Ivelin Andreev

@PASS SQLSATURDAY SOFIA | 14 OCT 2017

Machine Learning for IoT

Unpacking the Blackbox

About me

- **Software Architect @ ICB SOFTWARE INNOVATION**
  - 15 years professional experience
  - .NET Web Development MCPD
- **External Expert Horizon 2020**
- **External Expert Eurostars-Eureka & IFD**
- **Business Interests**
  - Web Development, SOA, Integration
  - Security & Performance Optimization
  - IoT, Computer Intelligence
- **Contact**

ivelin.andreev@icb.bg
www.linkedin.com/in/ivelin
www.slideshare.net/ivoandreev

Microsoft CERTIFIED Professional Developer

European Commission

eurostars™

Innovationsfonden

cost

PASS SQLSATURDAY SOFIA | 14 OCT 2017

Sponsors

Microsoft

SBTech we know sports

PASS

tech talks

adatis

ICB SOFTWARE INNOVATION

KPMG

SQL MASTER ACADEMY RAISE AN EXPERT

SUPERHOSTING.BG

SOFTWARE GROUP

software AG

PASS SQLSATURDAY

Agenda

Machine Learning

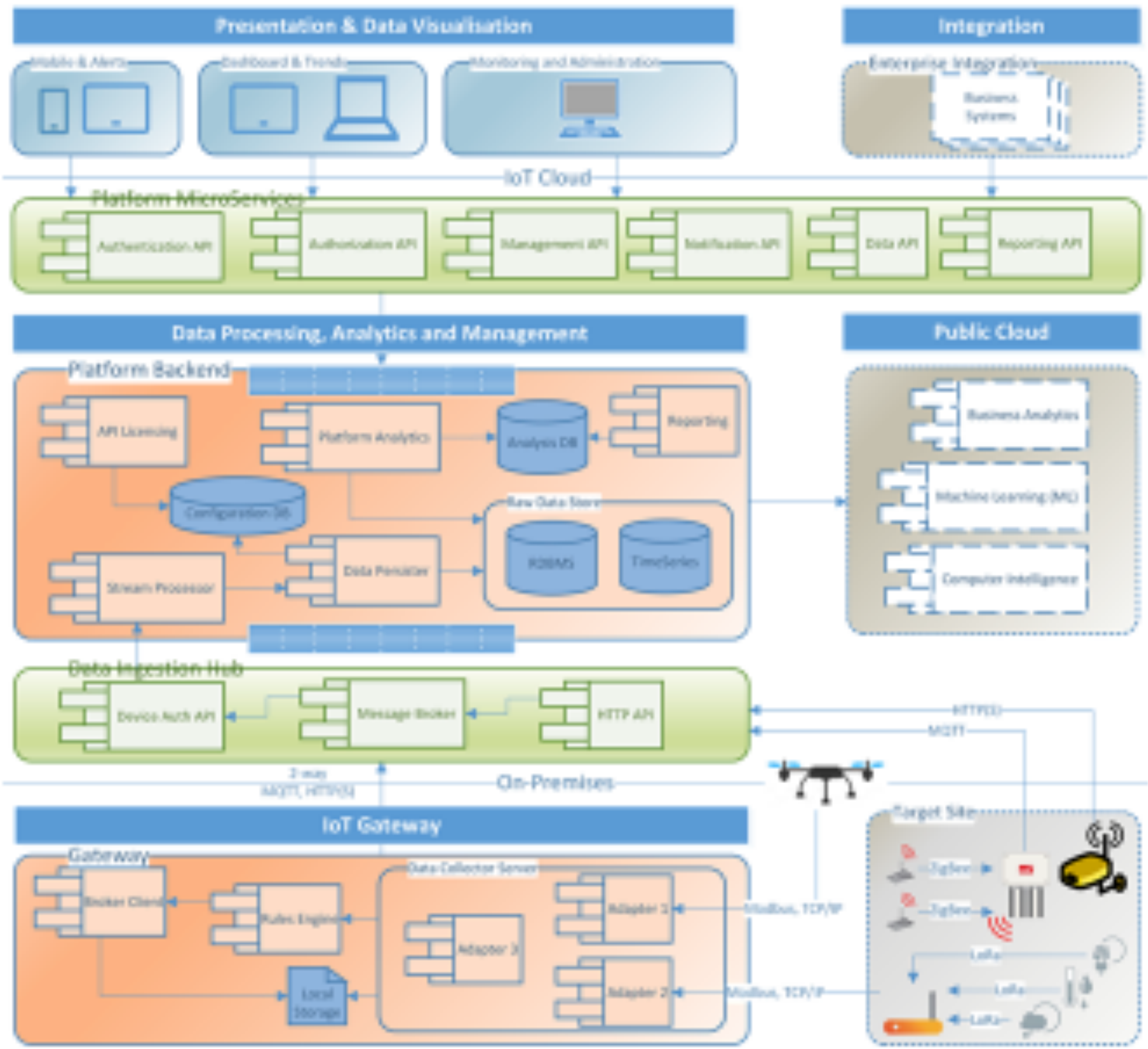
- Azure ML and Competitors
- Choosing the right algorithm
- Reasons your ML model may fail
- Algorithm performance
- Deep neural networks
- ML on-premises
- Demo

