* + 1. **Artificial intelligence**

Artificial Intelligence is, “*the effort to automate intellectual tasks normally performed by humans”.*

When you hard code rules in a program, that is called “Symbolic AI”. This type of AI occurred between the 50’s – 80’s. Symbolic AI is not good at image classification, speech recognition, or language translation. Drumroll ……. Machine Learning.

* + 1. **Machine Learning**

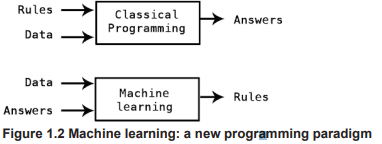
Charles Babbage, the inventory of the “Analytical Engine”. A general-purpose computer for calculating math problems.

*"The Analytical Engine has no pretensions whatever to originate anything. It can do whatever we know how to order it to perform… Its province is to assist us in making available what we are already acquainted with."*

Alan Turing, pioneer of AI, introduced the “Turing test” as well as key concepts used to shape AI.

Classical programming or “Symbolic AI” writes the rules and then produces answers.

Machine learning gives the machine the data and answers and the program comes up with the rules.



Machine learning is related to statistics except for the fact that machine learning tends to prove ideas empirically rather than theoretically. Because datasets are so large, it is unpractical to use Bayesian statistics.

* + 1. **Learning Representation from Data**

We need three things for machine learning:

* Input Data Points. i.e. sound files for speech recognition
* Examples of Expected Output. For speech recognition, it could be human-generated transcripts of our sound files.
* A way to measure if the algorithm is doing a good job, to measure its current output and its expected output. This is used as a feedback signal to adjust the way the algorithm works. i.e. “Learning”

With machine learning we want to meaningfully transform the data.

Machine learning is searching for useful representations of some input data, within a pre-defined space of possibilities, using guidance from some feedback signal.

* + 1. **“The “deep” in deep learning**

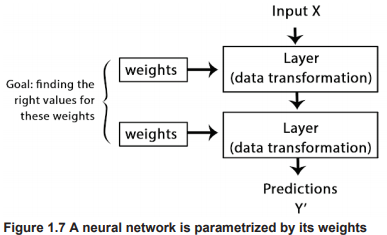
With deep learning, the learning occurs in successive “layers”. In “deep’ learning, the successive count of layers is what makes the process a “deep” learning process. Often these layers are in the hundreds or thousands. Machine learning only requires one or two layers of representation, hence being called “shallow learning”.

Deep learning requires “neural networks” to build these layers. *For our purposes, deep learning is merely a mathematical framework for learning representations from data.*

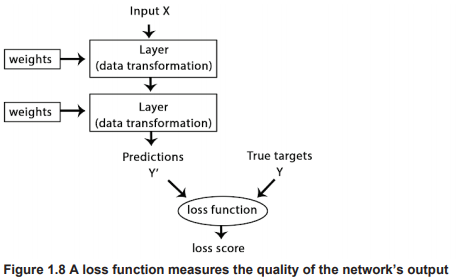
Each layer of a neural network used for deep learning distills information and purifies it from the input data and makes the input useful for the task.

* + 1. **Understanding how deep learning works in three figures**

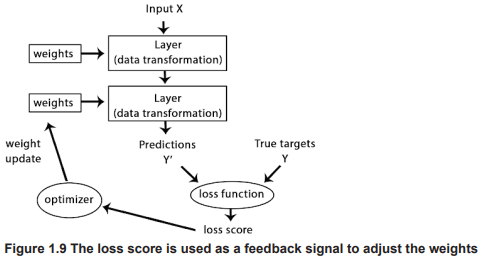
The information or specification of what each layer is doing is stored in each layer “weights”. Each layer is *“parametrized”* by its weights. The weights are also called the *“parameters”* of the layer. Learning is finding the correct weights for each layer for the task. It can be extremely complex given that changing one weight will affect the behavior of all the other layers and their weights.



There is a “loss function” or “objective function” that measures the accuracy of the output or prediction from the model. The model will produce a distance score from what the expected was and what it produced to measure accuracy.



The model adjusts lower the “loss score”, which is the job of the “optimizer”. The “optimizer” implements a “backpropagation” algorithm, a fundamental central algorithm to deep learning.



Initially, the weights of the network are assigned random values. With every new example, the network adjusts in the right direction and the “loss score” decreases. This “training loop” is done tens of thousands of times and yields a network with a minimized loss function. The result is a trained network. Once scaled up, the network seems to work and look like magic i.e. a “black box”.

* + 1. **What Deep Learning has achieved so far**

Deep learning has achieved:

* Near-human level image classification.
* Near-human level speech recognition.
* Near-human level handwriting transcription.
* Improved machine translation.
* Improved text-to-speech conversion.
* Digital assistants such as Google Now or Amazon Alexa.
* Near-human level autonomous driving.
* Improved ad targeting, as used by Google, Baidu, and Bing.
* Improved search results on the web.
* Answering natural language questions.
* Superhuman Go playing.
  + 1. **Don’t Believe the short-term hype**

Although cars are almost autonomous, we are nowhere near “human -level general intelligence” and any media equating it to should not be taken too seriously. The hype is detrimental to the industry when expectations are high and the promises are under delivered. Funding gets pulled from these projects and marks an “AI Winter”.

Every 20 years something like this happens. It can be traced back to the first “AI Winter” in the 60’s with the invention of symbolic AI. The same thing happened in the 80’s with an adaptation to the symbolic AI. Initial optimism of the technology flooded the industry with funding and when the technology could not deliver, funding eventually dried up.

