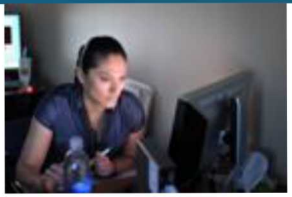
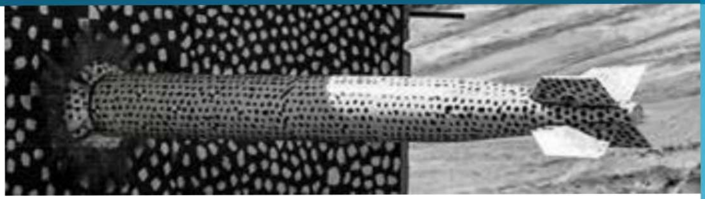
SAND2019-10566PE

Artificial Intelligence and Machine Learning: Overview





Raymond Byrne, David Stracuzzi, Warren Davis,
Jean-Paul Watson, Matthew Reno, Logan Blakely

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Course Outline

1. Introduction to Artificial Intelligence and Machine Learning (Ray Byrne) – 15 minutes
2. Machine Learning Overview (David Stracuzzi) – 45 minutes
3. Machine Learning Approaches and Data Considerations (Warren Davis) – 25 minutes
4. Optimization with Application to Machine Learning and Power Systems (JP Watson) – 10 minutes
5. Highlights of Artificial Intelligence/Machine Learning in Power Systems (Matt Reno/Logan Blakely) – 20 minutes
6. Q&A – 5 minutes

Introduction to Artificial Intelligence and Machine Learning

Raymond H. Byrne (rhbyrne@sandia.gov)

September 9, 2019

Why is Artificial Intelligence a Hot Topic?

Examples of Successful Artificial Intelligence (AI) Applications:

Self-driving cars, lane departure detection, etc.

Credit card fraud detection

amazon alexa

Personal assistants

cogito

Conversation optimization

BOXEVER

Customer interaction optimization (travel, etc.)

Music recommendations

JOHN PAUL

Existing client interactions – luxury travel concierge

amazon

Product recommendations

NETFLIX

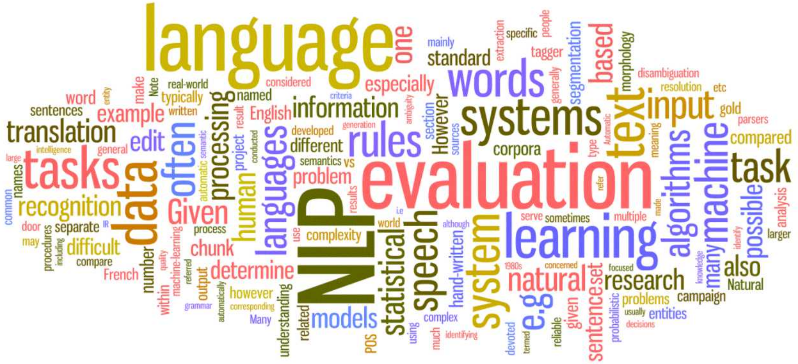
Product recommendations

nest

Thermostat control

R.L. Adams, "10 Powerful Examples Of Artificial Intelligence In Use Today," Forbes, www.forbes.com.

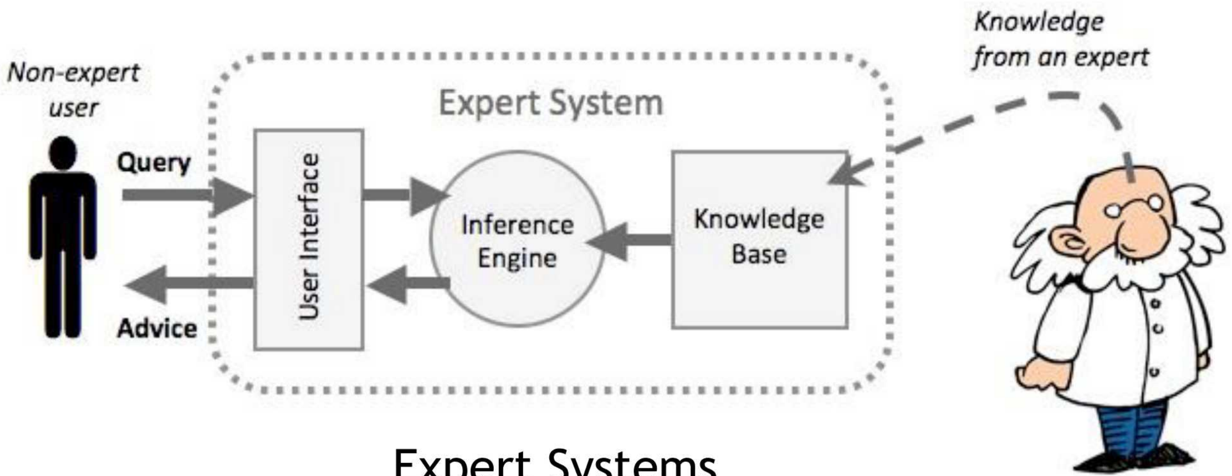
Families of AI Techniques



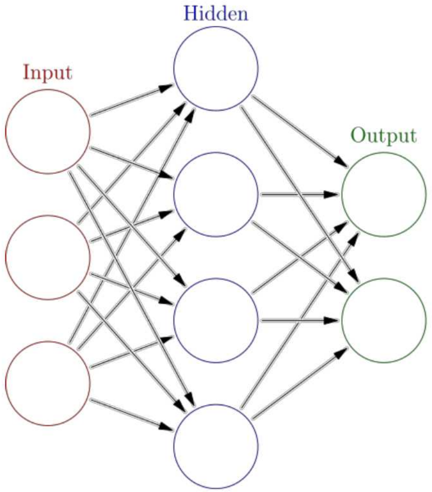
Natural Language

Robotics

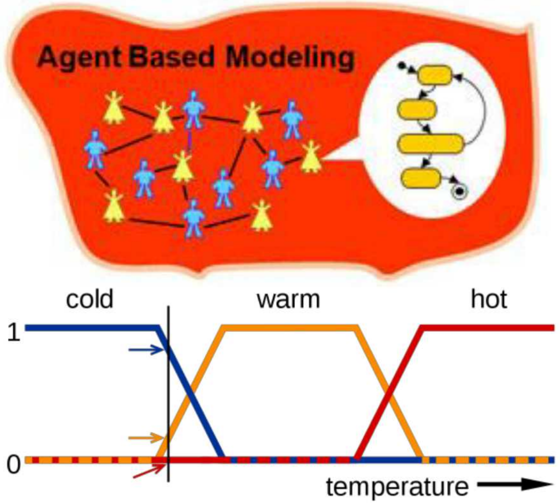
Perceptive Systems



Expert Systems



Learning Systems



Intelligent Software

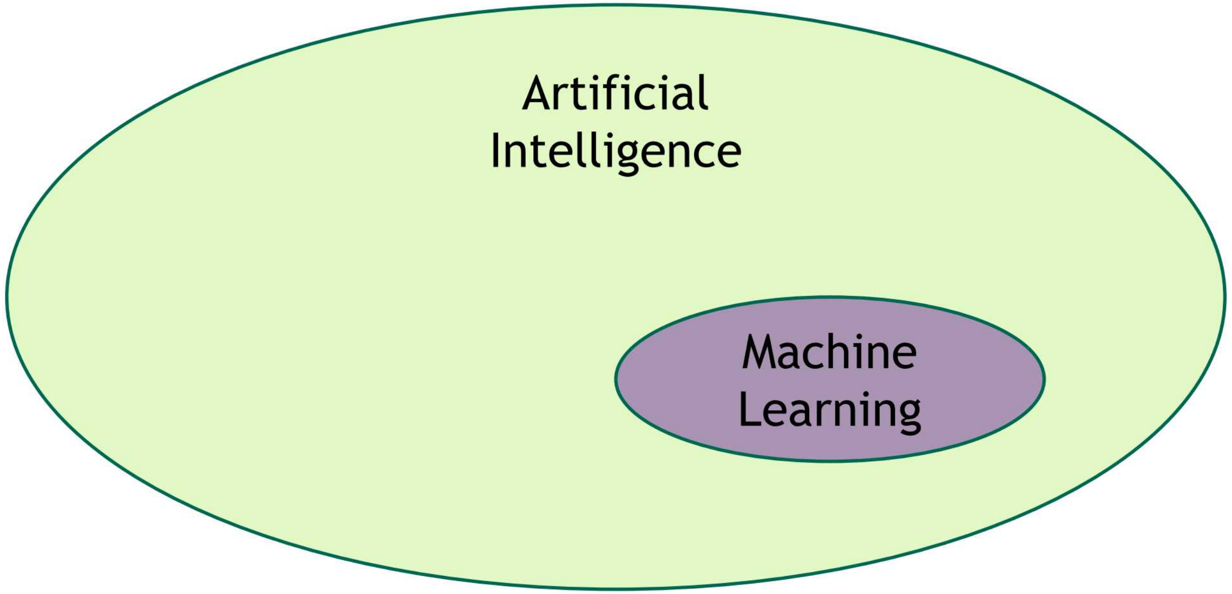
Huang Ling-fang, “Artificial Intelligence,” *2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE)*, Singapore, 2010, pp. 575-578.

AI versus Machine Learning

Machine learning is considered a subset of artificial intelligence

Artificial Intelligence: a branch of computer science which studies building machines capable of intelligent behavior

Machine Learning: a computer learns to perform a task, often without explicit instructions, by studying a training set of examples



K. Bakshi and K. Bakshi, "Considerations for artificial intelligence and machine learning: Approaches and use cases," *2018 IEEE Aerospace Conference*, Big Sky, MT, 2018, pp. 1-9.
P. Louridas and C. Ebert, "Machine Learning," in *IEEE Software*, vol. 33, no. 5, pp. 110-115, Sept.-Oct. 2016.

Machine Learning is a Subset of AI

Tables making comparisons are often incorrect … since machine learning is a subset of AI, every machine learning approach has some application to AI

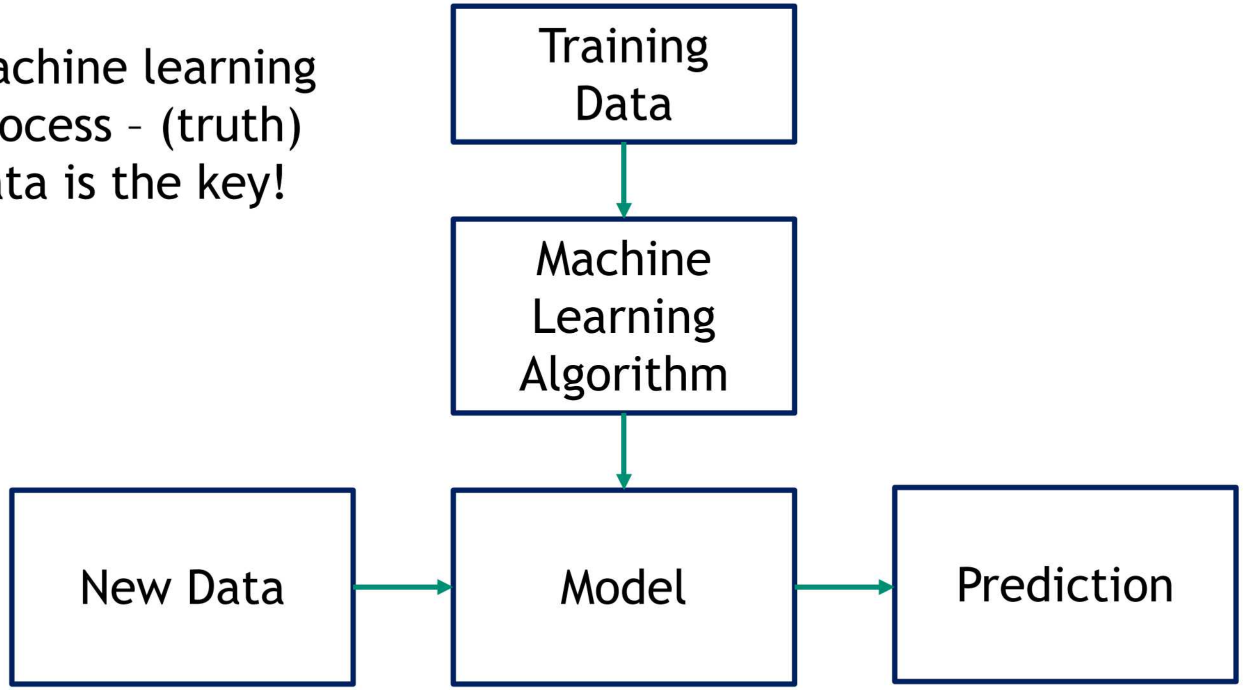
Machine Learning Example	AI application
Image segmentation and classification to visually identify manufacturing flaws	Image segmentation and classification is a key component in AI applications (e.g., humanoid robots, etc.)
Linear regression to predict future samples of a time series (e.g., GDP growth)	Numerous AI applications related to prediction (e.g., motion of images in a scene for autonomous navigation and obstacle avoidance)
Natural language (text and voice) processing for translation (e.g., Google translate) of business documents	Natural language processing is required for any AI application that involves language
Pattern recognition applied to credit card fraud detection	Pattern recognition applied to autonomous grasping (e.g., pick up the ball not like the others)
Product recommendations to improve customer experience and boost online sales	Product recommendations provided by an AI assistant

