Unit-2
Deep Learning Architectures

Machine Learning and Deep Learning:

Machine learning is a subfield of artificial intelligence that focuses on the development of algorithms and statistical models that enable computers to learn and make predictions or decisions without being explicitly programmed. It involves training algorithms on large datasets to identify patterns and relationships and then using these patterns to make predictions or decisions about new data.

What are the Different Types of Machine Learning?

Machine learning is further divided into categories based on the data on which we are training our model.

- **Supervised Learning** – This method is used when we have Training data along with the labels for the correct answer.
- **Unsupervised Learning** – In this task our main objective is to find the patterns or groups in the dataset at hand because we don't have any particular labels in this dataset.

Deep Learning

Deep learning, on the other hand, is a subset of machine learning that uses neural networks with multiple layers to analyze complex patterns and relationships in data. It is inspired by the structure and function of the human brain and has been successful in a variety of tasks, such as computer vision, natural language processing, and speech recognition.
Deep learning models are trained using large amounts of data and algorithms that are able to learn and improve over time, becoming more accurate as they process more data. This makes them well-suited to complex, real-world problems and enables them to learn and adapt to new situations.

Future of Machine Learning and Deep Learning

Both machine learning and deep learning have the potential to transform a wide range of industries, including healthcare, finance, retail, and transportation, by providing insights and automating decision-making processes.

- **Machine Learning**: Machine learning is a subset, an application of Artificial Intelligence (AI) that offers the ability of the system to learn and improve from experience without being programmed to that level. Machine Learning uses data to train and find accurate results. Machine learning focuses on the development of a computer program that accesses the data and uses it to learn from itself.
- **Deep Learning**: Deep Learning is a subset of Machine Learning where the artificial neural network and the recurrent neural network come in relation. The algorithms are created

exactly just like machine learning but it consists of many more levels of algorithms. All these networks of the algorithm are together called the artificial neural network. In much simpler terms, it replicates just like the human brain as all the neural networks are connected in the brain, which exactly is the concept of deep learning. It solves all the complex problems with the help of algorithms and its process.

Difference Between Machine Learning and Deep Learning

S. No.	Machine Learning	Deep Learning
1.	Machine Learning is a superset of Deep Learning	Deep Learning is a subset of Machine Learning
2.	The data represented in Machine Learning is quite different compared to Deep Learning as it uses structured data	The data representation used in Deep Learning is quite different as it uses neural networks(ANN).
3.	Machine Learning is an evolution of AI.	Deep Learning is an evolution of Machine Learning. Basically, it is how deep is the machine learning.
4.	Machine learning consists of thousands of data points.	Big Data: Millions of data points.
5.	Outputs: Numerical Value, like classification of the score.	Anything from numerical values to free-form elements, such as free text and sound.
6.	Uses various types of automated algorithms that turn to model functions and predict future action from data.	Uses a neural network that passes data through processing layers to, interpret data features and relations.
7.	Algorithms are detected by data analysts to examine specific variables in data sets.	Algorithms are largely self-depicted on data analysis once they're put into production.
8.	Machine Learning is highly used to stay in the competition and learn new things.	Deep Learning solves complex machine-learning issues.
9.	Training can be performed using the CPU (Central Processing Unit).	A dedicated GPU (Graphics Processing Unit) is required for training.
10.	More human intervention is involved in getting results.	Although more difficult to set up, deep learning requires less intervention once it is running.

Representation Learning:

- "***Representation learning*** [means] learning representations of the data that make it easier to extract useful information when building classifiers or other predictors"
- **Goal**: learn only one/few representations for each domain, then learn *simple* predictors for different tasks.

Example

- You have a security system at home which can notify you when you're away.
- It can be activated by burglars but also by earthquakes.
- Both events are quite rare.
- Sensory data: **alarm activation**
- Explanatory factors: **burglarized**, **earthquake**
- The alarm goes off (event **alarm activation** occurs)
- Given **alarm activation**, the events **earthquake** and **burglarized** are almost mutually exclusive ⇒ dependent

What makes a good representation?

1. **Smoothness:** if $x_1 \approx x_2$, then $g(x_1) \approx g(x_2)$. The most basic prior.
2. **"Less" supervised learning:** train semi-supervised / self-supervised etc.
3. **Invariances/Equivariance/Coherence:** Generally, small temporal/spatial changes should result in similar representations.
4. **Natural clustering**: When data is associated with categorical variables, the representations should reflect this

- **Idea:** humans have named categories and classes because of statistical structure
- Machine learning tasks involves predicting such categorical variables.

5. **Multiple explanatory factors:** recover many different explanatory factors, so that the representation is useful for many different tasks
6. **Disentangle underlying factors:** Each dimension of the representation should represent a separate and meaningful aspect of the data
7. **Hierarchical explanatory factors:** Some factors are more abstract than others and could be defined in terms of less abstract ones
8. **Sparsity:** for any observation *x,* only some factors are relevant ⇒ Most dimensions of $g(x)$ should be zero, or invariant to small variations of x .
9. **Simplicity of Factor Dependencies:** Factors should be related through simple, linear dependencies.