The best approach is to have modest expectations for the short term as the “deep learning” is a relatively new field (the idea was initially introduced in 2010), although AI has been studied for roughly 60 years now.

* + 1. **The promise of AI**

AI has yet to impact our daily lives than what our smart phone or Google can offer us publicly. Just like the introduction of the internet in 1995, most people couldn’t find use of it in their daily lives. AI can be viewed the same way. AI does not currently impact our daily lives but in the future, will have a profound daily impact projecting humanity forward. AI will assist in assisting humans in daily activities to discovering break thoughts across all scientific fields. AI may go through the bubble and crash much like the internet did in the 2000’s but when the dust settles, it will have the same effect and transform our world in a fantastic way.

* + 1. **Probabilistic Modeling**

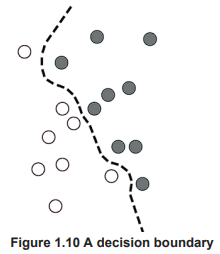
Probabilistic modeling finds it roots in the principles of statistics to data analysis. One of the most widely known algorithms is the Naïve Bayes algorithm. This algorithm dates to the 18th century well before computers. The Logistic Regression which is closely related is a classification algorithm for binary outcomes. The Logistic Regression also predates computers and is still widely used to this day. It is typically the first test a data scientist will try on a dataset to get a feel for the classification task at hand.

* + 1. **Early Neural Networks**

The initial idea for a neural network dates back to the 50’s but didn’t make its way into data science until the mid-80’s when researchers independently discovered “backpropagation”, allowing a way to train chains of parametric operations using gradient descent optimization and applied them to neural networks.

* + 1. **Kernel Methods**

Kernel methods are a group of classification algorithms, such as Support Vector Machine (SVM). SVM attempts to find a “decision boundary” between two sets of data points that belong to two different categories.



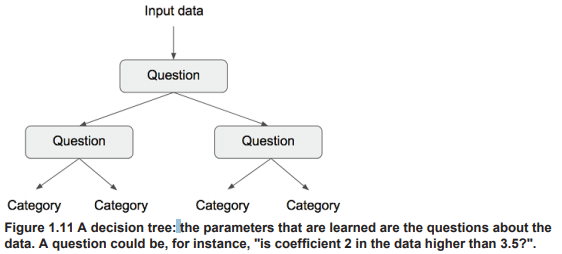
SVMs work in two steps:

1. The data is mapped to the high-dimensional representation where the boundary can be expressed as an “hyperplane”
2. Then the decision boundary is computed trying to maximize the distance between the hyperplane and the closest data points from each class in a step called “maximizing the margin”.

SVM excelled at simple classification problems and was widely popular because the theory and concept was well-understood and results were interpretable. However, scaling SVM is quite difficult give that SVM is a “shallow” neural network. What holds SVM back is when you have a many features you must manually decide which features to keep for the algorithm. Selection of the features is called “feature engineering”.

* + 1. **Decision trees, Random Forests, and Gradient Boosting Machines**

Decision Trees grew popular over kernel methods because they were easy to visualize and interpret. By 2010, decision trees were often preferred over kernel methods.



Random forests are an extension of decision trees as they generalize multiple iterations of the logic begin decision trees. By 2014 random forests dominated the scene for competitions on websites such as Kaggle.com. However, Gradient Boosting Machines took over. Gradient boosting works by iteratively training new models that specialize in addressing the weak points of the previous model. Gradient boosting machines is one of the best algorithms aside from deep learning for dealing with non-perceptual data today. It is one of the most commonly used techniques used in Kaggle competitions.

* + 1. **Back to Neural Networks**

In 2010, neural networks were mostly out of fashion until Dan Ciresan from IDSIA started winning image classification competitions using his early modern deep learning program. Since then, deep neural nets are the go-to algorithm for all computer vision tasks. Deep learning also found its way into natural language processing and have mostly replaced SVMs and decision trees in a wide range of applications. For example, the Large Hadron Collider (HLC) used to use decision tree-based models and now uses Keras-based neural networks due to their higher performance and ease of training on large datasets.

* + 1. **What Make Deep Learning Different**

Deep learning is great because it automates feature selection. Unlike a SVM where each layer of the process has a set importance no matter how far you refine the process, with deep learning processing the layers all have equal weight to the output of the data so each layer is necessary for the functioning of the process.

These are the two essential characteristics of how deep learning learns from data: the incremental, layer-by-layer way in which increasingly complex representations are developed these intermediate incremental representations are learned , and the fact jointly, each layer being updated both to follow the representational needs of the layer above and the needs of the layer below. Together, these two properties have made deep learning vastly more successful than previous approaches to machine learning.

* + 1. **The modern machine learning landscape**

In 2016, Kaggle is dominated by two approaches: gradient boosting machines, and deep learning. Practitioners of the former almost always use the excellent XGB library, which offers support for the two most popular languages of data science: Python and R. Meanwhile, most of the Kaggle entrants leveraging deep learning use the Keras library, due to its easy of use, flexibility and support of Python.

* 1. **Why Deep Learning, why now?**

Three forces driving deep learning to flourish:

* Hardware
* Datasets and benchmarks
* Algorithmic advances

Machine learning is not mathematics of physics, where major advances can be done with pen and paper. It is an engineering science. in the 90’s-20’s the bottleneck was cause by data and hardware. With the invention of the internet, all of that has changes.

* + 1. **Hardware**

It’s all about the GPU or in Google’s case, their “TPU” (tensor processing unit) for running deep neural networks.

* + 1. **Data**
    2. **Algorithms**