Deep Learning

Neural Networks

Chandra Lingam

Cloud Wave LLC

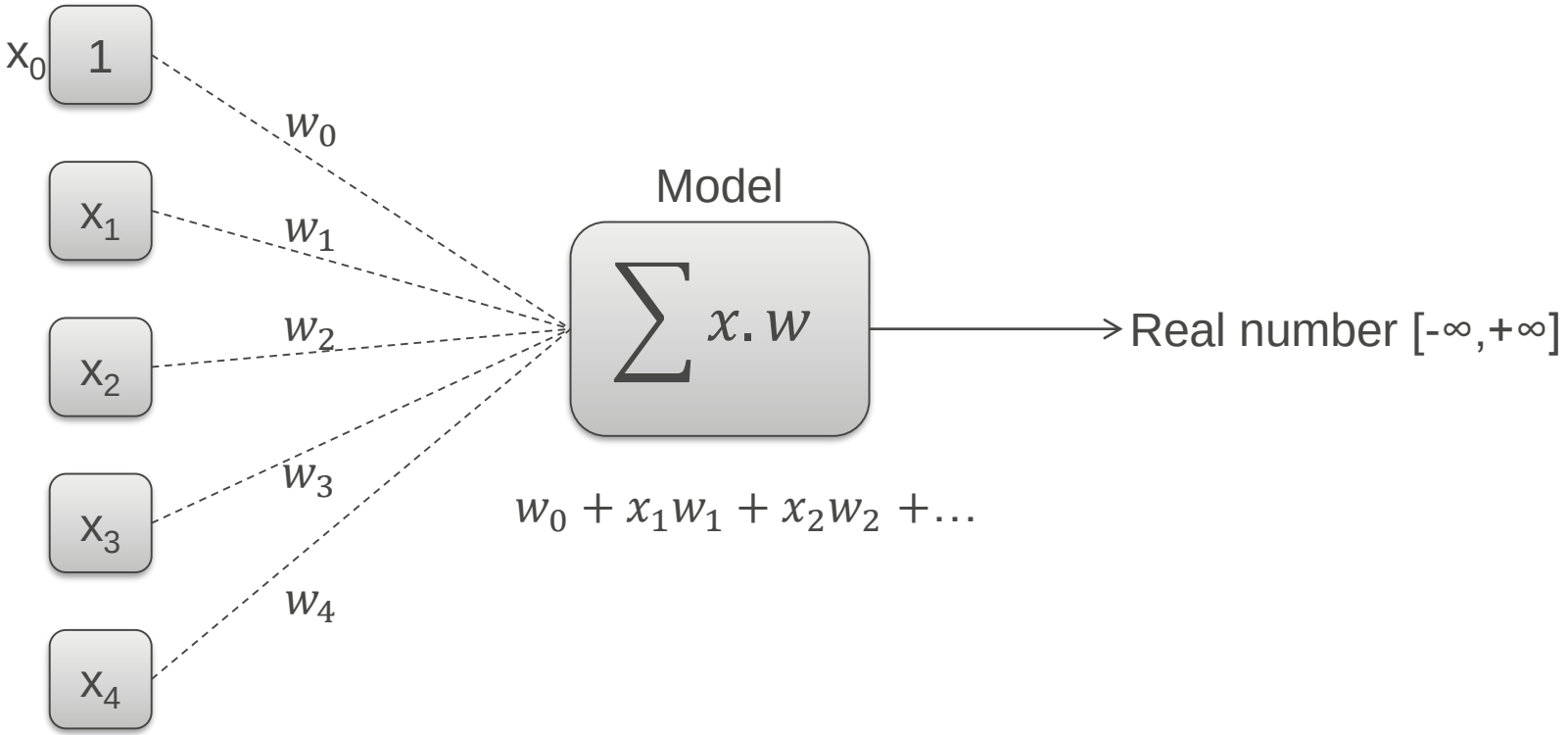
Linear Model

Model Training

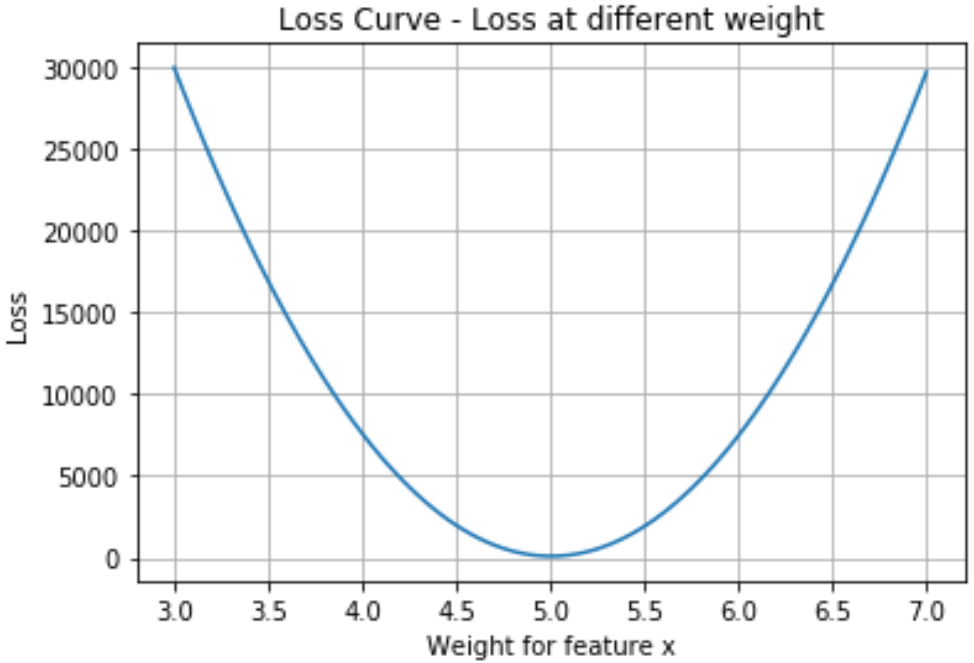
Gradient Descent Optimizer

Loss Function

Linear Regression



Loss Curve



Gradient Descent Variants

- RMSProp
- AdaGrad
- Adam
- …

Additional Reading:

https://ruder.io/optimizing-gradient-descent/

Gradient Descent Modes

Mode	Description
Batch	• Compute loss for all training examples • Adjust weight Example: 150 samples in training set. For each iteration, weight is adjusted once
Stochastic	• Compute loss for next example • Adjust weight Example: 150 samples in training set. For each iteration, weight is adjusted 150 times
Mini-batch	• Compute loss for specified number of examples • Adjust weight Example: 150 samples in training set. Mini-batch size is 15. For each iteration, weight is adjusted 10 times

