Event: Data Science On-Ramp

Presenter: Jon Tupitza, CTO Architect

BEOP.CTO.TP4
Owner: OCTO
Revision: 0001
Approved by: JAT
Effective: 08/30/2018

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

Data Science *On-Ramp*

Machine Learning & AI

Categories & Applications of Machine Learning Algorithms

Jon Tupitza
Practice Director, Data Platform & Predictive Analytics

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

BUCHANAN & EDWARDS

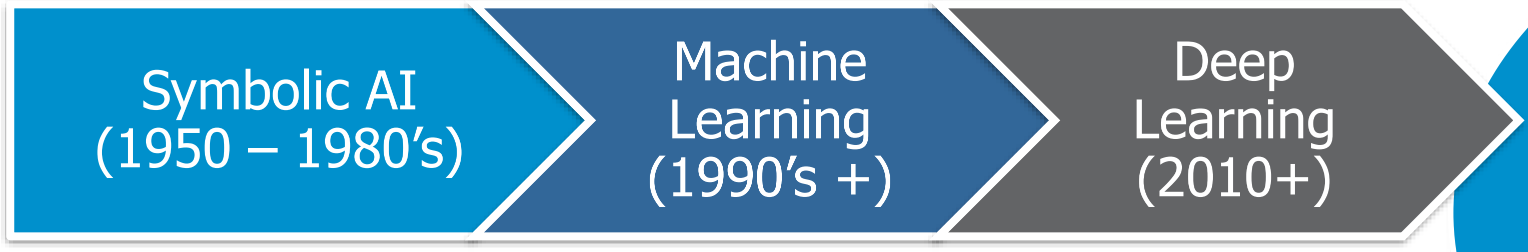
Take-Aways

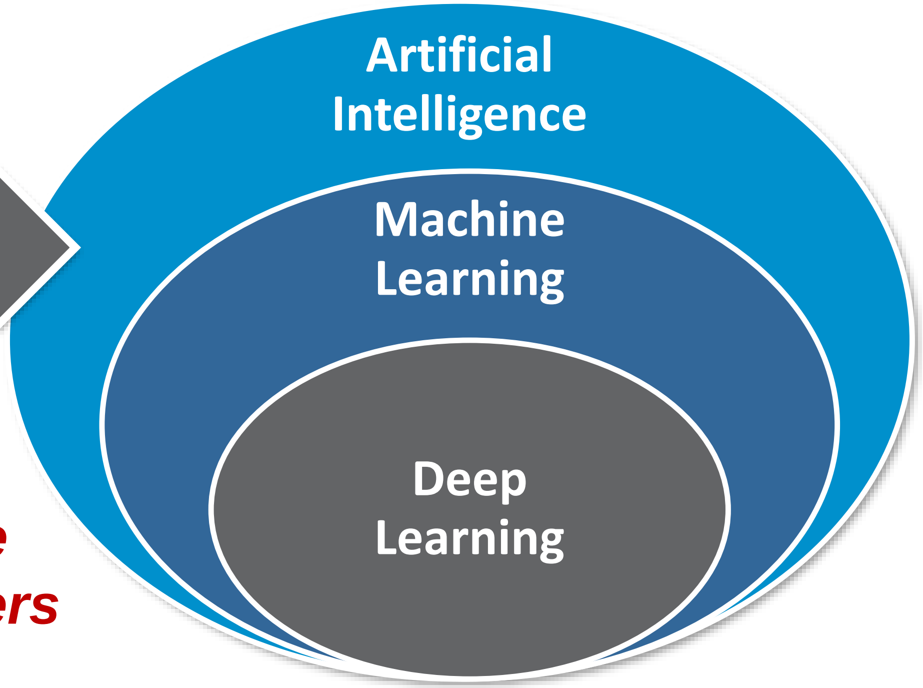
- Basic Understanding of Artificial Intelligence & Machine Learning
- Define the Data Science Process
- Identify Categories and Uses for Machine Learning Algorithms
- Fundamental Understanding of Deep Learning & Image Analysis

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.
BUCHANAN® & EDWARDS

Artificial Intelligence: The Origins of Machine Learning

- Early efforts at AI involved Defining Explicit Rules for Processing Data
- Machine Learning involves Inferring Rules from the Structure of Data
- Deep Learning extends Machine Learning to Hyper-Parametric Data
  - Learns from Successive Layers of Increasingly Meaningful Representations
  - Mimics Neurons in the Human Brain





The advancement of Artificial Intelligence has been made possible only through the Data Explosion of the Internet and the increasing power of modern computers

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

Machine Learning: Automates Problem-Solving

Using computational power to detect and exploit patterns in data

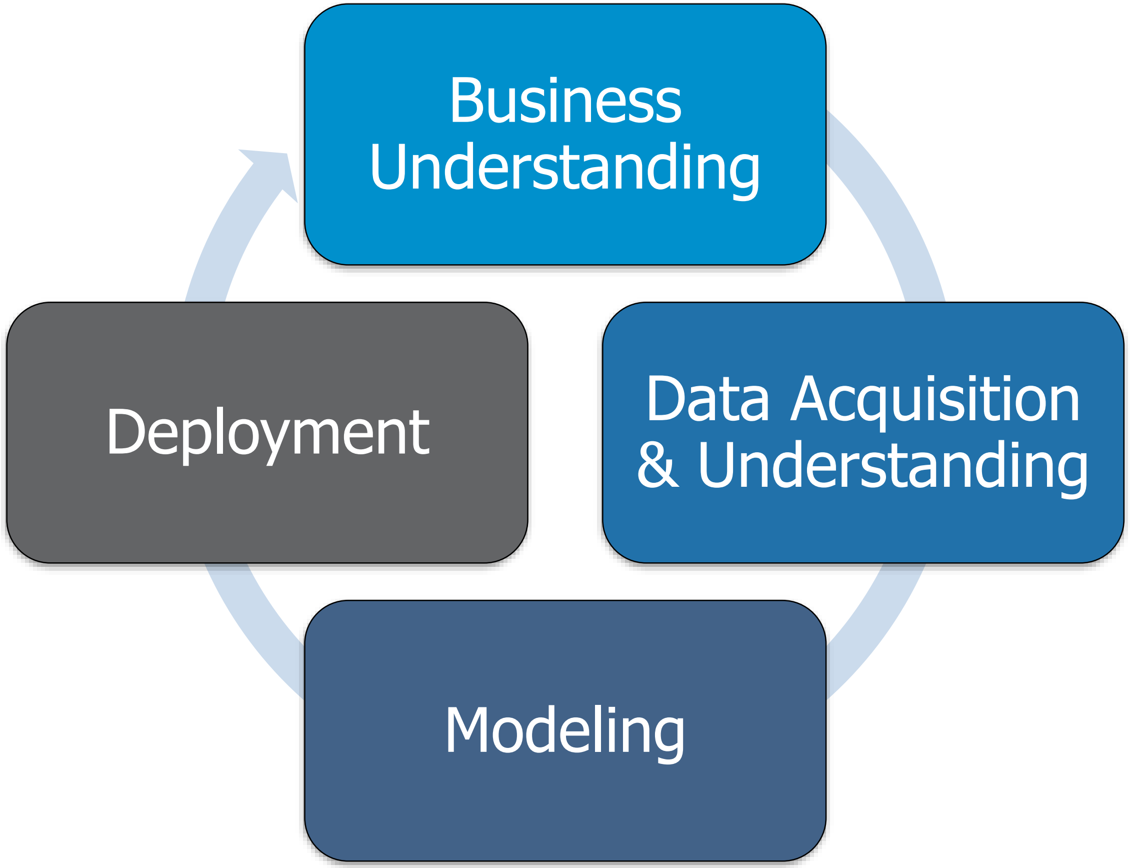
- Rather than trying to solve a problem, ***train*** a computer to solve the problem for you
- ***Fitting*** a model enables the system to predict outcomes by observing examples
- Well ***fit*** models generalize well. They produce reliable predictions when exposed to new data; not over-fitted!

