms our hypothesis. *eg :someone who buys a tie would look for men's shirts. So we might restrict our data collection to only men's apparel items. But women do wear ties*. An automation bias refers to the biases which crop up when the data we use is just the data that is easily automatable, the data that we can quickly get. *eg: If men's apparel and women's apparel are run by different parts of our organization, it is quite possible that this is why we might have collected only men's shirt sales* or *a fancy digital data collection system at some locations, but other locations are still sending you faxes.*



So what's the impact of biases in collecting data and labeling data? it attracts the entire pipeline. In machine learning, we begin with our data. So the biases in the original data are going to get reflected downstream in our models and consequently, they are going to result in biased outcomes. The result is that biases can appear at every point of the data pipeline. [*eg: model to find fraudulent transactions.]*



2 Avoid creating or reinforcing unfair bias

ML models learn from existing data collected from the real world, and so an accurate model may learn or even amplify problematic pre-existing biases in the data based on race, gender, religion, or other characteristics.

eg: a model to decide whether or not to make a loan was given

> Machine learning models are trained on real world data, you may have removed the gender from your loan application process.

> But the use of occupation as an input will mean that gender leaks through. Occupations tend to be associated with specific genders. So by using occupation as an input to our machine learning model, that decides whether or not to extend a loan, we may be providing a backdoor entry to the use of gender. This property of machine learning models leads to reinforcing existing biases in society. It's one of the things that we have to be on our guard against. It comes down to understanding our data, really understanding it, and understanding some of the implicit biases that ride along with that data.

Gender bias in occupations

Highest female bias

occupation	bias	occupation	bias
maid	59.2	librarian	20.1
waitress	52.5	obstetrician	16.9
midwife	50.9	secretary	13.7
receptionist	50.2	socialite	12.1
nanny	47.7	therapist	10.2
nurse	45.4	manicurist	10.1
midwives	43.8	hairdresser	9.7
housekeeper	36.6	stylist	8.6
hostess	32	homemaker	6.9
gynecologist	31.6	planner	5.8

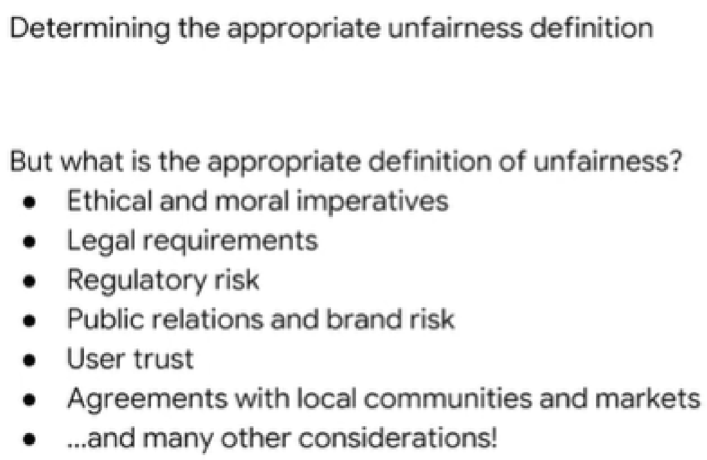
Highest male bias

occupation	bias	occupation	bias
undertaker	-73.4	captain	-53.4
janitor	-62.3	announcer	-51.1
referee	-60.7	architect	-50.7
plumber	-58	maestro	-50.6
actor	-56.9	drafter	-46.7
philosopher	-56.2	usher	-46.6
barber	-55.4	farmer	-45.4
umpire	-54.3	broadcaster	-45.2
president	-54	engineer	-45.1
coach	-53.8	magician	-44.8