Loss Plot with Multiple Features

Image Courtesy: Anantha Metals,
https://www.ananthaonline.com/

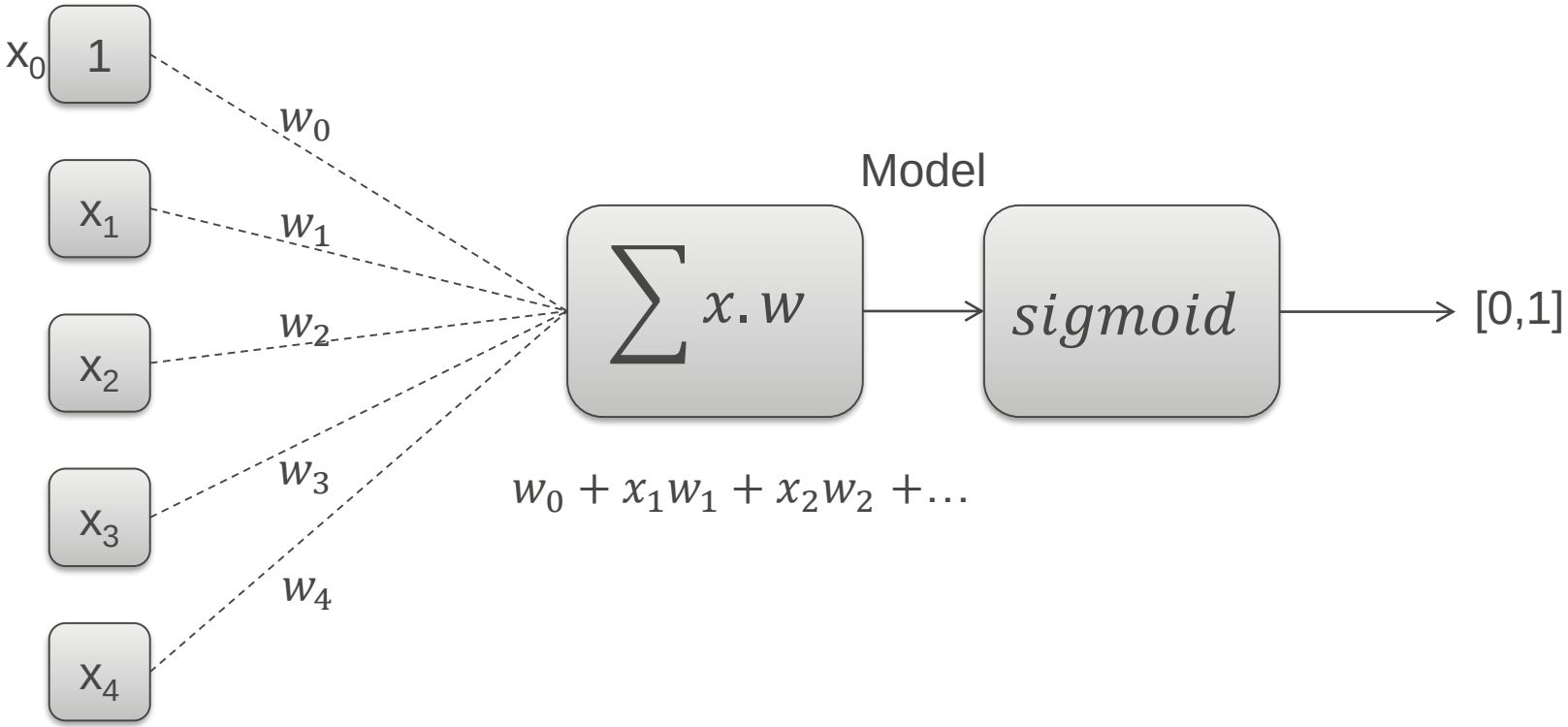
Linear Model - Summary

Loss Function

Learning Rate

Gradient Descent Optimizer

Logistic Regression (Binary Classification)

