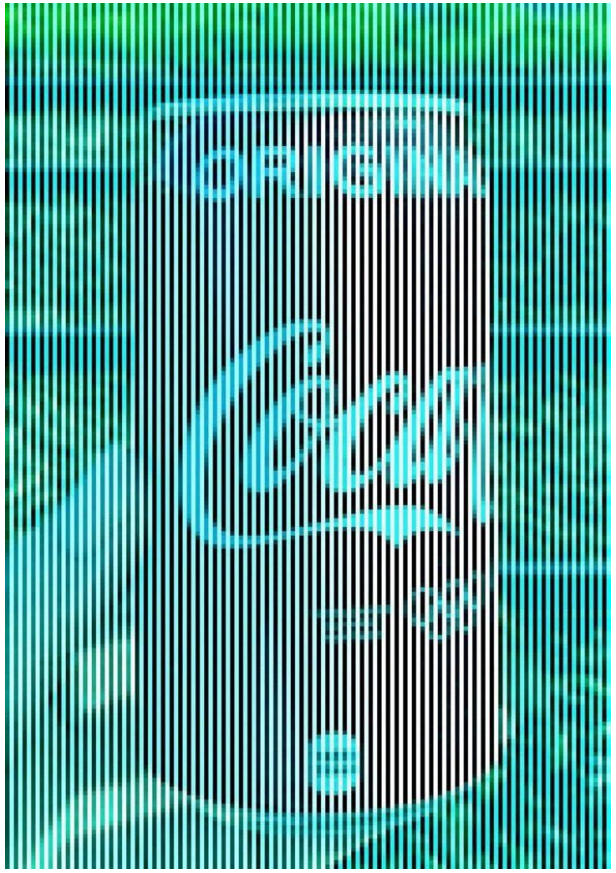
Machine Learning

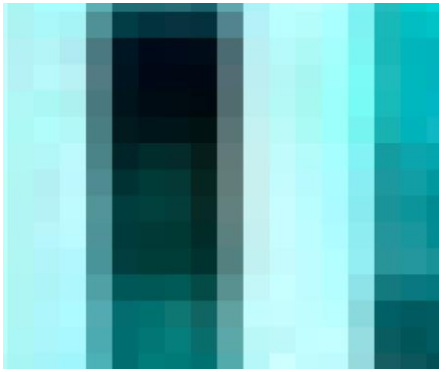
What is the intuition behind it and how do we apply it to problems in Neuroscience?

Think about complexity in languages, for instance in German:

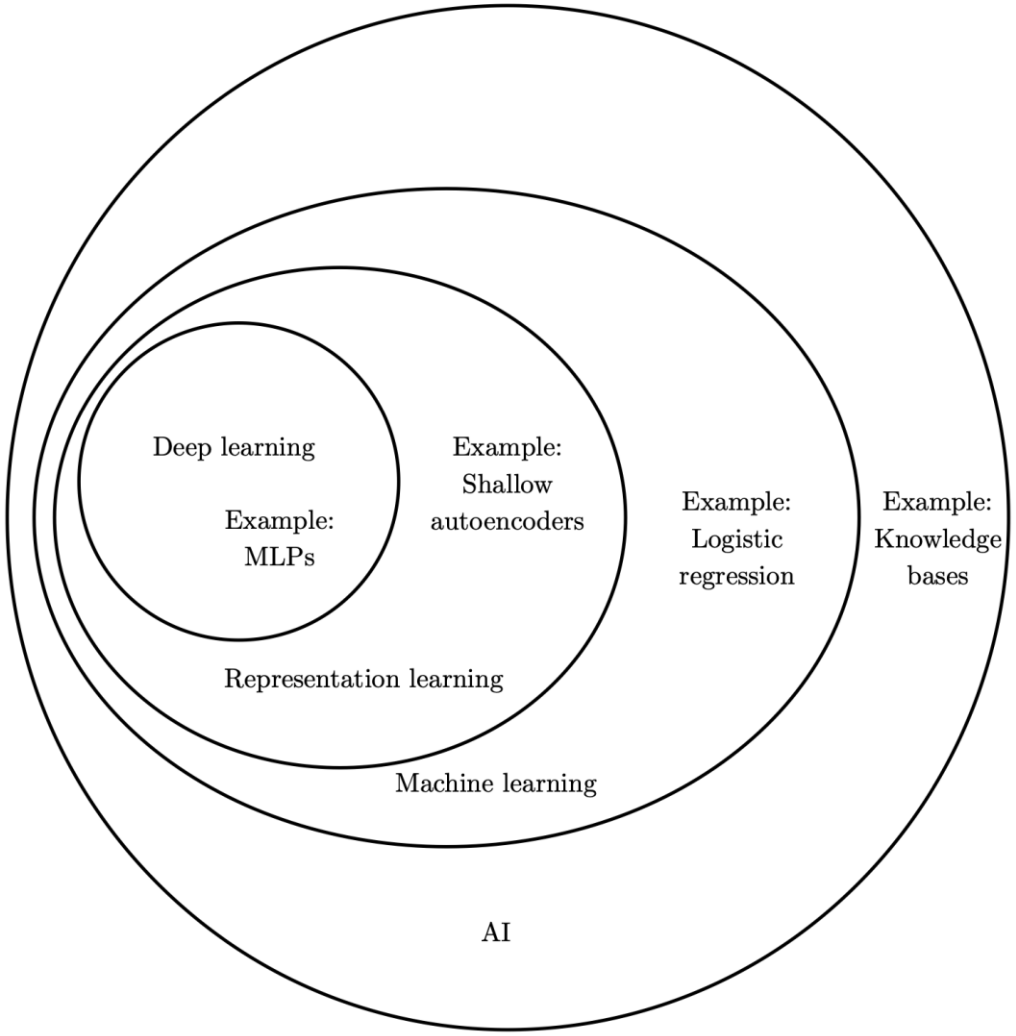
- you have to distinguish between “the” and “a” articles
- then each one has four case forms
- There are three genders
- singular and plural forms
- then the adjectives have to agree
- then there are verbs!

