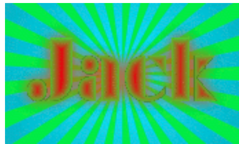
Introduction to Machine Learning

WHY DO WE WANT MACHINES TO LEARN?

JACK wants to buy a car. He tries to calculate how much he needs to save monthly for that. He went over dozens of ads on the internet and learned that new cars are around $20,000, used year-old ones are $19,000, 2-year old are $18,000 and so on.

Jack

JACK, our brilliant analytic, starts seeing a pattern: *so, the car price depends on its age and drops $1,000 every year, but won't get lower than $10,000.*



WHY DO WE WANT MACHINES TO LEARN?

- In machine learning terms, JACK finds *regression* – he predicted a value (price) based on known historical data.
- The problem is, they have different manufacturing dates, dozens of options, technical condition, seasonal demand spikes, and god only knows how many more hidden factors. An average JACK can't keep all that data in his head while calculating the price.

- People are dumb and lazy – we need robots to do the math for them. So, let's go the computational way here.
- Let's provide the machine some data and ask it to find all hidden patterns related to price.

WHY DO WE WANT MACHINES TO LEARN?

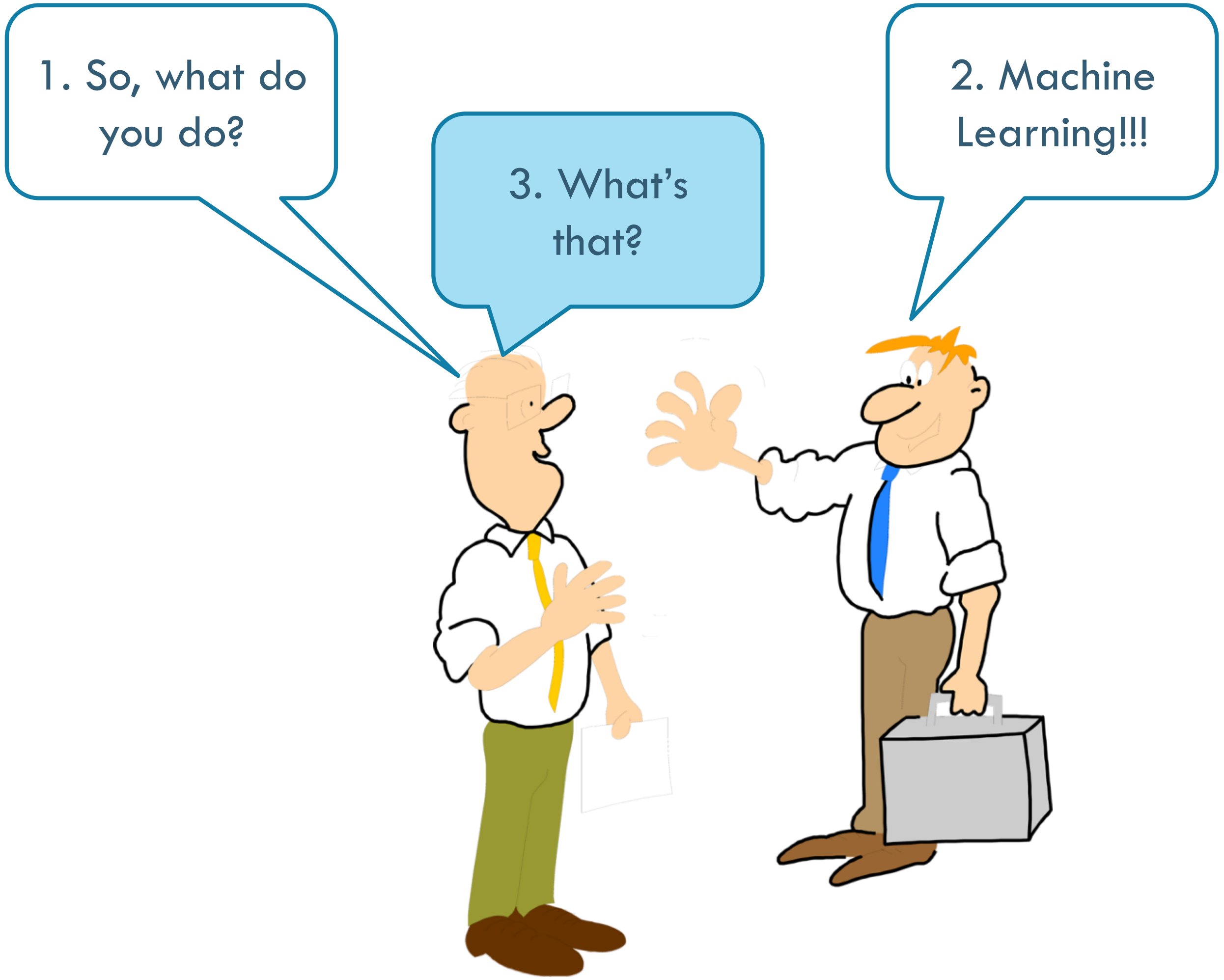
Aaaand it works. The most exciting thing is that the machine copes with this task much better than a real person does when carefully analyzing all the dependencies in their mind.

That was the birth of machine learning.