Real World Business Cases

- **Predictive maintenance**
  - Is it likely that the machine will break based on sensor readings?
- **Forest fire prediction**
  - Is it likely that a fire will start based on environment parameters?
- **Automated teller machine replenishment**
  - How to optimize ATM replenishment based on historical data for transactions?
- **Spear phishing identification**
  - Is a web site likely to be phishing based on URL and HTTP response?

ASPires

Distributed IoT Platform



Funded by

EUROPEAN CIVIL PROTECTION AND HUMANITARIAN AID OPERATIONS

2016/PREV/03 (ASPIRES)

Horizon 2020 European Union funding for Research & Innovation

Hochschule Fulda University of Applied Sciences

ICB SOFTWARE INNOVATION

Comicon

PASS SQLSATURDAY SOFIA | 14 OCT 2017


ML is not Black Magic
SQLSATURDAY

Supervised Learning

- Majority of practical ML uses supervised learning
- Mapping function approximated from past experience
  - Regression f(X) = Y, Y is a real number
  - Classification f(X) = Y, Y is a category label

- Training
  - Labeled positive and negative examples
  - From unseen input, predict corresponding output
  - Learning until acceptable performance is achieved

Unsupervised Learning

- Discover hidden relations and learn about the data
  - Clustering f(X) = [X1,..., Xk], k disjoint subsets
  - Association f(Xi, Xj) = R, relation
- Training
  - All examples are positive
  - No labeling / No teacher
  - No single correct answer
- Practical usage
  - Derive groups, not explicitly labeled
  - Market basket analysis (association among items)

Machine Learning with Microsoft Azure

- Primary goal: makes deployable and scalable web services from the ML modules.
- Though experience for creating ML models is great, it is not intended to be a place to create and export models