Width and Depth of Neural Networks:

The architecture of neural networks often specified by the width and the depth of the networks. The depth h of a network is defined as its number of layers (including output layer but excluding input layer); while the width dm of a network is defined to be the maximal number of nodes in a layer.

Neural network with "many layers" is called a deep neural network. On the other hand, the width is the name of a property of a layer in a neural network: **it is equal to the number of neurons in that particular layer**. So, it may be apt to use the phrase "the width of a layer in a neural network

Activation Functions:

- Activation function is one of the building blocks on Neural Network
- *It's just a thing function that you use to get the output of node. It is also known as **Transfer Function**.*

- When our brain is fed with a lot of information simultaneously, it tries hard to understand and classify the information into *"useful"* and *"not-so-useful"* information. We need a similar mechanism for classifying incoming information as *"useful"* or *"less-useful"* in case of Neural Networks.

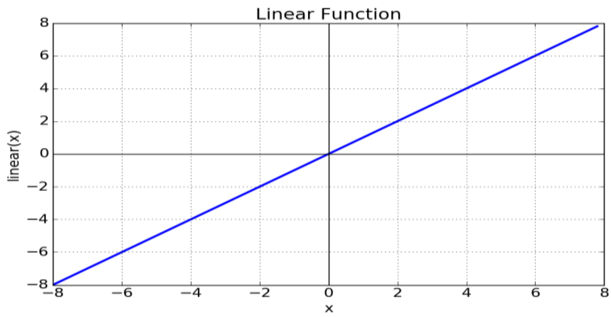
- This is important in the way a network learns because not all the information is equally useful. Some of it is just noise. This is where activation functions come into picture. The activation functions help the network use the important information and suppress the irrelevant data points.

- The Activation Functions can be basically divided into 2 types-
  - Linear Activation Function
  - Non-linear Activation Functions

Linear Activation Function:

Here function is a line or linear. Therefore, the output of the functions will not be confined between any range. It doesn't help with the complexity or various parameters of usual data that is fed to the neural networks. Range is -infinity to infinity

Equation : f(x) = x

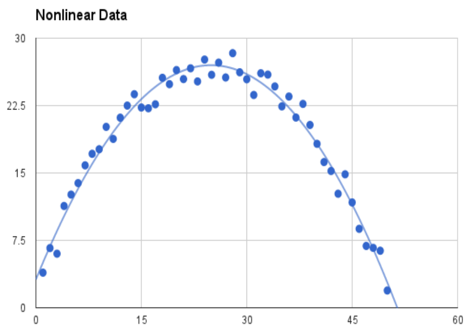


Non-linear Activation Functions:

The Nonlinear Activation Functions are the most used activation functions

- It makes it easy for the model to generalize or adapt with variety of data and to differentiate between the output.

- They allow backpropagation because now the derivative function would be related to the input, and it's possible to go back and understand which weights in the input neurons can provide a better prediction



- Let us go through these activation functions, learn how they work and figure out which activation functions fits well into what kind of problem statement.
  - Step Function
  - Sigmoid / Logistic function
  - Tanh or Hyperbolic Tangent Activation function
  - ReLU (Rectified Linear Unit) Activation Function
  - Leaky ReLU (LRELU)
  - Parametric Relu Function (PRELU)
  - Exponential Linear Unit (ELU)
  - Softmax Output Activation Function
  - Elastic Rectified Linear Unit (ERELU)

The input is fed to the input layer, the neurons perform a linear transformation on this input using the weights and biases.

$$x = (weight * input) + bias$$

Post that, an activation function is applied on the above result.

$$Y = Activation(\Sigma(weight * input) + bias)$$

Finally, the output from the activation function moves to the next hidden layer and the same process is repeated. This forward movement of information is known as the ***forward propagation***.

What if the output generated is far away from the actual value?
Using the output from the forward propagation, error is calculated. Based on this error value, the weights and biases of the neurons are updated. This process is known as ***back-propagation***.

Binary Step Function

The first thing that comes to our mind when we have an activation function would be a threshold based classifier i.e. whether or not the neuron should be activated based on the value from the linear transformation.

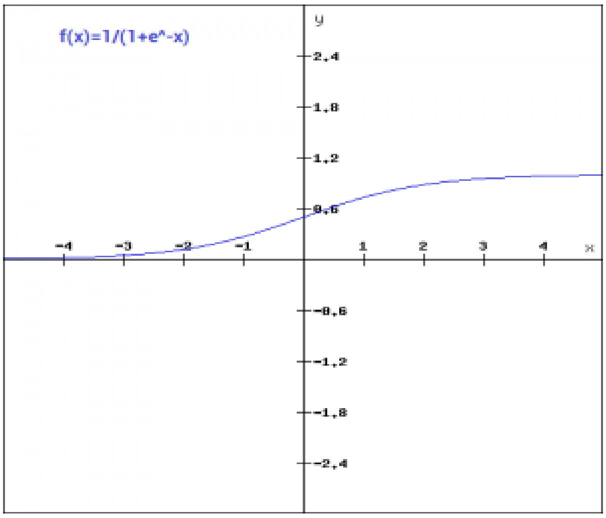
In other words, if the input to the activation function is greater than a threshold, then the neuron is activated, else it is deactivated, i.e. its output is not considered for the next hidden layer. Let us look at it mathematically-