Jack

A SHORT STORY – JACK & JILL

YEAR 2010

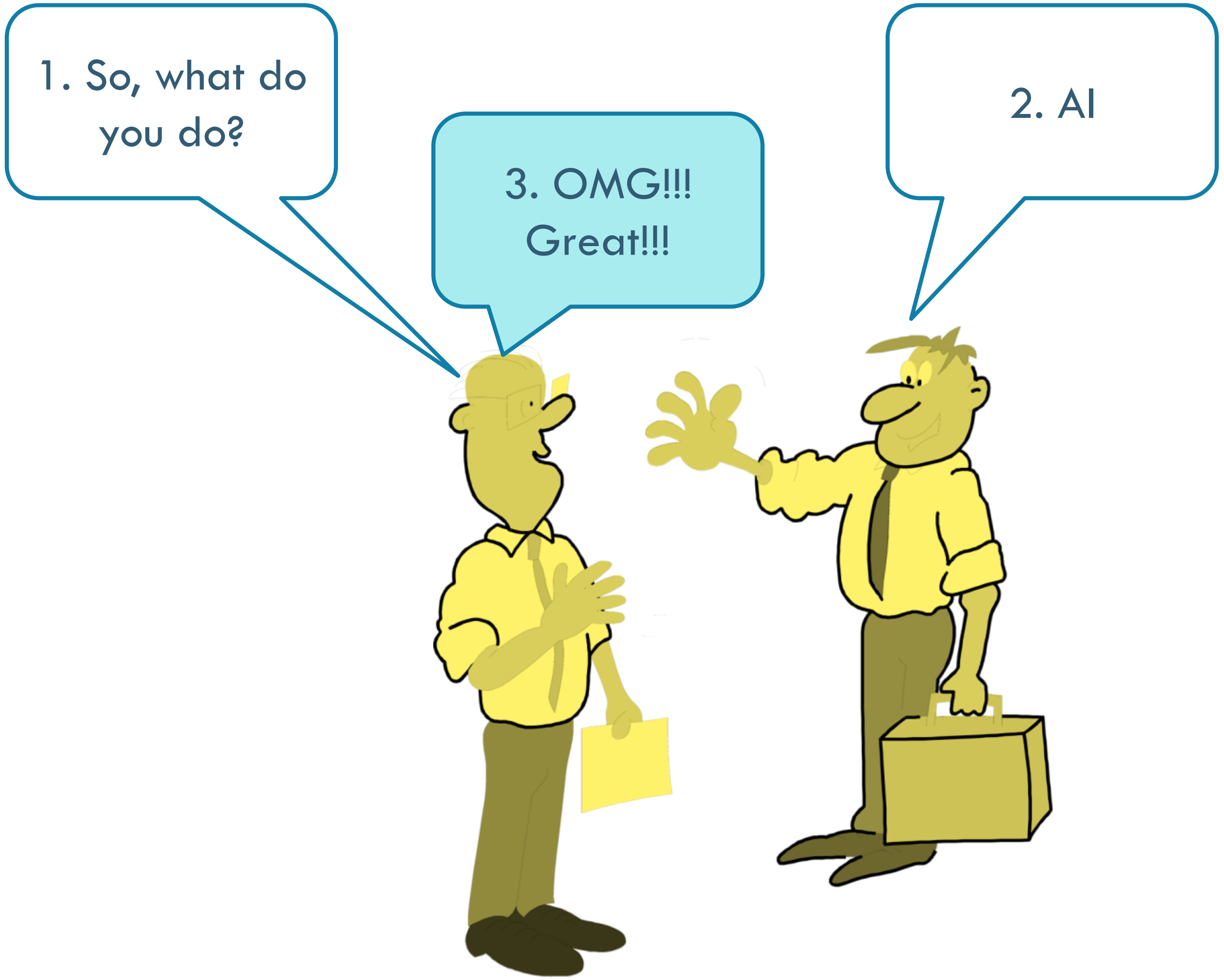


A SHORT STORY – JACK & JILL

YEAR 2010
+
10 SECONDS

A SHORT STORY – JACK & JILL

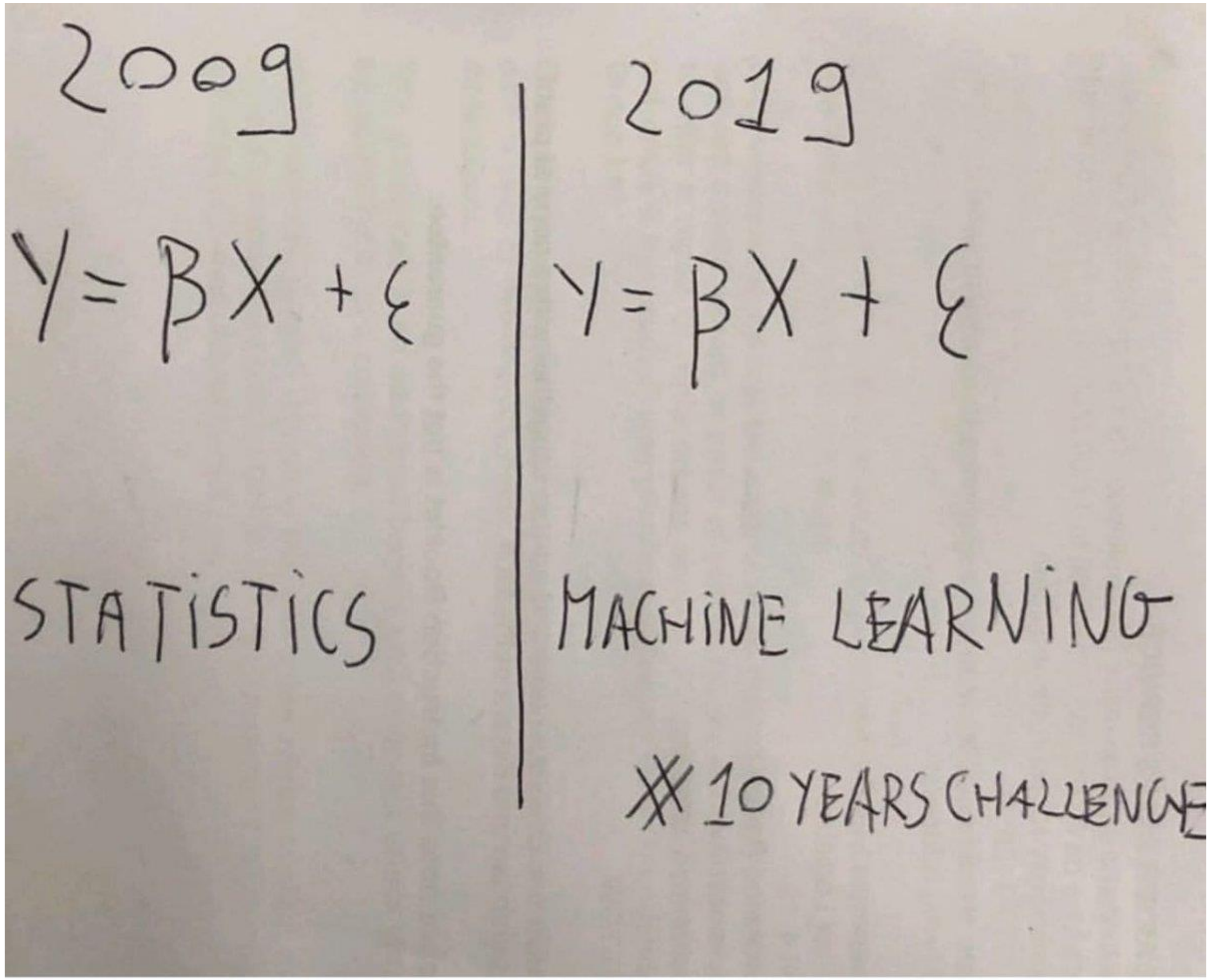
YEAR 2019



People Love to hear AI and paid more if you say AI.

<u>Moral:</u>
Same Thing Different Words

10 YEARS CHALLENGE

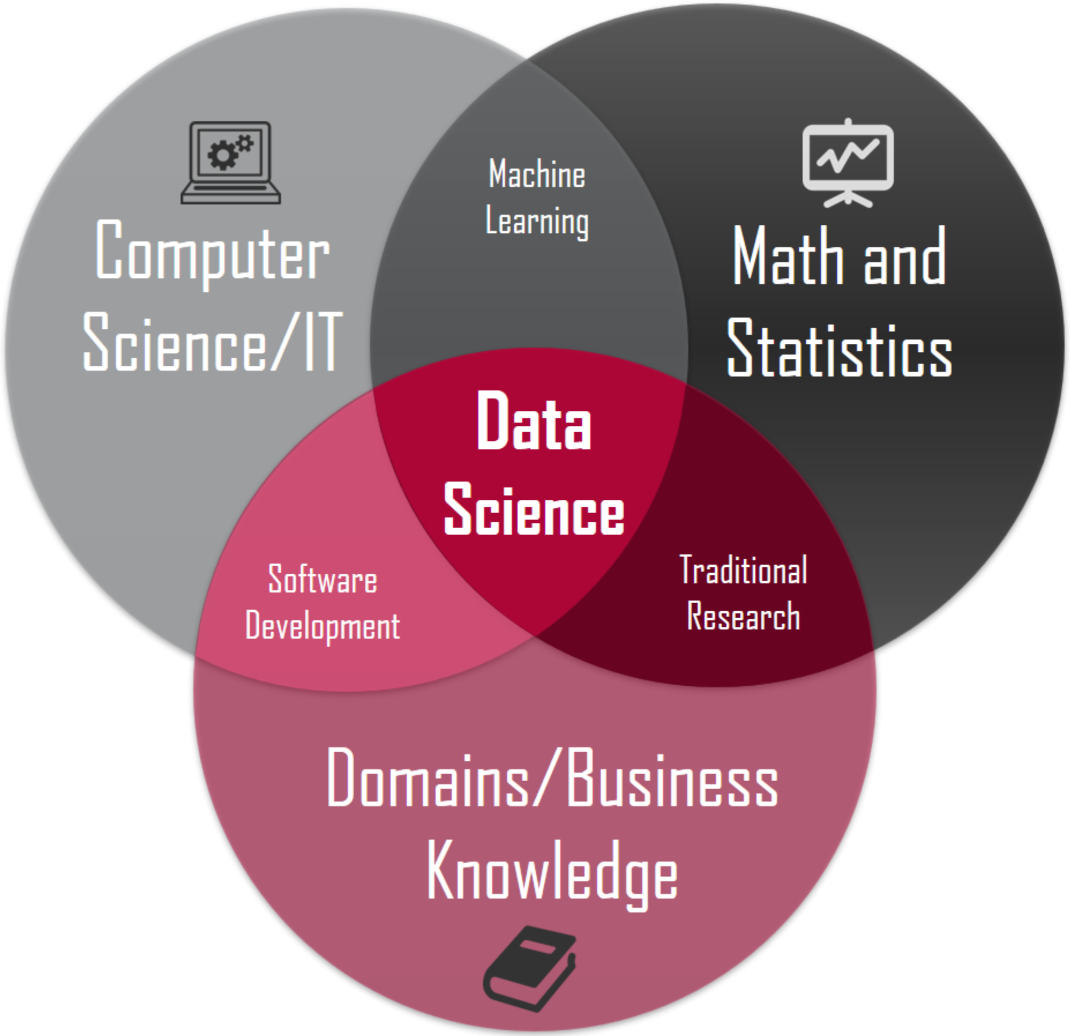


We have been doing ML for 100s of years from now. We're doing faster and bigger now!

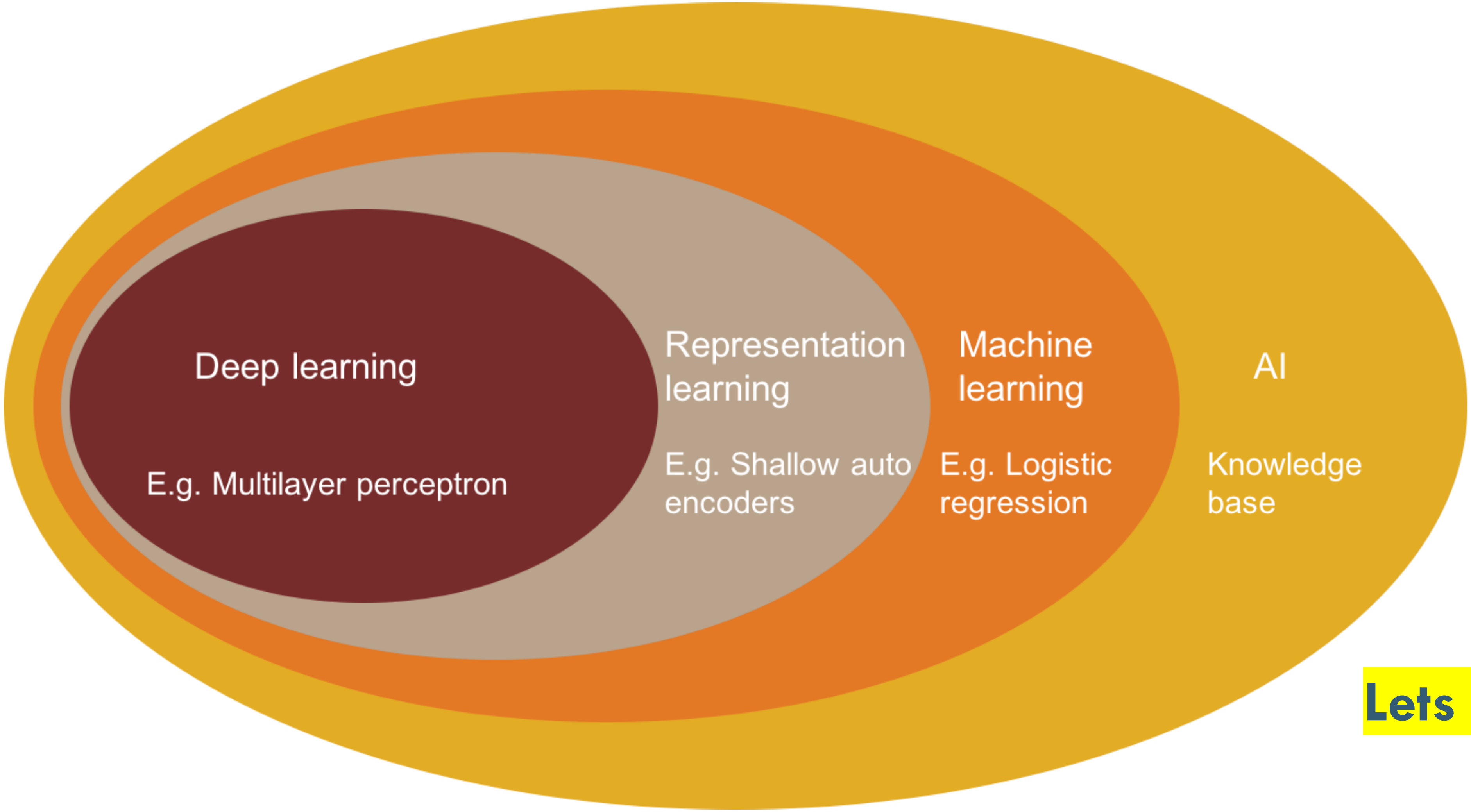
Marketing has changes and we're still doing the same stuff!