Main ML Market Players

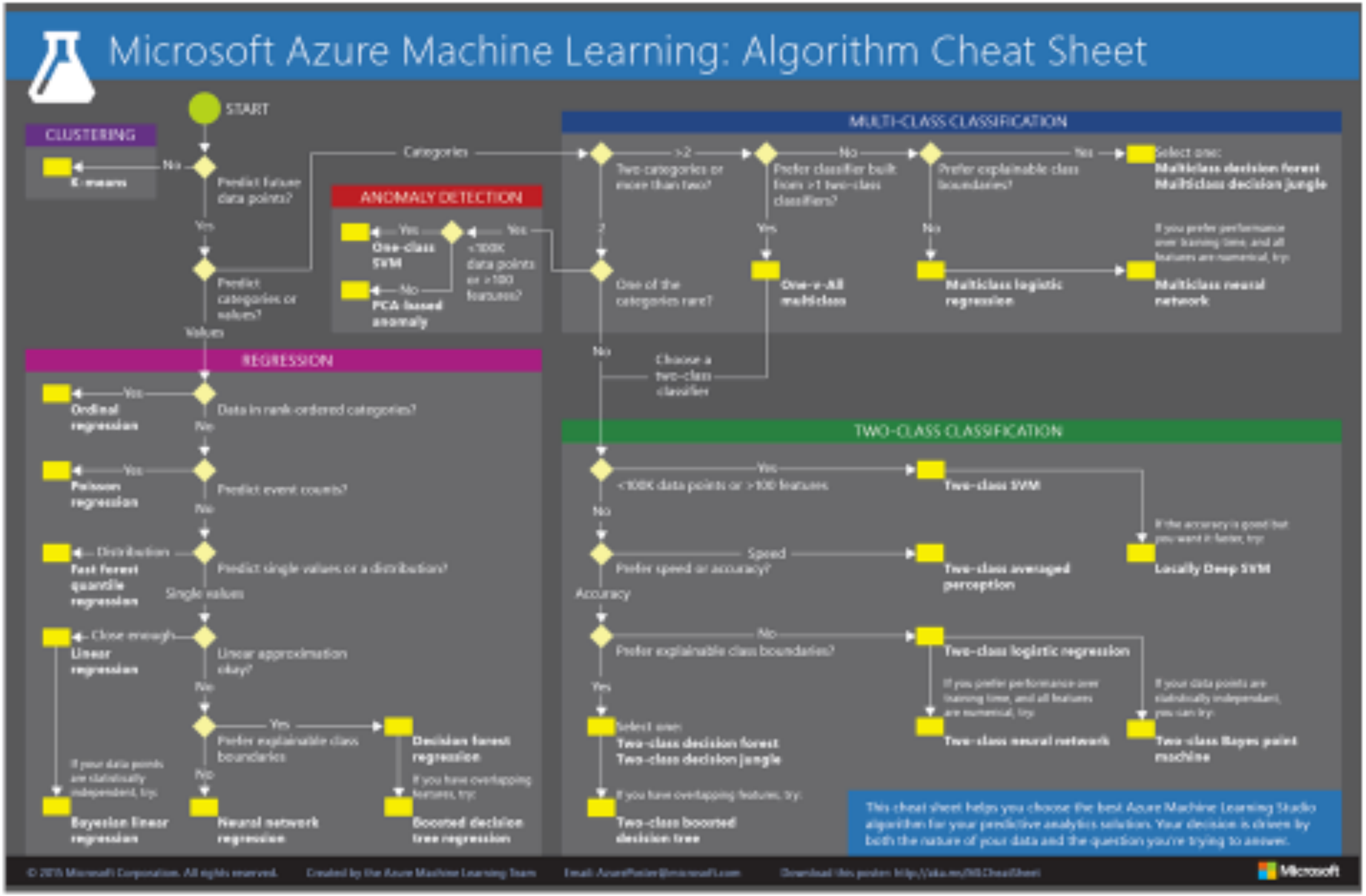
	Azure ML	BigML	Amazon ML	Google Prediction	IBM Watson ML
Flexibility	High	High	Low	Low	Low
Usability	High	Med	High	Low	High
Training time	Low	Low	High	Med	High
Accuracy (AUC)	High	High	High	Med	High
Cloud/ On-premises	+/-	+/+	+/-	+/-	+/-
Algorithms	Classification Regression Clustering Anomaly detect Recommendations	Classification Regression Clustering Anomaly Recommend	Classification Regression	Classification Regression	Semantic mining Hypothesis rank Regression
Customizations	Model parameters R-script Evaluation support	Own models C#, R, Node.js	Few parameters		

Pricing?

	Azure ML	BigML	Amazon ML	Google Prediction	IBM Watson ML
Model Building	$1.00 / h $10.00 / user / month	$30 - $10'000 / month	$0.42 / h	$0.002 / MB	$0.45 / h $10.00 / service
Retraining (per 1000)	$0.50	-	N/A	$0.05	-
Prediction (per 1000)	$0.50	-	$0.10	$0.50	$0.50
Compute (per hour)	$2.00	-	$0.42	-	-
Free Usage	1'000 / month 2h compute	Dataset Size Max 16MB	N/A	10'000/month	5'000 / month 5h compute
Notes		Private deployment $55'000 / year		Shutdown April 30, 2018 Cloud ML Engine (TensorFlow)	

Azure ML Flow

1. Dataset
2. Training Experiment
3. Predictive Experiment
4. Publish Web Service
5. Retrain Model



PASS SQLSATURDAY
SOFIA | 14 OCT 2017

Do you need Math to work with ML?