- Spam & Fraud Detection
- Decision Engines: Approval or Disapproval
- Recommendation Engines: Which one to choose
- Predictive Maintenance: When will it break?
- Image Analysis: Classification & Object Detection
- Text Analysis: Term & Document Comprehension
- Natural Language Processing & Machine Translation

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

The Team Data Science Process

Largely Heuristic! Based on Conducting Experiments (i.e., Scientific Method)



Business Understanding	• Identify the Problem Domain • Identify the Solution Scenario
Acquire & Understand Data	• Load, Prepare & Explore Data • Identify Impactful Features
Develop Machine Learning Models	• Select & Engineer Features • Train, Evaluate & Tune Models
Deployment	• Publish Models as Webservices • Consume Models Visually and Programmatically

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.
BUCHANAN & EDWARDS

Machine Learning: Categories and Algorithms

Supervised Learning:
Classification and Regression

- Learn by historic example to predict the future
- Requires both Features and a Label (Observations where the answers are known)

Train: Give the computer samples from which to learn

Test: Give the computer different samples from which to assess accuracy

Evaluate: Analyze test results to determine accuracy of the model

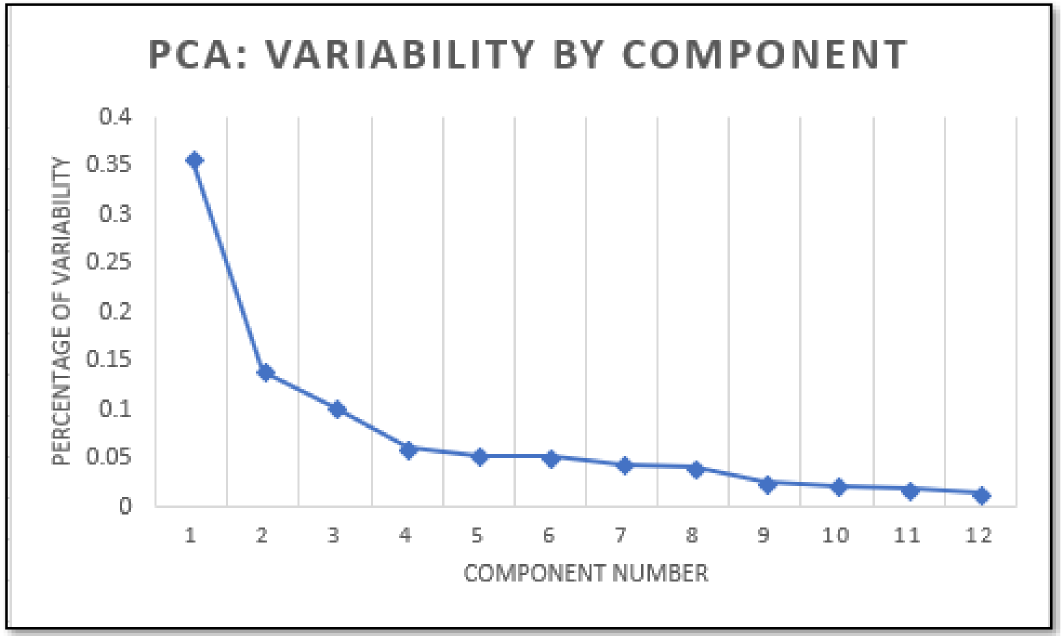
Unsupervised Learning:
Clustering
(K-Means & Hierarchical)

- Understanding the past by gaining insight from historic data
- Involves Feature-sets having no Labels (Observations where answers aren’t known)
- Examine data to reveal its intrinsic [but hidden] structure
- Make recommendations by grouping people, things, or events together

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

BUCHANAN® & EDWARDS

Dimensionality Reduction: Principal Component Analysis



- Reduces the Number of Features
  - Simplifies data analysis
  - Transforms high-dimensional data into a smaller collection of more influential components
- Descriptive Analytic Method:
  - Being a form of Unsupervised Learning, it serves to categorize observations in the absence of a (dependent) predictor variable
- Addresses Non-linear Correlations
  - Summarizes the influence of non-linear relationships
  - Extracts variability from correlated features into powerful components