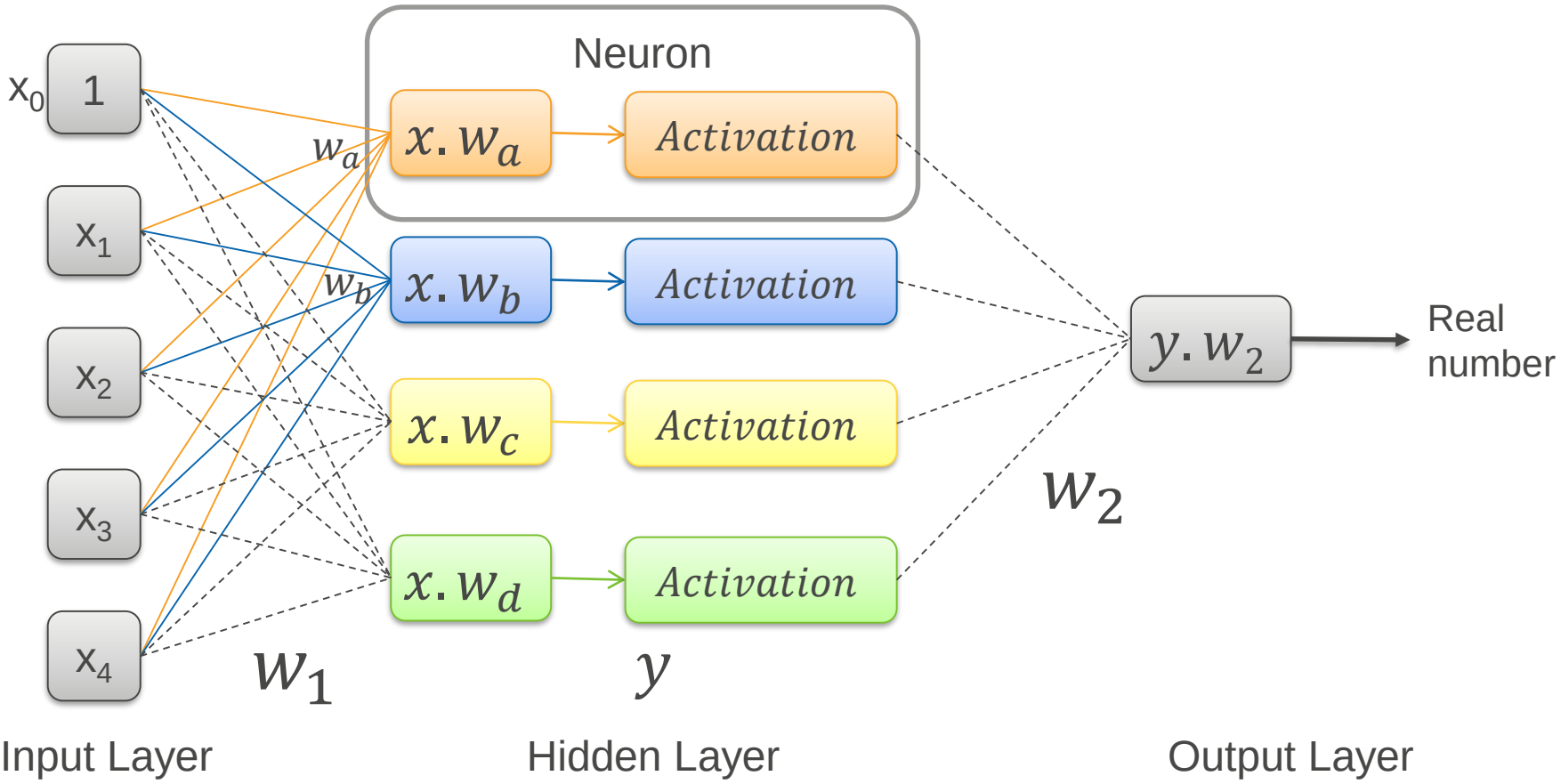


Summary - Linear Models

- Features and Weights
- Loss Function, Gradient Descent
- Simple to understand and use
- Difficult to use with Non-linear datasets
- Requires complex feature engineering
- Data needs to be in similar range and scale – preferably standardized or normalized

****Foundation for deep learning******

Neural Networks – Regression



Neural Networks

- Automatic feature-engineering – mixes features to create new ones
- Handles non-linear datasets
- Easy to overfit (apply regularization, reduce model complexity and so forth)

