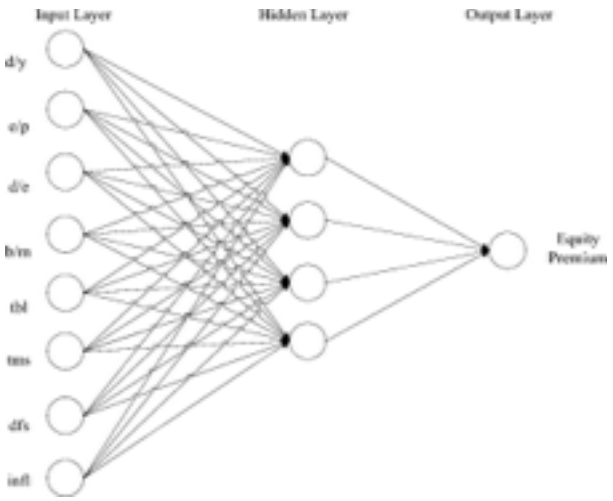
Artificial Neural Network & Back-propagation Algorithm

artificial neural network

An artificial neural network is modeled after that of a biological nervous system. It is an interconnected feed-forward network. Which means, a signal (or input) is received on one end of the model and then passed along through an indeterminate number of units (or layers) until it reaches the end (the output).

In an artificial neural network, there is an input layer; where the signal begins, one or more hidden layer(s), and finally an output layer; where the signal ends. Every layer starting from the input layer is fully connected to the next layer. Also, every connection from one layer to the next layer has a value associated with it, called a **weight**. These weights adapt during the training of the network, therefore allowing the network to learn as it receives more and more inputs. Every node within the neural network has an activation function assigned to it as well. It is this activation function that determines the output from the node to be carried to the next layer.



Source: wikipedia.org/wiki/Artificial_neural_network

Back-propagation Algorithm

Back-propagation is a training algorithm that can be employed by a neural network. Back propagation networks learn by example; given some training data, the network modifies itself so that when training is completed, it will be able to give correct results for a test data point. The algorithm works in two steps: the Feed-Forward step and the Back-Propagation step.

The Feed-Forward step will take an input and traverse it along the network and generate an output at the output layer. The Back Propagation step will take that output and calculate an error based on what the target value was on the original input. Based on this error, it will then modify the weights that are feeding into the node that produced the error. This is completed for all output nodes, and then the algorithm back propagates to the previous hidden layer of inputs and repeats the process of calculating an error and updates the corresponding weights. This process will continue until it reaches the input layer and stop.

Implementation

We used the artificial neural network as our model for predicting the domestic and international box office(2 target value).

We also ignore some aspects of the movie since it's trivial or not in the "Opus" database, these are release date, quality of script, competition from similar movies and etc.

Inputs:

We have 25 inputs which including:
9 creative type; and
11 genre of the movie; and
2 actor profitability; and
1 director profitability; and
the budget before the production; and
whether it's franchise or not.
(see Appendix for details).

Hidden layers:

3 hidden layers are employed for this project.

Outputs(target value):

2 outputs: Domestic Box Office & International Box Office.

Appendix:

1. inputs list for artificial neural network.

0	[Contemporary Fiction]	Binary value.
1	[Dramatization]	Binary value.
2	[Factual]	Binary value.
3	[Fantasy]	Binary value.
4	[Historical Fiction]	Binary value.
5	[Kids Fiction]	Binary value.
6	[Multiple Creative Types]	Binary value.
7	[Science Fiction]	Binary value.
8	[Super Hero]	Binary value.
9	[Adventure]	Binary value.
10	[Black Comedy]	Binary value.
11	[Comedy]	Binary value.
12	[Concert/Performance]	Binary value.
13	[Documentary]	Binary value.
14	[Drama]	Binary value.
15	[Horror]	Binary value.
16	[Musical]	Binary value.
17	[Romantic Comedy]	Binary value.
18	[Thriller/Suspense]	Binary value.
19	[Western]	Binary value.
20	[actor_1_profitability]	Real number
21	[actor_2_profitability]	Real number
22	[director_profitability]	Real number
23	[budget]	Real number
24	[franchise]	Binary value.