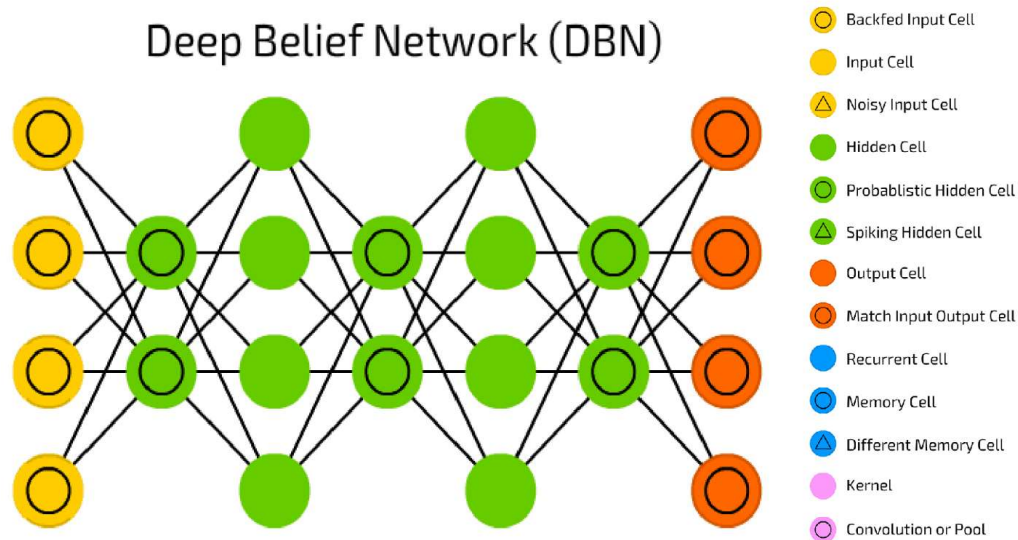


Deep Belief Network



Today we will understand Deep Belief Network(**DBN**) A type of **Unsupervised Pretrained Networks (UPNs)**

Lets define DBN first:

DBN is a class of deep neural network which comprises of multiple layer of graphical model having both directed and undirected edges. It is composed of multiple layers of hidden units, where each layers are connected with each others but units are not.

To understand Deep belief networks we need to understand two important caveat of DBN

1. Belief Net
2. RBM : Restricted Boltzmann Machine

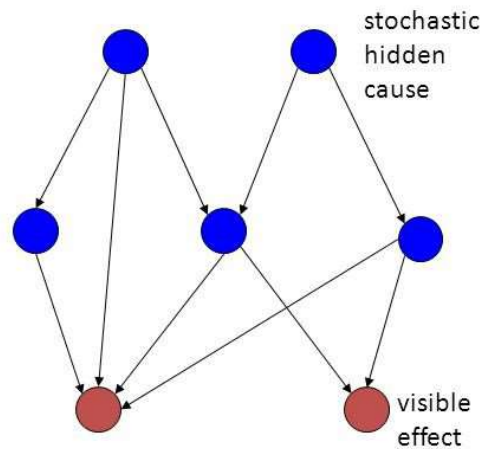
1. Belief Network :

It consists of stochastic binary unit layers where each connected layers have some weight. The stochastic binary units in belief networks have a state of 0 or 1 and the probability of becoming 1 is determined by a bias and weighted input from other units.

Let's understand Deep Belief Net by this Image (G. Hinton)

Belief Nets

- A belief net is a directed acyclic graph composed of stochastic variables.
- Can observe some of the variables and we would like to solve two problems:
- **The inference problem:** Infer the states of the unobserved variables.
- **The learning problem:** Adjust the interactions between variables to make the network more likely to generate the observed data.



Use nets composed of layers of stochastic binary variables with weighted connections. Later, we will generalize to other types of variable.

Dr. Geoffrey Hinton Says :

The two most significant properties of deep belief nets are:

- There is an efficient, layer-by-layer procedure for learning the top-down, generative weights that determine how the variables in one layer depend on the variables in the layer above.
- After learning, the values of the latent variables in every layer can be inferred by a single, bottom-up pass that starts with an observed data vector in the bottom layer and uses the generative weights in the reverse direction.