f(x) = 1, x>=0
= 0, x<0

Sigmoid

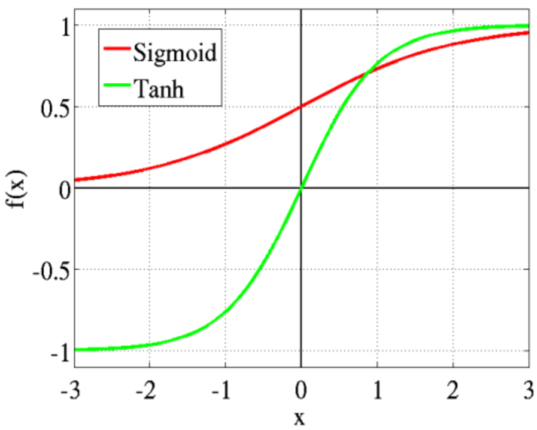
The next activation function that we are going to look at is the Sigmoid function. It is one of the most widely used non-linear activation function. Sigmoid transforms the values between the range 0 and 1. Here is the mathematical expression for sigmoid-

f(x) = 1/(1+e^-x)



<u>Tanh or hyperbolic tangent Activation Function</u>

- Tanh is also like logistic sigmoid but better. The range of the tanh function is from (-1 to 1). tanh is also sigmoidal (s - shaped).
- The advantage is that the negative inputs will be mapped strongly negative and the zero inputs will be mapped near zero in the tanh graph.
- The function is **differentiable**.
- The function is **monotonic** while its function **derivative is not monotonic**.
- The tanh function is mainly used classification between two classes.
- Both tanh and logistic sigmoid activation functions are used in feed-forward nets.



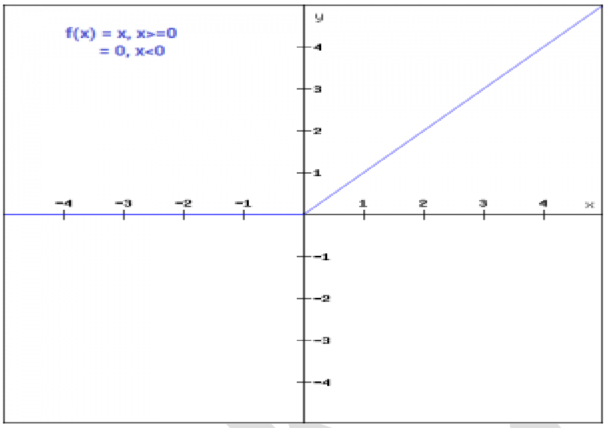
ReLU

The ReLU function is another non-linear activation function that has gained popularity in the deep learning domain. ReLU stands for Rectified Linear Unit. The main advantage of using the ReLU function over other activation functions is that it does not activate all the neurons at the same time.

This means that the neurons will only be deactivated if the output of the linear transformation is less than 0. Range is 0 to infinity. The function and its derivative both are monotonic.

The plot below will help you understand this better-

f(x)=max(0,x)



The limitations faced by this function are:

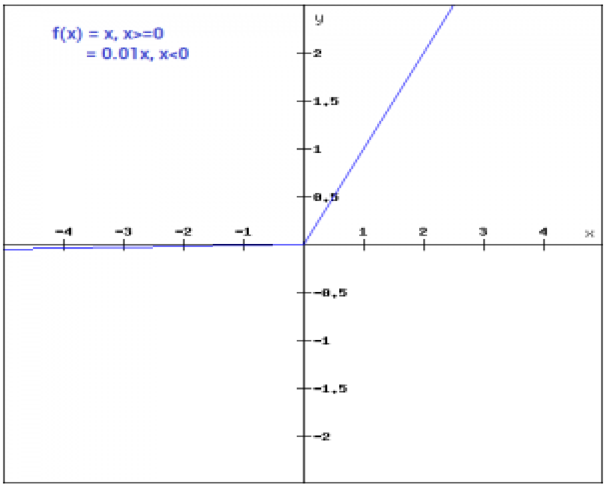
- The Dying ReLU problem.

Leaky ReLU (LReLU)

Leaky ReLU function is nothing but an improved version of the ReLU function. To solve the dying ReLu problem we are going for LReLU. As we saw that for the ReLU function, the gradient is 0 for x<0, which would deactivate the neurons in that region. The leaky helps to increase the range of the ReLU function as a small positive slope in the negative area.