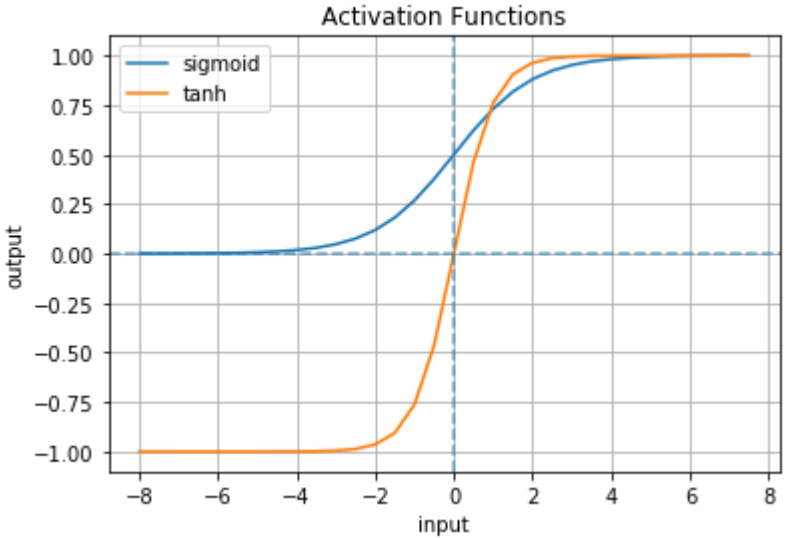
Activation Functions

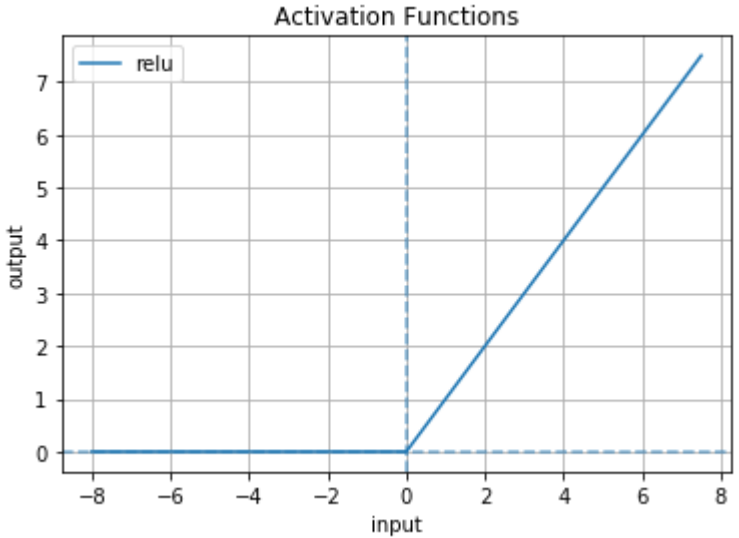
Introduce non-linearity in the model

Improves ability of model to fit complex non-linear datasets

Three popular activation functions: sigmoid, tanh, relu

Activation Functions





Sigmoid: for any input, output is bounded between 0,1