Patterns and generalisation

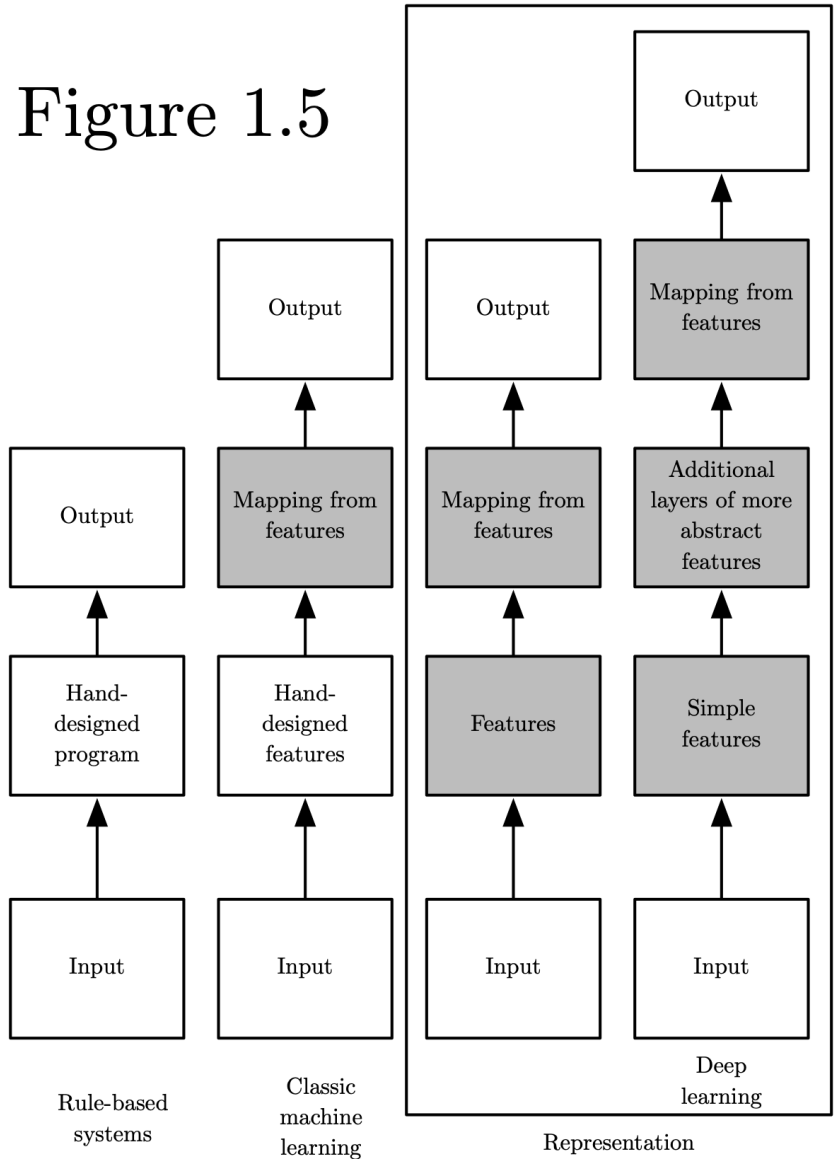




AI and machine learning



Categories of learning and intelligent systems

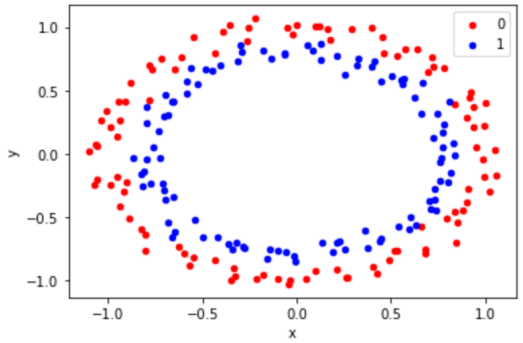


Machine learning

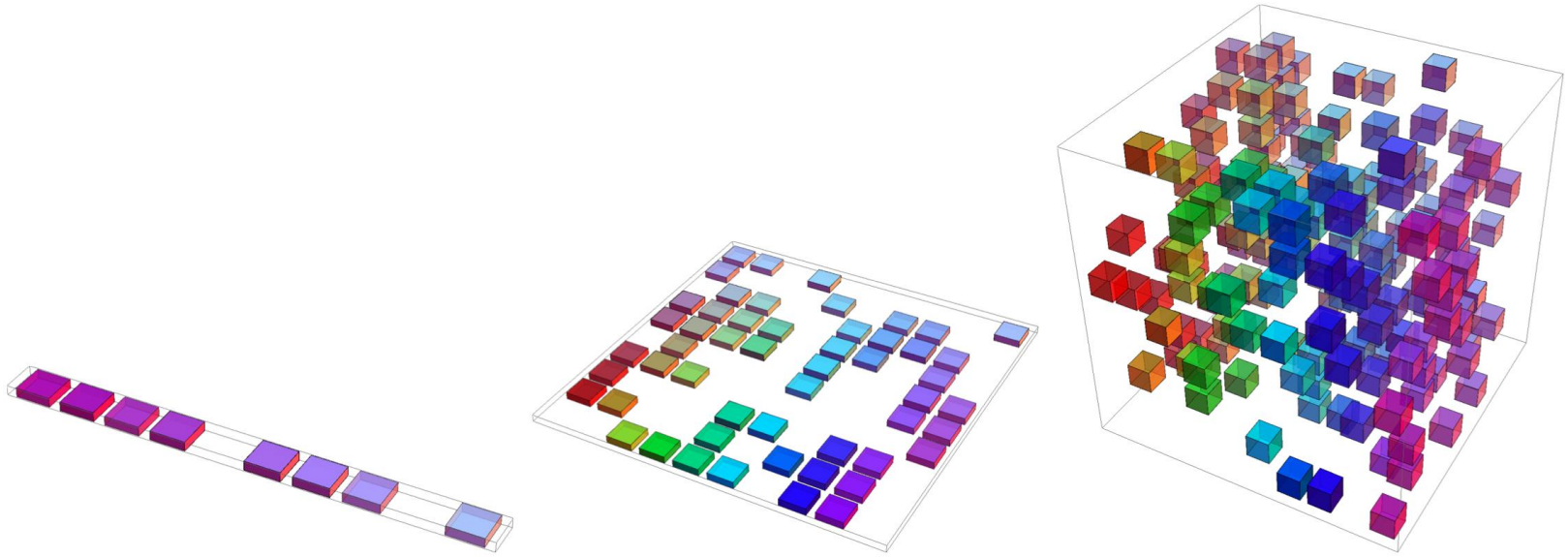
- Most of machine learning methods designed as a learning process to learn a function `f*` for which:

$$f^*(x) \approx f^*(x + \varepsilon)$$

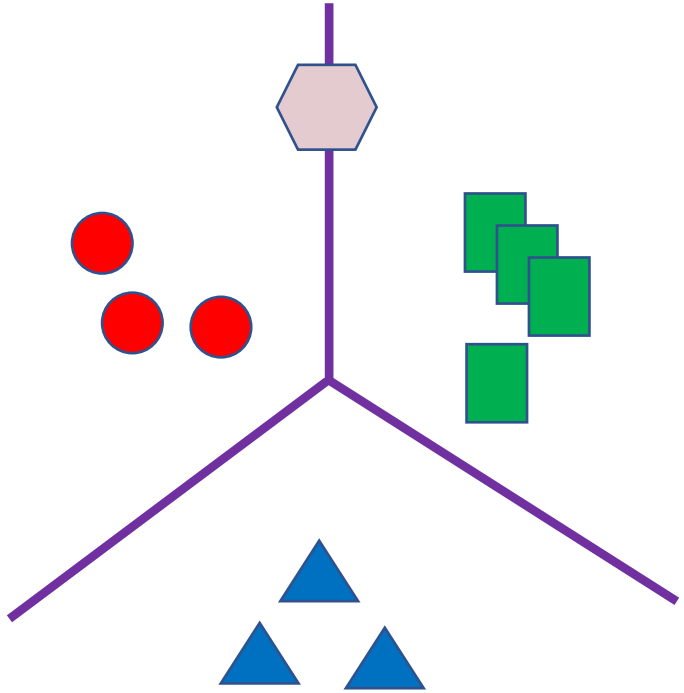
- In other words, this means if we know the answer for a training sample x, then that answer is *probably* good in the neighborhood of x.



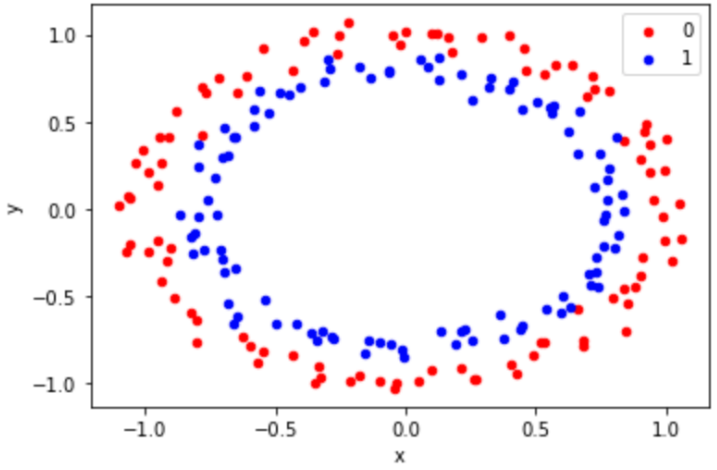
Curse of dimensionality



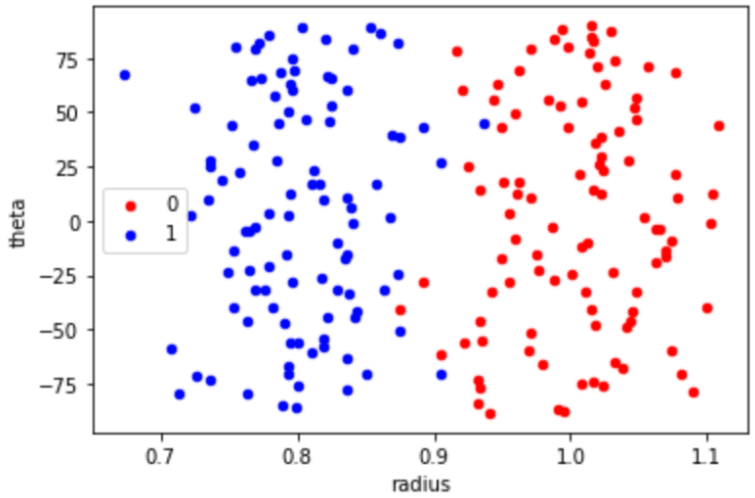
K-nearest neighbors



Data/sample representation



(x,y)



$$r = \sqrt{x^2 + y^2}$$

$$\theta = \tan^{-1}\left(\frac{y}{x}\right).$$

Training set and testing set

- Machine learning is about learning some properties of a data set and applying them to new data.

- This is why a common practice in machine learning to evaluate an algorithm is to split the data at hand into two sets, one that we call the training set on which we learn data properties and one that we call the testing set on which we test these properties.