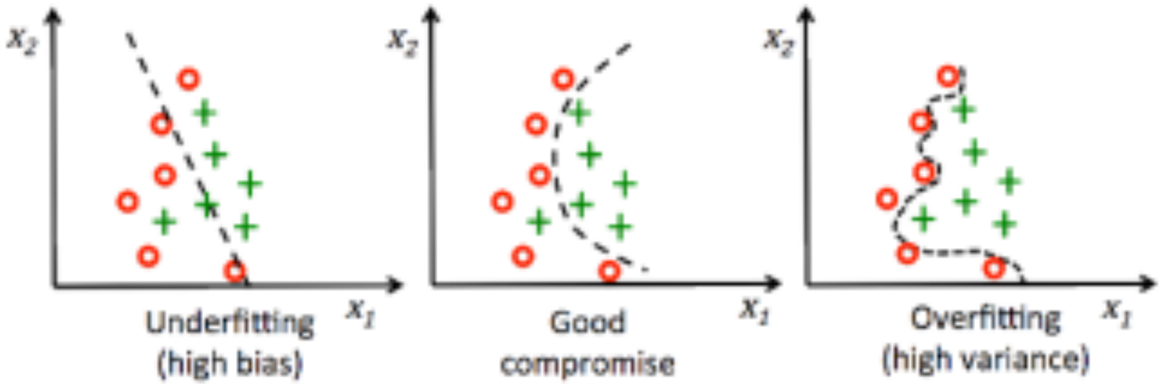
Answer: *It is not required, but would definitely help*

Data Science is what is really necessary

Interdisciplinary field about processes and methods for extracting relations and knowledge from data