2. Restricted Boltzmann Machine :

Boltzmann Machine is a stochastic recurrent neural network with stochastic binary units and undirected edges between units. Due to Boltzmann Machine limitation to scale RBM was introduced which consists of hidden layers unit having restricted connection between each hidden units. This structure help RBM to learn efficiently.

See below to understand more :

Restricted Boltzmann Machines

- Restrict the connectivity to make learning easier.
 - Only one layer of hidden units.
 - Deal with more layers later
 - No connections between hidden units.
- In an RBM, the hidden units are conditionally independent given the visible states.
 - So can quickly get an unbiased sample from the posterior distribution when given a data-vector.
 - This is a big advantage over directed belief nets

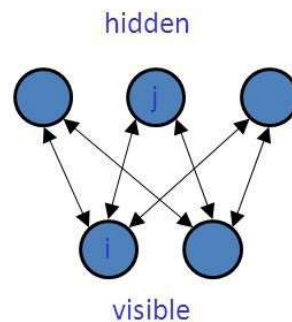
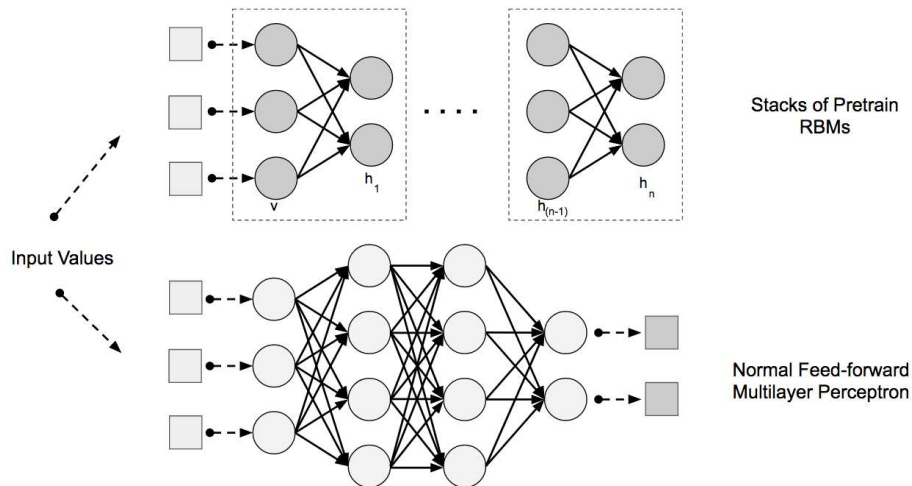


Image Source : [Geoffrey Hinton,cs.toronto.edu](http://Geoffrey.Hinton.cs.toronto.edu)

DBN Architecture :

DBNs are composed of layers of Restricted Boltzmann Machines (RBMs) for the pre-train phase and then a feed-forward network for the fine-tune phase. As shown in the network architecture of a DBN below :

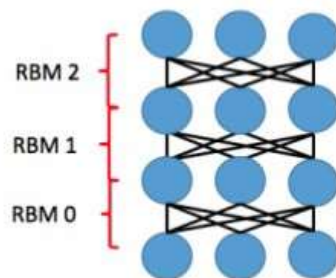


How DBN Works (Basic Learning) :

Now that we have understood the basics of **Belief Net** and **RBM**, let's try to understand how DBN actually learns. As DBN is multi-layer belief networks, where each layer is Restricted Boltzmann Machine stacked against each other to form the Deep belief Network. The first step of training DBN is to learn a layer of features from the visible units, using Contrastive Divergence (CD) algorithm. Then, the next step is to treat the activations of previously trained features as visible units and learn features of features in a second hidden layer. Finally, the whole DBN is trained when the learning for the final hidden layer is achieved.

Greedy DBN Learning :

Greedy Layer-wise Deep Training



- **Idea:** DBNs can be formed by "stacking" RBMs
- Each layer is trained as a Restricted Boltzmann Machine.
- Train layers sequentially starting from bottom (observed data) layer. (Greedy layer-wise)
- Each layer learns a higher-level representation of the layer below. The training criterion does not depend on the labels. (Unsupervised)