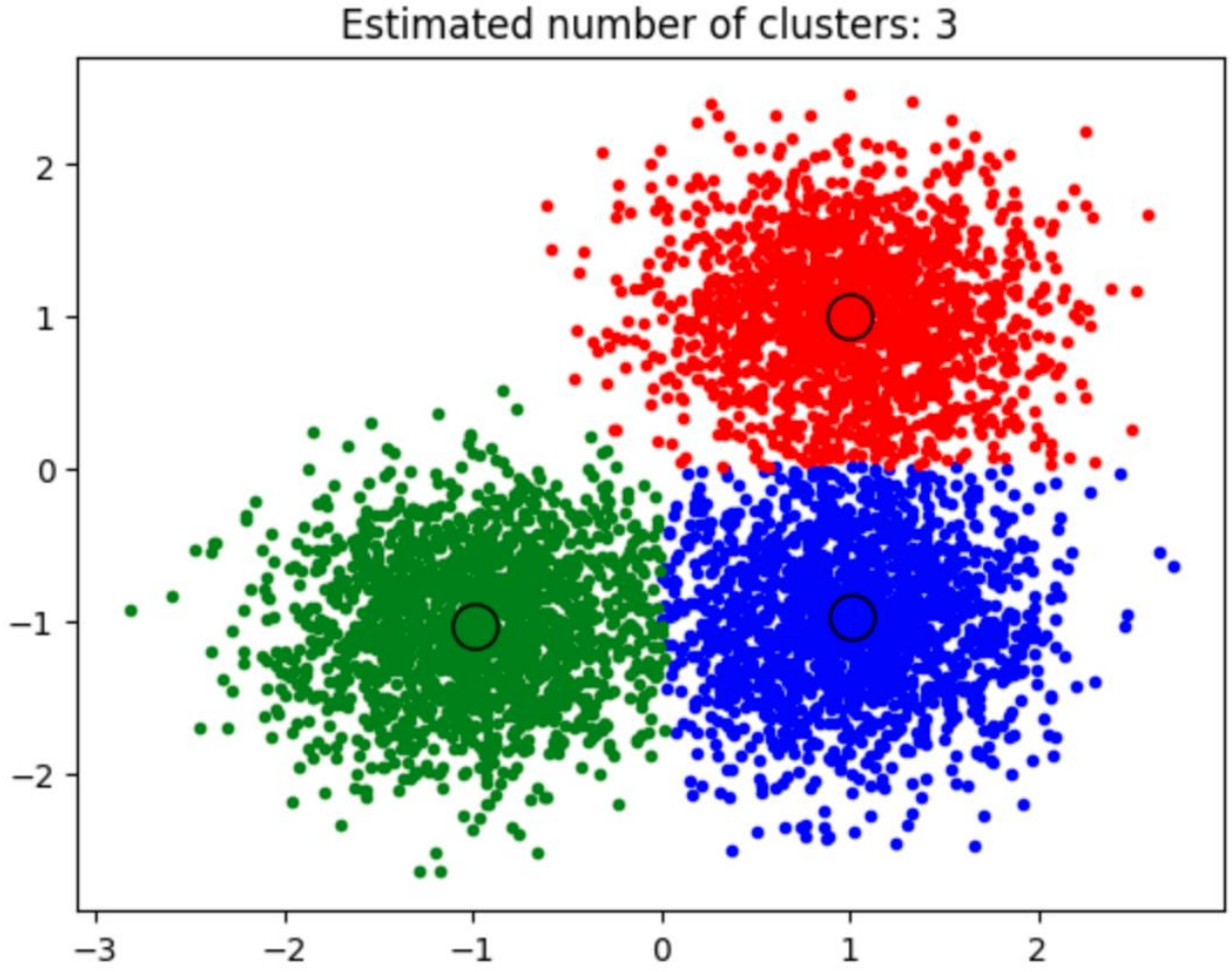
Clustering

- The problem that needs to be solved in clustering:
  - For example, given a dataset, if we knew that there were 3 types of data, but did not have access to a taxonomist (i.e. labels) to directly identify them: we could try a clustering task.
  - This means splitting the data into well-separated group called clusters.

K-means clustering

- The $KMeans$ algorithm clusters data by trying to separate samples in n groups of equal variance, minimising a criterion known as the inertia or within-cluster sum-of-squares.
- This algorithm requires the number of clusters to be specified. It scales well to large number of samples and has been used across a large range of application areas in many different fields.
- The k-means algorithm divides a set of N samples X into K disjoint clusters C, each described by the mean μ_j of the samples in the cluster.

K-means

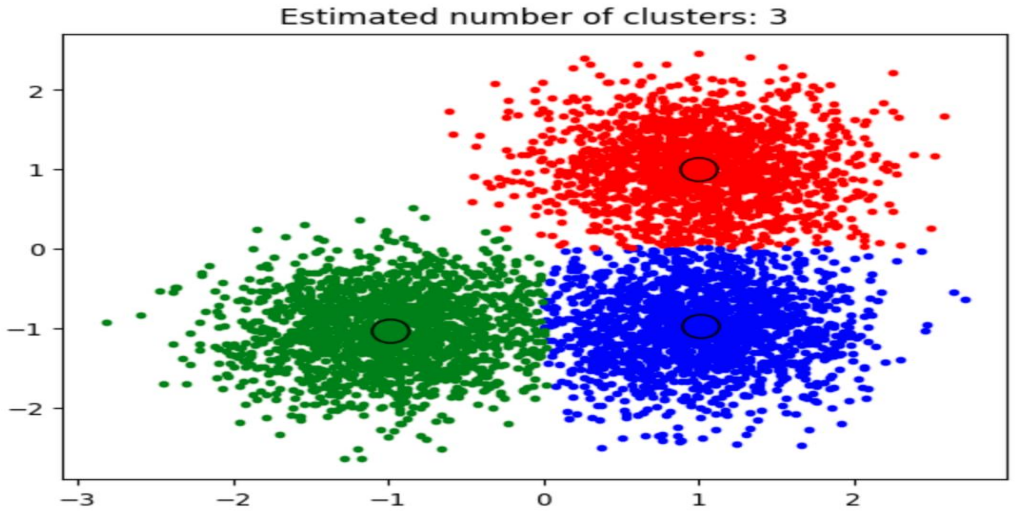


K-Means

- The means are commonly called the cluster "centroids"; note that they are not, in general, points from X, although they live in the same space.
- The K-means algorithm aims to choose centroids that minimise the inertia, or within-cluster sum of squared criterion:

$$\sum_{i=0}^{n} \min(\|\mu_j - \mu_i\|^2)$$

$$\mu_j \in C$$



Convergence in K-means

- Given enough time, K-means will always converge, however this may be to a local minimum. This is highly dependent on the initialisation of the centroids.
- As a result, the computation is often done several times, with different initialisations of the centroids.

Neuroscience in ML

- Machine learning could help to provide a systems neuroscience-level view of the brain.
- It can help to view the network, architecture, functions, and representations that the brain utilises.
- The precise mechanisms by which the processes/interactions are physically realised in a biological substrate are often less relevant at the implementation level (of AI models).

Transferrable ideas from neuroscience

- By focusing on the computational and algorithmic levels, we can obtain transferrable insights into general mechanisms of brain function while leaving room to accommodate the distinctive opportunities and challenges that arise when building intelligent machines.