In addition, there are many fields of AI that are not application specific and machine learning is not the primary focus or methodology … examples include research on planning and cognitive architectures

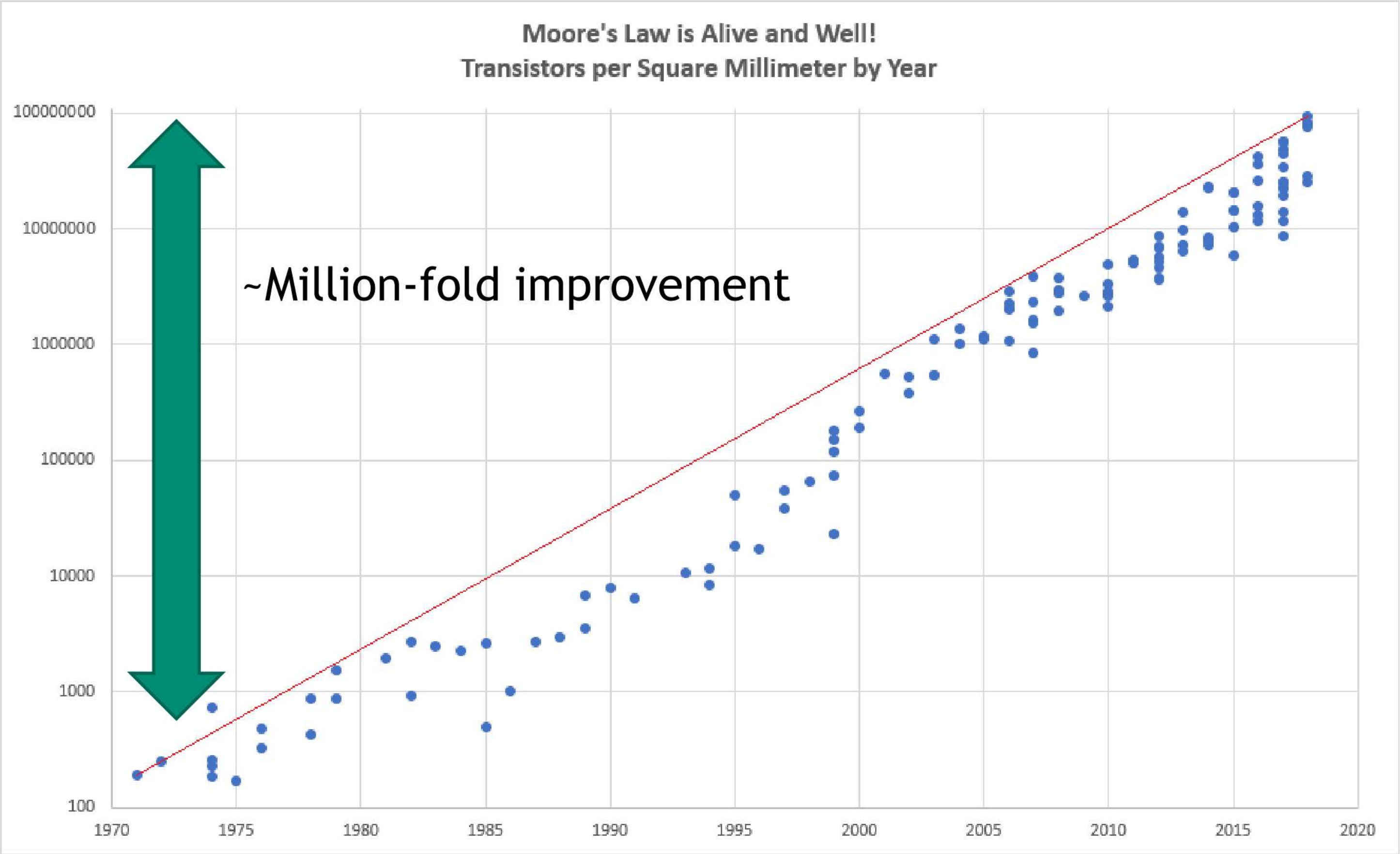
Machine Learning

While there are many machine learning techniques, the basic process flow is the same for all approaches

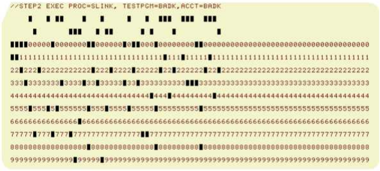
Machine learning process – (truth) data is the key!



Enabling Technological Advances



Moore's Law: the number of transistors on an IC would double every few years.



~80 bytes/punch card
133 bytes/sec

1951 – 7200 bytes/sec

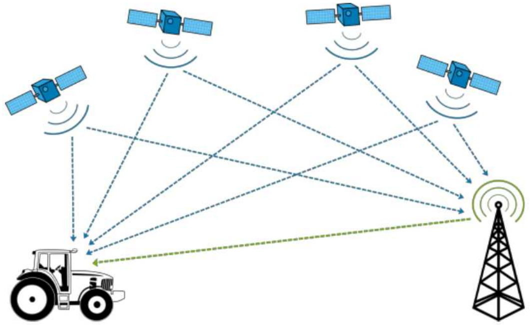
1980 – 5MB, 0.625MB/sec
1990 – 400MB, 0.7MB/sec
2008 – 750GB, 64MB/sec

today – solid state drive
4TB, 500MB/sec

E. P. DeBenedictis, "It's Time to Redefine Moore's Law Again," in *Computer*, vol. 50, no. 2, pp. 72-75, Feb. 2017.

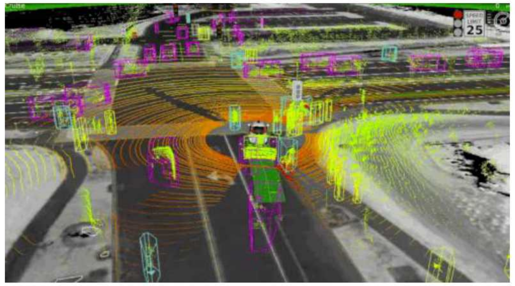
Enabling Technological Advances (continued)

Low cost, high performance sensors, platforms



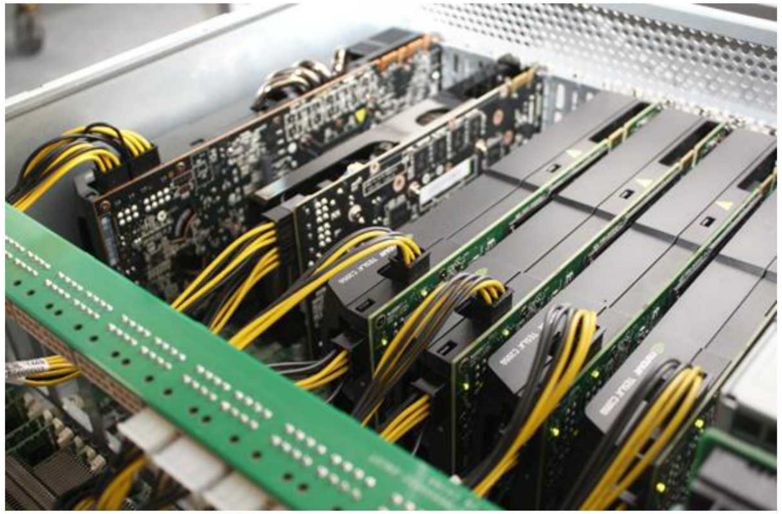
GPS

Image sensors

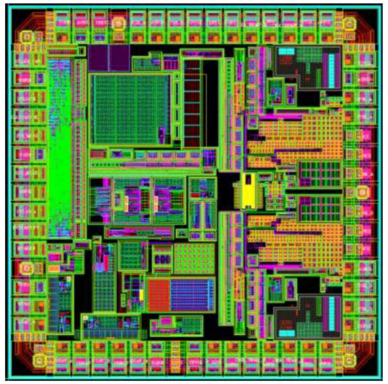


LIDAR (Light Detection and Ranging)

UAVs



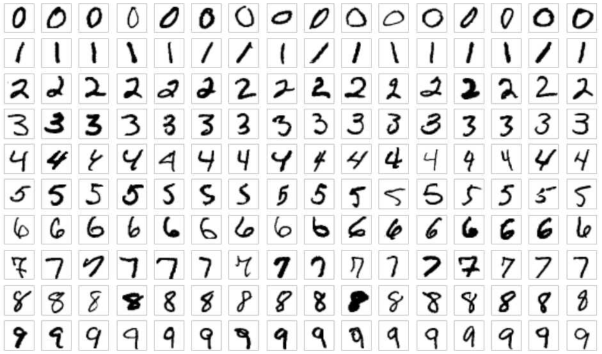
Graphical Processing Units (GPUs)



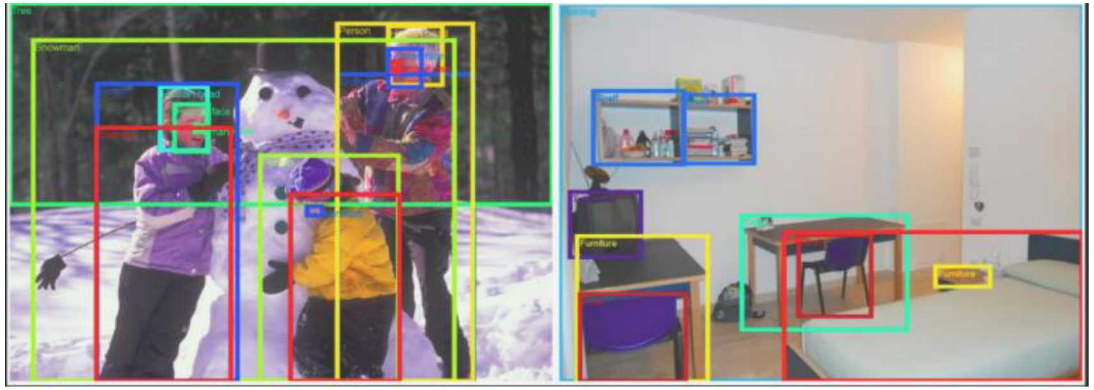
Application Specific Integrated Circuits (ASICs)

Enabling Technological Advances (continued)

Truth data for training/validation – there are a large number of datasets available for image processing, natural language processing, and audio/speech processing

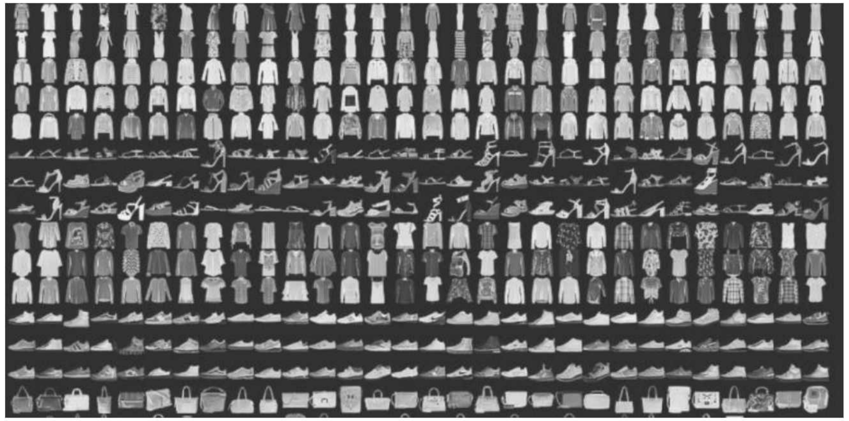


MNIST – 70,000 handwritten digits



Open Images Dataset - 9 Million annotated images

The Street View House Numbers (SVHN), 600,000 images



Fashion-MNIST, 70,000 images

“25 Open Datasets for Deep Learning Every Data Scientist Must Work With”, https://www.analyticsvidhya.com/blog/2018/03/comprehensive-collection-deep-learning-datasets/