Same Thing Different Words

KEY DIFFERENCE – VENN DIAGRAM



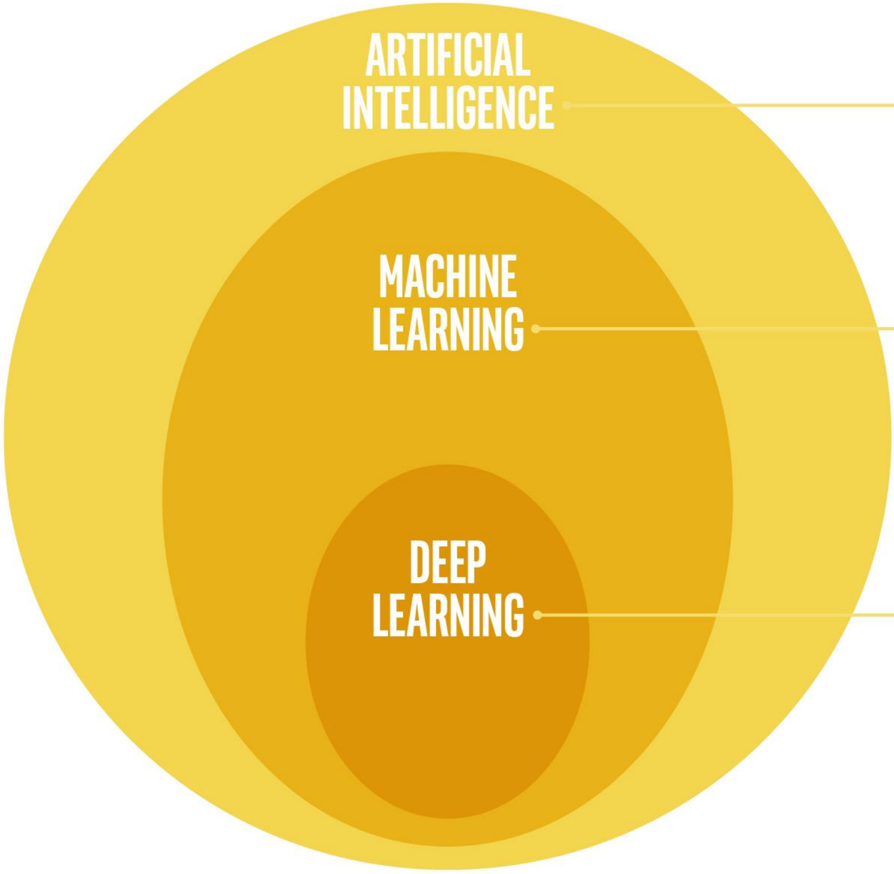
NEW AI VENN DIAGRAM



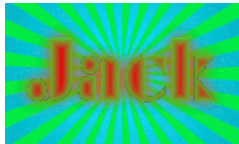
Lets Define AI

TERMINOLOGIES

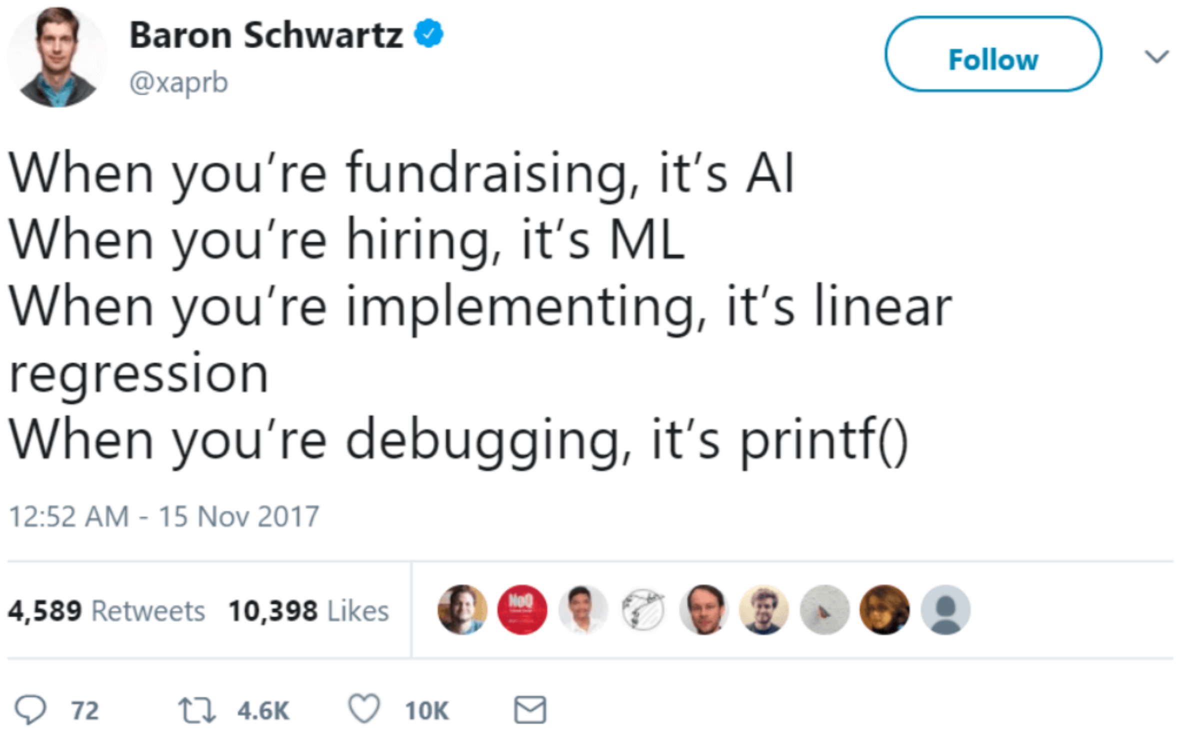
ARTIFICIAL INTELLIGENCE TERMS



- **AI** is an umbrella term for machines capable of perception, logic, and learning.
- **Machine learning** employs algorithms that learn from data to make predictions or decisions, and whose performance improves when exposed to more data over time.
- **Deep learning** uses many-layered neural networks to build algorithms that find the best way to perform tasks on their own, based on vast sets of data.



DEFINING AI



Baron Schwartz
@xaprb

Follow

When you're fundraising, it's AI
When you're hiring, it's ML
When you're implementing, it's linear regression
When you're debugging, it's printf()

12:52 AM - 15 Nov 2017

4,589 Retweets **10,398** Likes

72 4.6K 10K

DEFINING AI

Broad & Changes over time!

What Human Find Easy but Computers Find Hard

AI meeting human is State of Art

CHESS

Its not AI Anymore, every app has it

Definition of AI is Fluid

Now, AI is beating humans in MARIO, PACMAN and many more

LEARNING VS INTELLIGENCE

Artificial intelligence is the name of a whole knowledge field, similar to biology or chemistry.

Machine Learning is a part of artificial intelligence. An important part, but not the only one.

Neural Networks are one of machine learning types. A popular one, but there are other good guys in the class.

Deep Learning is a modern method of building, training, and using neural networks. Basically, it's a new architecture. Nowadays in practice, no one separates deep learning from the "ordinary networks". We even use the same libraries for them. To not look like a dumbass, it's better just name the type of network and avoid buzzwords.

Machine can	Machine cannot
Forecast	Create something new
Memorize	Get smart really fast
Reproduce	Go beyond their task
Choose best item	Kill all humans

TERMINOLOGIES

Definition [T. Mitchell]:

Machine Learning is the study of computer algorithms that improve their performance in a certain task through experience.

Example: Chess

Task → Play Chess.

Experience → Self-Play

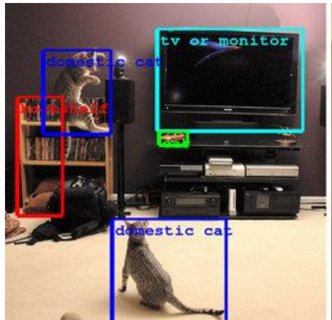
Performance measure → Games won against humans.

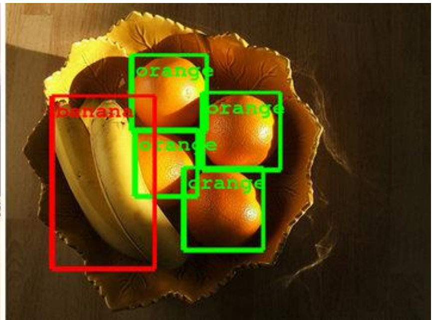
Example: Object Recognition

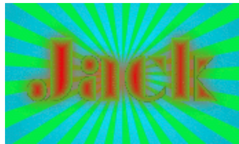
Task → Determine which objects are visible in images.