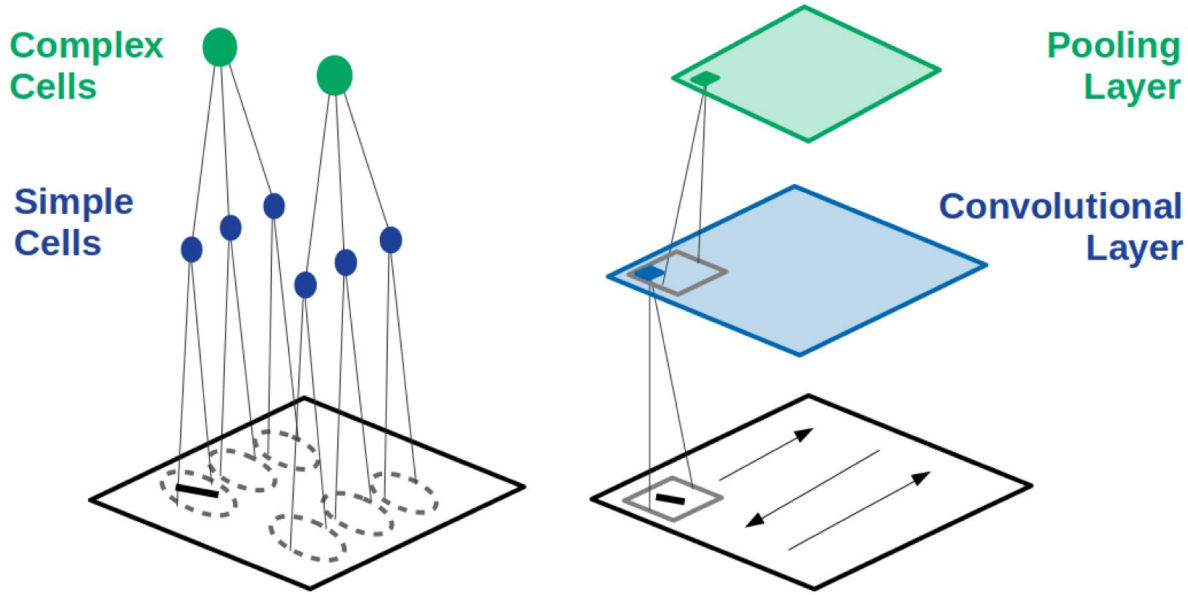
- We can model the neuronal networks of ion channels and electrical impulses, peak voltages and synapses as sophisticated Digital Logic circuits or electrical equivalents and use Machine Learning to model the behaviour of a network of neurons after application of a single stimulus.

Deep learning and biological systems

- In both biological and artificial systems, successive non-linear computations transform raw visual input into an increasingly complex set of features, permitting object recognition that is invariant to transformations of pose, illumination, or scale.

CCNs and visual cortex

- Hubel and Wiesel discovered that simple cells (left, blue) have preferred locations in the image (dashed ovals) wherein they respond most strongly to bars of particular orientation. Complex cells (green) receive input from many simple cells and thus have more spatially invariant responses.



Attention-based models

- The brain does not learn by implementing a single, global optimisation principle within a uniform and undifferentiated neural network.
- Biological brains are modular, with distinct but interacting subsystems underpinning key functions such as memory, language, and cognitive control.

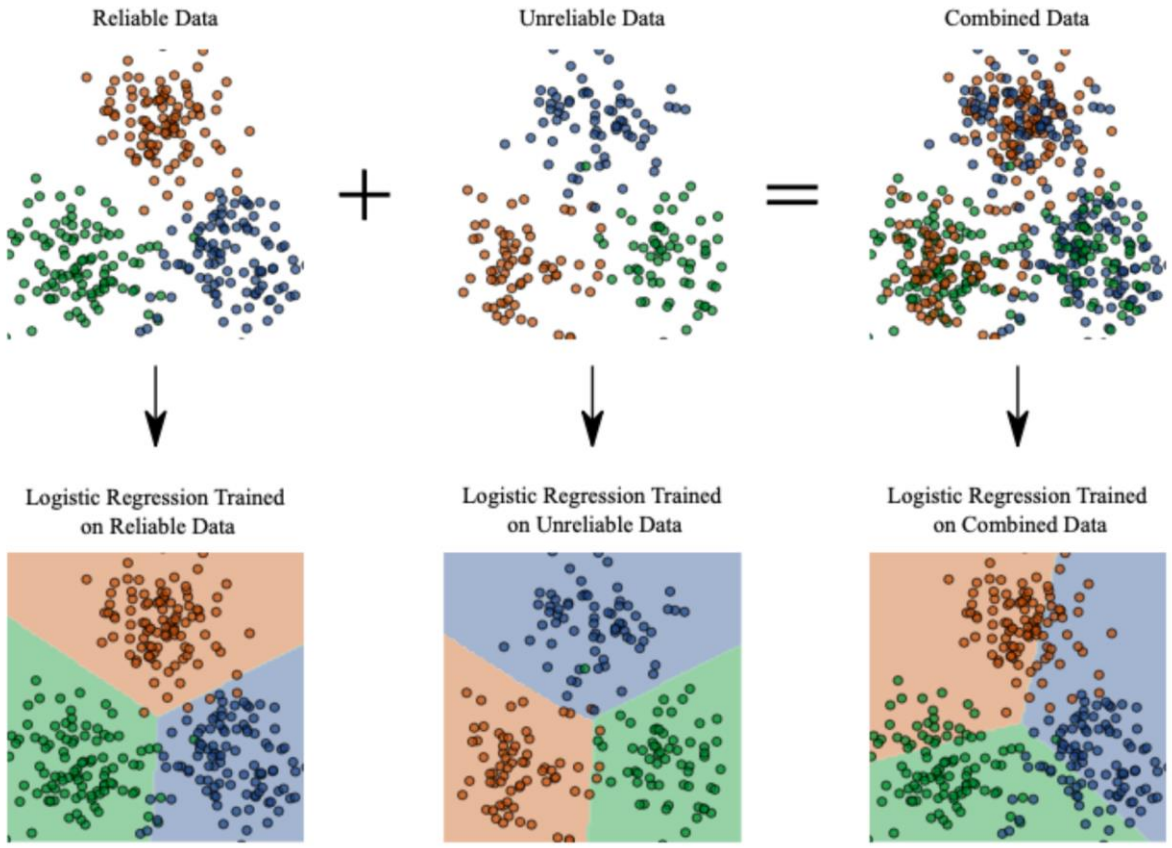
- One illustrative example is recent AI work on attention. Up until quite lately, most CNN models worked directly on entire images or video frames, with equal priority given to all image pixels at the earliest stage of processing. The primate visual system works differently.