23. Activity: Understanding Learned Relationships using the Embedding Projector Review

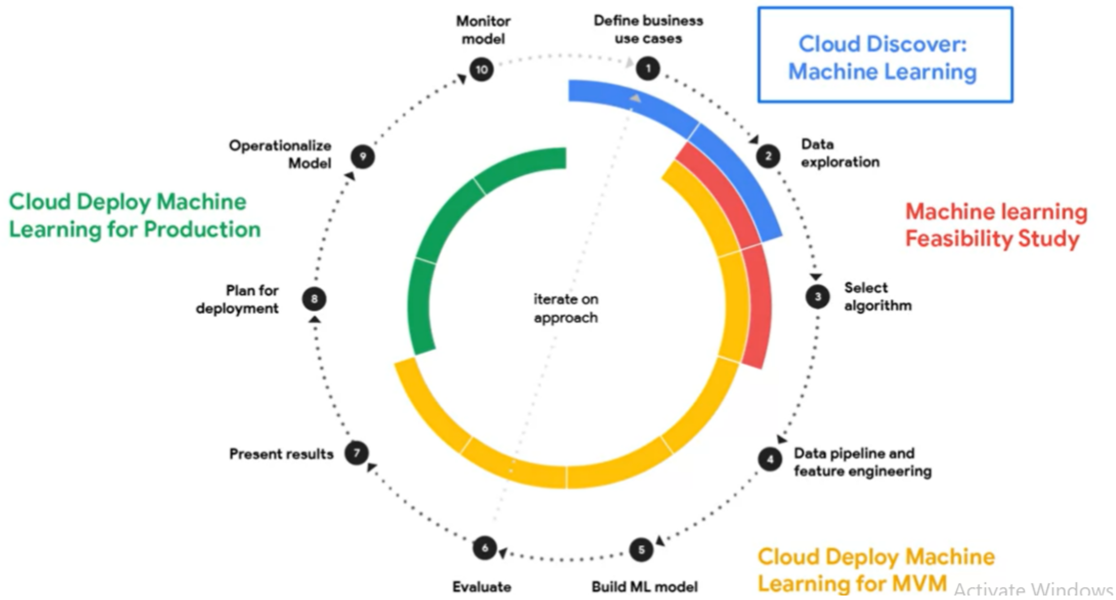
24. We've built a model, how do we know whether the model is fair? One challenge is that there's no one standard definition of fairness. Whether decisions are made by humans or by machines. Both in law and philosophy, this is a question that engenders lots of debate. This because defining fairness is difficult. *eg: blue & orange people loan case scenario [see: Week 2> Fairness in ML Part 1]*. As in eg. the group-unaware constraint holds that the thresholds are the same. The demographic parity constraint, requires that the same percentage within each subgroup will receive the loan, and the equal opportunity constraint requires that the same fraction within each subgroup, who can pay them off, will receive the loan. Now, none of these definitions is necessarily better than the other. Each of them has different mathematical properties, and you cannot satisfy them all at the same time, and which constraint you choose depends a lot on the context of the machine learning use case. But you've got to choose.

25. Allison Woodruff & her team asked different groups what is the appropriate definition of unfairness? The result varies greatly depending on who you're speaking to and their ethical, legal and economic priorities. Unfairness, is not immediately obvious, and it requires asking nuanced, social, political and ethical questions. It takes a holistic analysis to consider which definition of fairness is appropriate for your project. They asked different groups, what is the appropriate definition of unfairness? The result varies greatly depending on who you're speaking to and their ethical, legal and economic priorities. Unfairness, is not immediately obvious, and it requires asking nuanced, social, political and ethical questions. It takes a holistic analysis to consider which definition of fairness is appropriate for your project.

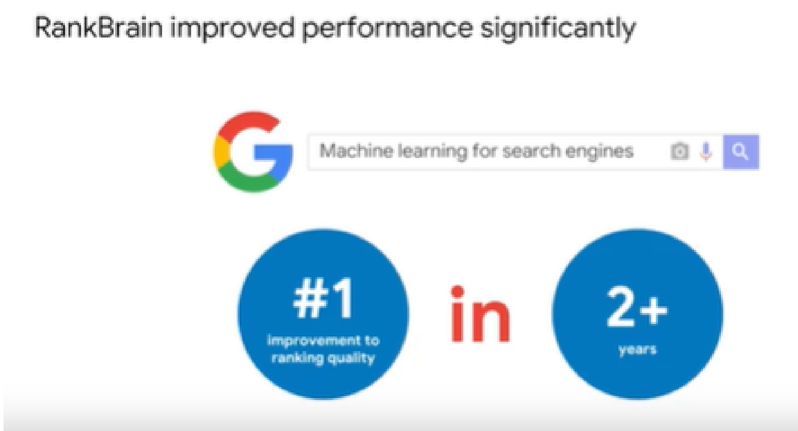
    Checklist of data prone to bias related issues.: biometrics, race, skin color, religion, sexual orientation, socioeconomic status, income, country, location, health, language or dialect? For example, zip code or other geospatial data, is often correlated with socioeconomic status, and/or income. Image or video data, can reveal information about race, gender, and age.