Experience → Annotated training data

Performance measure → Object recognized correctly



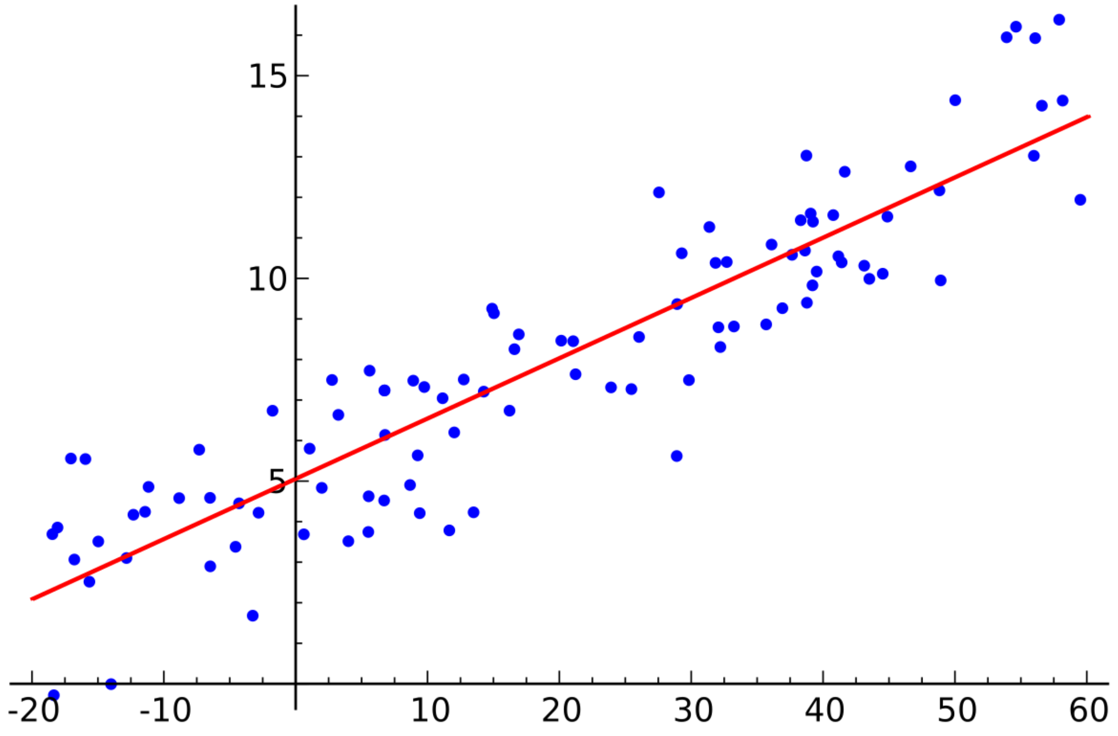




SAME THING DIFFERENT WORDS

Y=a + bx

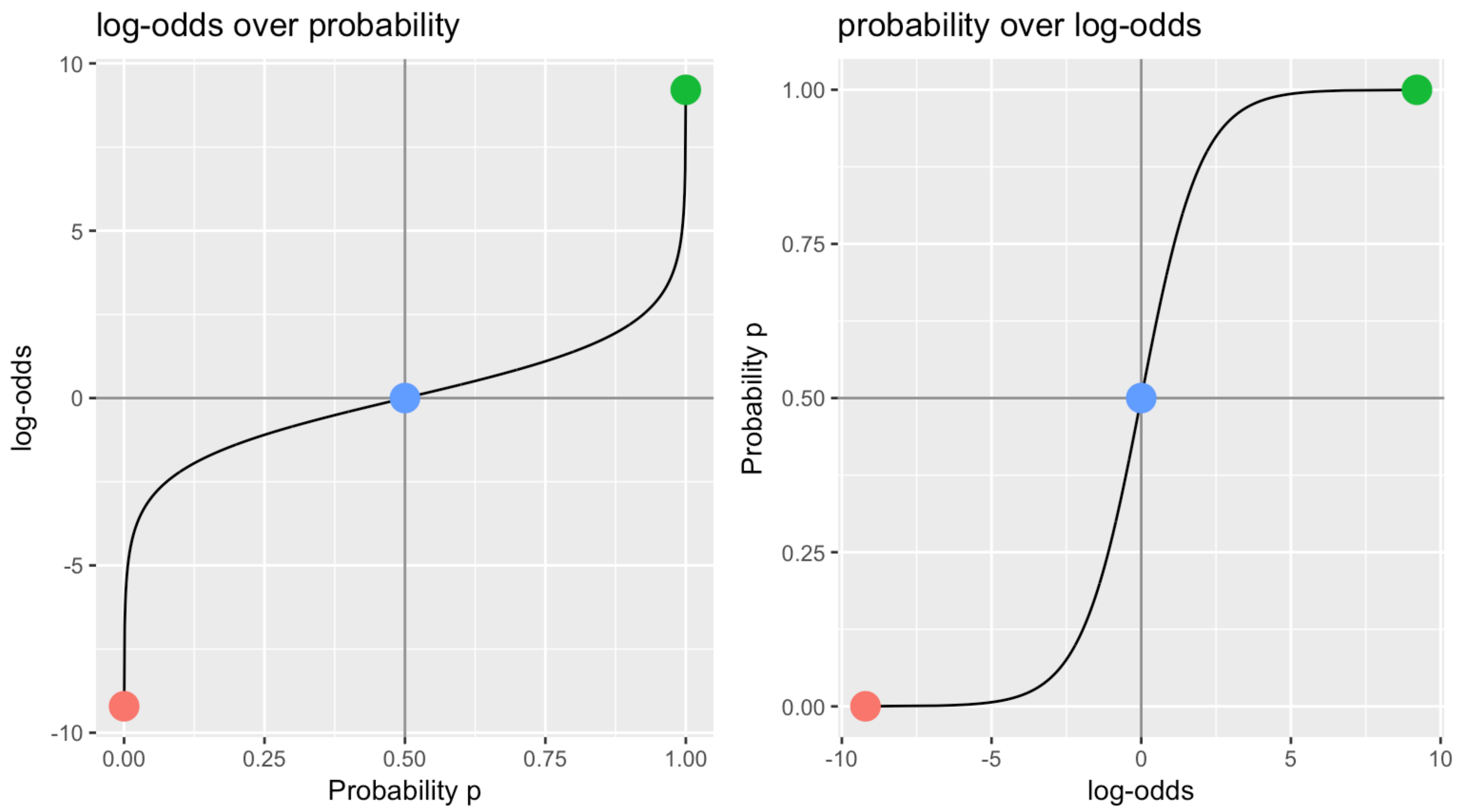
Traditional	Newer
Intercept	Bias
Coefficients	Weights



SAME THING DIFFERENT WORDS

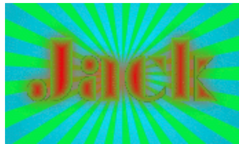
$$Y = \frac{1}{1 + e^{-x}}$$

Traditional	Newer
Inverse Logit	Sigmoid
Prediction	Inference



$$\text{left-hand function} = \text{logit}(p) = \text{log-odds} = \log\left(\frac{p}{1-p}\right)$$

$$\text{right-hand function} = \text{inverse-logit}(\alpha) = \frac{\exp(\alpha)}{\exp(\alpha)+1} = \frac{1}{1+\exp(-\alpha)}$$



THREE COMPONENTS OF MACHINE LEARNING

The only goal of machine learning is to predict results based on incoming data. That's it.
All ML tasks can be represented this way, or it's not an ML problem from the beginning.

The greater variety in the samples you have, the easier it is to find relevant patterns and predict the result. Therefore, we need three components to teach the machine:

- Data
- Features
- Algorithms

COMPONENTS OF MACHINE LEARNING - DATA

The only goal of machine learning is to predict results based on incoming data. That's it. All ML tasks can be represented this way, or it's not an ML problem from the beginning.

The more diverse the data, the better the result.

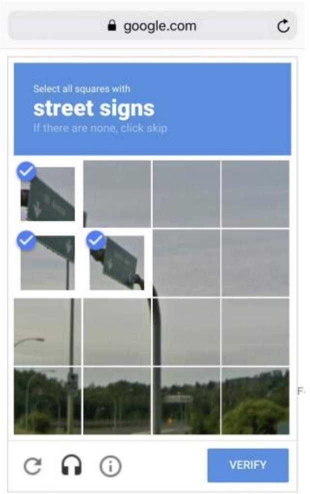
Want to detect spam?	Get samples of spam messages.
Want to forecast stocks?	Find the price history.
Want to find out user preferences?	Parse their activities on Facebook

There are two main ways to get the data,

Manual

Automatic.

Manually collected data contains far fewer errors but takes more time to collect → that makes it more expensive in general.	Automatic approach is cheaper → you're gathering everything you can find and hope for the best.



Google use their own customers to label data for them for free. Free labor!

It's extremely tough to collect a good collection of data (usually called a dataset).
They are so important that companies may even reveal their algorithms, but rarely datasets.

COMPONENTS OF MACHINE LEARNING - FEATURES

- Features Also known as parameters or variables.
- In other words, these are the factors for a machine to look at.

car mileage	user's gender	stock price	word frequency	Salary	profession	education	City	State	Country	Etc…..
32	F	123	3	90900	SE	BE	Chennai	TN	INDIA	….

When data stored in tables it's simple — features are column names.

But what are they if you have 100 Gb of cat pics? We cannot consider each pixel as a feature.

That's why selecting the right features usually takes way longer than all the other ML parts.

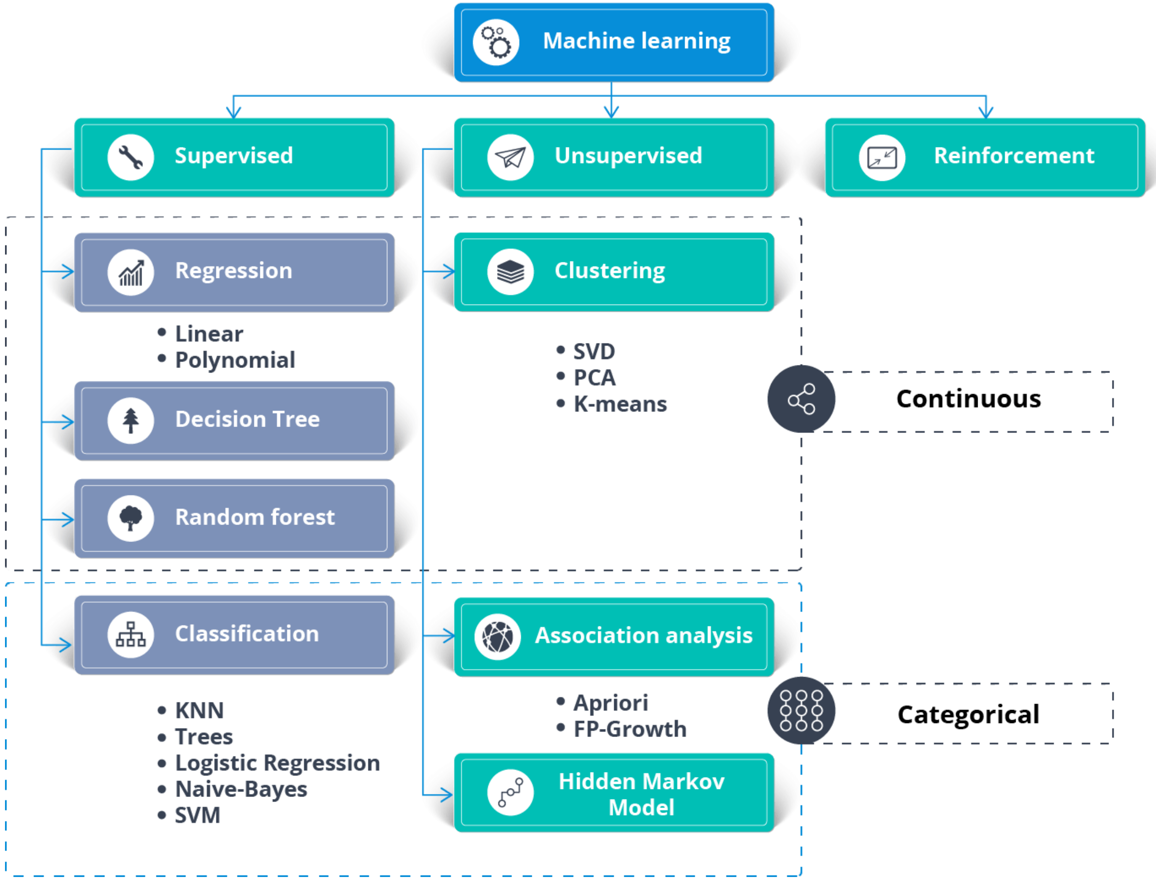
That's also the main source of errors.

COMPONENTS OF MACHINE LEARNING - ALGORITHMS

- Algorithms Most obvious part. Any problem can be solved differently.

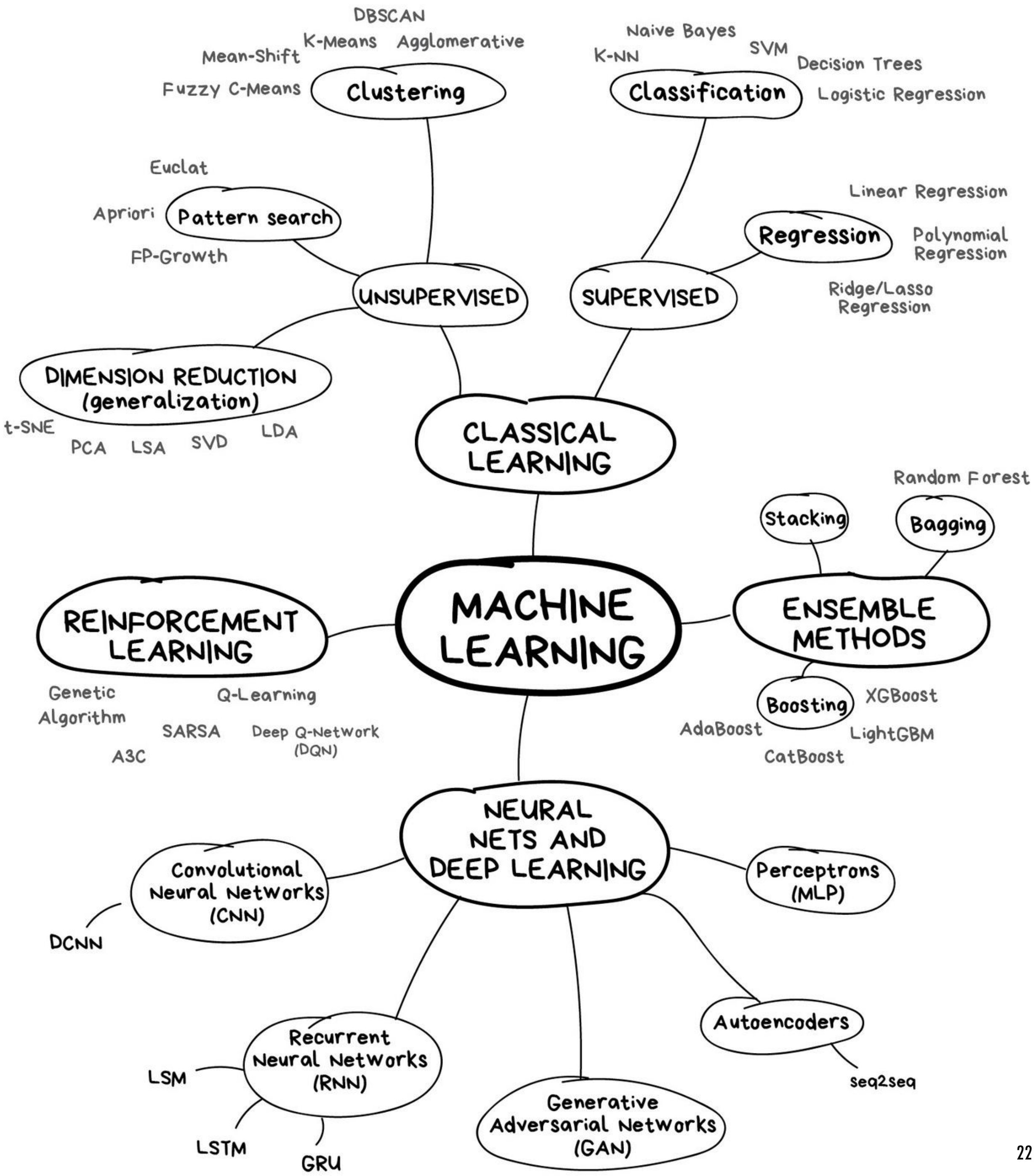
- The method you choose affects the precision, performance, and size of the final model.