A Brief History of AI – The Turing Test

Proposed by Alan Touring in 1950

Three players

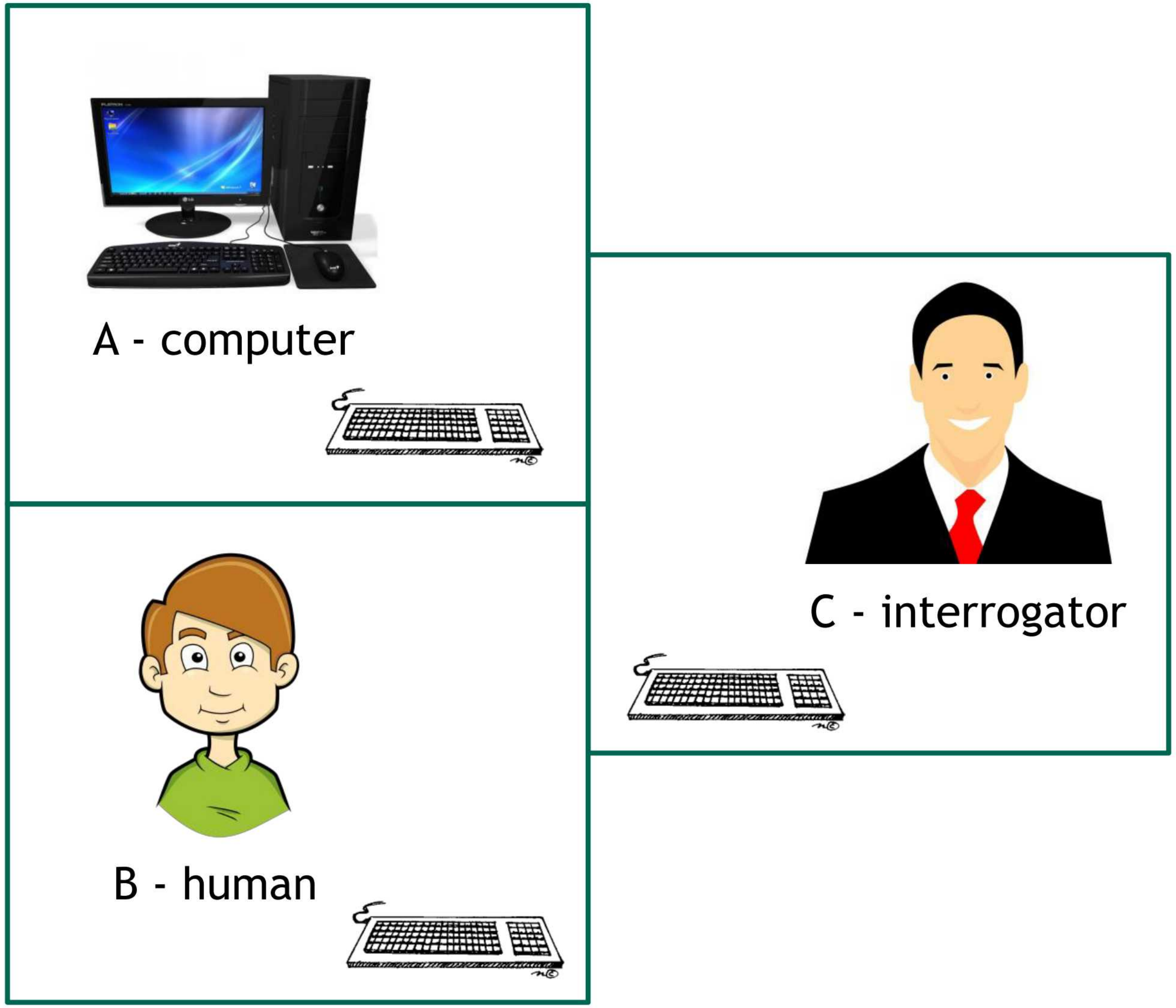
- A – computing machine
- B – human being
- C – interrogator

All communication is through a textual device (e.g., keyboard)

Can the interrogator identify the human and computer?

Turing predicted that a computer could convince ~33% of the judges after 5 minutes of questioning by the year 2000

June 2014, A chatbot called Eugene Goostman, which simulates a 13-year-old Ukrainian boy, convinced 30% of the judges



S. Guccione and G. Tamburrini, "Turing's Test Revisited," *Proceedings of the 1988 IEEE International Conference on Systems, Man, and Cybernetics*, Beijing, China, 1988, pp. 38-41.

A Brief History of AI – the Dartmouth Workshop

The term "artificial intelligence" was first coined by John McCarthy, Marvin Minsky, Nathaniel Rochester, and Claude Shannon in 1956

They proposed a summer workshop on artificial intelligence at Dartmouth College

Topics included:

- ***Automatic Computers*** – "If a machine can do a job, then an automatic calculator can be programmed to simulate the machine. The speeds and memory capacities of present computers may be sufficient to simulate many of the higher functions of the human brain, but the major obstacle is not the lack of machine capacity, but our inability to write programs taking full advantage of what we have."
- ***How Can a Computer be Programmed to Use a Language*** – "It may be speculated that a large part of human thought consists of manipulating words according to rules of reasoning and rules of conjecture. From this point of view, forming a generalization consists of admitting a new word and some rules whereby sentences containing it imply and are implied by others. This idea has never been very precisely formulated nor have examples been worked out."
- ***Neuron Nets*** – "How can a set of (hypothetical) neurons be arranged so as to form concepts. Considerable theoretical and experimental work has been done on this problem …"
- **Theory of the Size of Calculation** – you have to understand the size of the calculation to measure the efficiency of an algorithm
- **Self Improvement** – a truly intelligent machine will carry out self-improvement
- **Abstractions** – machine methods of forming abstractions from sensory and other data
- **Randomness and Creativity** – conjectured that creative thinking involves some randomness

J. McCarthy, M.L. Minsky, Nathaniel Rochester, and C.E. Shannon, "A proposal for the Dartmouth summer research project on artificial intelligence", submitted to the Rockefeller Foundation, August 31, 1955.

A Brief History of AI - Timeline

AI Winter: period of significantly reduced research funding. One cause was outlandish claims that were impossible to meet.

1940 | 1950 | 1960 | 1970 | 1980 | 1990 | 2000 | 2010 | 2020

- 1950 Alan Turing proposes "Turing Test"
- 1956 Dartmouth Workshop
- 1958 LISP developed
- 1965 First Expert System
- 1974-80 AI Winter Funding Cuts
- 1981 First Commercial Expert System
- 1987-93 AI Winter Funding Cuts
- 1997 IBM's Deep Blue defeats Garry Kasparov
- ~2006 AI for image processing takes off
- 2008 Google's speech recognition app on iPhone
- 2011 Siri released with iPhone 4S
- 2011 IBM's Watson beats two legendary humans at Jeopardy
- 2014 Chatbot Eugene Goostman passes "Turing Test"
- 2015 Google DeepMind's AlphaGo beats human champion

SIRI was spun out of the DARPA funded CALO (Cognitive Assistant that Learns and Organizes) project, 2003-2008.

Research in Machine Learning Applied to Energy Systems

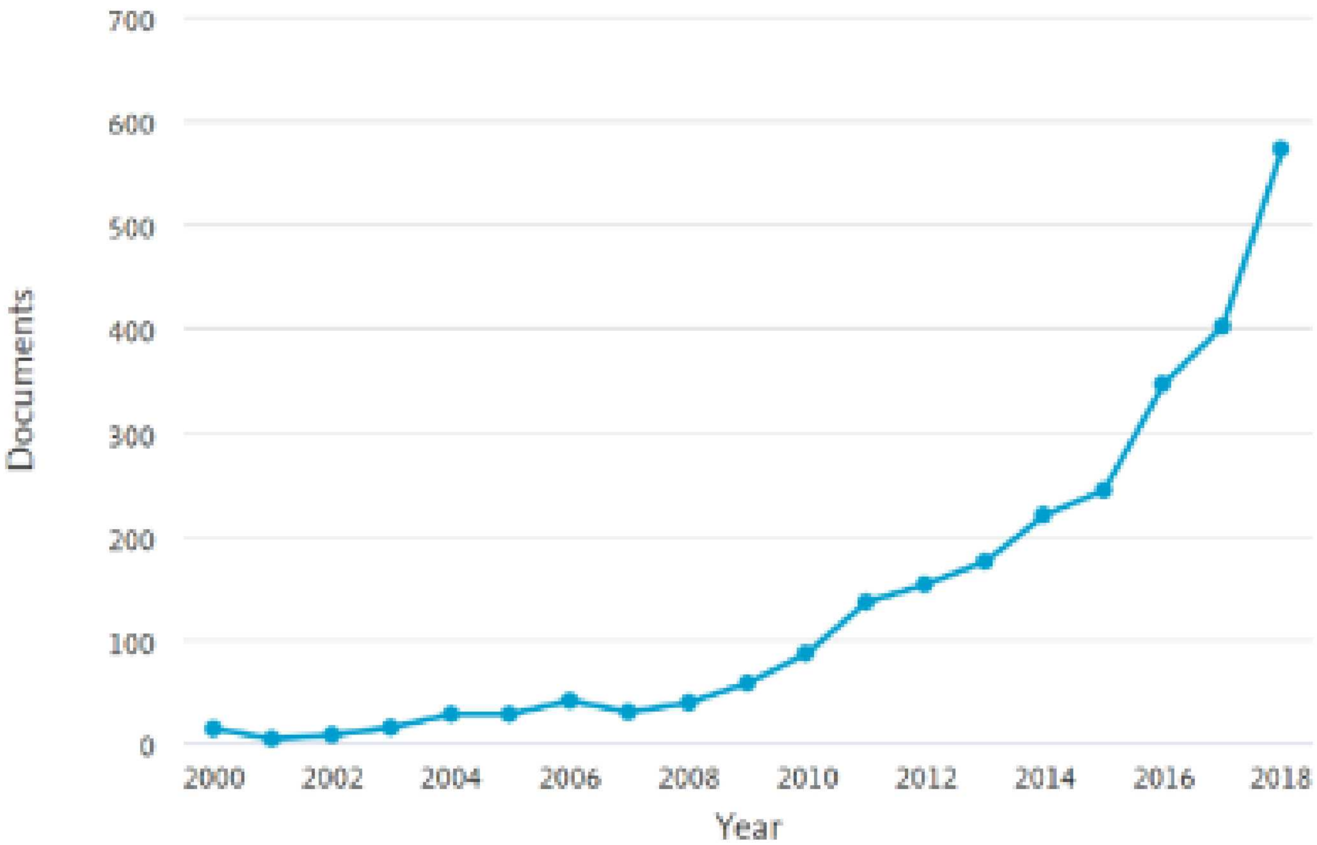


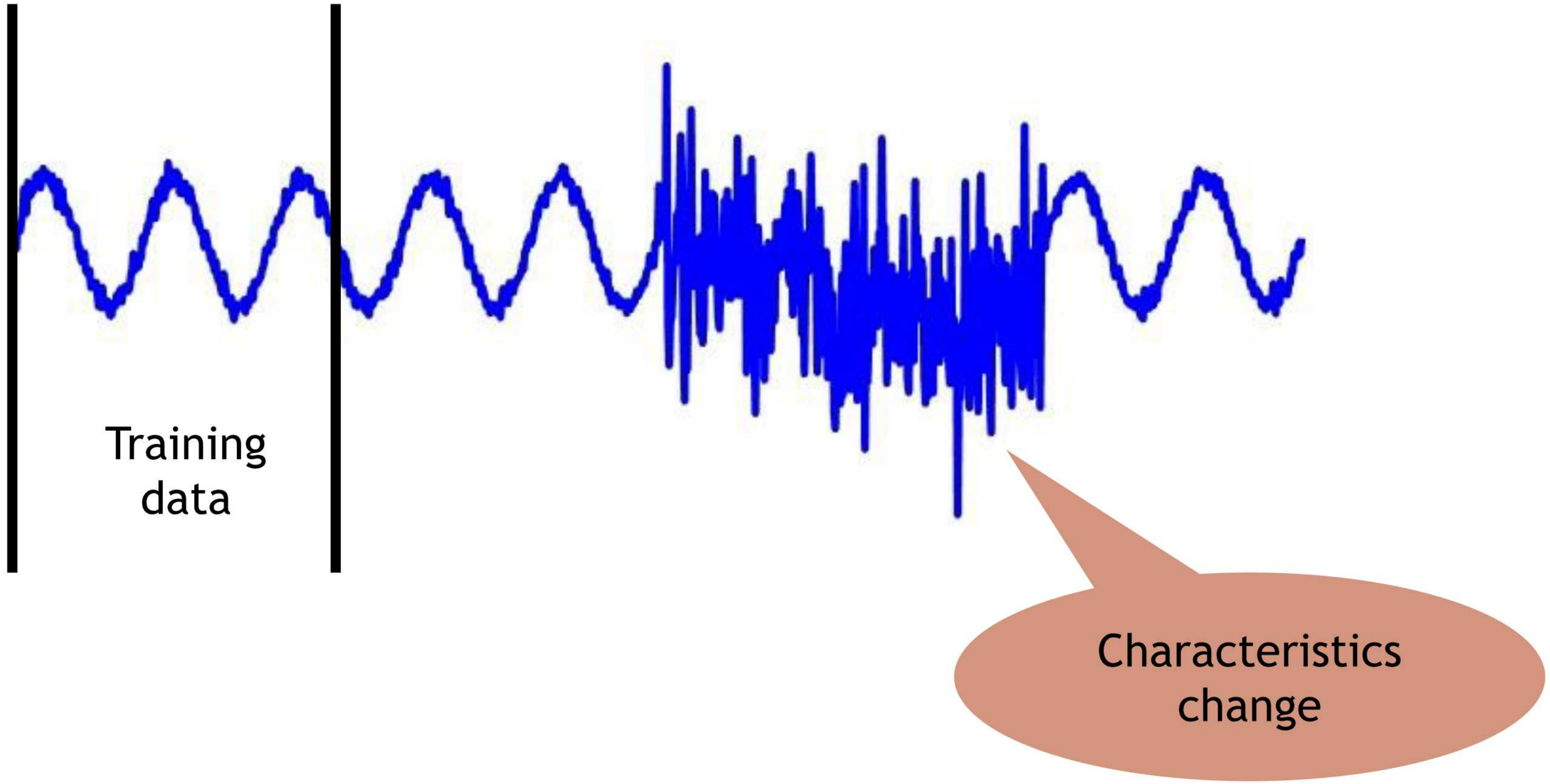
Figure 2. The growth in the number of articles during the past two decades.