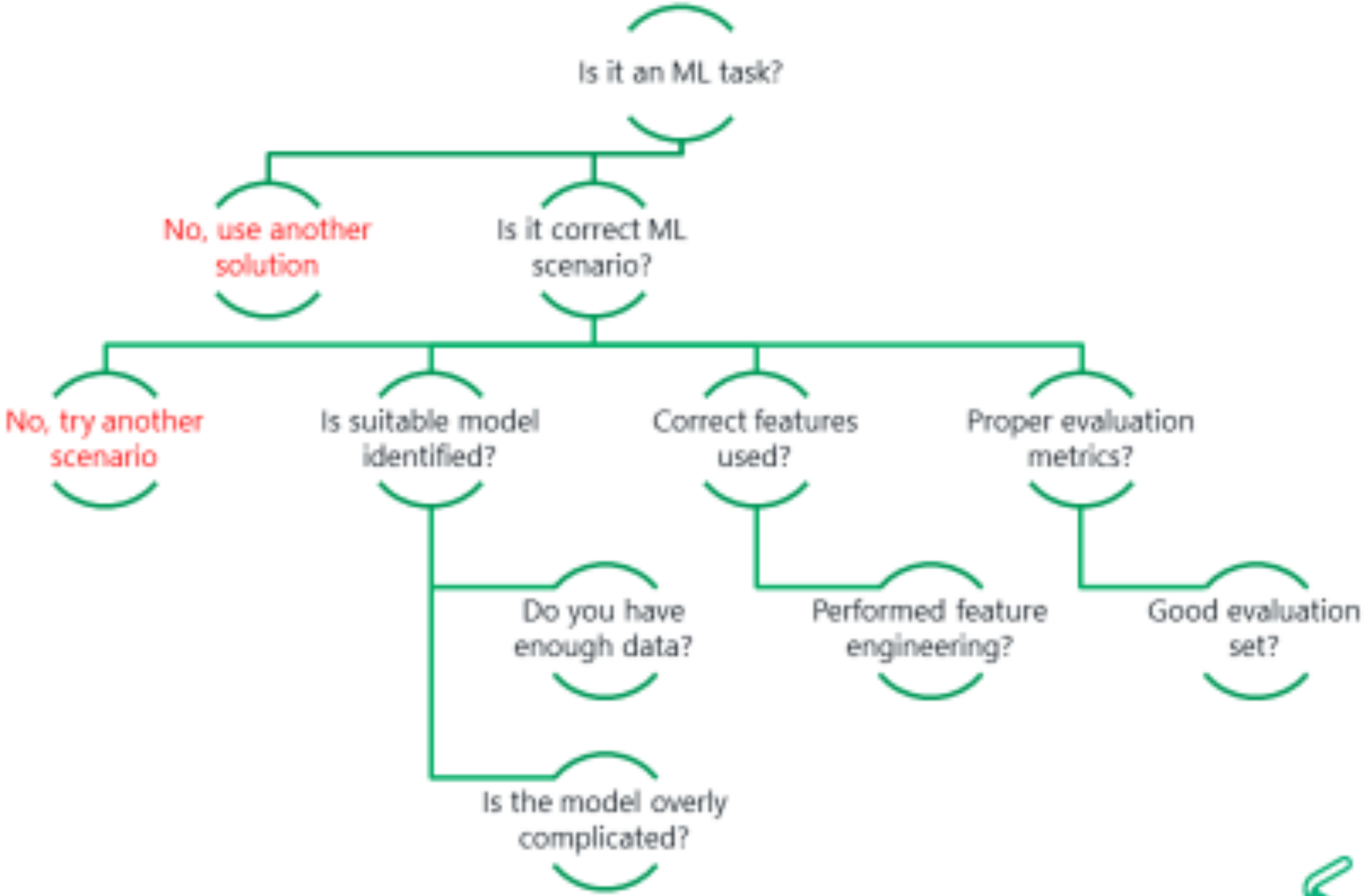
Working around Math

- Select the **right algorithm**
  - Use cheat-sheets instead
- Choose **parameter settings**
  - Experiment
  - Use parameter-sweep
- Identify **underfitting and overfitting**
  - Compare training and validation scores




I selected an appropriate
ML algorithm but…
why my ML model fails?

ML Decision Tree



PASS SQLSATURDAY SOFIA | 14 OCT 2017

Diagnose Steps (part 1)

1. Is it an ML task? Are you sure ML is the best solution?
   - **Hard**: X is independent of Y: X <Name, Age, Income>, Height=?
   - **Easy**: X is a set with limited variations. Configure Y=F(X)
2. Appropriate ML scenario?
   - **Supervised** learning (classification, regression, anomaly detection)
   - **Unsupervised** learning (clustering, pattern learning)
3. Appropriate model?
   - Data **size** (small data -> linear model, large data -> consider non-linear)
   - **Sparse** data (require normalization to perform better)
   - **Imbalanced** data (special treatment of the minority class required)
   - Data **quality** (noise and missing values require loss function – i.e. L2)
4. Enough training data?
   - **Investigate** how precision improves with more data

Diagnose Steps (part 2)

5. Model overly complicated
   - Start **simple first**, increase complexity and evaluate performance
   - **Avoid overfitting** to training set
6. Feature quality
   - Have you identified all **useful features**?
   - Use **domain knowledge** of an expert to start
   - **Include any feature** that could be found and investigate model performance
7. Feature engineering
   - The **best strategy** to improve performance and reveal important input
   - **Encode** features, **normalize** [0:1], **combine** features
8. Combine models
   - If multiple models have **similar performance** there is a chance of improvement
   - Use one model for one subset of data and another model for the other

Diagnose Steps (part 3)

9. Model Validation

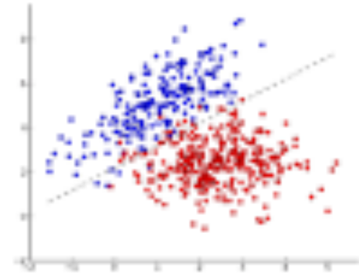
- Use **appropriate performance indicator** (Accuracy, Precision, Recall, F1, etc.)
- How well does the model **describe data**? (AUC)
- Data typically divided into **Training and Validation**
- Evaluated accuracy on **disjoint dataset** (other than training dataset)
- Tune model **hyper parameters** (i.e. number of iterations)