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

Supervised Learning: Classification

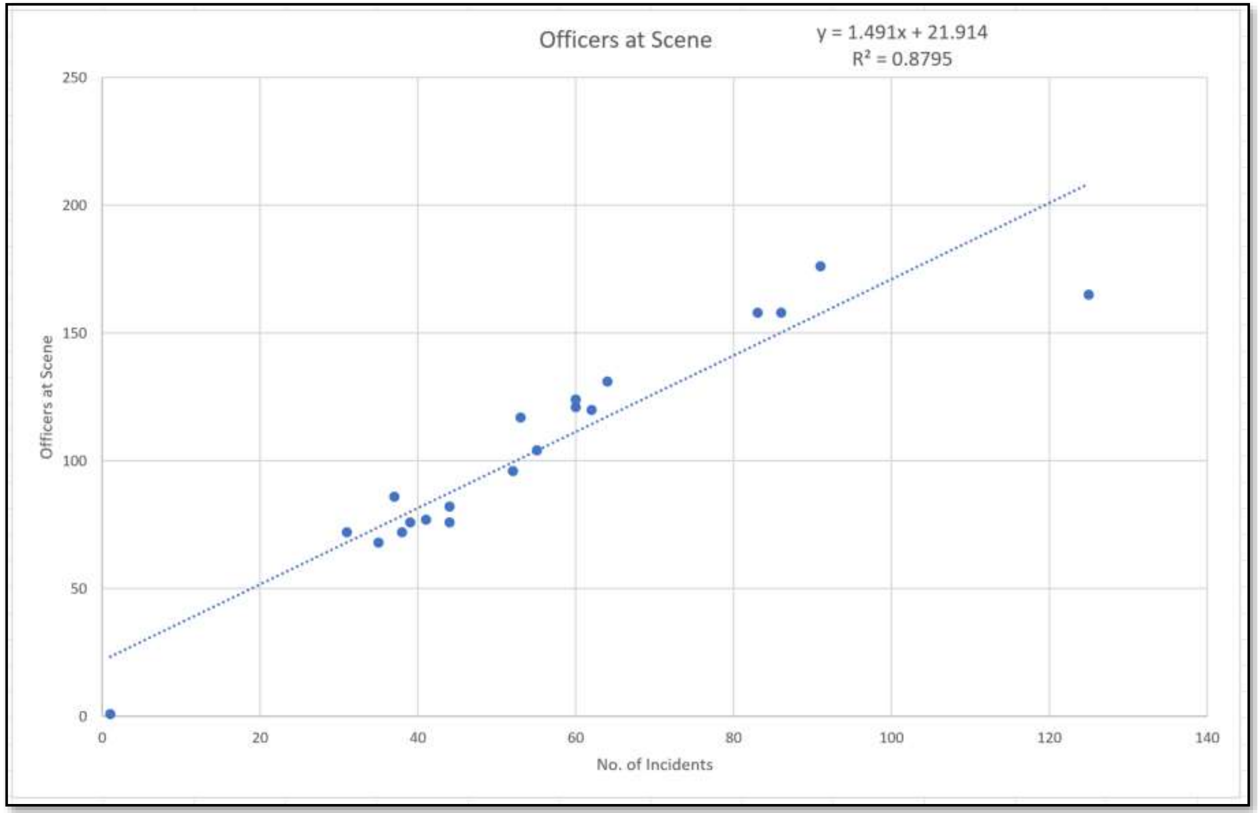
- Requires both Features and a Label (where the answers are known)
  - Identify features most collinear to the label; eliminating those that aren’t
- Produces discrete-value outcomes; i.e., categorical values
  - Binomial (2-class) Classification: Boolean Decisions; e.g., Loan approvals
  - Multinomial (Multi-class) Classification:
    - Concatenate multiple binomial classifiers: Combinations and Permutations
    - Learning from a finite set of values
- Determines into which category each observation should be classified:
  - Based on performance criteria, which grade should be awarded
  - Based on attributes, which type is it? Is it an Animal, Vegetable or Mineral?

	H_0 True	H_0 False
Reject H_0	Type I Error	Correct Rejection
Fail to Reject H_0	Correct Decision	Type II Error

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

Supervised Learning: Regression

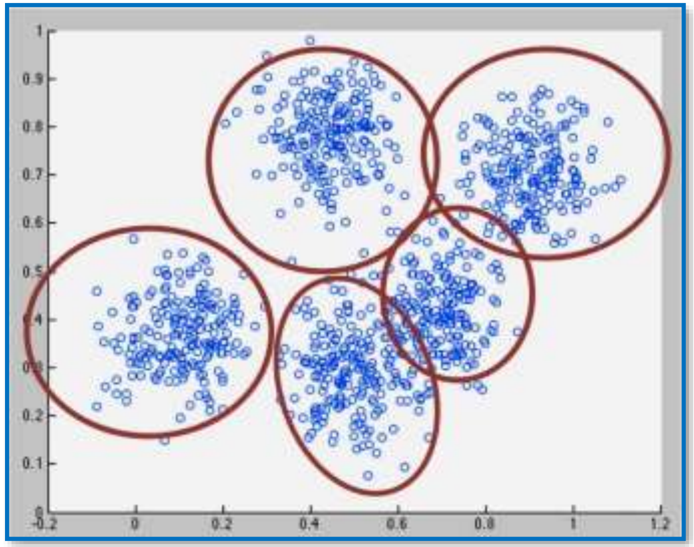
- Requires both Features and a Label (where the answers are known)
- For determining if a correlation exists between 2 or more variables:
  - Predictor (X); e.g., # *Incidents*
  - Response (Y); e.g., # *Officers*
- Produces continuous-value outcomes; i.e., real numbers
  - Linear Regression produces continuous values: e.g., can be any numeric value
  - Logistic Regression produces discrete values: e.g., must be a digit between 0 – 9



Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

Unsupervised Learning: K-Means Clustering

- Used where Feature-sets have No Labels (the answers aren't known)
- Appropriate for Separating Overlapping Spheroid Clusters
- This Algorithm Minimizes the Sum of Distances from Clusters' Centers:
  1) The "hidden" underlying structures in the data are discovered
  2) Based on that structure, each of the observations (data points) are iteratively moved closer to conceptual center-points that represent the centers of ***k*** groups.
  3) The centers are then moved to optimize the centrality of their location within each group
  4) Points are moved closer to the closest center-point
  5) The entire process is iterated until the sum of the distances from their centers is minimized