A. Mosavi, M. Salimi, S. F. Ardabili, T. Rabczuk, S. Shamshirband, and A. R. Varkonyi-Koczy, “State of the Art of Machine Learning Models in Energy Systems, a Systematic Review,” *Energies*, vol. 12, no. 7, Apr. 2019.

Limitations of Machine Learning

Performance of a ML algorithm can be very good if the characteristics of the training data match the observed data

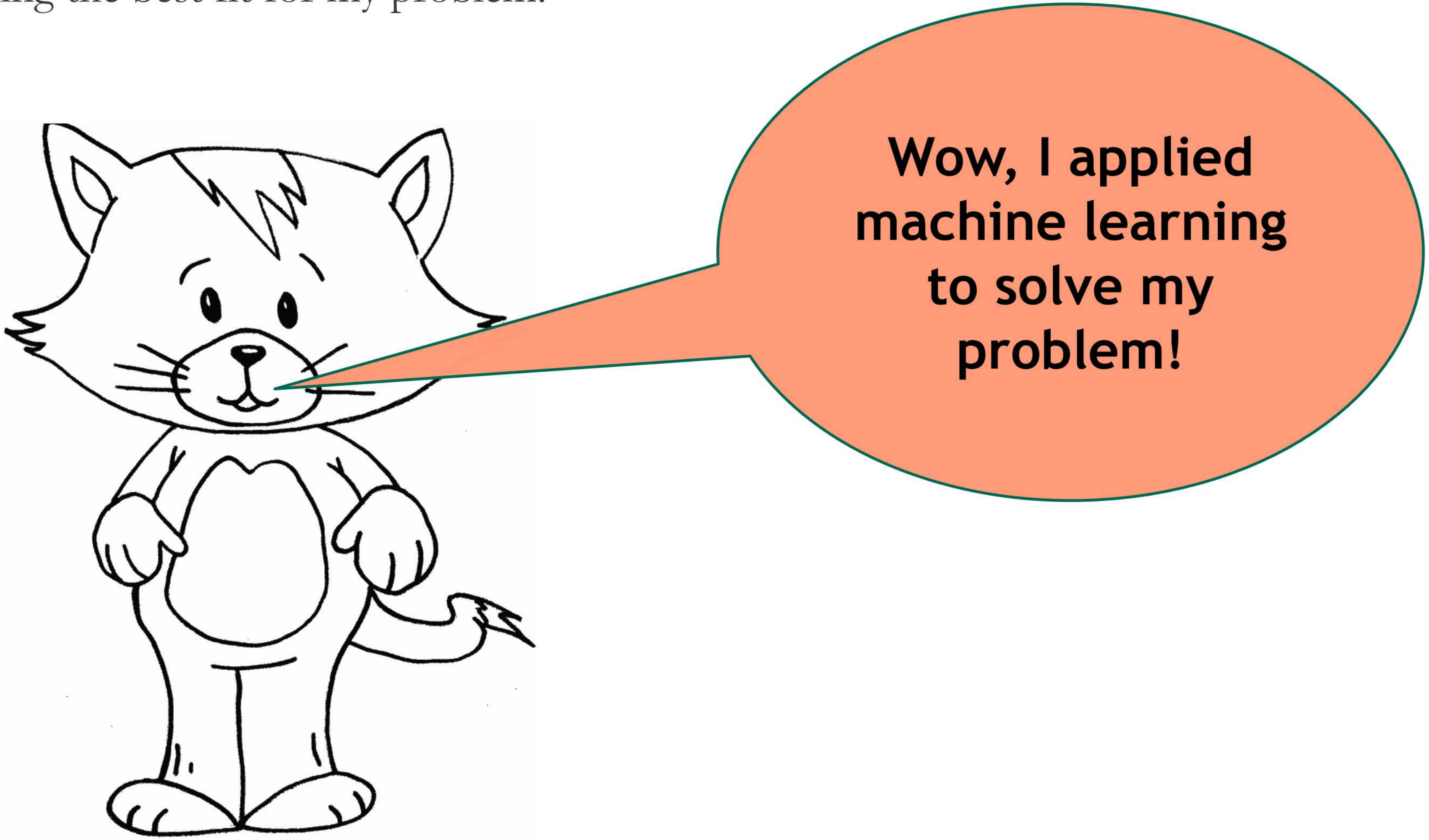
If the characteristics of the data change over time, and this is not captured in the training data, the performance of the ML algorithm can vary widely



Limitations of Machine Learning

For some problems, there is a known non-machine learning solution that is efficient, elegant, and robust

Is machine learning the best fit for my problem?



Course Outline

1. Introduction to Artificial Intelligence and Machine Learning (Ray Byrne) – 15 minutes
2. Machine Learning Overview (David Stracuzzi) – 45 minutes
3. Machine Learning Approaches and Data Considerations (Warren Davis) – 25 minutes
4. Optimization with Application to Machine Learning and Power Systems (JP Watson) – 10 minutes
5. Highlights of Artificial Intelligence/Machine Learning in Power Systems (Matt Reno/Logan Blakely) – 20 minutes
6. Q&A – 5 minutes

Machine Learning Overview

David J. Stracuzzi (djstrac@sandia.gov)

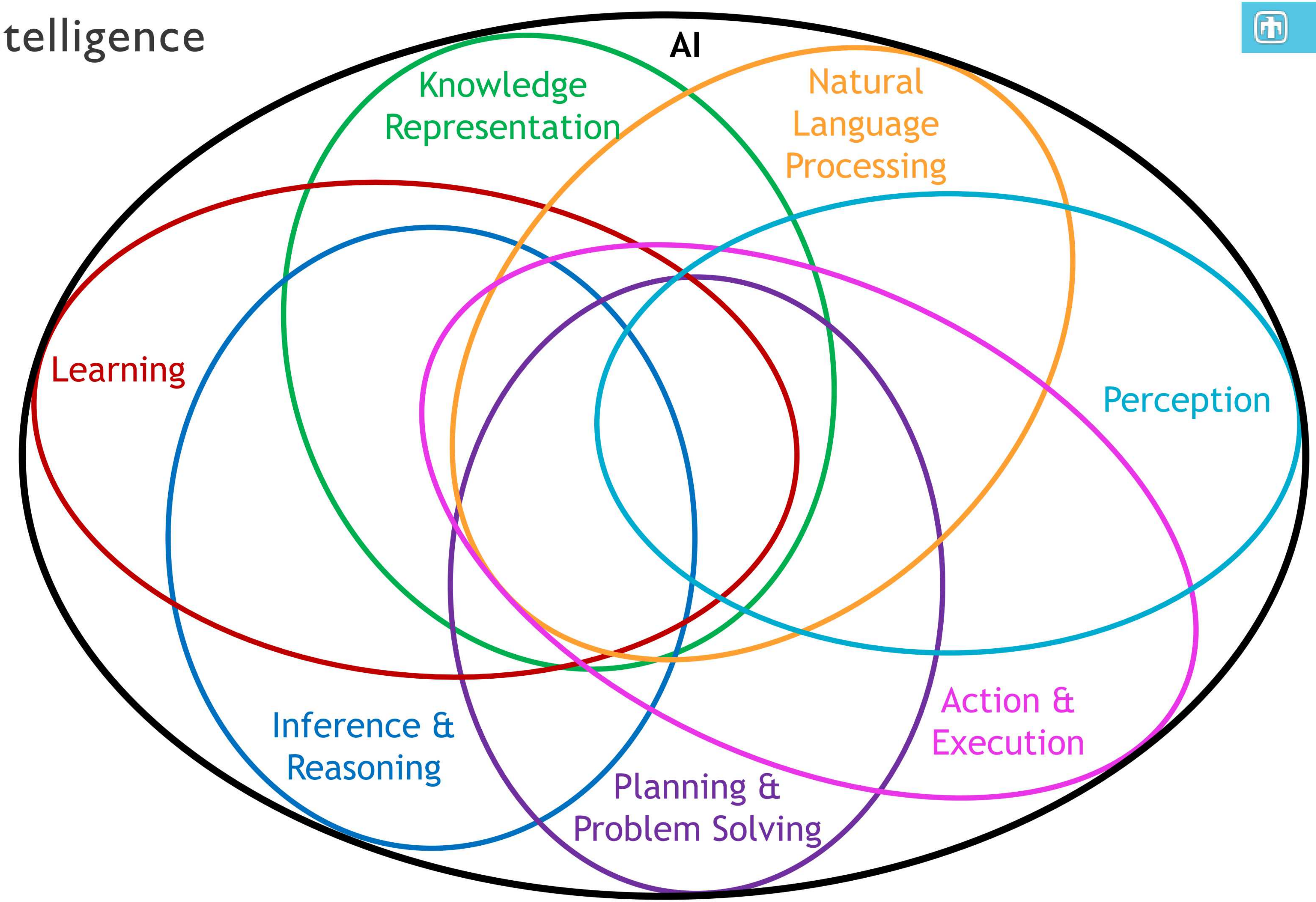
September 9, 2019

Artificial Intelligence

"The automation of activities that we associate with human thinking …"
(Bellman, 1978)

"The art of creating machines that perform functions that require intelligence when performed by people."
(Kurzweil, 1990)

"The study of the computations that make it possible to perceive, reason, and act."
(Winston, 1992)



What is Machine Learning?

Machine Learning coined in 1959 by Arthur Samuel while trying to use data to improve performance of a checkers playing program.

IBM, 1956

Samuel, A.L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*.

What is Machine Learning?

A computer program is said to ***learn*** *from*

experience E with respect to some class of

tasks T and performance measure P

if its performance at tasks in T, as measured by P,

improves with experience E.