- There is one important nuance though: if the data is crappy, even the best algorithm won't help. Sometimes it's referred as "garbage in – garbage out". So don't pay too much attention to the percentage of accuracy, try to acquire more data first.

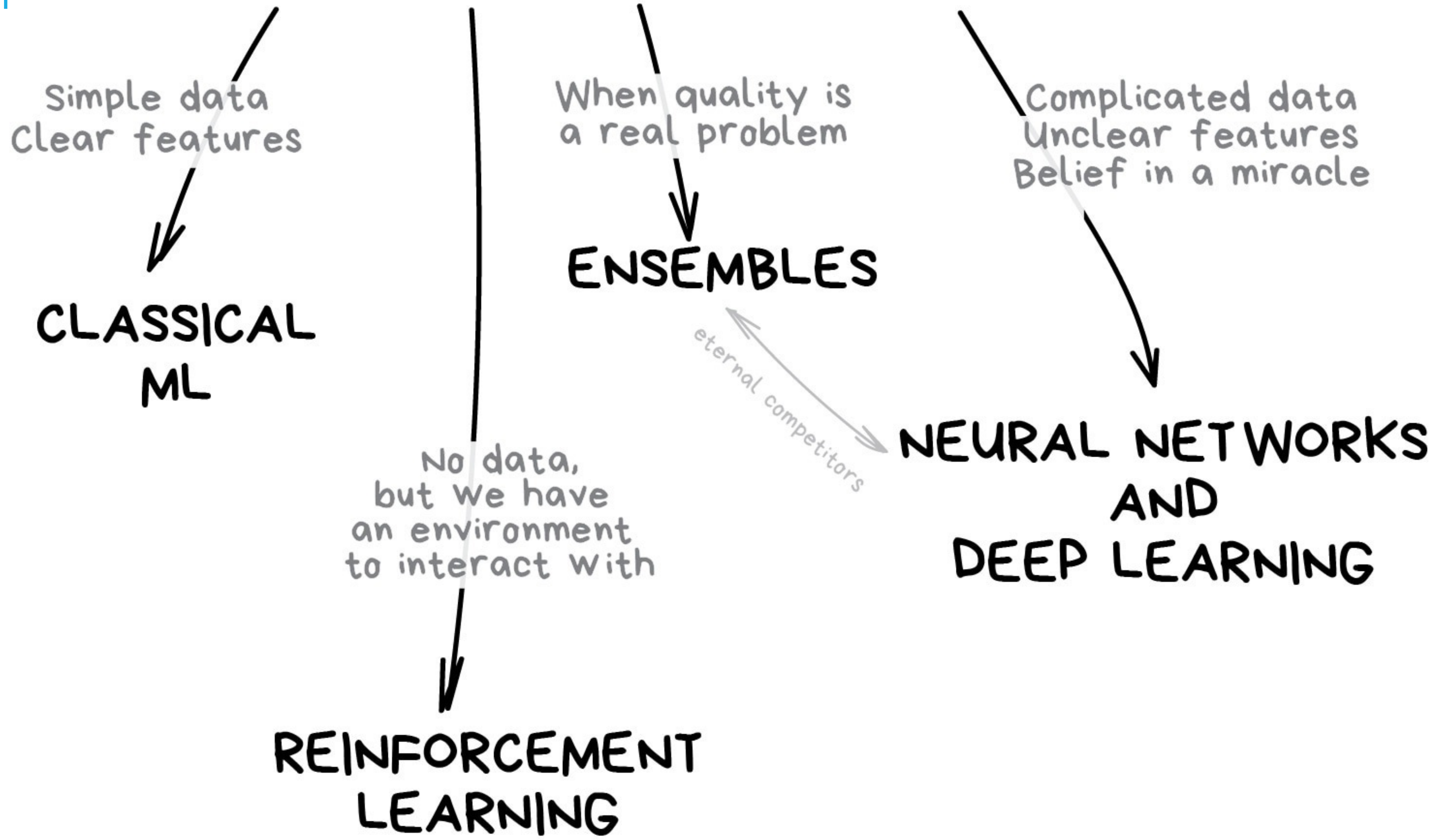


THE MAP OF THE MACHINE LEARNING WORLD

Always important to remember → there is never a sole way to solve a problem in the machine learning world. There are always several algorithms that fit, and you have to choose which one fits better.

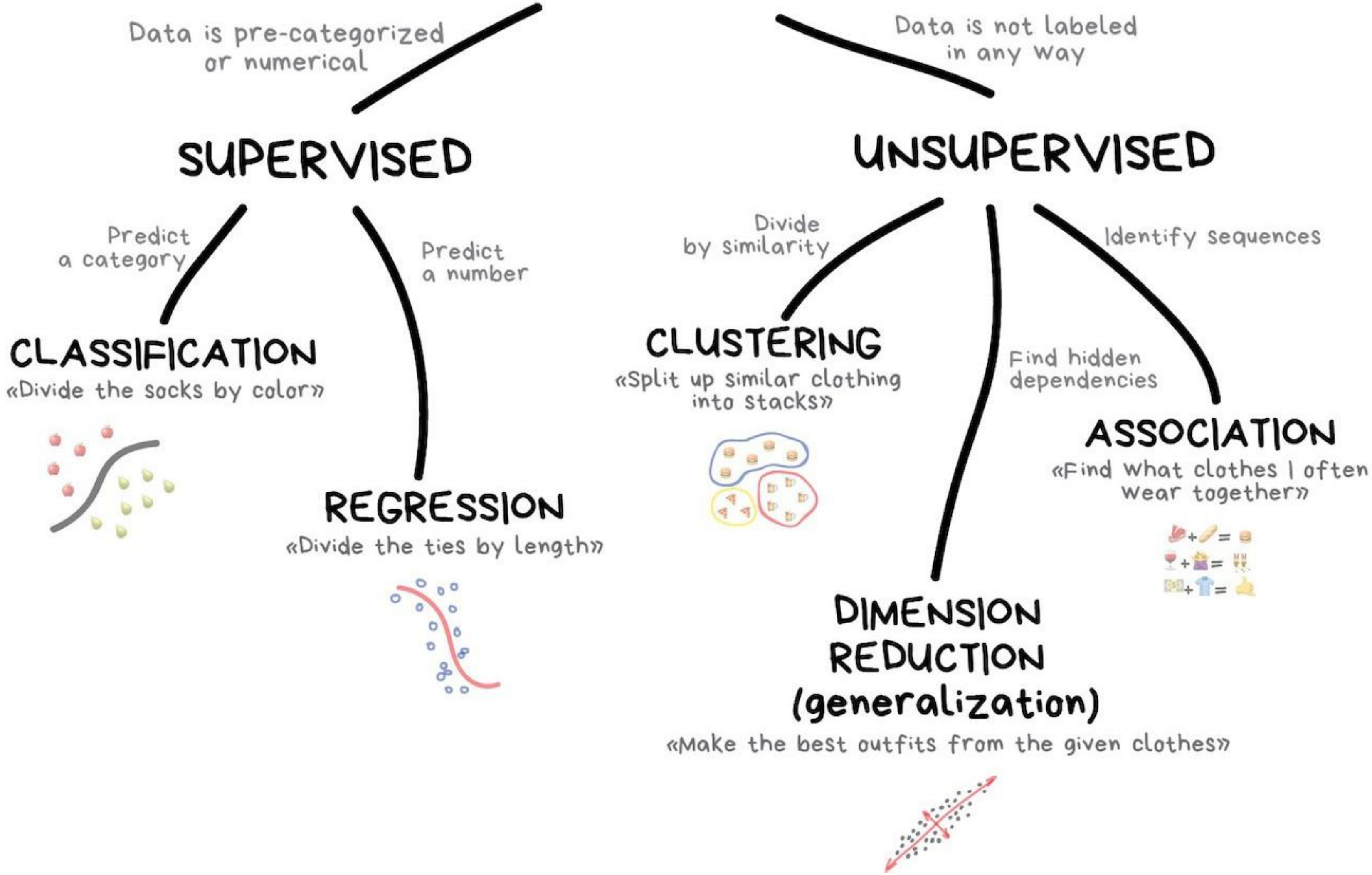


MAIN TYPES OF MACHINE LEARNING



Let's start with a basic overview. Nowadays there are four main directions in machine learning.

1.CLASSICAL MACHINE LEARNING



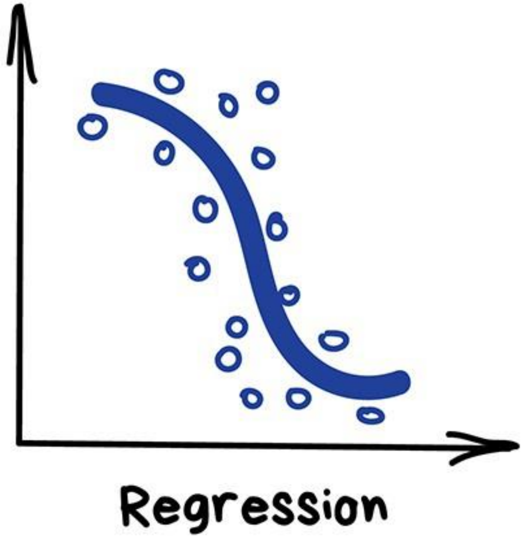
Classical approaches are so natural that you could easily explain them to a toddler. They are like basic arithmetic — we use it every day, without even thinking.