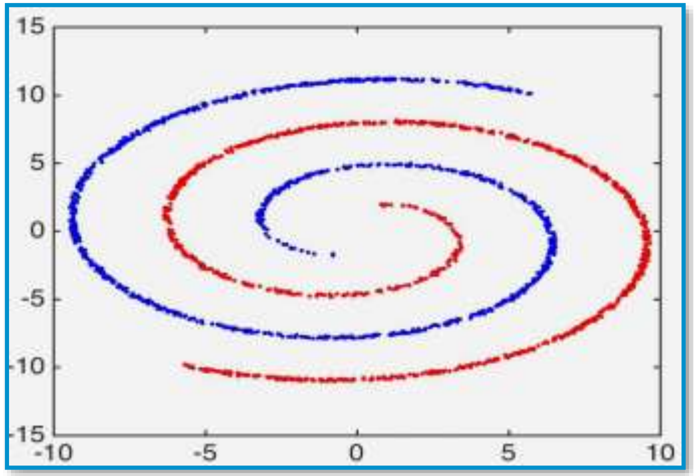


Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

BUCHANAN® & EDWARDS

Unsupervised Learning: Hierarchical Clustering

- Used where Feature-sets have No Labels (the answers aren't known)
- Groups Observations by Association using Like Attributes
  - Documents or Emails by Topic: Detect cyber-crime, perform research
  - People by their Behaviors: Categorize Customers, Identify Criminals
  - Entities by their Properties: Animal, Vegetable or Mineral?
- Can be used to Identify Which Attributes are the Most Influential
- Appropriate for Separating Linear Clusters
  1) Places each observation into its own cluster
  2) Merge the clusters having the closest two points
  3) Continue until all clusters belong to closest groups
  4) You Must Stop before it becomes a single cluster



Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

Deep Learning: Algorithms and Applications

- Deep Neural Networks (DNN) & Convolutional Neural Networks (CNN)
  - Neural Networks mimic layers in the human brain
  - Each layer learns a higher abstraction of the preceding layer
  - Typically involves fitting many [100+ Millions] of parameters
  - Requires voluminous data sets, and extensive computational capabilities
- Applications:
  - Image and Video Analysis (CNNs): Object Recognition and Classification
  - Natural Language Processing (NLP): Machine Translation, Text Extraction, Document Comprehension
  - Audio Analysis: Voice Recognition, Speech-to-Text

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

BUCHANAN® & EDWARDS

Supervised Machine Learning: Image Classification

- Pre-categorized dataset
- Model learns patterns that exist within categories
- Application: predict the category of a new piece of data

TRAINING

Apple

Orange

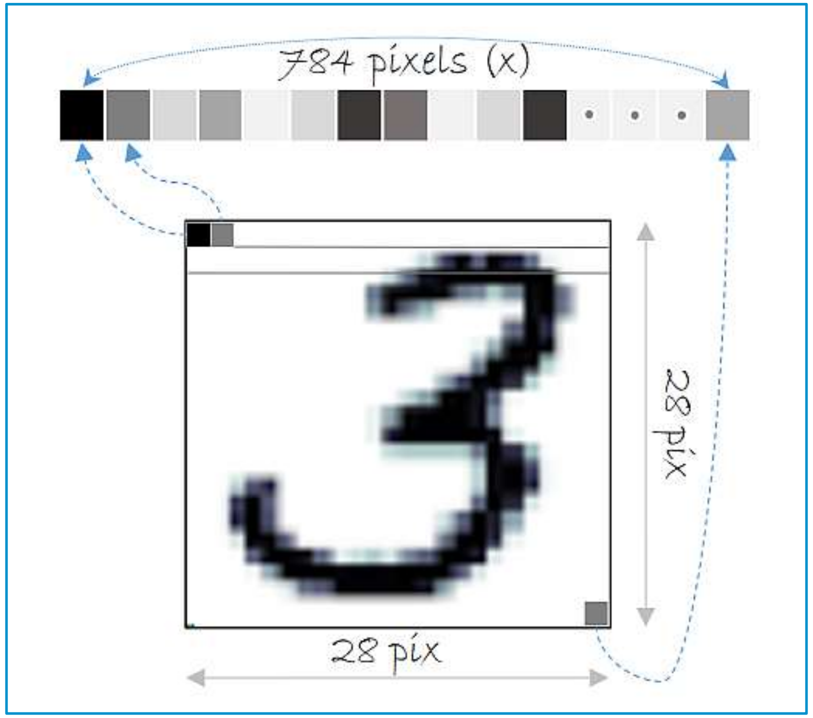
TESTING

???

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

BUCHANAN® & EDWARDS

Historic Classification Methods: Logistic Regression



Features = height*width*channels

- 64 * 64 * 3 = **12,288**
- 92 * 92 * 3 = **25,382**
- 128 * 128 * 3 = **49,152**

- Multi-Class Logistic Regression
  - A fundamental [statistical] machine learning method that uses a combination of features to render the probability of predicting classes
  - Builds upon simple logistic regression to render multiple classifications; e.g., 0 – 9
- Reading and Storing Data
  - Flattens the input image to serialize its pixels into a one “row” data vector
  - Looses feature correlations pertaining to the spatial relationships present in the image

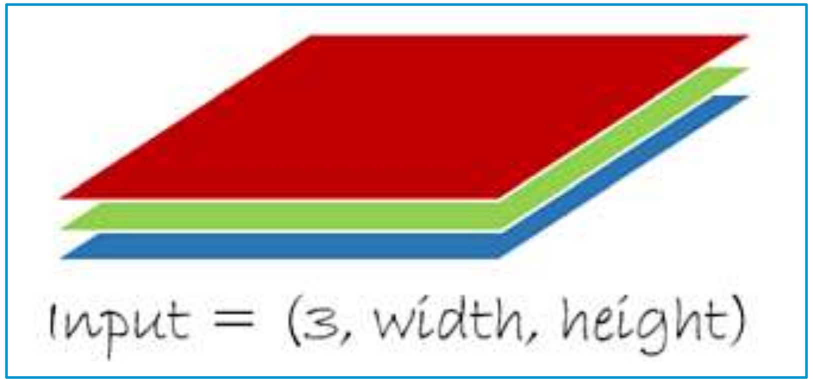
Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

Convolution: What is it?

- Convolution defined:
  1. Something that is complex and difficult to know
  2. A coil or twist, especially one of many
  3. A sinuous fold in the surface of the brain
  4. A function derived from two given functions by integration which expresses how the shape of one is modified by the other.