– Tom Mitchell, *Machine Learning*, 1997

Many Types of Tasks and Methods

Tasks:

- Supervised vs Unsupervised
- Classification
- Clustering
- Regression
- Anomaly Detection
- Time Series Analysis
- Policy Learning
- Transfer Learning

Methods:

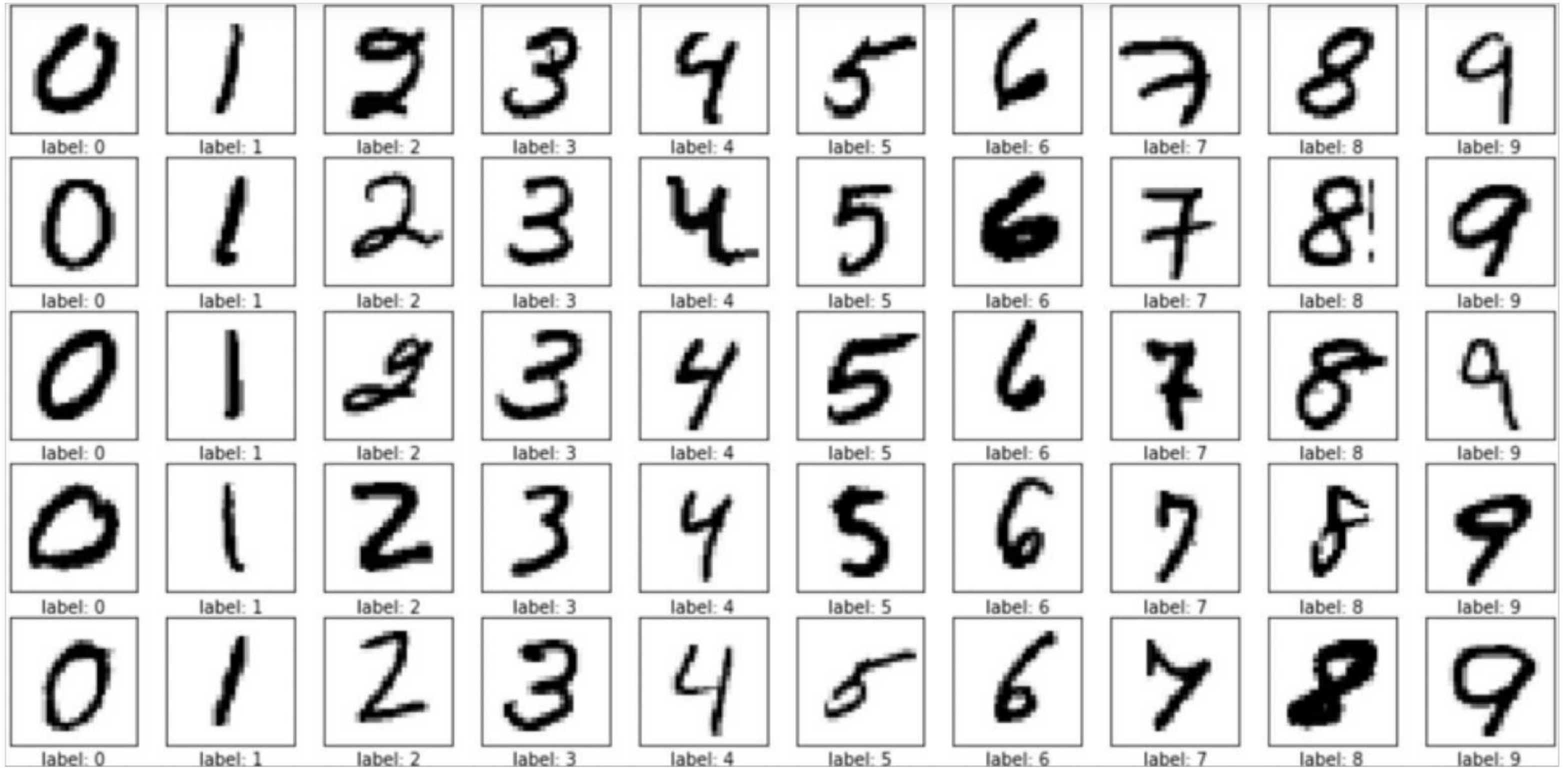
- Decision Trees
- Rule-Based Methods
- Neural Networks
- Inductive Logic
- Support Vector Machines
- Bayesian Methods
- Genetic Algorithms
- Statistical Algorithms
- Ensembles

Example Problem: Handwriting Recognition

Task (T): Recognizing and classifying handwritten numbers within images

Performance measure (P): Percent of numbers correctly classified

Experience (E): Database of handwritten numbers with given classifications



Example adapted from Tom Mitchell, *Machine Learning*, 1997

Data from MNIST database, http://yann.lecun.com/exdb/mnist/

How Does Machine Learning Work?

Data / Experience

$X = \{x_1, x_2, \ldots, x_n\}$		Y

Model

$$Y=f(x)$$

Loss Function

$$\varepsilon = \frac{1}{N}\sum_{i=0}^{n} g(f(x_i) - Y_i)$$

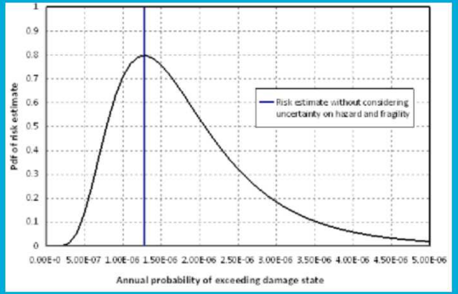
Learning Algorithm

```
if (settings[0].compareTo("s")==0)  {
    if (name.compareTo("") !=0) {
        name += "_";
    }
    name+- etr.getString(settings[1]);
} else if (setting [0].compareTo("d") == 0){
    if (name.compareTo("") !=0) {
        name += "_";
    }
    name += DateUtils.format(etr.getDate(settings[1])
} else if (setting [0].compareTo("d") == 0){
    if (name.compareTo("") !=0) {
        name += "_";
    }
```

Parameterized Model

$$f(x) = \theta_1 x_1 + \theta_2 x_2 + \cdots$$

Predictions & Evaluation

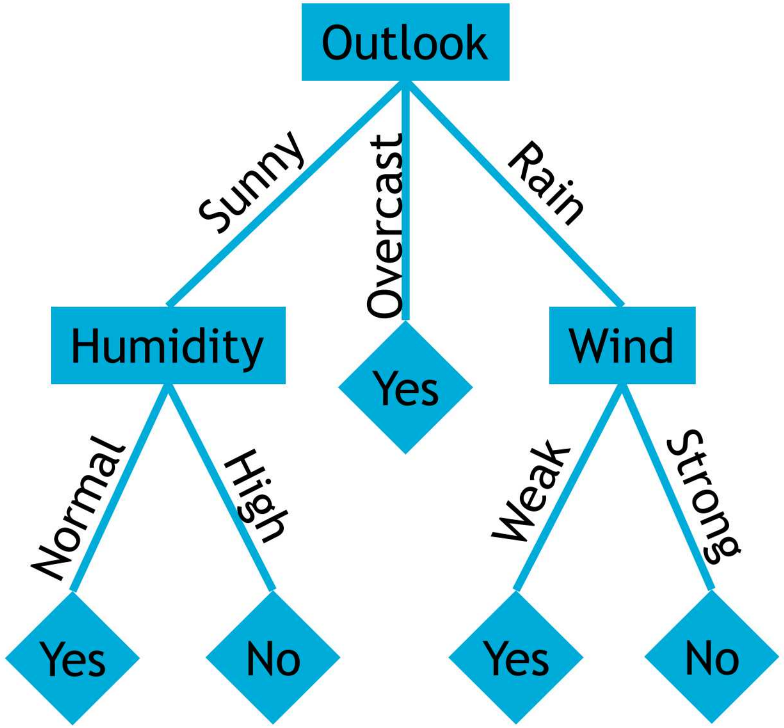


Learning Example : Decision Trees

Task: Determine if Bill will play tennis given weather observations

Performance Metric: Prediction accuracy

Experience: Past observations



Day	Outlook	Temperature	Humidity	Wind	Play Tennis?
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Example from Mitchell, T.M. (1997). *Machine Learning*.

Learning Example: Data Preprocessing and Feature Engineering

- Many learning algorithms take a set or sequence of ***vectors*** as input
  - Raw data needs to be encoded in this format
  - For many data types, there are existing encoding conventions
- ***Feature engineering*** uses domain knowledge to create these encodings
  - Highly manual and time consuming
  - Quality of learned model often dependent on feature encodings

Example: Play Tennis?

Outlook: {sunny, overcast, rain} or
{sunny, partly cloudy, mostly cloudy, cloudy, drizzle, rain, downpour} or
RGB image from TennisCam

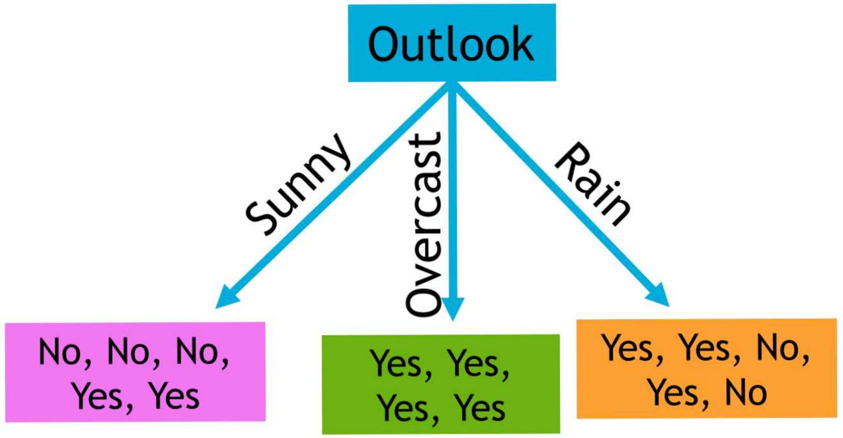
Temperature: {hot, mild, cool} or
{hot, warm, mild, cool, cold} or
{-20F, -19F, … , 114F, 115F} or
continuous

Learning Example: Decision Trees

General Approach:

- Split the data based on information theory (entropy)
- Entropy measures the distribution of positive and negative examples in each block
- Greedy search through attribute (feature) space

Gain = Entropy all data - Sum of Entropies after split



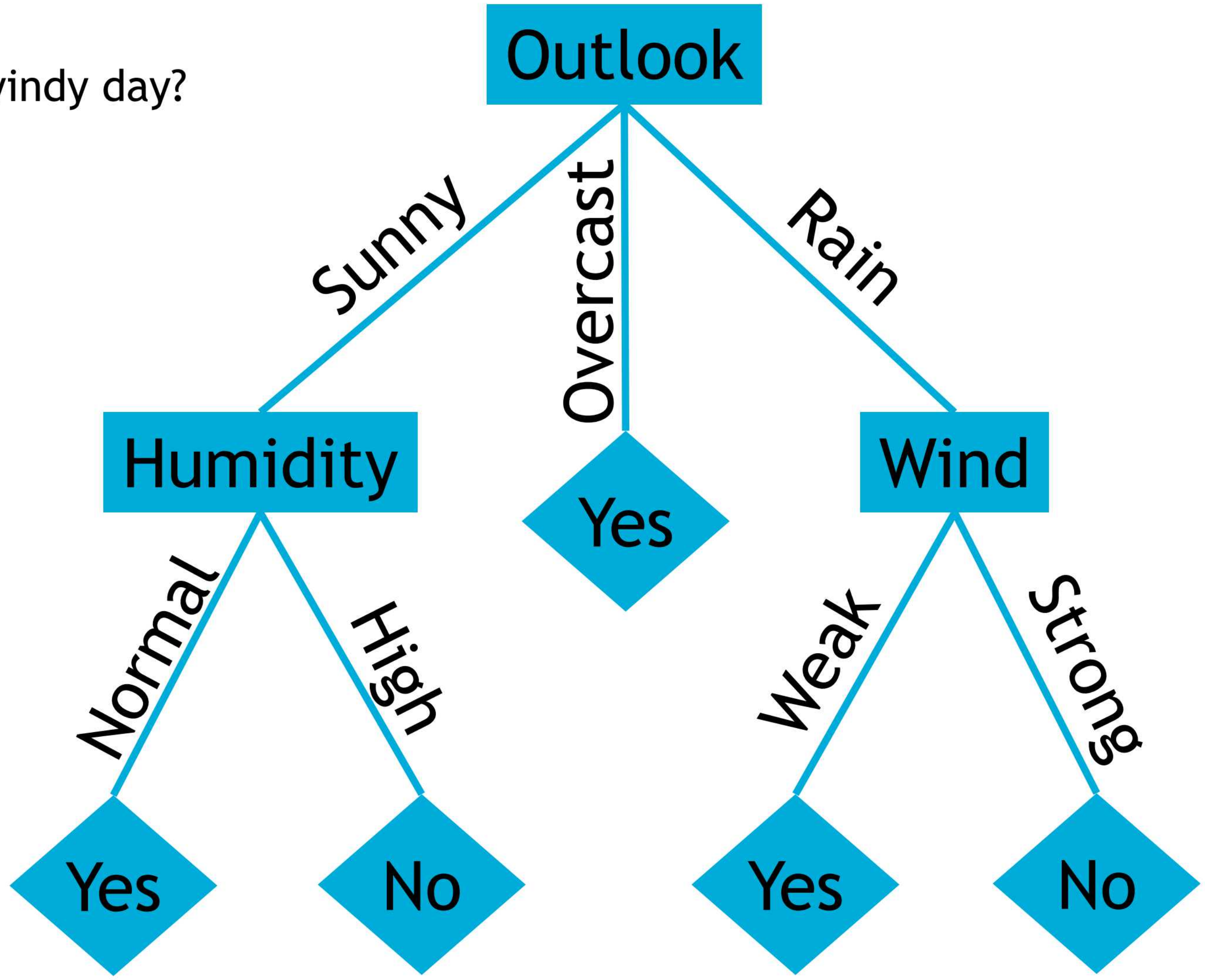
Day	Outlook	Temperature	Humidity	Wind	Play Tennis?
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No
	G=0.247	G=0.029	G=0.152	G=0.048	