1.1 SUPERVISED LEARNING - TRAIN ME!

- In this case, the machine has a "supervisor" or a "teacher" who gives the machine all the answers, like whether it's a cat in the picture or a dog. The teacher has already divided (labeled) the data into cats and dogs, and the machine is using these examples to learn. One by one. Dog by cat.
- Supervised Learning models are trying to find parameter values that will allow them to perform well on historical data. Then they are used for making predictions on unknown data, that was not a part of training dataset.
- Clearly, the machine will learn faster with a teacher, so it's more commonly used in real-life tasks.
- There are two types of such tasks: **classification** – an object's category prediction, and **regression** – prediction of a specific point on a numeric axis.

1.11 SUPERVISED LEARNING – REGRESSION

"Draw a line through these dots.
Yep, that's the machine learning"



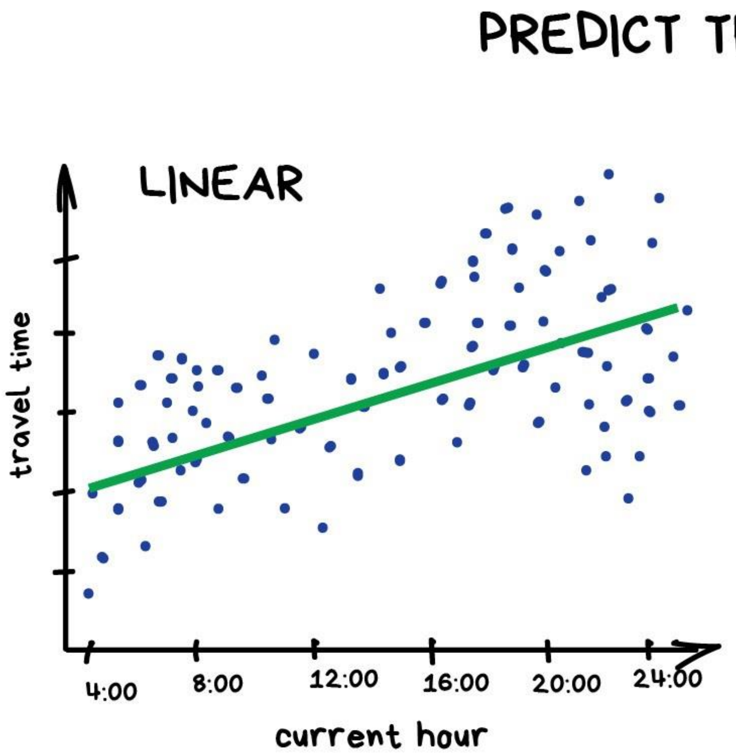
Objective:

- Stock price forecasts
- Demand and sales volume analysis
- Medical diagnosis
- Any number-time correlations
- Car price by its mileage
- Traffic by time of the day
- Etc

Popular algorithms are Linear and Polynomial regressions.

Everyone who works with finance and analysis loves regression. It's even built-in to Excel. And it's super smooth inside — the machine simply tries to draw a line that indicates average correlation. Though, unlike a person with a pen and a whiteboard, machine does so with mathematical accuracy, calculating the average interval to every dot.

1.11 SUPERVISED LEARNING – REGRESSION

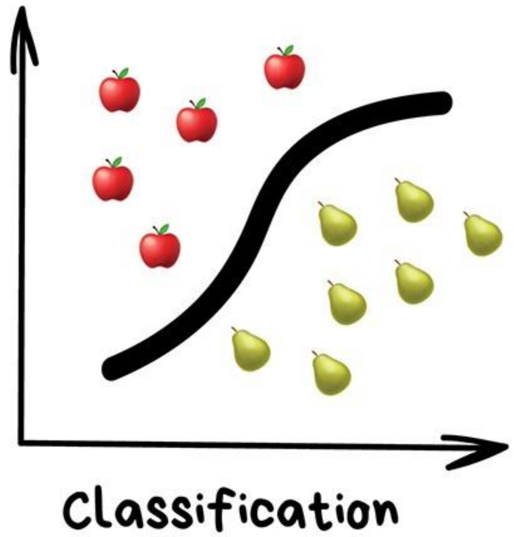


When the line is straight — it's a linear regression, when it's curved – polynomial. These are two major types of regression. The other ones are more exotic. Logistic regression is a black sheep in the flock. Don't let it trick you, as it's a classification method, not regression.

1.12 SUPERVISED LEARNING – CLASSIFICATION

“Splits objects based at one of the attributes known beforehand. Separate socks by based on color, documents based on language, music by genre"

Objective:
– Spam filtering
– Language detection
– A search of similar documents
– Sentiment analysis
– Recognition of handwritten characters and numbers
– Fraud detection
– Etc



Popular algorithms: Naive Bayes, Decision Tree, Logistic Regression, K-Nearest Neighbours, Support Vector Machine

In classification, you always need a teacher. The data should be labeled with features so the machine could assign the classes based on them. Everything could be classified — users based on interests (as algorithmic feeds do), articles based on language and topic (that's important for search engines), music based on genre (Spotify playlists), and even your emails.

1.2 UNSUPERVISED LEARNING - I AM SELF SUFFICIENT IN LEARNING

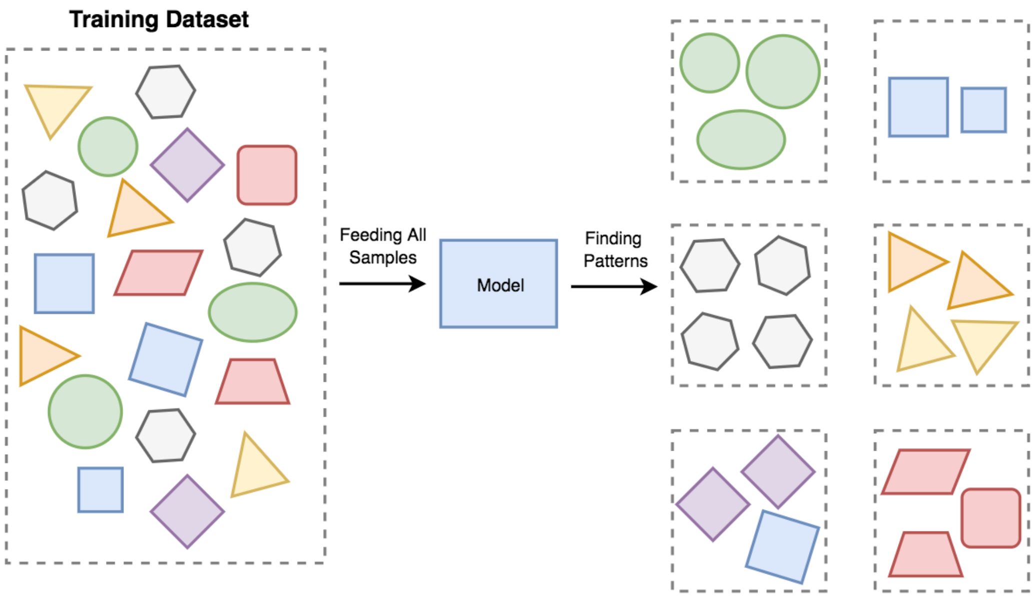
Labeled data is luxury.

you can try to use unsupervised learning

Unsupervised was invented a bit later, in the '90s. It is used less often, but sometimes we simply have no choice.

What if I want to create, let's say, a bus classifier?

- Should I manually take photos of million buses on the streets and label each of them? No way, that will take a lifetime.



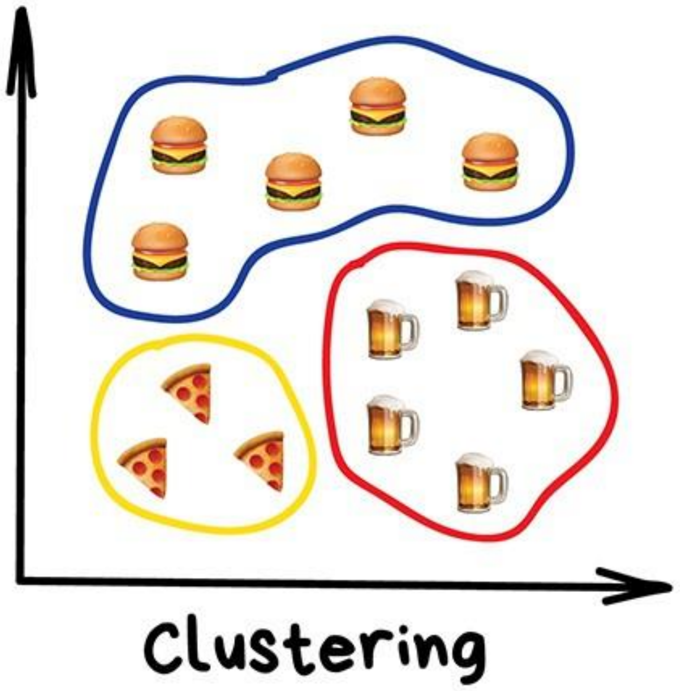
It's usually useful for **exploratory data analysis** but not as the main algorithm.

Group of algorithms that try to **draw inferences from non-labeled data** (without reference to known or labeled outcomes). In Unsupervised Learning, **there are no correct answers**. Models based on this type of algorithms can be used for discovering unknown data patterns and data structure itself.

1.21 CLUSTERING

“Divides objects based on unknown features. Machine chooses the best way"

Clustering is a classification with no predefined classes. It’s like dividing socks by color when you don't remember all the colors you have. Clustering algorithm trying to find similar (by some features) objects and merge them in a cluster. Those who have lots of similar features are joined in one class. With some algorithms, you even can specify the exact number of clusters you want.