Neuroimaging and continual learning

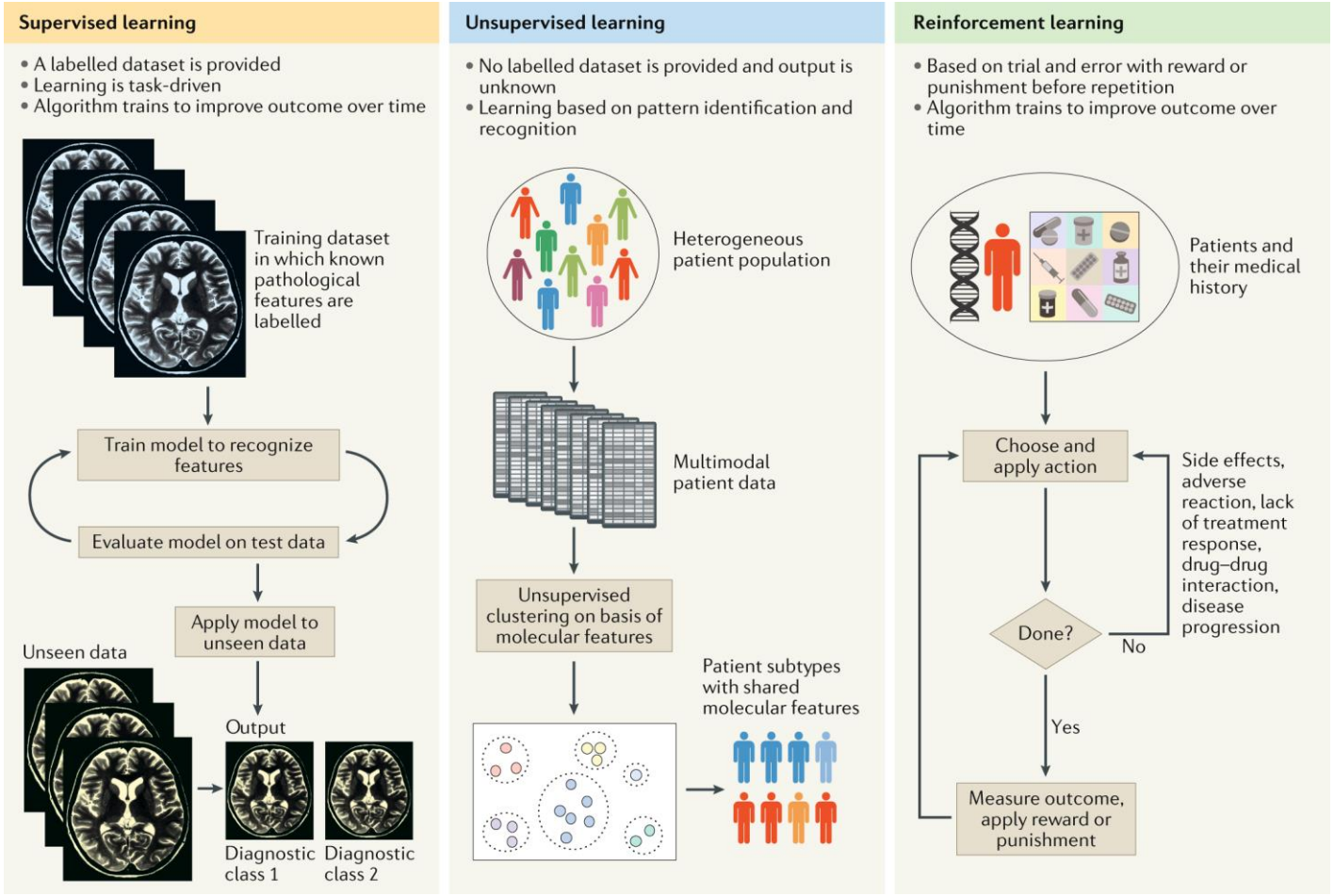
- In neuroscience, advanced neuroimaging techniques (e.g., two-photon imaging) now allow dynamic in vivo visualization of the structure and function of dendritic spines during learning, at the spatial scale of single synapses.

- This approach can be used to study neocortical plasticity during continual learning.

Neuroplasticity and machine learning models



Applications of ML to diagnosis and treatment of neurodegenerative diseases



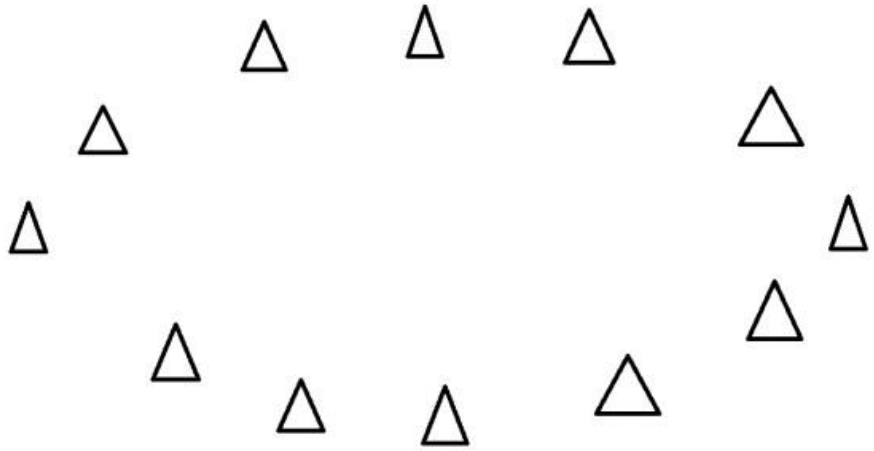
Source : Myszczynska, M.A., Ojamies, P.N., Lacoste, A.M.B. *et al.* Applications of machine learning to diagnosis and treatment of neurodegenerative diseases. *Nat Rev Neurol*16, 440–456 (2020).

Principal component analysis (PCA)

- PCA is used to decompose a multivariate dataset in a set of successive orthogonal components that explain a maximum amount of the variance.
- In scikit-learn, PCA is implemented as a transformer object that learns n components in its fit method, and can be used on new data to project it on these components.