Appropriate Algorithms are
Determined by Data

Types of Algorithms

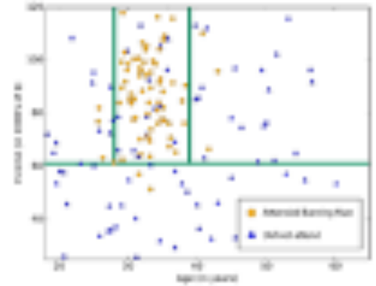
Linear Algorithms

- **Classification** - classes separated by straight line
- **Support Vector Machine** – wide gap instead of line
- **Regression** – linear relation between **variables** and **label**



Non-Linear Algorithms

- **Decision Trees** and **Jungles** - divide space into regions
- **Neural Networks** – complex and irregular boundaries

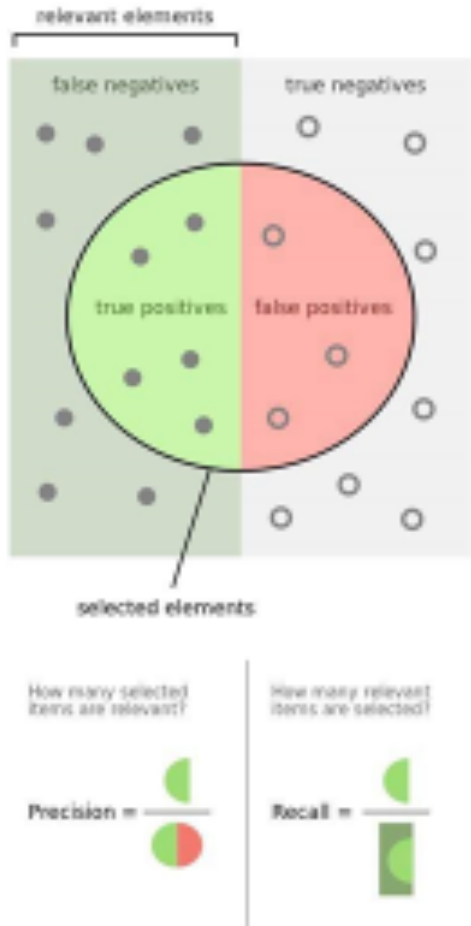


Special Algorithms

- **Ordinal Regression** – ranked values (i.e. race)
- **Poisson Regression** - discrete distribution (i.e. nr. of events)
- **Bayesian** – normal distribution of errors (bell curve)

Threshold Selection

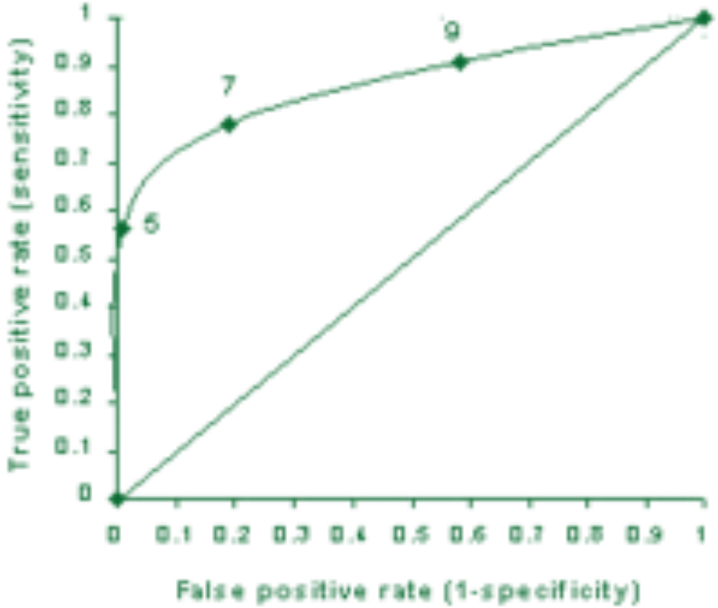
- **Probability Threshold**
  - Cost of one error could be much higher that cost of other
  - (i.e. Spam filter – it is more expensive to miss a real mail)
- **Accuracy**
  - For symmetric 50/50 data
- **Precision**
  - (i.e. 1000 devices, 6 fails, 8 predicted, 5 true failures)
  - Correct positives (i.e. 5/8 = 0.625, <u>FP are expensive</u>)
- **Recall**
  - Correctly predicted positives (i.e. 5/6=0.83, <u>FN are expensive</u>)
- **F1** (balanced error cost)
  - Balanced cost of Precision/Recall