Learning Example: Decision Trees

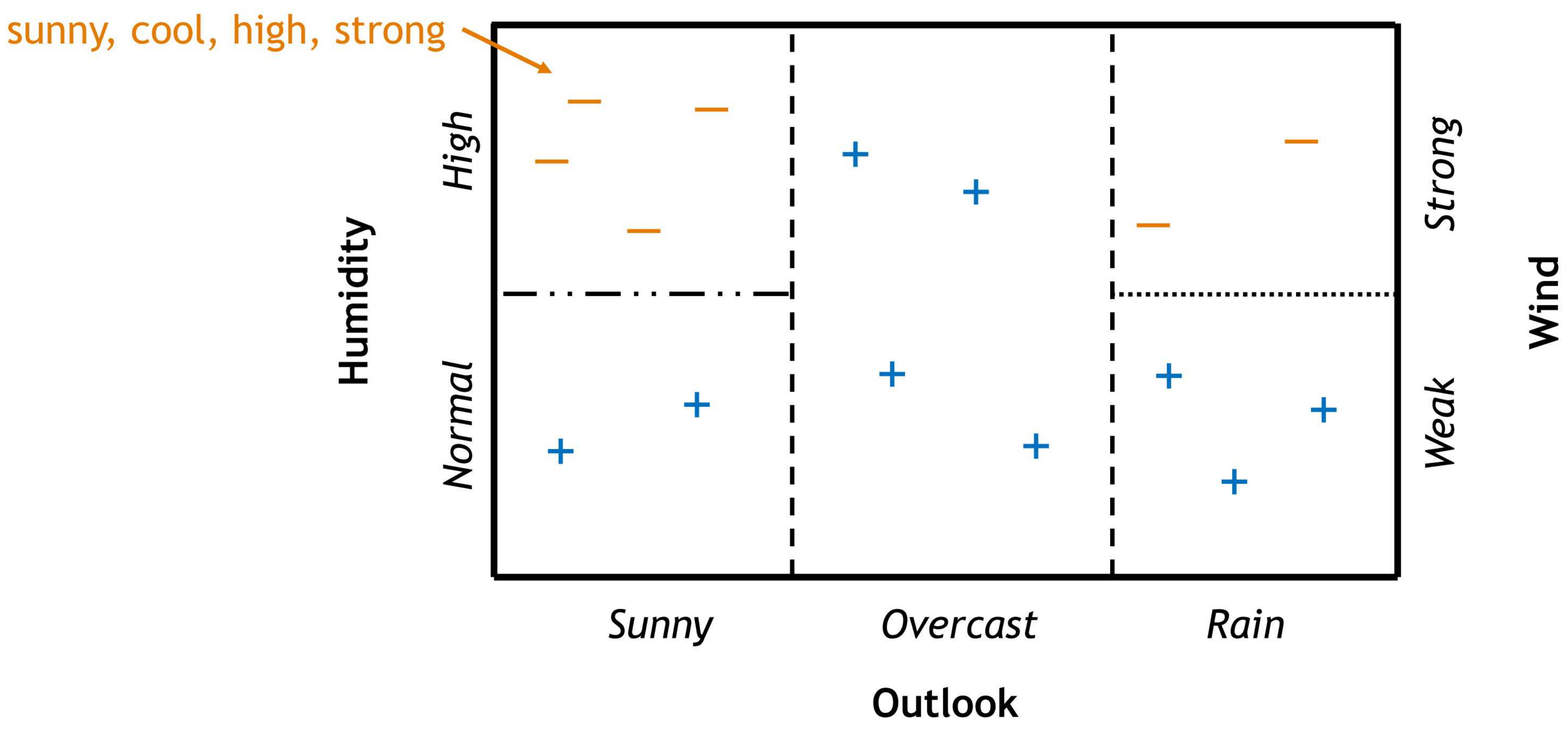
(Day 15) What will happen on a sunny, cool, humid, windy day?

Many design decisions affect performance:

- Training data (number and quality of examples)
- Which variables describe the data
- Splitting criterion
- Binary versus multivariate splits
- What to do with numeric variables
- Stopping criterion



Decision Tree Hypothesis Space



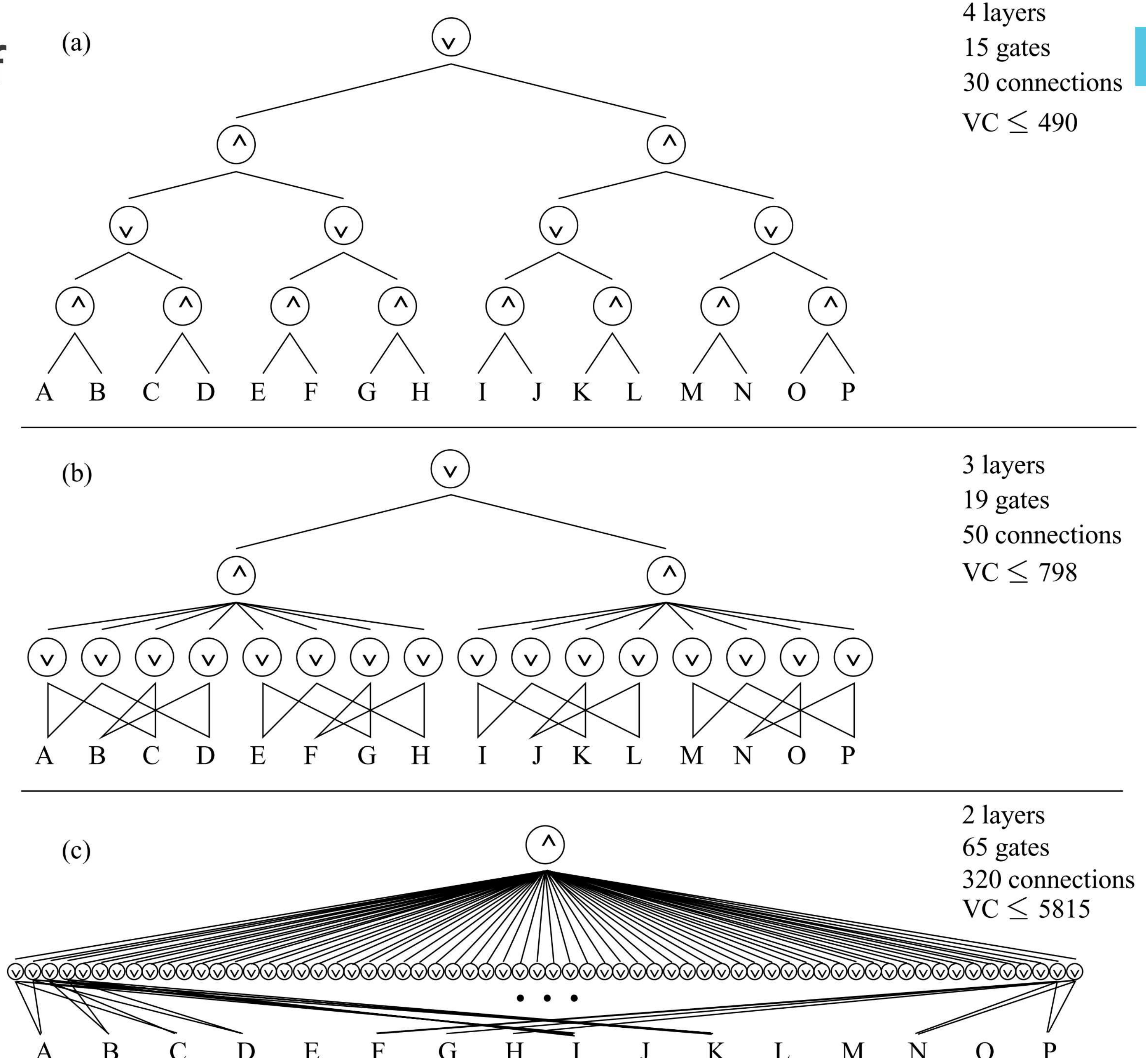
Note: Original data was in 5 dimensions. Only showing 3 here compressed into 2.

Bias-Variance Trade-Off

Always choose the simplest model that can fit the data.

- Circuits (a), (b), (c) represent same logical function
- Can view gates and connections as learnable parameters
- All things equal, (a) is a much easier learning problem and most likely to generalize well.

Many theoretical constructs attempt to explicitly manipulate this trade, yet it remains a vexing problem.



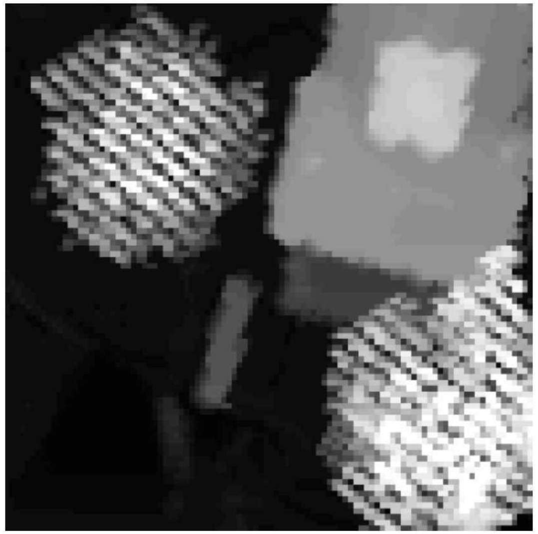
Learning Example: Image Analysis

Task: Classify pixels as tree, grass, roof, water, concrete, or boat

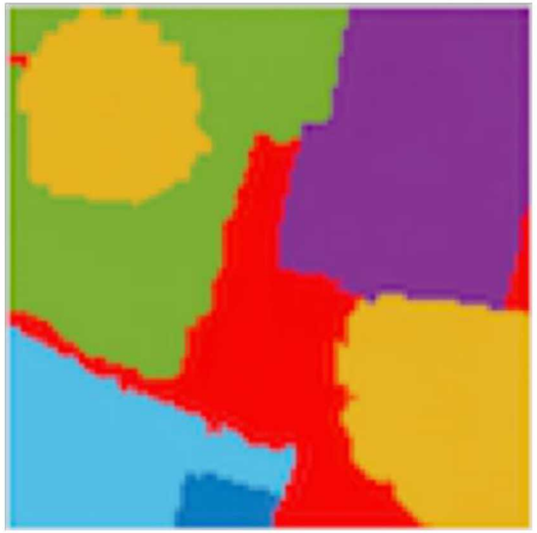
Performance Metric: Accuracy

Experience: Labeled pixels

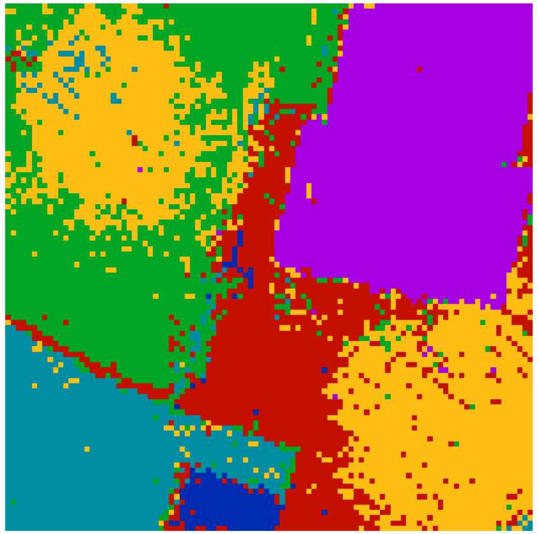
RGB Color



Height



Labels



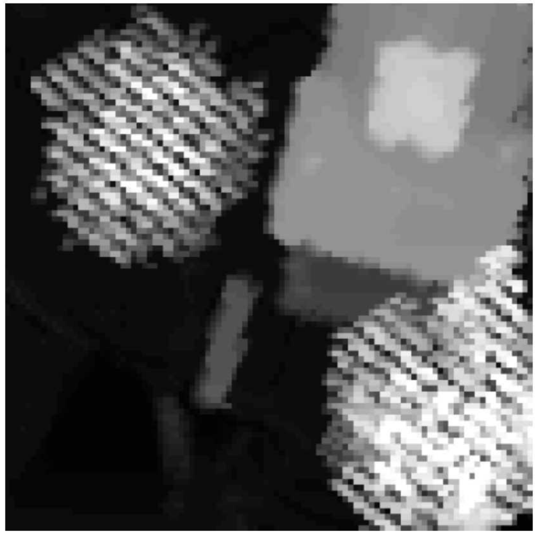
Predictions

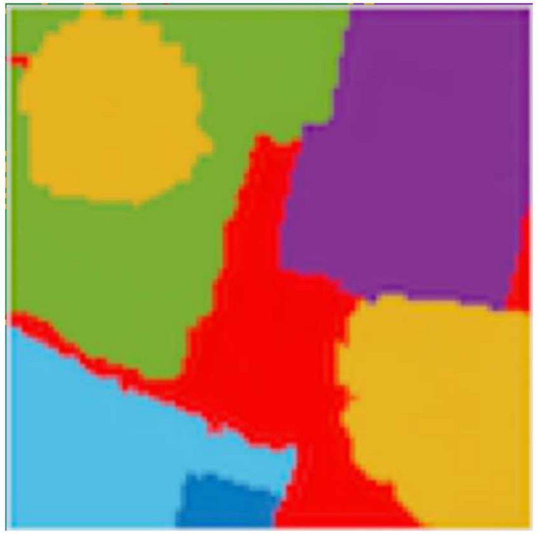
Learning Example: Image Analysis

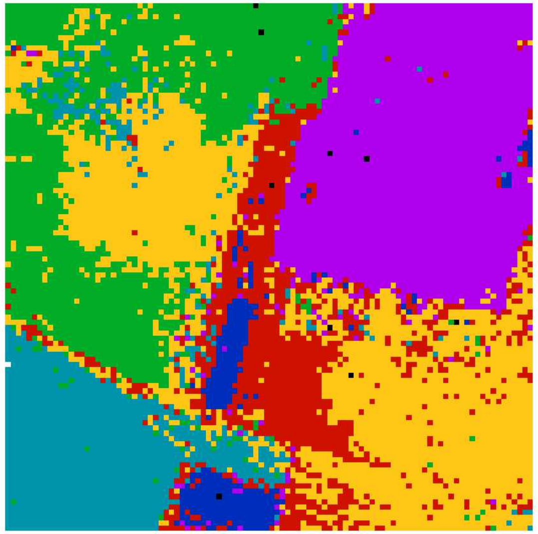
As we develop an application, we need to ask:

- How else might we formulate the problem?
- What input variables might provide the most information?
- How good are my labels?
- What is the most appropriate knowledge representation?
- What is the most appropriate performance metric?
- Given the task and the data, what learning algorithms are likely to perform well?