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.
BUCHANAN® & EDWARDS

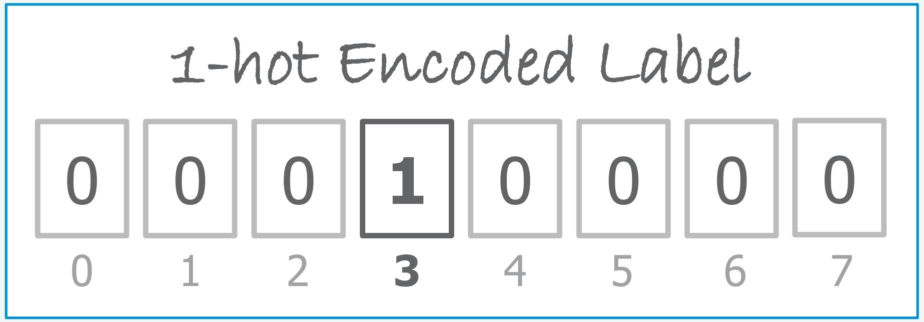
Convolutional Neural Networks: Data Representation



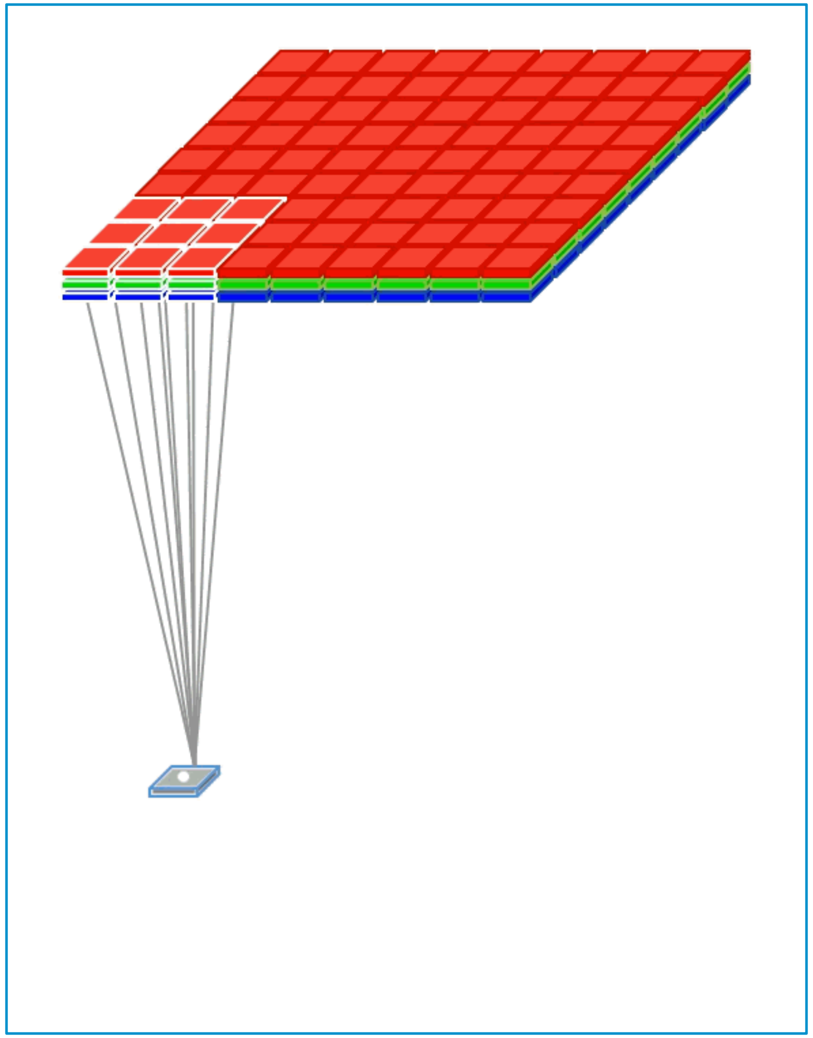
- Input data is shaped into a 3D matrix so as to preserve spatial relationships between pixels
  - Number of Channels, Height, Width
  - Color images are typically represented using 3 channels; Red, Green and Blue (RGB)

- CNTK stores data using CTF; a simple text format involving 2 fields (label and features) that uses **1-hot encoding** to represent each class:
  0 = Neutral, 1 = Upset, 2 = Fear, 3 = Happiness, 4 = Sadness, 5 = Surprise …
- If there are 8 possible classes the output of label class 3 becomes: 00010000
- If the image has 92 * 92 * 3 features the output contains 25,392 integers conveying RGB values

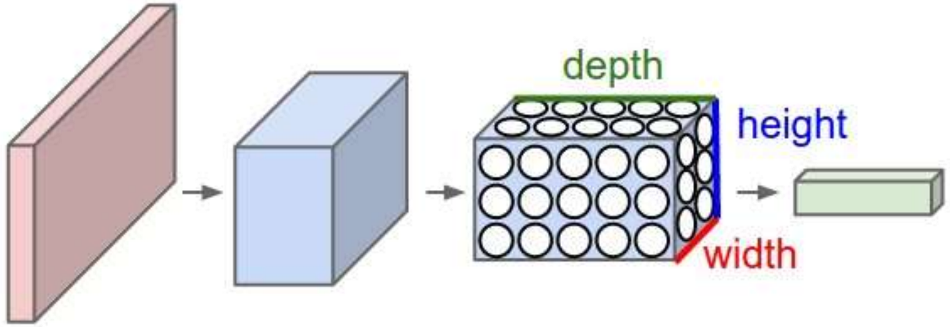
```
| labels 00010000 | features 0 255 0…
```



Convolutional Neural Networks: Convolutional Layers

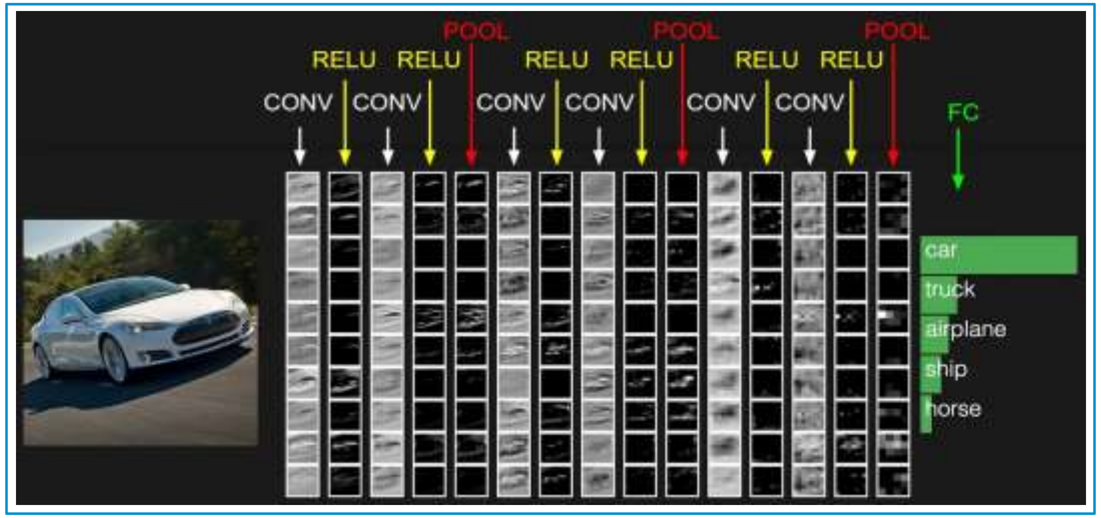


- Explicitly assumes that inputs are images
- Vastly reduces the number of parameters
  - Increases efficiency of forward function
- Enables processing larger images, using larger filter sizes, learning of more filters, and creation of deeper architectures (e.g., 152-layers)
- Made up of 3D neurons, each of which has learnable weights and biases:



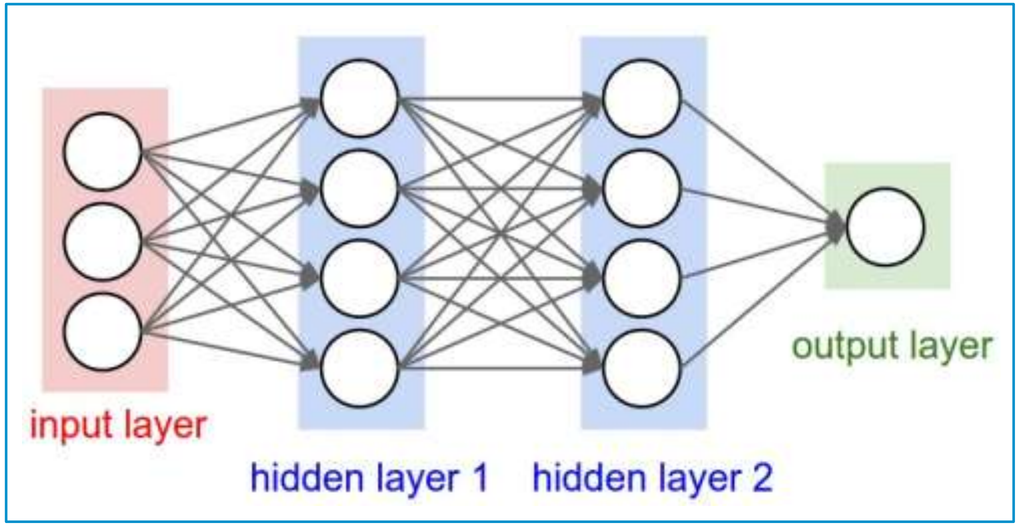
Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

Convolutional Neural Networks: Architectures



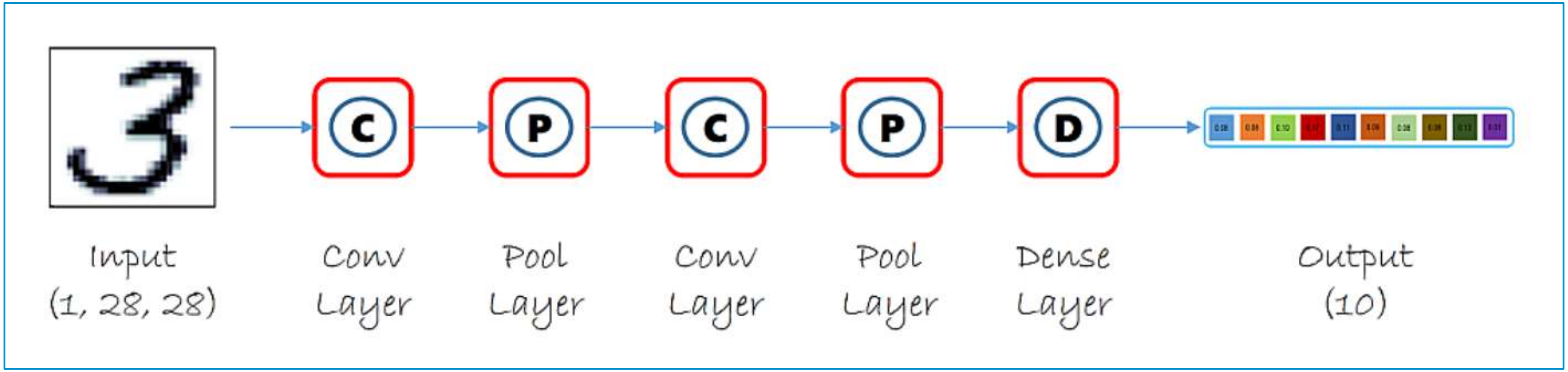
- Stack layers to transform the image volume into an output volume
  - Convolutional Layers
  - ReLu Layers (Activation Function)
  - Pooling Layers
  - Fully-Connected Layer

- Input is transformed by a set of hidden layers
  - Each hidden layer contains a set of neurons
  - Each neuron is fully connected to all of the neurons in the previous layer
  - Neurons is each single layer function independently; don't share connections
- The last fully-connected (output) layer represents the class scores