26. Activity: Applying Fairness Concerns with the What-If tool Review
27. What are the use cases of ML? [27-32]



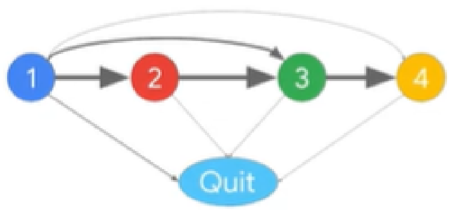
Above figure - processes in Google Cloud Discover

1. ML can be used to simplify many rule-based systems
2. Automate business processes with ML
3. Predict business metrics
4. Understand unstructured data with ML
5. Personalize applications with ML
6. Improve user experiences with ML



28. [1] The first way to think about machine learning use cases is, as a way to simplify rule-based systems. That was the essential idea behind RankBrain, our deep neural network for search ranking. It outperformed many human-built signals and we could replace many of the hand-coded rules with machine learning. The neural network ended up improving our search quality dramatically. Plus, the system could continually improve itself based on what users actually preferred. *eg: Acme widgets smoke alarms systems.*
29. *eg: Netmarble. Here unusual action is skipping a step to complete a level.*

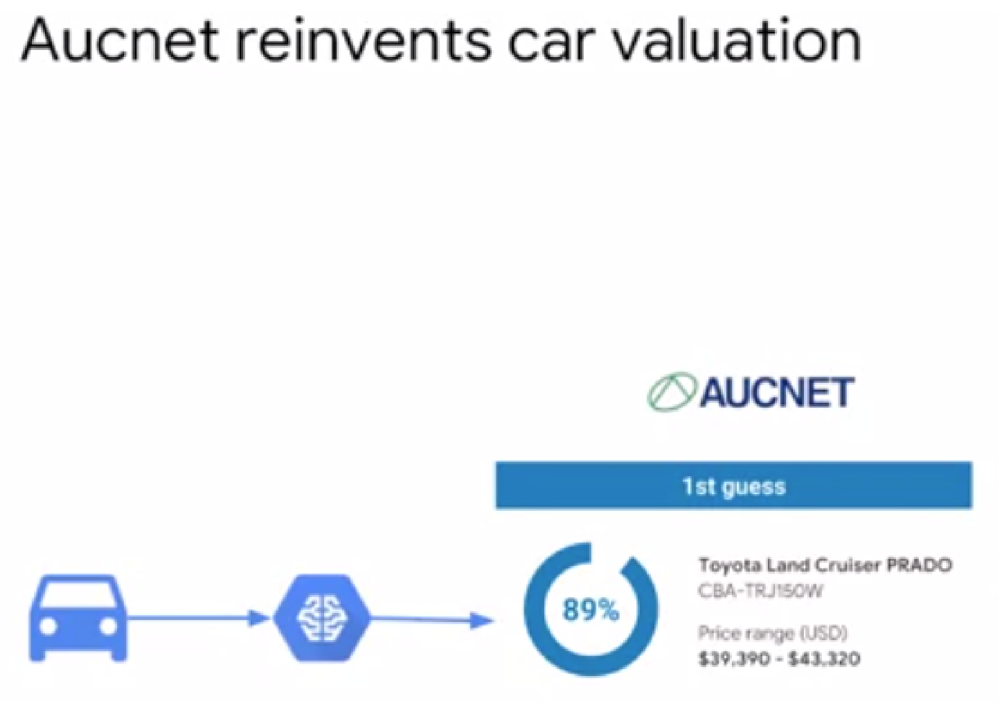
Detecting unusual player actions



Possible sequences of actions in a game

> So, the key idea here, is to do a pilot project, and that success will embolden other departments to carry out their own machine learning projects. Be the pioneer.

30. [2] eg: Aucknet, Ocado, Ananda Developers



> The purpose of streamlining a business process is not to reduce costs, the purpose is to reduce drudgery, increase accuracy, gain more visibility into the business. In other words, it's about doing more, not about spending less.

To buy the full note click the link below.

Managing Machine Learning Projects with Google Cloud

Hello, thank you for visiting my BMC page. You will also find my study notes on the subjects of Machine Learning, Data Science and AI. I also love creating Notion templates and sharing them on this

https://www.buymeacoffee.com/josephben/e/37717

▼ Formatting guidelines:

Bold: Important

Italic: Examples

Bold Italic: Key terms

Every numbered point is emphasizing a key idea, and that is highlighted in blue text. If those ideas are questions, then their answers have been represented in purple text. Green text denotes lab or activity.

Using this note in conjunction with the course will help you grasp the note better.