## Recurrent Neural Network

### Description

* Unlike standard neural networks, recurrent neural networks use previous inputs in their current predictions. Standard neural networks assume that inputs are independent. Only works with previous elements, not next elements.
* Recurrent neural networks share the same weight parameters in each layer of the network.
* Training is done via BPTT (Backpropagation through time)
* BPTT leads to exploding and vanishing gradients. If exploding, than the weights become too large and the model is unstable. If vanishing, the weights go to 0.
* You can have different frameworks for RNNS
  + One to one, one to many, many to one, many to many
* Activations
  + Sigmoid, tanh, relu
* Variations
  + Bidirectional Recurrent Neural Networks (BRNN)
    - Use future inputs in current prediction
  + Long short-term memory (LSTM)
    - Problem with RNNs is that they have a short term memory. It is hard for them to use inputs that happened far in the past
    - LSTM has cells that have three gates
      * Input
      * Output
      * Forget
  + Gated recurrent units (GRUs)
    - Uses ‘hidden states’ instead of ‘cell state’, and uses reset and update gates instead of the three gates in LSTM
    - Seeks to solve the same problem as LSTM

## Gradient Boosting - <https://www.youtube.com/watch?v=3CC4N4z3GJc>

### Description

* Gradient Boost for regression is used to predict a continuous variable
* Build successive decision trees weighted by some factor to predict the residuals of the previous prediction
* Add the results from the successive trees to get the prediction
* Trees are weighted equally

## Explainable Boosting Machine - <https://interpret.ml/docs/ebm.html>

### Description