Leaky ReLU is defined to address this problem. Instead of defining the Relu function as 0 for negative values of x, we define it as an extremely small linear component of x. Here is the mathematical expression-

$$f(x)= 0.01x, x<0$$
$$= x, x>=0$$

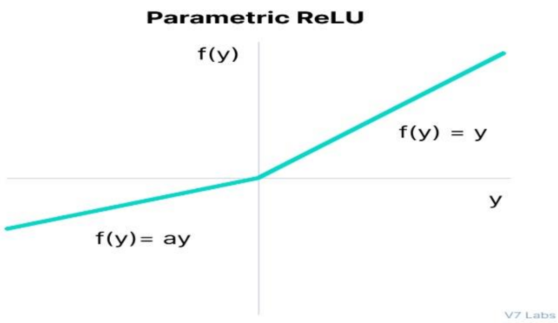


Parametric ReLU Function: (PReLU)

- Parametric ReLU is another variant of ReLU that aims to solve the problem of gradient's becoming zero for the left half of the axis.
- This function provides the slope of the negative part of the function as an argument *a*. By performing backpropagation, the most appropriate value of *a* is learnt.

$$\textit{Parametric ReLU}$$

$$f(x) = max\ (ax, x)$$

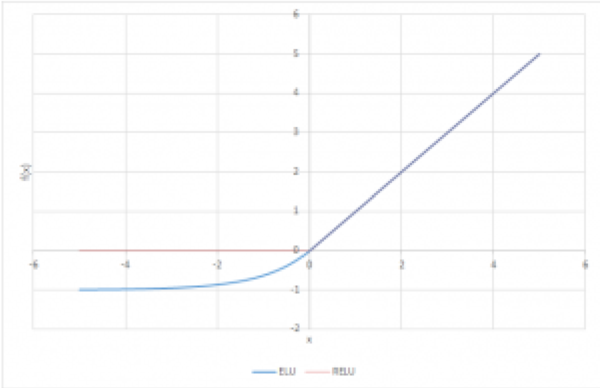


Exponential Linear Unit

Exponential Linear Unit or ELU for short is also a variant of Rectified Linear Unit (ReLU) that modifies the slope of the negative part of the function. Unlike the leaky relu and parametric ReLU functions, instead of a straight line, ELU uses a log curve for defining the negative values. Avoids dead ReLU problem by introducing log curve for negative values of input. It helps the network nudge weights and biases in the right direction. It increases the computational time because of the exponential operation included

It is defined as

f(x) = x, x>=0
= a(e^x-1), x<0



Softmax Output Activation Function

- The softmax function outputs a vector of values that sum to 1.0 that can be interpreted as probabilities of class membership.
- It is related to the argmax function that outputs a 0 for all options and 1 for the chosen option.
- Softmax is a "*softer*" version of argmax that allows a probability-like output of a winner-take-all function.
- The softmax function is calculated as follows:

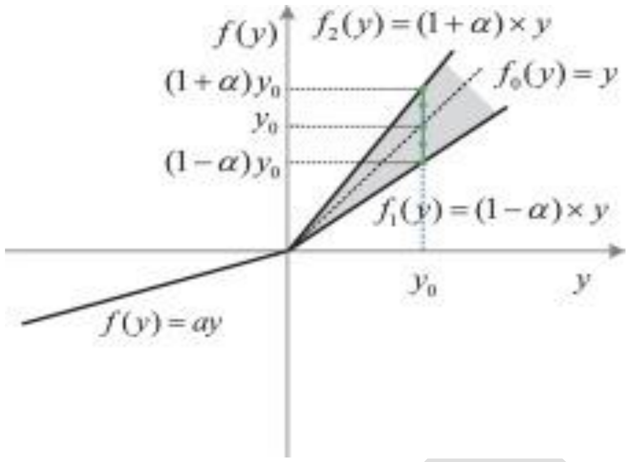
e^x / sum(e^x)

- Softmax function is described as a combination of multiple sigmoids.

Elastic Rectified Linear Unit (EReLU)

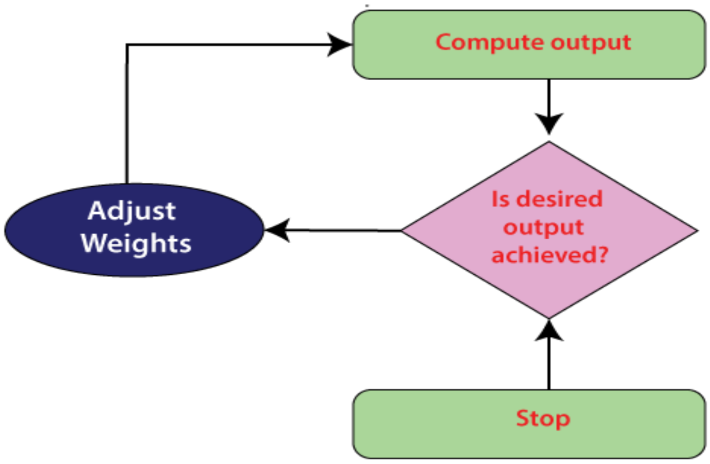
- EReLU is proposed for modifying the positive parts, unlike the situation which occurs with Randomized Leaky Rectified Linear Units (RReLU).
- EReLU changes the positive part of the input in the training stage.
- The input features are denoted with $x_{i,j}^{(c)}$ in the (*i, j*) element of the (*c*)th feature channel.
- The coefficient $k_{i,j}^{(c)}$ represents the slope for a positive input, sampled from a uniform distribution.
- The hyperparameter ϵ determines the range of random variation in the slope.
- In the testing stage, the coefficient $k_{i,j}^{(c)}$ is replaced by $E\left(k_{i,j}^{(c)}\right)$.
- As the average of the uniform distribution is 1, EReLU acts in the same way as ReLU in the testing stage.
- EReLU had a higher performance than ReLU without the need for additional parameters.
- Elastic Rectified Linear Unit (**EReLU**) is a novel activation function for deep neural networks that scales each positive value within a moderate range like a spring during training, improving model fitting with no extra parameters and little overfitting risk.
- It focuses on processing the positive part of input and becomes a standard ReLU during test time.