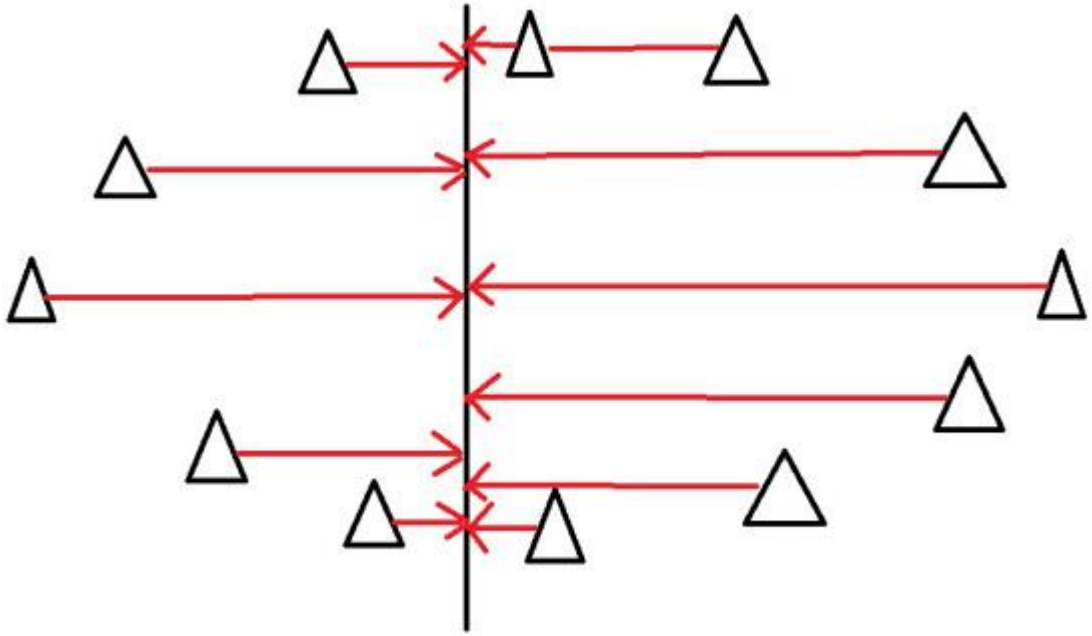
What are principal components?

- They are the underlying structure in the data.
- They are the directions where there is the most variance, the directions where the data is most spread out.

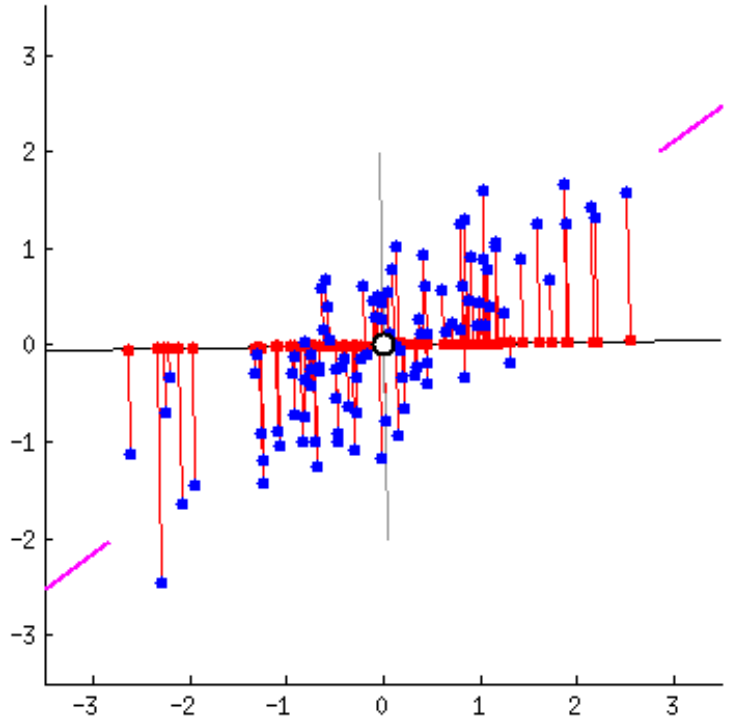


Direction where there is most variance

- Imagine that the triangles are points of data. To find the direction where there is most variance, find the straight line where the data is most spread out when projected onto it. A vertical straight line with the points projected on to it will look like this:



PCA



Eigenvectors and eigenvalues

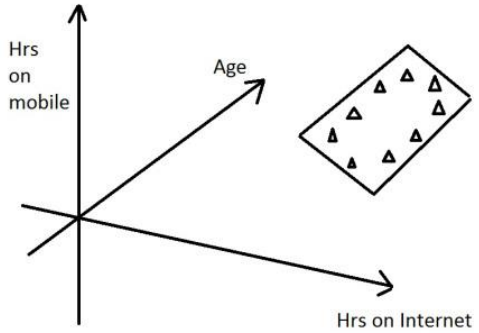
- Eigenvectors and values exist in pairs: every eigenvector has a corresponding eigenvalue.
- An eigenvector is a direction, in the previous example the eigenvector was the direction of the line (vertical, horizontal, 45 degrees etc.).
- An eigenvalue is a number, telling you how much variance there is in the data in that direction.
- In the previous example the eigenvalue is a number telling us how spread out the data is on the line. The eigenvector with the highest eigenvalue is therefore the principal component.

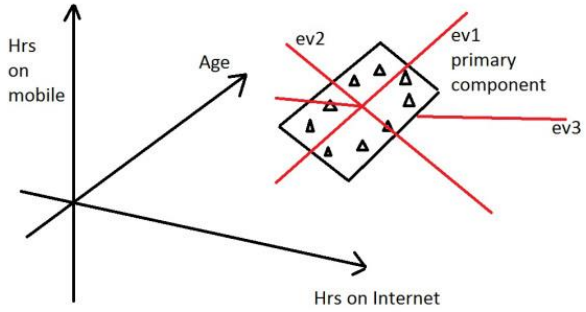
Dimensions and eigenvectors

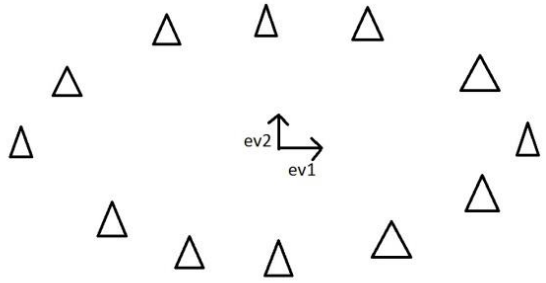
- The amount of eigenvectors/values that exist equals the number of dimensions the data set has.
- For example, if we measuring age and hours on the internet. there are 2 variables, it’s a 2 dimensional data set, therefore there are 2 eigenvectors/values.
- If we measure age, hours on internet and hours on mobile phone there’s 3 variables, 3-D data set, so 3 eigenvectors/values.
- The reason for this is that eigenvectors put the data into a new set of dimensions, and these new dimensions have to be equal to the original amount of dimensions.