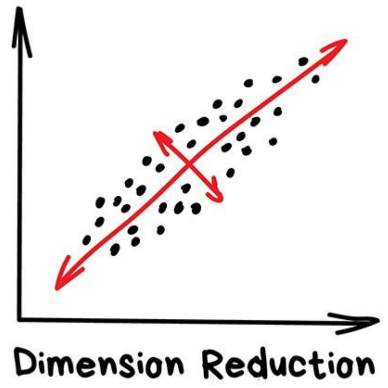


Nowadays used:
- For market segmentation (types of customers, loyalty)
- To merge close points on a map
- For image compression
- To analyze and label new data
- To detect abnormal behavior

Popular algorithms: K-means_clustering, Mean-Shift, DBSCAN

1.22 DIMENSIONALITY REDUCTION (GENERALIZATION)

"Assembles specific features into more high-level ones"



Nowadays is used for:
Recommender systems (★)
Beautiful visualizations
Topic modeling and similar document search
Fake image analysis
Risk management

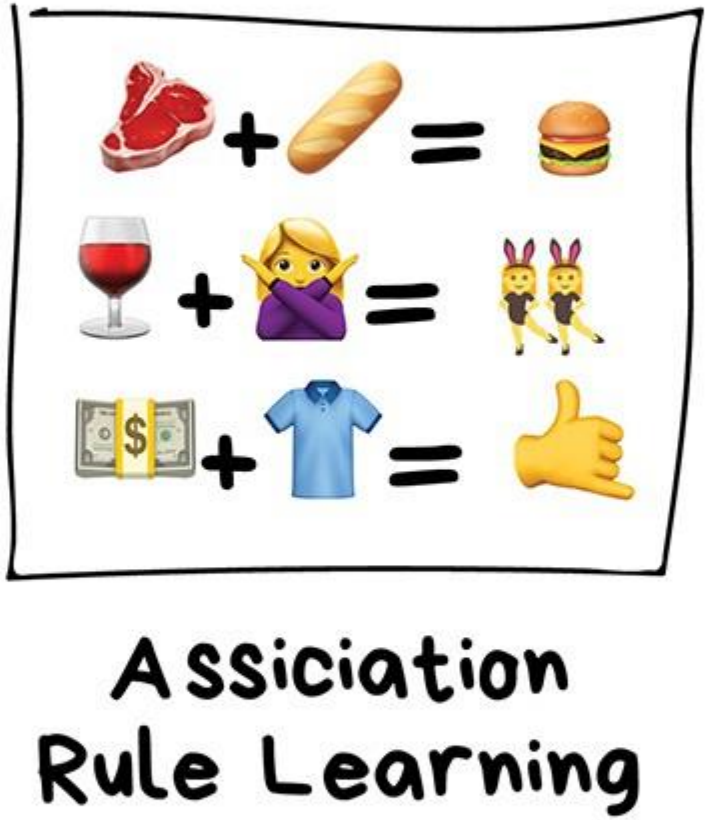
Popular algorithms: Principal Component Analysis (PCA), Singular Value Decomposition (SVD), Latent Dirichlet allocation (LDA), Latent Semantic Analysis (LSA, pLSA, GLSA), t-SNE (for visualization)

1.23 ASSOCIATION RULE LEARNING

"Look for patterns in the orders' stream"

Nowadays is used:
Market Basket Analysis is a popular application of Association Rules.
To analyze goods bought together
To place the products on the shelves
To analyze web surfing patterns

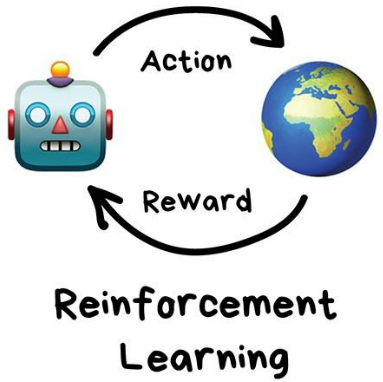
Popular algorithms: [Apriori, Euclat, FP-growth]

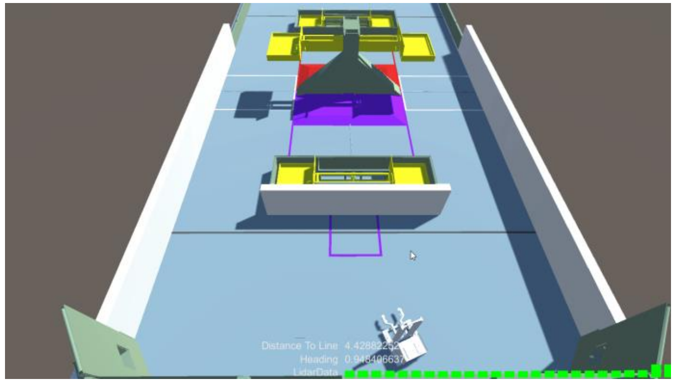


1.3 REINFORCEMENT LEARNING – MY LIFE MY RULES! (HIT & TRIAL)

"Throw a robot into a maze and let it find an exit"

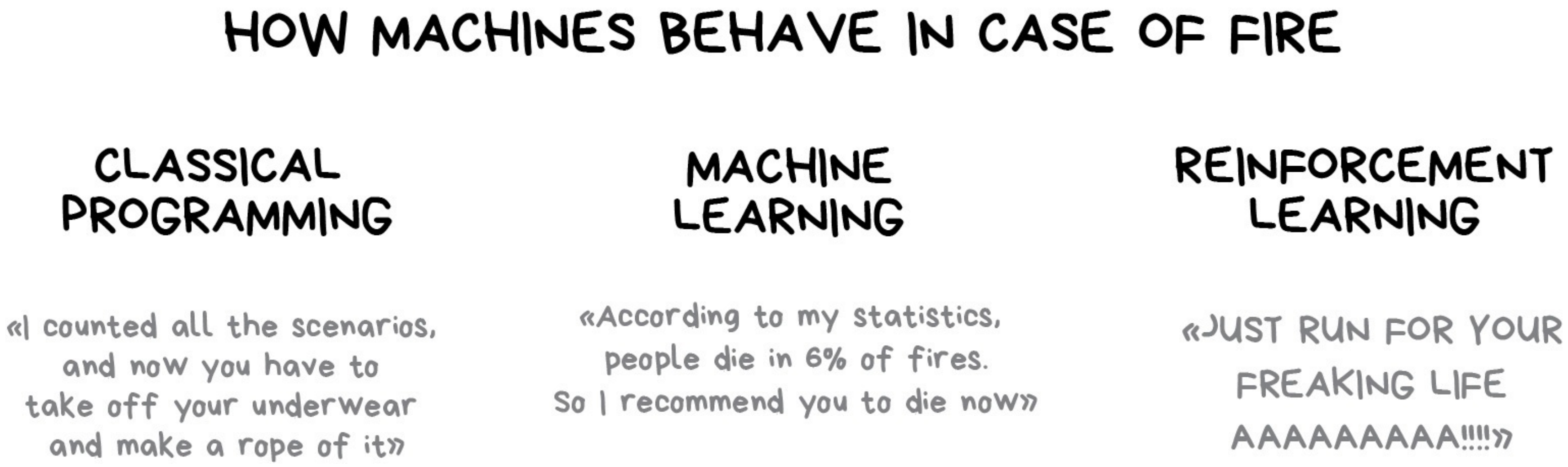
Reinforcement learning is used in cases when your problem is not related to data at all, but you have an environment to live in. Like a video game world or a city for self-driving car.





Nowadays used for:
Self-driving cars
Robot vacuums
Games
Automating trading
Enterprise resource management

Popular algorithms: Q-Learning, SARSA, DQN, A3C, Genetic algorithm

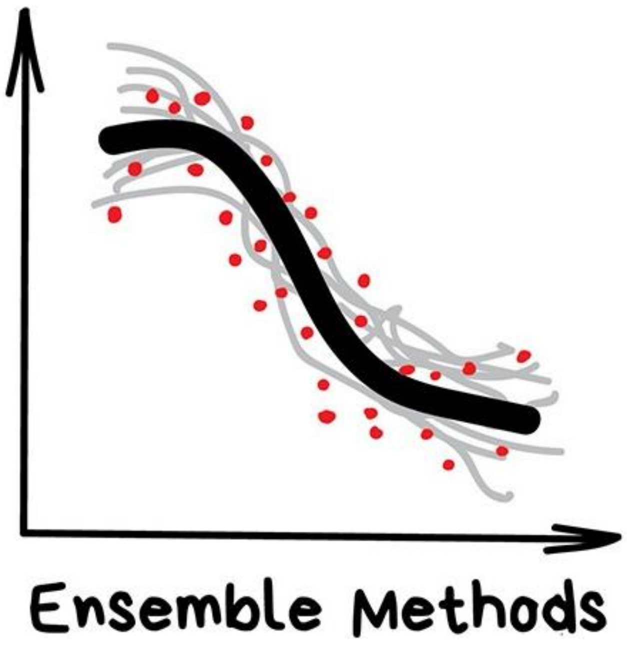


1.4 ENSEMBLE METHODS

"Bunch of stupid trees learning to correct errors of each other"

Nowadays is used for:
Everything that fits classical algorithm approaches (but works better)
Search systems (★)
Computer vision
Object detection

Popular algorithms: Random Forest, Gradient Boosting



It's time for modern, grown-up methods. Ensembles and neural networks are two main fighters paving our path to a singularity. Today they are producing the most accurate results and are widely used in production.

Despite all the effectiveness the idea behind these is overly simple. If you take a bunch of inefficient algorithms and force them to correct each other's mistakes, the overall quality of a system will be higher than even the best individual algorithms.

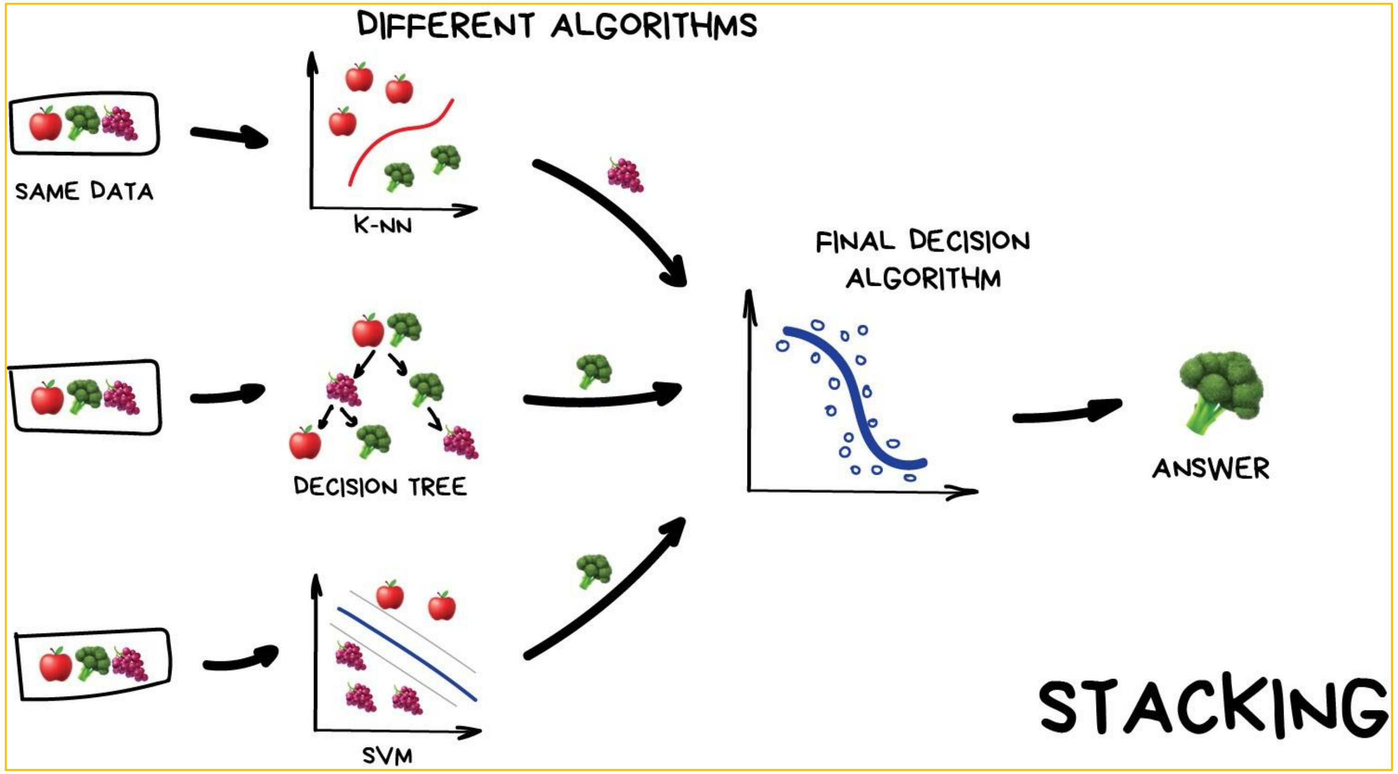
1.41 STACKING

There are three battle-tested methods to create ensembles.