Tanh: for any input, output is bounded between -1,1

ReLU : for any input x, output is max(0,x)

Deep Learning

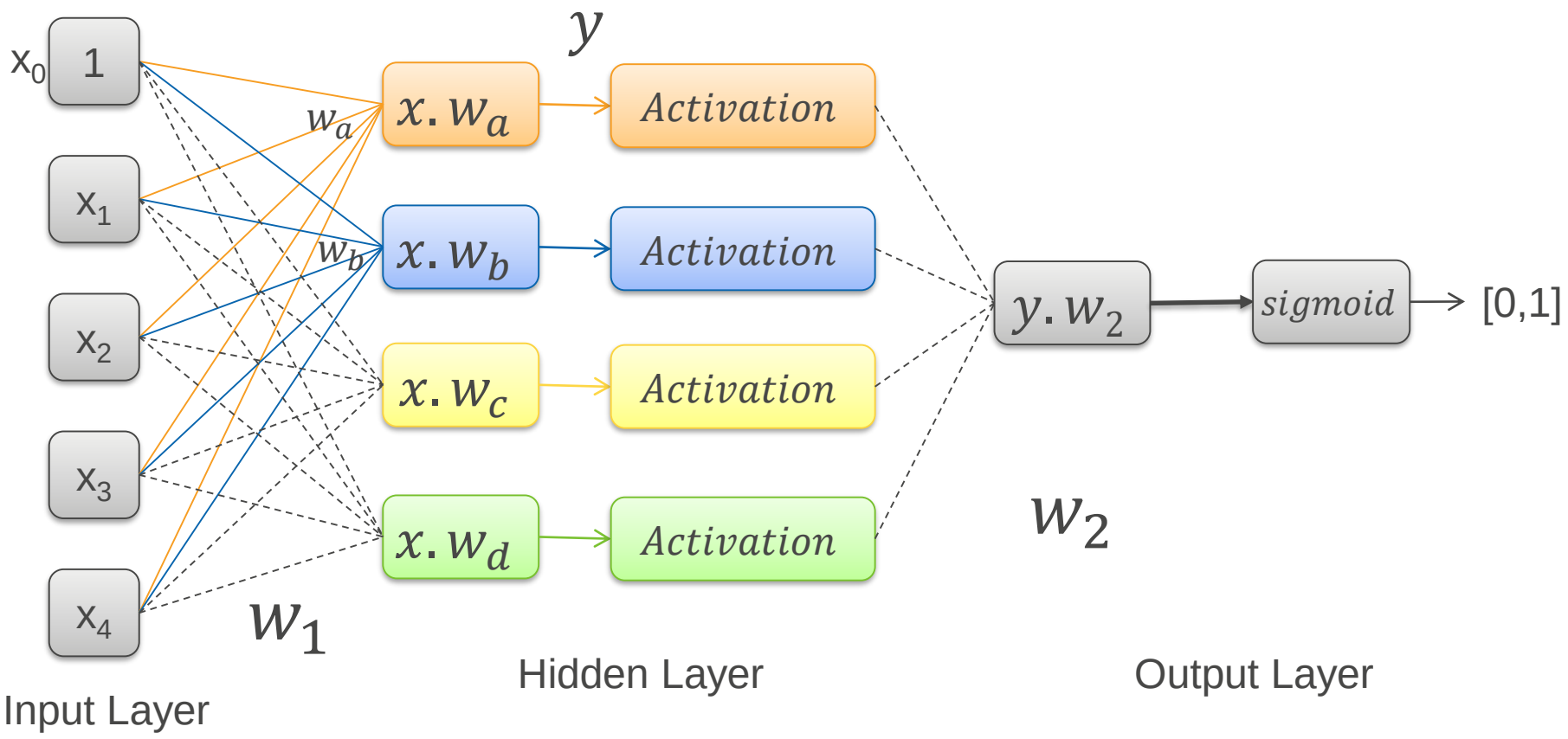
“Traditional ML algorithms appear to saturate on how much they can learn. So, having massive amounts of data does not translate to ‘more learning’”