How Good the Model is? (Performance)

- **Binary classification** outcomes {negative; positive}
- **ROC Curve**
  - TP Rate = True Positives / All Positives
  - FP Rate = False Positives / All Negatives
- **Example**

	TP Rate	FP Rate	1-FP Rate
5	0.56	0.99	0.01
7	0.78	0.81	0.19
9	0.91	0.42	0.58



- **AUC** (Area Under Curve)
  - KPI for model performance and model comparison
  - 0.5 = Random prediction, 1 = Perfect match
- For **multiclass** – average from all RoC curves

How Good the Model is? (Regression)

- **Coefficient of Determination** (R^2)
  - Single numeric KPI – how well data fits model
  - $R^2>0.6$ – good, $R^2>0.8$ – very good $R^2=1$ – perfect
- **Mean** Absolute **Error** / Root Mean Squared Error
  - **Deviation** of the estimates from the observed values
  - Compare model errors measure in the **SAME** units
- **Relative** Absolute **Error** / Relative Squared Error
  - **% deviation** from real value
  - Compare model errors measure in the **DIFFERENT** units

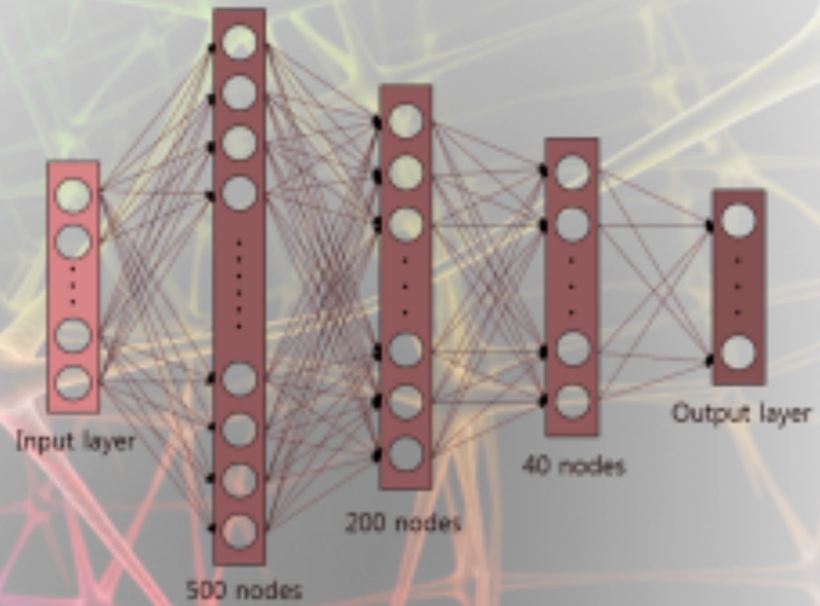
Neural Networks & Deep Learning

- Neural-Networks are considered universal function approximators
- They can compute and learn any function

PASS SQLSATURDAY
SOFIA | 14 OCT 2017

Neural Network Architecture

- Nodes, organized in layers, with weighted connections
  - Acyclic directed graph
- **Layers**
  - Input (1), Output (1)
  - Shallow - 1 hidden layer
  - Deep – multiple hidden layers
- **MS NET# language**
  - Define DNN layers
  - Bundles (connections)
  - Activation functions



Artificial Neuron Activation

- Calculates weighted sum of inputs
- Adds bias (function shift)
- Decides whether it shall be activated

Natural Questions

- Why do we have so many?
- Why some work better than other?
- Which one to use ?