- EReLU mainly deals with the positive part of input.



Unsupervised training of neural networks:

Learning is a fundamental component required by every human being in the creation of intelligence. Humans derive their intelligence from the brain's capacity to learn from experience and utilizing that to adapt when confronted with existing and new circumstances. Reproduction of human intelligence in machines and computers is the objective of artificial intelligence techniques, one of which is an Artificial Neural Network. ANNs are models defined to mimic the learning capability of human brains. Like humans, validation, training, and testing are significant components in making such computational models. Artificial Neural Networks acquire information by getting some datasets (might be labeled or unlabeled) and computationally changing the network's free parameters adapted from the environment through simulation. Based on the learning rules and training process, learning in ANNs can be sorted into supervised, reinforcement, and unsupervised learning.



Supervised learning:

In **supervised learning**, the artificial neural network is under the supervision of an educator (say a system designer) who utilizes his or her knowledge of the system to prepare the network with labeled data sets. Thus, the artificial neural networks learn by receiving input and target the sets of a few observations from the labeled data sets. It is the process of comparing the input and output with the objective and computing the error between the output and objective. It utilizes the error signal through the idea of backward propagation to alter the weights that interconnect the network neuron with the point of limiting the error and optimizing performance. Fine-tuning of the network proceeds until the set of weights that limit the discrepancy between the output and the targeted output. The supervised learning process is used to solve classification and regression problems. The output of a supervised learning algorithm can either be a classifier or predictor. The application of this process is restricted when the supervisor's knowledge of the system is sufficient to supply the network's input and targeted output pairs for training.

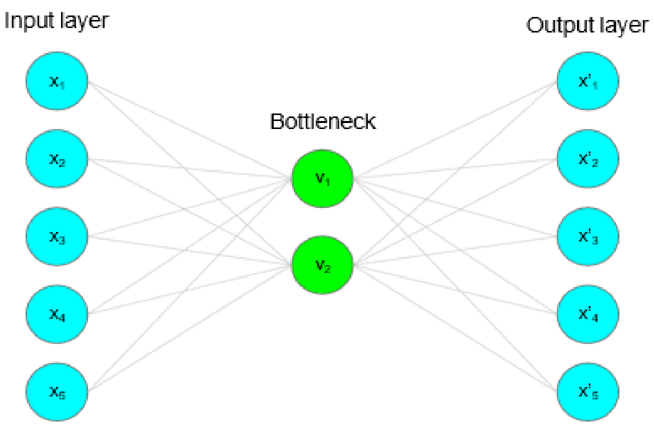
Unsupervised learning:

Unsupervised learning is used when it is absurd to augment the training data sets with class identities(labels). This difficulty happens in situations where there is no knowledge of the system, or the cost of obtaining such knowledge is too high. In unsupervised learning, as its name suggests, the ANN is not under the guidance of a "teacher." Instead, it is provided with unlabelled data sets (contains only the input data) and left to discover the patterns in the data and build a new model from it. In this situation, ANN figures out how to arrange the data by exploiting the separation between clusters within it.

Reinforcement learning:

Reinforcement learning is another type of unsupervised learning. It includes cooperation with the system, getting the condition of such a system, choosing an activity to change this state, sending the action to a system and accepting a numerical reward or a penalty in the form of feedback which can be positive or negative with the target of learning a policy. Activities that boost the reward are chosen by trial and error techniques. The figure illustrates the block diagram to describe the concept of reinforcement learning.
Include example any of those two(Dimensionality Reduction, Auto Encoders).

Auto Encoders



Definition: Create an architecture with a bottleneck, which ensures a lower-dimensional representation of the original data.

Autoencoder is a type of neural network where the output layer has the same dimensionality as the input layer. In simpler words, the number of output units in the output layer is equal to the number of input units in the input layer. An autoencoder replicates the data from the input to the output in an unsupervised manner and is therefore sometimes referred to as a replicator neural network.

The autoencoders reconstruct each dimension of the input by passing it through the network. It may seem trivial to use a neural network for the purpose of replicating the input, but during the replication process, the size of the input is reduced into its smaller representation. The middle layers of the neural network have a fewer number of units as compared to that of input or output layers. Therefore, the middle layers hold the reduced representation of the input. The output is reconstructed from this reduced representation of the input.