BUCHANAN® & EDWARDS

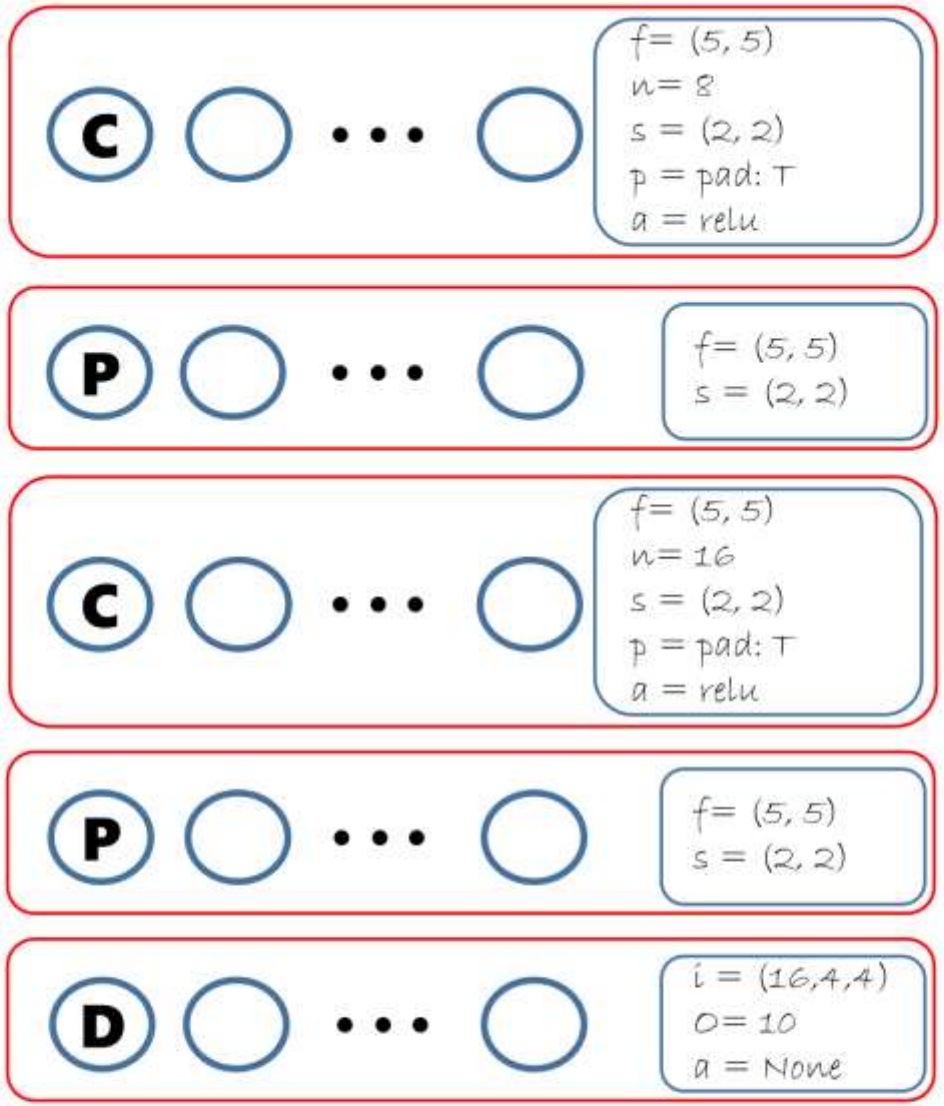
CNN Architectures: Typical Convolution Network



Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

CNN Architectures: Typical Convolution Network Code

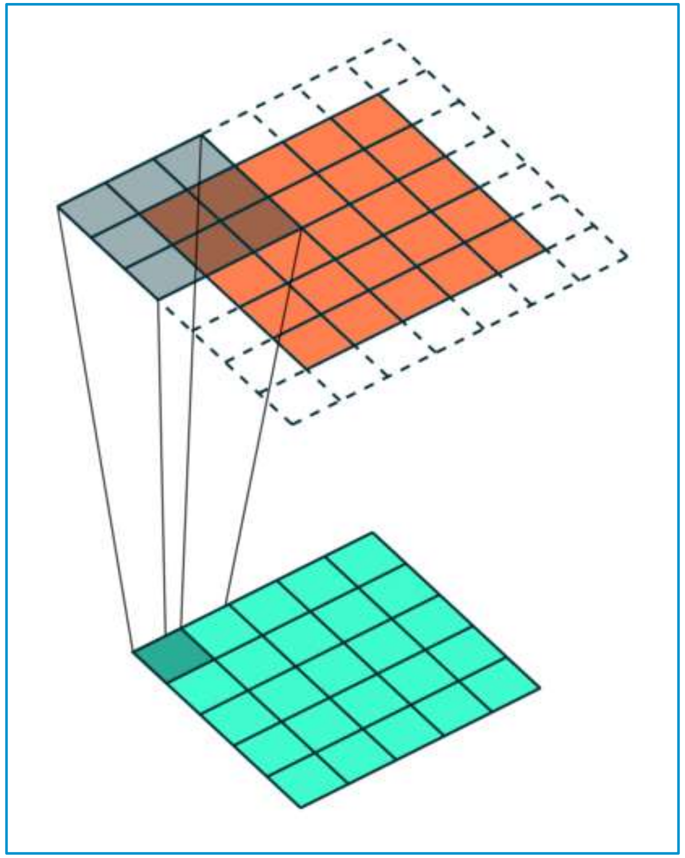
```
def create_model(features):
    with default_options(activation = relu):
        h = features
        h = Convolution2D(filter_shape=(5,5),
                          num_filters=8,
                          strides=(1,1), pad=True)(h)

        h = MaxPooling(filter_shape=(2,2),
                       strides=(2,2))(h)

        h = Convolution2D(filter_shape=(5,5),
                          num_filters=16,
                          strides=(2,2), pad=True)(h)

        h = MaxPooling(filter_shape=(2,2),
                       strides=(2,2))(h)

        r = Dense(num_output_classes,
                  activation = None)(h)
        return r

z = create_model(input)
```



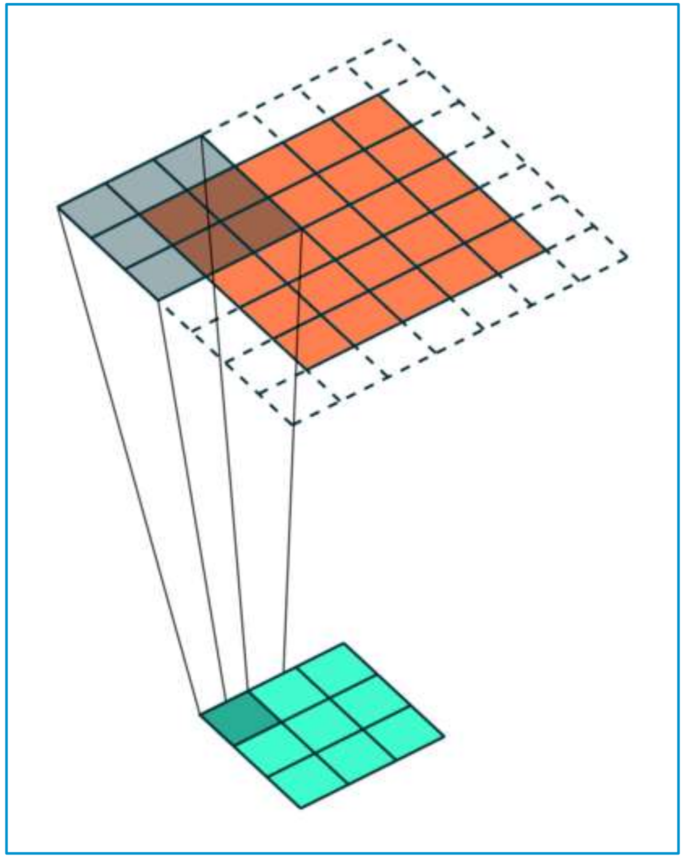
Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