Stacking Output of several parallel models is passed as input to the last one which makes a final decision.

Emphasis here on the word "different". Mixing the same algorithms on the same data would make no sense. The choice of algorithms is completely up to you. However, for final decision-making model, regression is usually a good choice.

Stacking is less popular in practice, because two other methods are giving better accuracy.

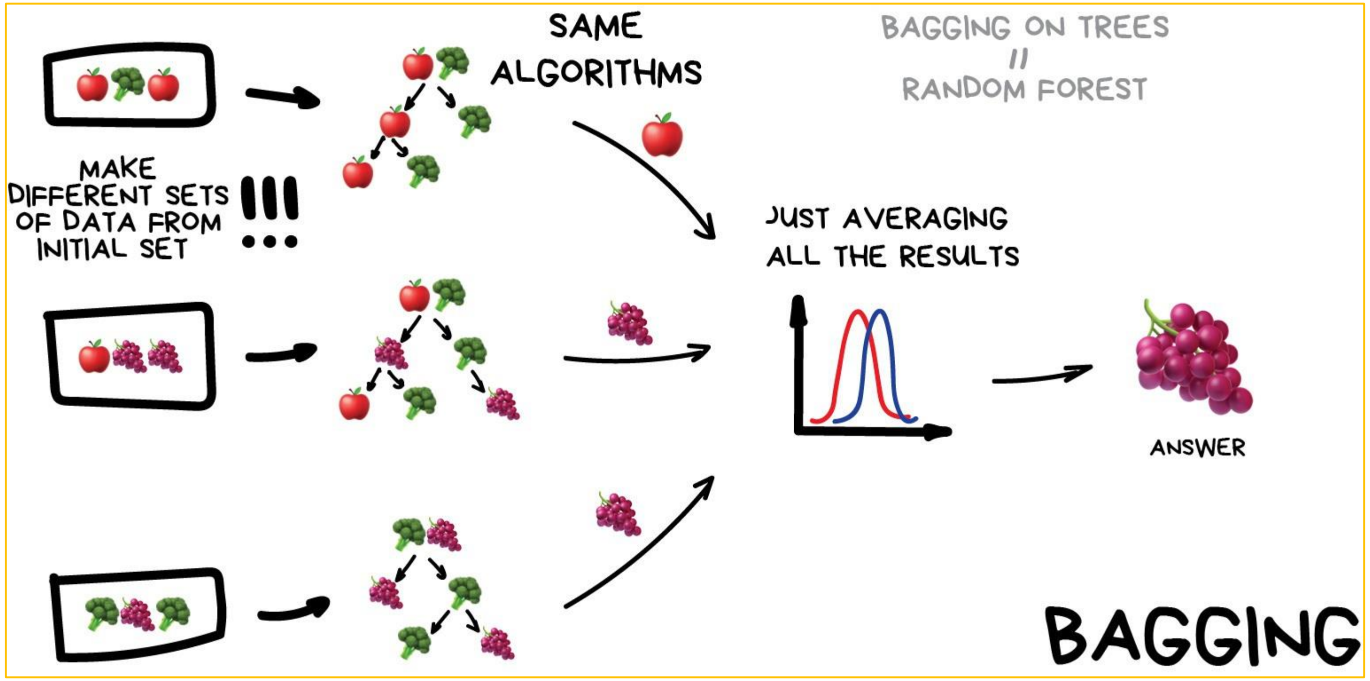


1.42 BAGGING

There are three battle-tested methods to create ensembles.

Bagging aka Bootstrap Aggregating. Use the same algorithm but train it on different subsets of original data. In the end — just average answers.

Data in random subsets may repeat. For example, from a set like "1-2-3" we can get subsets like "2-2-3", "1-2-2", "3-1-2" and so on. We use these new datasets to teach the same algorithm several times and then predict the final answer via simple majority voting.



1.42 BAGGING

There are three battle-tested methods to create ensembles.

The most famous example of bagging is the Random Forest algorithm, which is simply bagging on the decision trees. When you open your phone's camera app and see it drawing boxes around people's faces — it's probably the results of Random Forest work. Neural networks would be too slow to run real-time yet bagging is ideal given it can calculate trees on all the shaders of a video card or on these new fancy ML processors.

In some tasks, the ability of the Random Forest to run in parallel is more important than a small loss in accuracy to the boosting, for example. Especially in real-time processing. There is always a trade-off.

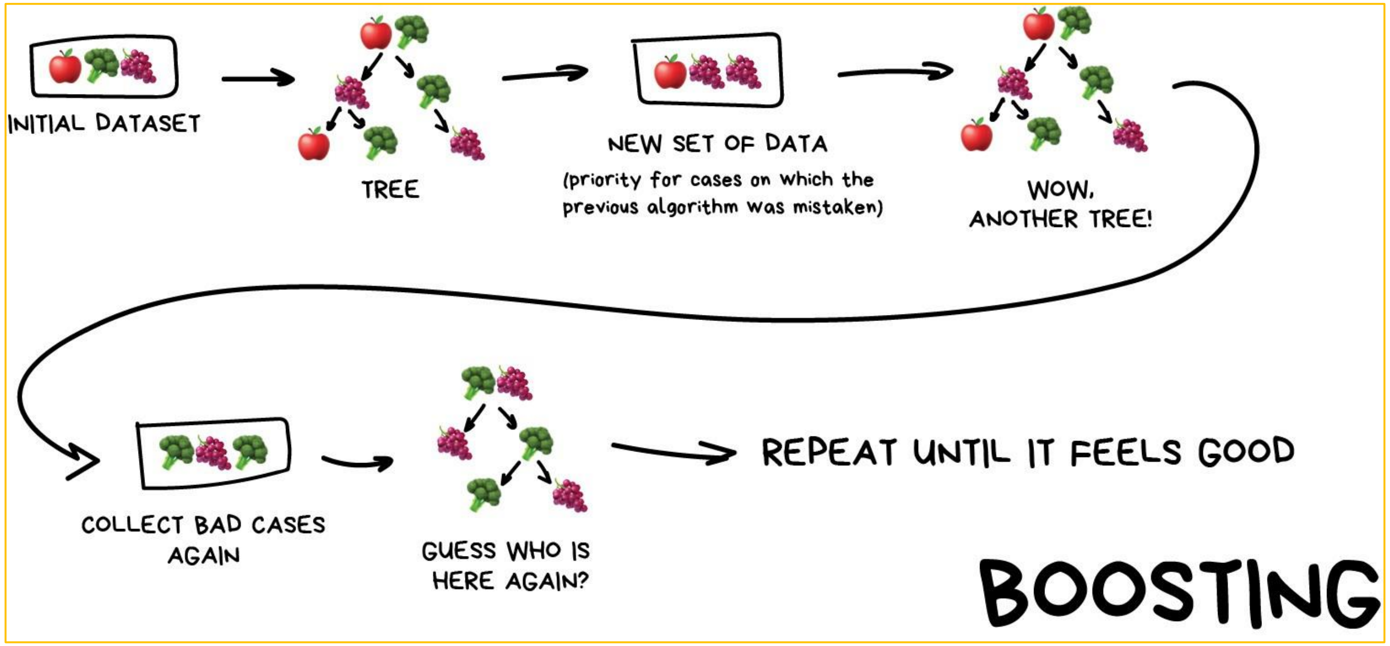
Nowadays there are three popular tools for boosting, [CatBoost vs. LightGBM vs. XGBoost]

1.43 BOOSTING

There are three battle-tested methods to create ensembles.

Boosting Algorithms are trained one by one sequentially. Each subsequent one paying most of its attention to data points that were mispredicted by the previous one. Repeat until you are happy.

Same as in bagging, we use subsets of our data but this time they are not randomly generated. Now, in each subsample we take a part of the data the previous algorithm failed to process. Thus, we make a new algorithm learn to fix the errors of the previous one.



Faster than neural networks.

If you want a real example of boosting — open Facebook or Google and start typing in a search query. Can you hear an army of trees roaring and smashing together to sort results by relevancy? That's because they are using boosting.

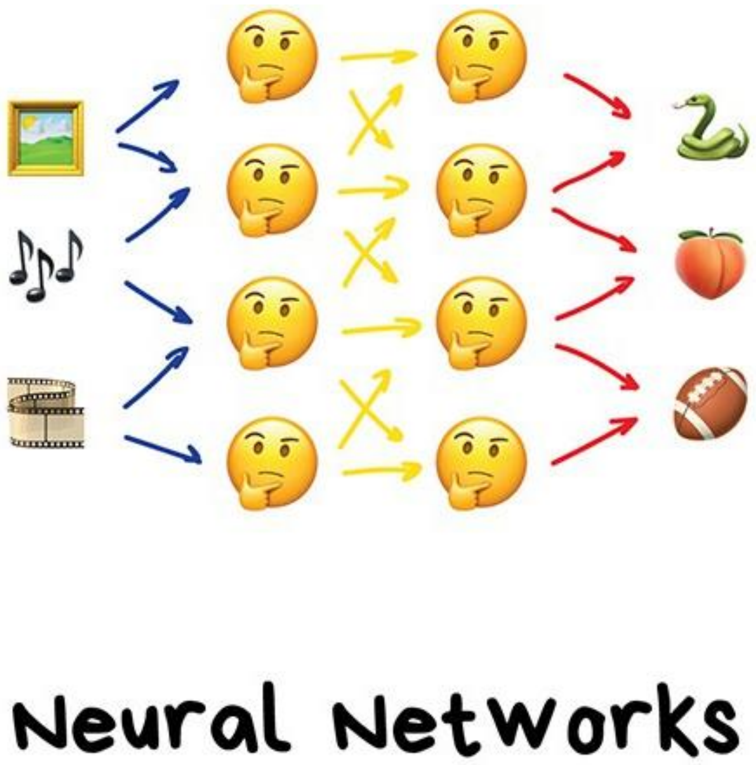
1.5 NEURAL NETWORKS AND DEEP LEANING

"We have a thousand-layer network, dozens of video cards, but still no idea where to use it. Let's generate cat pics!"

Used today for:

- Replacement of all algorithms above
- Object identification on photos and videos
- Speech recognition and synthesis
- Image processing, style transfer
- Machine translation

Popular architectures: Perceptron, Convolutional Network (CNN), Recurrent Networks (RNN), Autoencoders




நன்றி
Thank You!