Architecture of autoencoders

An autoencoder consists of three components:

- **Encoder**: An encoder is a feedforward, fully connected neural network that compresses the input into a latent space representation and encodes the input image as a compressed representation in a reduced dimension. The compressed image is the distorted version of the original image.
- **Bottleneck**: This part of the network contains the lower dimensionality when compared to the input layer and then it is fed into the decoder.
- **Decoder**: Decoder is also a feedforward network like the encoder and has a similar structure to the encoder. This network is responsible for reconstructing the input back to the original dimensions from the code.

Reconstructed Error = Reconstructed – Original

How can we train an autoencoder?

- Backpropagation
- Minimise reconstruction error

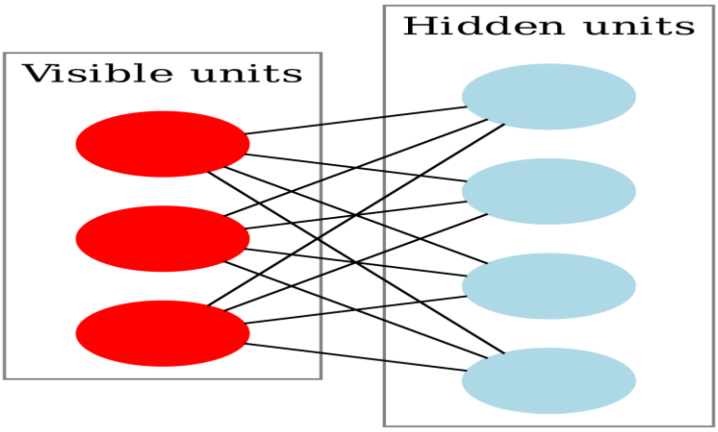
$$E(\underbrace{\boxed{x}}_{\text{Original data}}, \underbrace{\boxed{\hat{x}}}_{\text{Reconstructed data}})$$

Restricted Boltzmann Machine:

A restricted Boltzmann machine (RBM) is a generative stochastic artificial neural network that can learn a probability distribution over its set of inputs. Restricted Boltzmann Machine (RBM) is a type of artificial neural network that is used for unsupervised learning. It is a type of generative model that is capable of learning a probability distribution over a set of input data. It is an algorithm which is useful for dimensionality reduction, classification, regression, collaborative filtering, feature learning, and topic modeling.

RBM was introduced in the mid-2000s by Hinton and Salakhutdinov as a way to address the problem of unsupervised learning. It is a type of neural network that consists of two layers of neurons – a visible layer and a hidden layer. The visible layer represents the input data, while the hidden layer represents a set of features that are learned by the network.

The RBM is called "restricted" because the connections between the neurons in the same layer are not allowed. A restricted term refers to that we are not allowed to connect the same type layer to each other. In other words, the two neurons of the input layer or hidden layer can't connect to each other. Although the hidden layer and visible layer can be connected to each other. This allows the RBM to learn a compressed representation of the input data by reducing the dimensionality of the input



It is a network of neurons in which all the neurons are connected to each other. In this machine, there are two layers named visible layer or input layer and hidden layer. The visible layer is denoted as **v** and the hidden layer is denoted as the **h.** In Boltzmann machine, there is no output layer. Boltzmann machines are random and generative neural networks capable of learning internal representations and are able to represent and (given enough time) solve tough combinatorial problems.

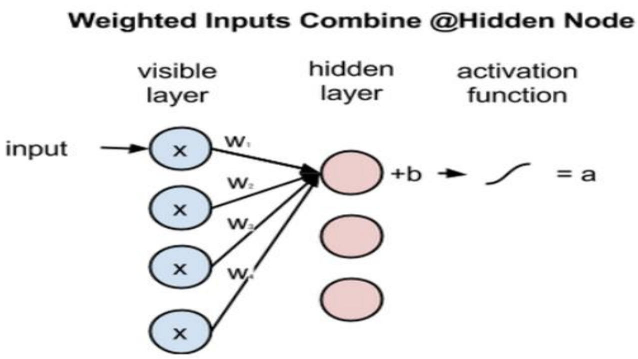
The Boltzmann distribution (also known as **Gibbs Distribution**) which is an integral part of Statistical Mechanics and also explain the impact of parameters like Entropy and Temperature on the Quantum States in Thermodynamics. Due to this, it is also known as **Energy-Based Models (EBM)**.

Working:

Each visible node takes a low-level feature from an item in the dataset to be learned.

At node 1 of the hidden layer, **x is multiplied** by a ***weight*** and added to a ***bias***. The result of those two operations is fed into an ***activation function***, which produces the node's output, or the strength of the signal passing through it, given input x.