Dimension Reduction

- PCA can be used to reduce the dimensions of a data set.
- Dimension reduction reduces the data down into it's basic components, stripping away any unnecessary parts.







Logistic regression

- We can generalise linear regression to the (binary) classification setting by making some changes.

- We compute a linear combination of the inputs, as before, but then we pass this through a function that ensures $0 \leq \mu(x) \leq 1$ by defining:

$$\mu(\mathbf{x}) = \text{sigm}(\mathbf{w}^T \mathbf{x})$$

The Sigmoid function

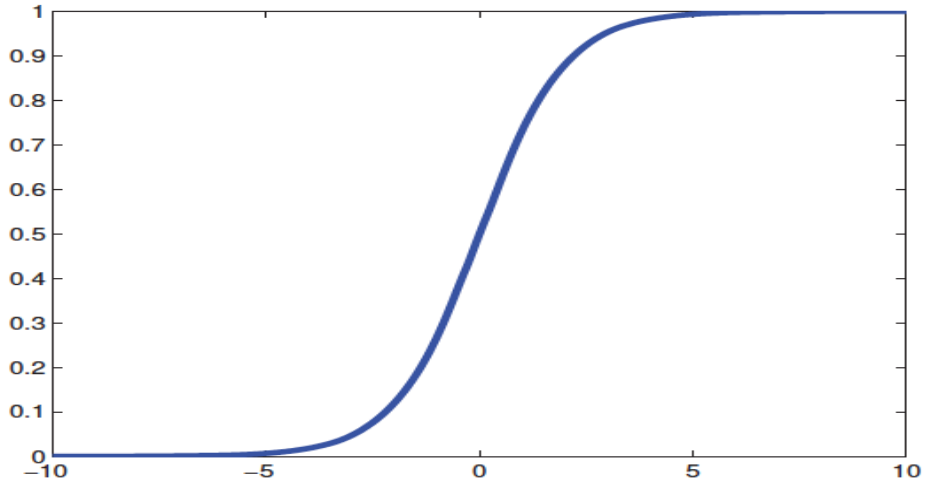
- where sigm(η) refers to the sigmoid function, also known as the *logistic* or *logit* function.
- This is defined as

$$\text{sigm}(\eta) \triangleq \frac{1}{1+\exp(-\eta)} = \frac{e^{\eta}}{e^{\eta}+1}$$

- The term "$sigmoid$" means S-shaped for a plot.
- It is also known as a squashing function, since it maps the whole real line to [0, 1].

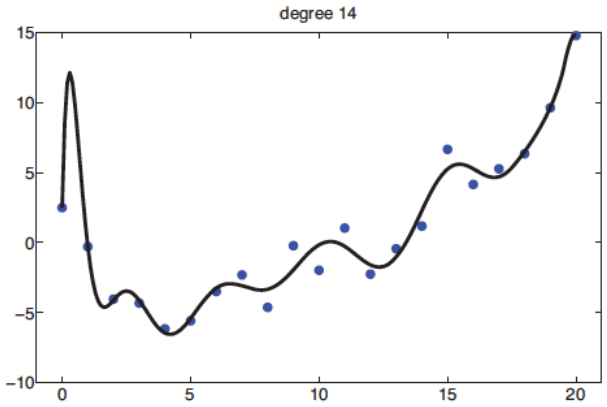
Logistic regression

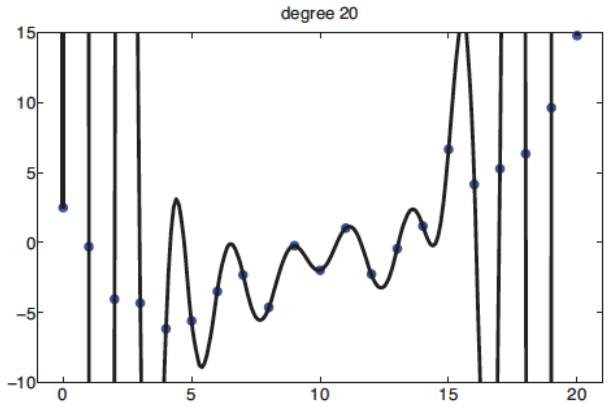
- The process of applying linear combination of the inputs, as before, but then we pass this through a logistics function is called logistic regression due to its similarity to linear regression (although it is a form of classification, not regression!).



Overfitting

- When we fit highly flexible models, we need to be careful that we do not overfit the data, that is, we should avoid trying to model every minor variation in the input, since this is more likely to be noise than true signal.





- This is illustrated in the figure above, where we see that using a high degree polynomial results in a curve that is very “wiggly”. It is unlikely that the true function has such extreme oscillations.
- Using such a model might result in accurate predictions of future outputs.

Information Theory

- The basic intuition behind information theory is that learning that an unlikely event has occurred is more informative than learning that a likely event has occurred.

- Likely events should have low information content, and in the extreme case, events that are guaranteed to happen should have no information content whatsoever.
- Less likely events should have higher information content.

Information Theory

- Independent events should have additive information.
- For example, finding out that a tossed coin has come up as heads twice should convey twice as much information as finding out that a tossed coin has come up as heads once.

- To satisfy all three of these properties, the self-information

of an event x = x is defined as:

$$I(x) = -\log p(x)$$

Shannon's entropy

- Self-information deals only with a single outcome.
- We can quantify the amount of uncertainty in an entire probability distribution using the Shannon entropy:

$$H = -\sum_{x} p(x) \,.\log p(x)$$

Goals for the coming months:

- Learn about GANs (Generative Adversarial Networks), Cyclic GANs and Computer Vision fundamentals.
- Understand the algorithmic constraints on PCA analysis and dimensionality reduction when the dataset in question has a very high number of unique, independent principle components
- Use the data from the new paper and train the data set on a cyclic self correcting GAN to see if there is a significant correlation between orientation maps of the visual cortex of mouse and the evoked response.