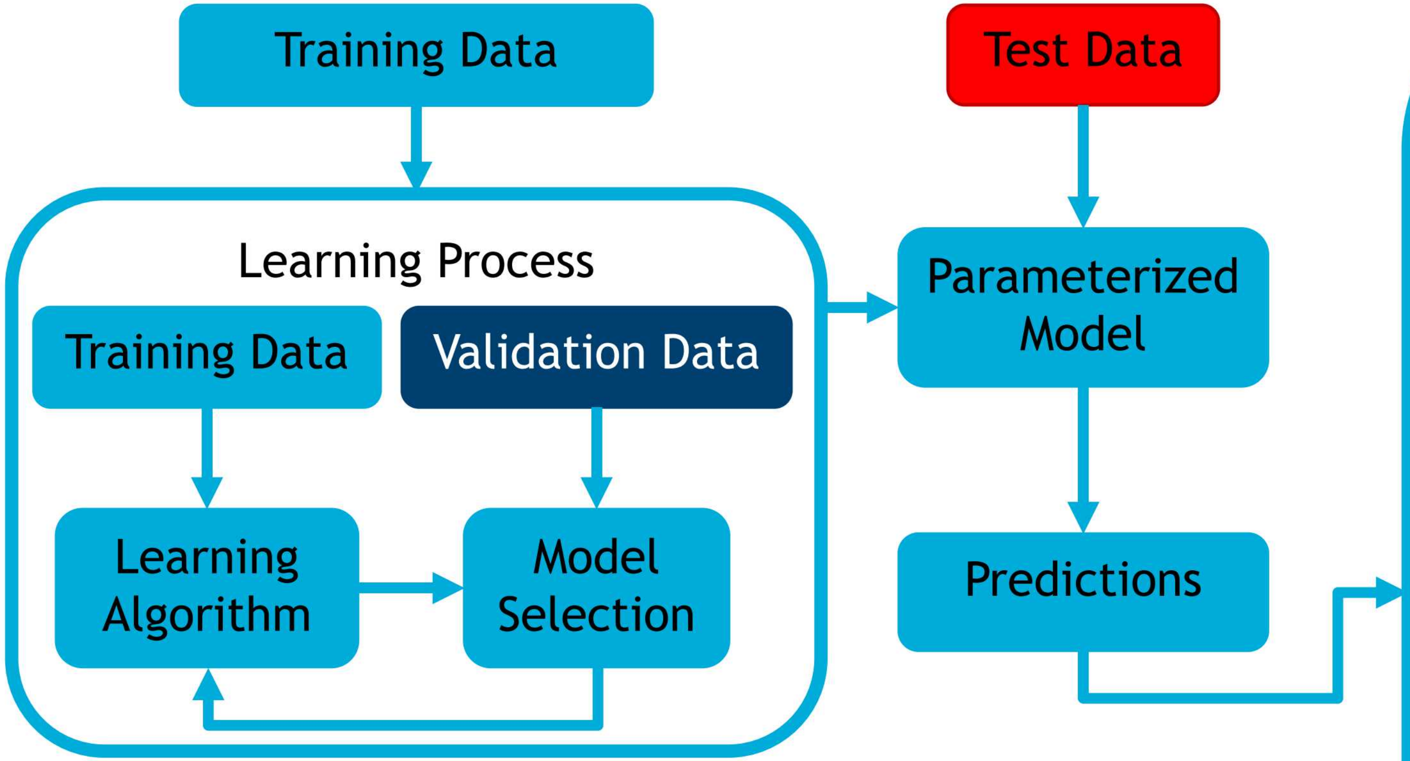

RGB Color


Height


Labels


Predictions with Color Only (78%)

Evaluation



What makes evaluation hard?

- Many ways to formulate error and performance metrics
- Highly dependent on the data, task, and goals
- Extrapolation ability is difficult to evaluate
- Hard to determine if/when we are extrapolating

Performance (Loss) Metrics

- Accuracy = (TP + TN)/n
- Precision = TP / (TP+FP)
- Recall (Sensitivity) = TP / (TP+FN)
- F-score = (P*R) / (P+R)
- Confusion Matrices
- Log Loss = $\frac{-1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij} \times \log(p_{ij})$
- ROC Curves: calibrate classification thresholds
- P-R Curves: similar to ROC; lots of negatives
- Regression metrics:
  - Root Mean Squared Error
  - Mean Absolute Error
  - R^2 – variance explanation

All of these can be applied with cross validation, random resampling, and stratification

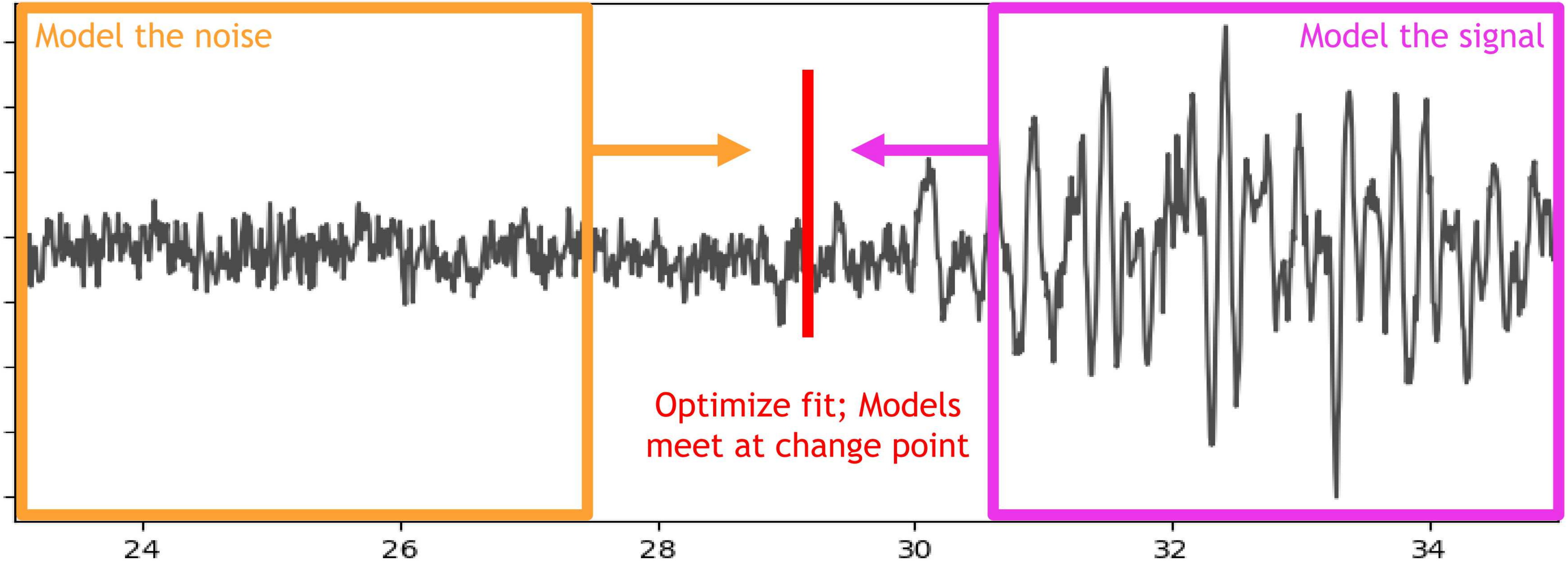
Learning Example: Time Series Application

Task: Change detection

Want to know, as precisely as possible, when the signal first arrived

Performance Metric: *No Ground Truth!!* Internal distance metrics only

Experience: Waveform data, containing both signal and noise



Uncertainty

Measurement Errors

Regularization Effects

Data / Experience

$X = \{x_1, x_2, \ldots, x_n\}$		Y

Model

Y=f(x)

Loss Function

$$\varepsilon = \frac{1}{N}\sum_{i=0}^{n} g(f(x_i) - Y_i)$$

Learning Algorithm

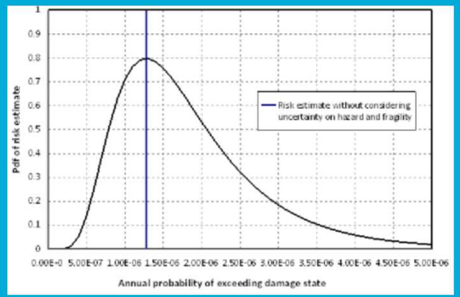
```
if (settings[0].compareTo("s")==0) {
    if (name.compareTo("") !=0) {
        name += "_";
    }
    name+- etr.getString(settings[1]);
} else if (setting [0].compareTo("d") == 0){
    if (name.compareTo("") !=0) {
        name += "_";
    }
    name += DateUtils.format(etr.getDate(settings[1])
} else if (setting [0].compareTo("d") == 0){
    if (name.compareTo("") !=0) {
        name += "_";
    }
```