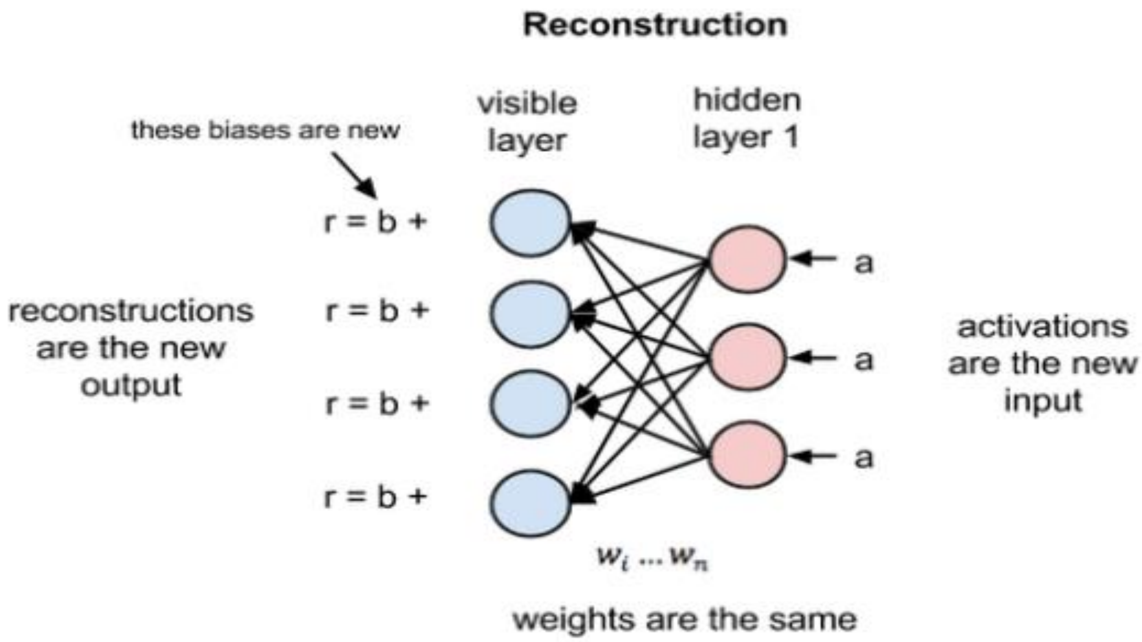
```
        activation f((weight w * input x) + bias b ) = output
a
```



Reconstruction:

To reconstruct data by themselves in an unsupervised fashion (unsupervised means without ground-truth labels in a test set), making several forward and backward passes between the visible layer and hidden layer no. 1 without involving a deeper network. In the reconstruction phase, the activations of hidden layer no. 1 become the input in a backward pass.

Because the weights of the RBM are randomly initialized, the difference between the reconstructions and the original input is often large. You can think of reconstruction error as the difference between the values of r and the input values, and that error is then backpropagated against the RBM's weights, again and again, in an iterative learning process until an error minimum is reached.



Deep Learning Applications:

1. **Healthcare**: The healthcare sector has long been one of the prominent adopters of modern technology to overhaul itself. As such, it is not surprising to see Deep Learning finding uses in interpreting medical data for

- the diagnosis, prognosis & treatment of diseases
- drug prescription
- analysing MRIs, CT scans, ECG, X-Rays, etc., to detect and notify about medical anomalies
- personalising treatment
- monitoring the health of patients

2. **Personalized Marketing**: Personalized marketing is a concept that has seen much action in the recent few years. Marketers are now aiming their advertising campaigns at the pain points of individual consumers, offering them exactly what they need. And Deep Learning is playing a significant role in this.
Today, consumers are generating a lot of data thanks to their engagement with social media platforms, IoT devices, web browsers, wearables and the ilk. However, most of the data generated from these sources are disparate (text, audio, video, location data, etc.).
To cope with this, businesses use customisable Deep Learning models to interpret data from different sources and distil them to extract valuable customer insights. They then use this information to predict consumer behaviour and target their marketing efforts more efficiently.

3. **Financial Fraud Detection**: virtually no sector is exempt from the evil called "fraudulent transactions" or "financial fraud". However, it is the financial corporations (banks, insurance firms, etc.) that have to bear the brunt of this menace the most. Not a day goes by when

criminals attack financial institutions. There are a plethora of ways to usurp financial resources from them.
Thus, for these organizations, detecting and predicting financial fraud is critical, to say the least. And this is where Deep Learning comes into the picture.
Financial organizations are now using the concept of anomaly detection to flag inappropriate transactions. They employ deep learning algorithms, such as logistic regression (credit card fraud detection is a prime use case), decision trees, random forest, etc., to analyze the patterns common to valid transactions. Then, these models are put into action to flag financial transactions that seem potentially fraudulent.
Some examples of fraud detection being deterred by Deep Learning include:

- identity theft
- insurance fraud
- investment fraud
- fund misappropriation

4. **Natural Language Processing**: NLP or Natural Language Processing is another prominent area where Deep Learning is showing promising results.
Natural Language Processing, as the name suggests, is all about enabling machines to analyze and understand human language. The premise sounds simple, right? Well, the thing is, human language is punishingly complex for machines to interpret. It is not just the alphabet and words but also the context, the accents, the handwriting and whatnot that discourage machines from processing or generating human language accurately.
Deep Learning-based NLP is doing away with many of the issues related to understanding human language by training machines (Autoencoders and Distributed Representation) to produce appropriate responses to linguistic inputs.
One such example is the personal assistants we use on our smartphones. These applications come embedded with Deep Learning imbued NLP models to understand human speech and return appropriate output. It is, thus, no wonder why Siri and Alexa sound so much like how people talk in real life.
Another use case of Deep Learning-based NLP is how websites written in one human language automatically get translated to the user-specified language.