CNN Filter Positioning: Padding and Stride

- The `padding` parameter, when set to `true`, adds a 1-pixel border to enhance the filter's ability to capture details along the edge of the image (receptive field)
- The `strides` parameter controls how far the filter should step to the right and down while traversing each row of the image to reduce the number of parameters



Padding with No Stride



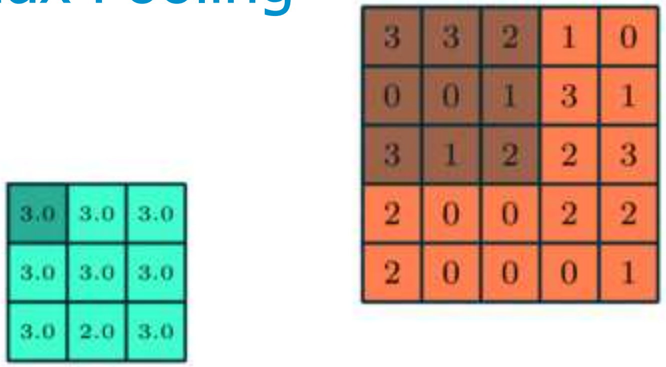
Padding with Stride = 2

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

BUCHANAN® & EDWARDS

Convolutional Neural Networks: Pooling Layers

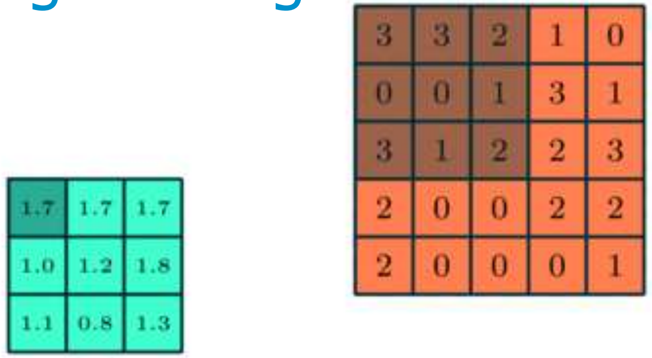
Max Pooling



- Controls the number of parameters when building especially deep networks
- Each Convolutional Layer can have a corresponding Pooling Layer that reduces the dimensionality of the previous layer
- Increases the model's tolerance to changes in the object location within the image

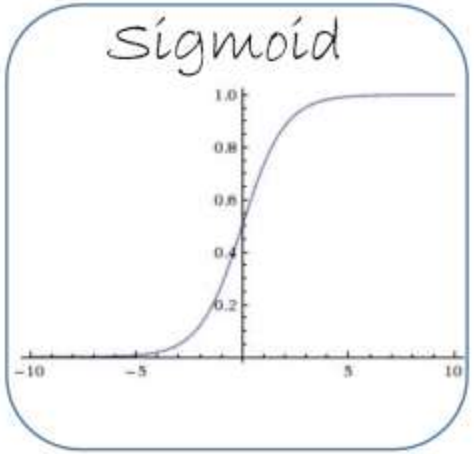
- Max Pooling:
  - Outputs the maximum of the input values corresponding to the filter position of the input
- Average Pooling:
  - Outputs the average of the input values corresponding to the filter position of the input

Avg. Pooling

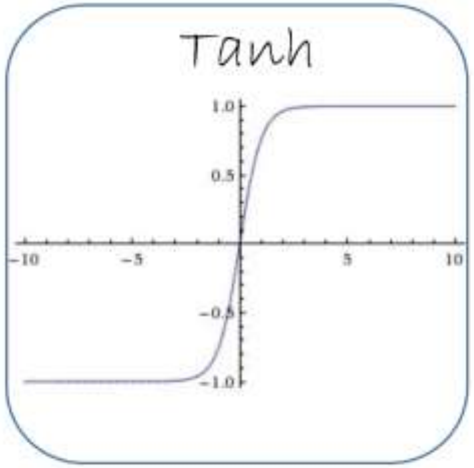


Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

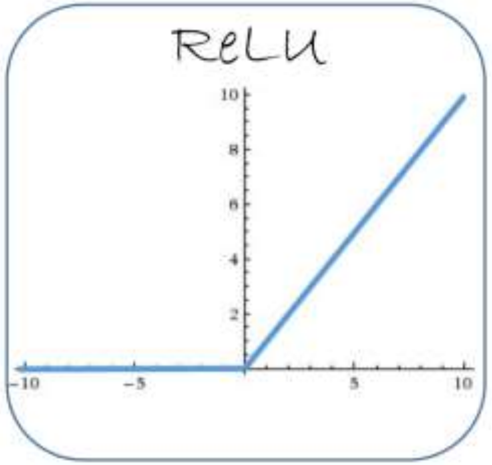
Convolutional Neural Networks: Activation Functions



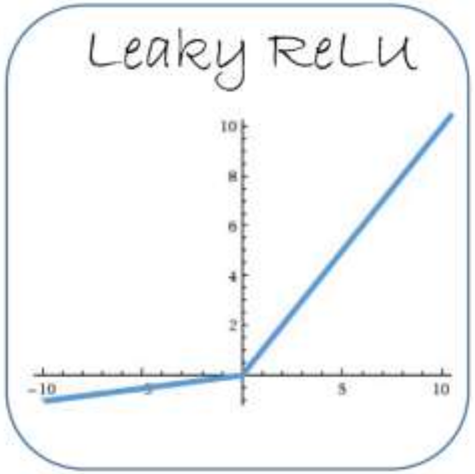
$\sigma(x) = 1/(1 + e^{-x})$



$\tanh(x) = (e^{x} - e^{-x})/(e^{x} + e^{-x})$



$f(x) = max(0, x)$



$f(x) = 1(x < 0)(\alpha x) + 1(x \geq 0)(x)$

- a.k.a. Non-Linearities
- Normalizes values: Maps the real value to a range; e.g., 0 - 1

Buchanan & Edwards Proprietary: Printed copies of this document are UNCONTROLLED. Verify that this is the correct version before use.

BUCHANAN® & EDWARDS