“Small NN can learn better. Medium NN can learn even more. Large NNs can keep learning with more data (several hidden layers)”

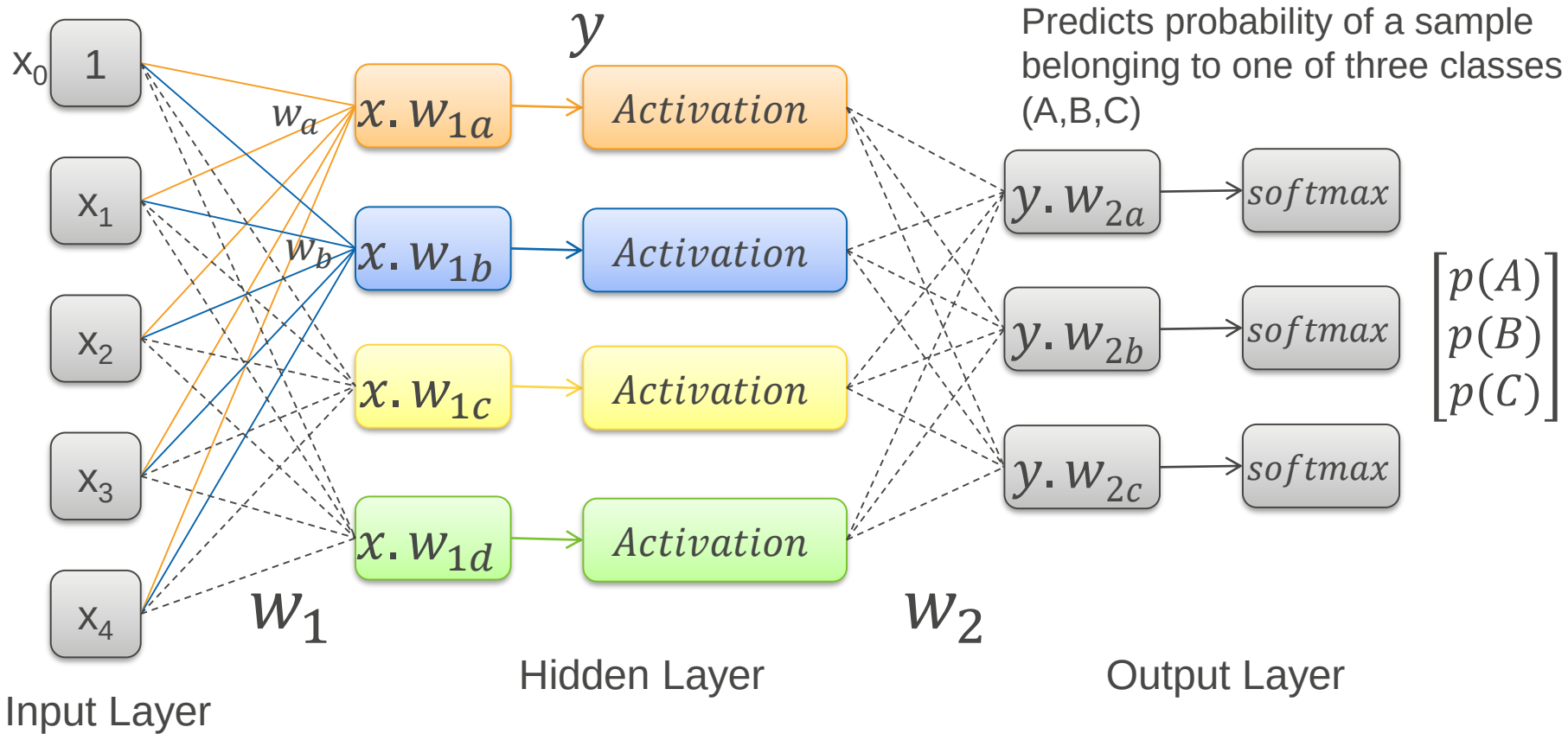
Reference: Dr. Andrew Ng

"Nuts and bolts of building AI applications using Deep Learning“.
https://www.youtube.com/watch?v=wjqaz6m42wU

Neural Networks – Binary Classification



Neural Networks – Multiclass Classification



Popular Neural Network Architectures

General Purpose

- Fully connected network
- Example: treats each pixel as a separate feature

Convolutional Neural Network (CNN or convnet)

- Useful for image analysis
- Example: considers pixel and its surrounding pixels

Popular Neural Network Architectures

Recurrent Neural Network (RNN)

- Looks at history
- Useful for timeseries forecasting, natural language processing
- Example: time series forecasting – model looks at current value and historical values

Lab – Regression with SKLearn Neural Network

- Kaggle Bike Rental Data
- Train using SKLearn's MLPRegressor (Multi-layer Perceptron)
- Data Preparation:
    - One Hot Encode all categorical features
    - Standardize or Normalize all numeric features

Deep Learning Libraries

Keras API

“Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano.

It was developed with a focus on enabling fast experimentation.”

Reference: Keras.io, https://keras.io/

Lab – Regression with TensorFlow Neural Network

- Kaggle Bike Rental Data
- Train using Keras API with TensorFlow as Backend
- SageMaker Notebook comes pre-installed with TensorFlow, Apache MxNet, Pytorch and other DeepLearning Libraries