5. **Autonomous Vehicles:** The concept of building automated or self-governing vehicles goes back 45 years when the Tsukuba Mechanical Engineering Laboratory unveiled the world's first semi-automatic car. The car, a technological marvel then, carried a pair of cameras and an analogue computer to steer itself on a specially designed street.
However, it wasn't until 1989 when ALVINN (Autonomous Land Vehicle in a Neural Network), a modified military ambulance, used neural networks to navigate by itself on roads.
Since then, deep learning and autonomous vehicles have enjoyed a strong bond, with the former enhancing the latter's performance exponentially.
Autonomous vehicles use cameras, sensors – LiDARs, RADARs, motion sensors – and external information such as geo-mapping to perceive their environment and collect relevant data. They use this equipment both individually and in tandem for documenting the data.
This data is then fed to deep learning algorithms that direct the vehicle to perform appropriate actions such as

- accelerating, steering and braking
- identifying or planning routes
- traversing the traffic
- detecting pedestrians and other vehicles at a distance as well as in proximity
- recognising traffic signs

6. **Fake News Detection**: The concept of spreading fake news to tip the scales in one's favour is not old. However, due to the explosive popularity of the internet, and social media platforms, in particular, fake news has become ubiquitous.
Fake news, apart from misinforming the citizens, can be used to alter political campaigns, vilify certain situations and individuals, and commit other similar morally illegible acts. As such, curbing any and all fake news becomes a priority.
Deep Learning proposes a way to deal with the menace of fake news by using complex language detection techniques to classify fraudulent news sources. This method essentially works by gathering information from trusty sources and juxtaposing them against a piece of news to verify its validity.
This paper explains how a combination of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) can validate digital news with high accuracy.

7. **Facial Recognition:** Facial Recognition is the technological method of identifying individuals from images and videos by documenting their faces. It uses advanced biometric technology to record a person's face and match it against a database to extract their identity.
Facial Recognition is an old technology, first conceptualized in the 1960s. However, it is the integration of neural networks in facial recognition that exponentially increased its detection accuracy.
Deep Learning enforced Facial Recognition works by recording face embeddings and using a trained model to map them against a huge database of millions of images.
For instance, DeepFace is a facial recognition method that uses Deep Learning (hence the name) to identify persons with a recorded 97% accuracy rate. It uses a nine-layer neural network for its purpose and has been trained using four million images of about 4000 individuals.

8. **Recommendation Systems:** Have you ever stopped to think about how Spotify knows which genres you listen to or, how Netflix recommends shows that match your preferences exactly? The short answer is Deep Learning. And the long answer, well, it is still deep learning but with some added explanation.
As discussed earlier, Deep Learning models process user data acquired from different sources and compile them to extract consumer info. This information then goes into deep learning-based recommender systems to generate appropriate suggestions for the users.
Deep Learning empowered suggestions, although widely used by audio/video streaming services, are not just limited to them. Social media networks use similar systems to recommend relevant posts, videos, accounts and more to users in their feeds.

9. **Smart Agriculture:** Artificial Intelligence and its subsets are fortifying a lot of industries and sectors, and agriculture is no different.
Of late, smart farming has become an active agricultural movement to improve upon the various aspects of traditional agriculture. Farmers are now using IoT devices, satellite-based soil-composition detection, GPS, remote sensing, etc., to monitor and enhance their farming methods.
Deep Learning algorithms capture and analyse agriculture data from the above sources to improve crop health and soil health, predict the weather, detect diseases, etc.
Deep learning also finds uses in the field of crop genomics. Experts use neural networks to determine the genetic makeup of different crop plants and use it for purposes like

- increasing resilience to natural phenomena and diseases
- increase crop yield per unit area

- breeding high-quality hybrids

10. **Space Travel:** For most of us, space travel is something we associate with the most advanced technology available to humankind. We think of humanoid robots, hyper-intelligent AIs, hi-tech equipment, etc., working relentlessly in space to assist the astronauts in their painstaking endeavours.

However, while most of this stuff is over-the-top, it does signal one aspect of space flight – that it is technologically demanding.

Scientists and engineers need to implement the latest and most efficient technologies – both hardware and software – to ensure the safety, integrity and success of space missions.

Thus, it goes without saying that AI, Machine Learning and Deep Learning are crucial components of everything astronomy.

For instance, ESA states that Deep Learning can be (and is, to some extent) used in

- automating the landing of rockets
- building space flight systems that can make intelligent decisions without human intervention.