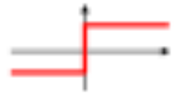
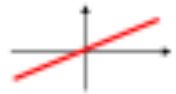
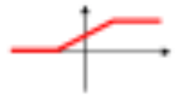
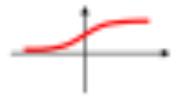
Activation Functions

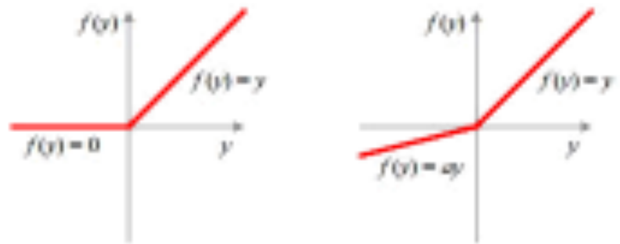
Goal

- Convert input -> output signal
- Output signal is input to next layer
- Approximate target function faster

Samples

- **ReLu, PReLu** – good to start with
- **TanH** and **Sigmoid** – outdated
- **Softmax** – output layer, classification
- **Linear func** – output layer, regression

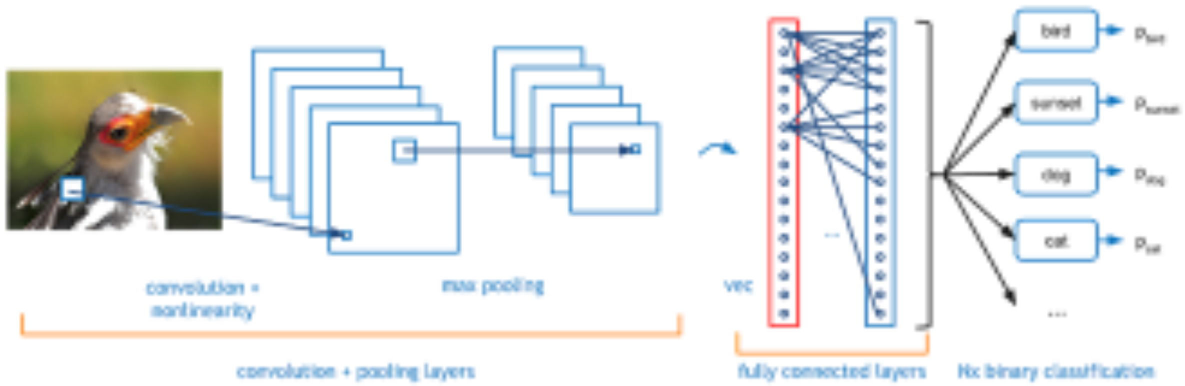
Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	



ReLU vs. PReLU. For PReLU, the coefficient is adaptively learned.

Image Recognition with DNN

- Convolution
- Non-Linearity (i.e. ReLU)
- Downsampling
- Classification



Microsoft Cognitive Toolkit

- The Microsoft way to run **ML on-premises**
- Toolkit for **learning and evaluating DNN**
  - **Production-ready**
  - **Open source** (GitHub since Jan 2016)
  - **Cross platform** (Windows, Linux, Docker)
  - **Highly optimized** for multiple GPUs/machines
- Definition in C++, Python, C# (beta), BrainScript
- Convolutional NN are extremely **good for feature extraction**

Key ML Modules

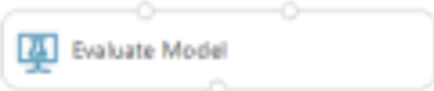
Building blocks for creating experiments

Split data into training and verification sets

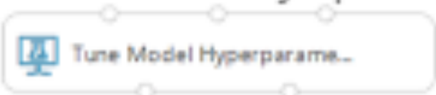
Trains model from untrained model and dataset

Number of arguments = algorithm flexibility

Confirm model results against data set



Get key quality factors of the results or evaluate against another model



Sweep model parameters for optimal settings

PASS SQLSATURDAY SOFIA | 14 OCT 2017

Takeaways

Machine Learning

Cortana Intelligence Gallery (3700+ **Sample Azure ML Projects**)

Evaluating **model performance**

Azure ML documentation (full)

Azure ML **video guidelines**

ML **algorithms by use case**

Artificial Neural Networks

Intuitive Explanation of **Convolutional Neural Networks**

Biginners guide to Convolutional Neural Networks

Understanding activation functions

Activation functions, **which is better**