- To change the backend, use the appropriate Jupyter Notebook kernel in SageMaker

Lab – Binary Classification with Neural Network

- Mobile Operator Customer Churn Prediction
- Predict probability of a customer joining a competitor
- Build a neural network for binary classification with Keras
- Data transformation to handle highly correlated features, categorical and numeric data

Adapted from Predicting Customer Churn with Amazon Machine Learning by Denis V. Batalov, and SageMaker Examples

Lab – Multiclass Classification with Neural Network

- Iris Plant Classification
- Transform target using one hot encoding

Additional Resources - Books

Introduction to Machine Learning with Python

by Andreas C. Müller and Sarah Guido

Solid introduction to scikit-learn and machine learning

Deep Learning with Python

by François Chollet

Great introduction to Deep Learning with Keras

Additional Resources - Videos

MIT 6.S191: Introduction to Deep Learning

by Alexander Amini

Excellent introduction to deep learning, loss function, optimizers

MIT 6.S191: Convolutional Neural Networks

MIT 6.S191: Recurrent Neural Networks

by Ava Soleimany

Excellent overview of CNN and RNNs

Additional Resources - Videos

NIPS 2016 tutorial: "Nuts and bolts of building AI applications using Deep Learning"

by Andrew Ng

Practical tips, tricks and industry experience

Transfer Learning

by Andrew Ng

Additional Resources - Articles

The importance of hyperparameter tuning for scaling deep learning training to multiple GPUs

by Sina Afrooze

Guide To Multi-Class Multi-Label Classification With Neural Networks In Python

by Tobias Sterbak