Parameterized Model

$$f(x) = \theta_1 x_1 + \theta_2 x_2 + \cdots$$

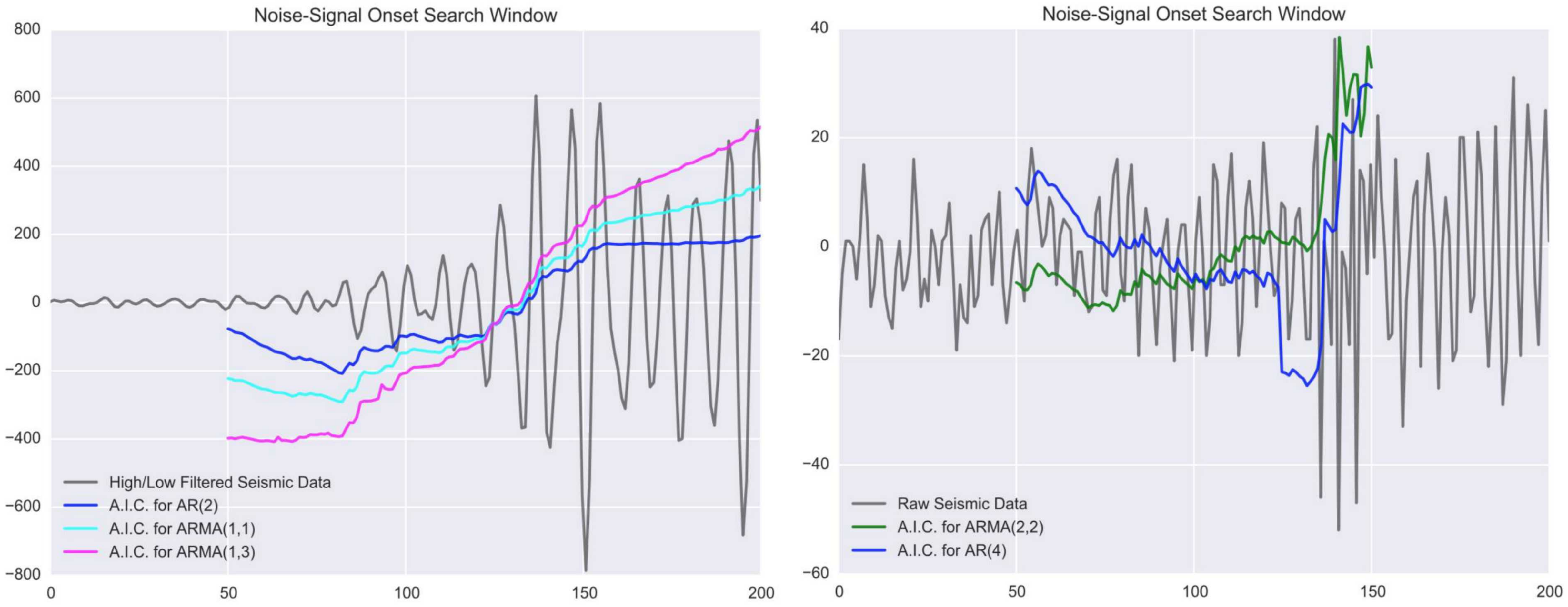
Inference Errors

Predictions

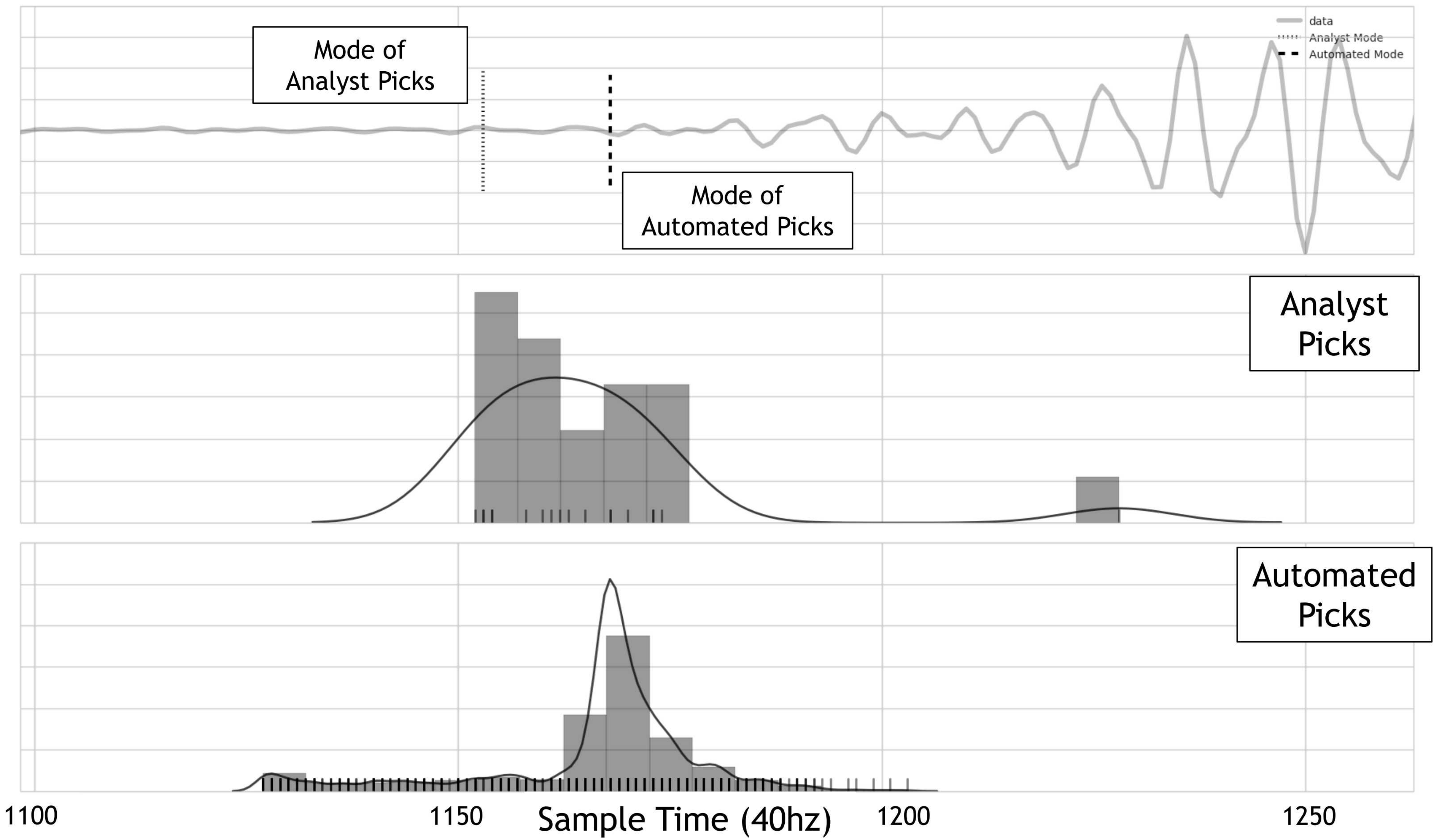


Model Form Uncertainty

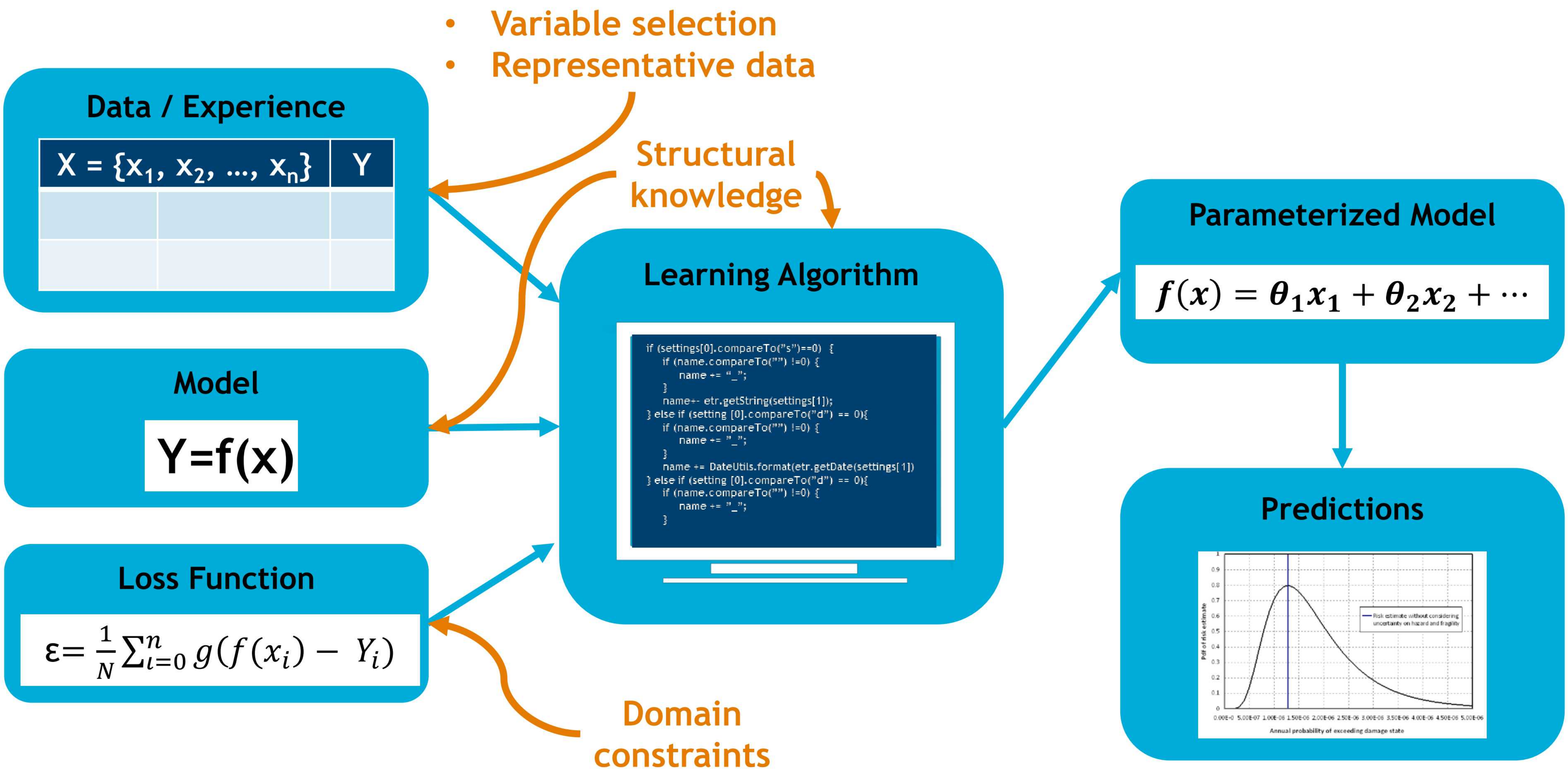
Uncertainty Example: Seismic Onset Detection



Uncertainty Example: Seismic Onset Detection



Domain Knowledge

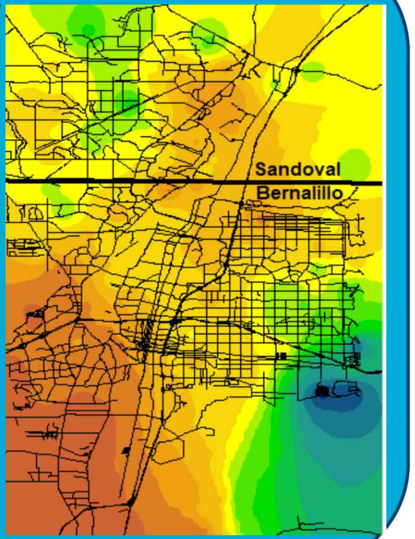


Interactions Between Domain Theories and Machine Learning

Sparse measurements

Example: Given sparse network of rainfall sensors and doppler radar, compute rainfall distribution map for entire region.

Becomes input to hydrology simulations.

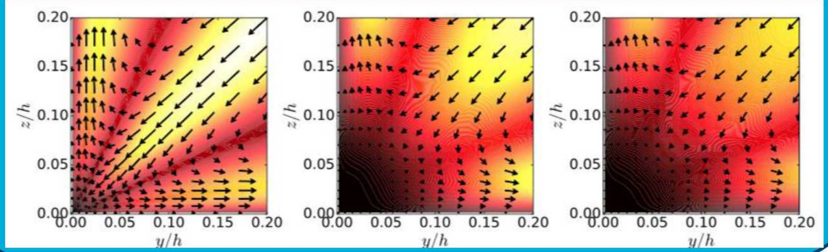


Predictive Model Induction

Example: Given URL format rules and known examples of benign and malicious links, learn to distinguish between the two.

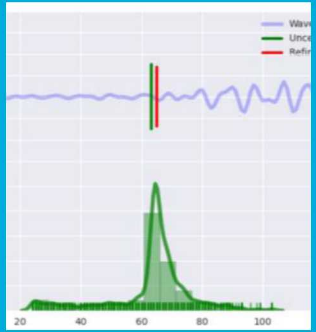
Error correction

Example: Compare mod/sim results to observations, learn an error term for the simulation model. Error term may use observables not included in the model.



Event detection

Example: Given seismograph network, identify all onset times and estimate relative detection quality.

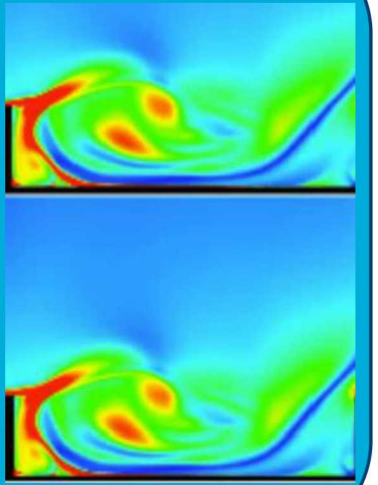


Becomes input to slowness inversion.

Surrogate models

Example: Given high-fidelity model and simulations, predict physical model outputs for given conditions.

Use learned model as inexpensive proxy for the high-fidelity model.



References

Textbooks on AI & ML

Hastie, T., Tibshirani, R. and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer

Mitchell, T.M. (1997). *Machine Learning*. McGraw-Hill

Russell, S. and Norvig, P. (2009). *Artificial Intelligence: A Modern Approach (3rd Ed)*. Pearson

Sutton, R.S. and Barto, A.G. (2018). *Reinforcement Learning: An Introduction (2nd Ed)*. MIT Press

Learning Theory & Information Theory

Kearns, M.J. and Vazirani, U. (1994). *An Introduction to Computational Learning Theory*. MIT Press

Rissanen, J. (2007). *Information and Complexity in Statistical Modeling*. Springer

Vapnik, V.N. (1999). *The Nature of Statistical Learning Theory (2nd Ed)*. Springer

Bayesian Methods

Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., Vehtari, A. and Rubin, D.B. (2013). *Bayesian Data Analysis (3rd Ed)*. CRC Press

Kruschke, J.K. (2015). *Doing Bayesian Data Analysis (2nd Ed)*. Academic Press.

References

Evaluation

Cohen, P.R. (1995). *Empirical Methods for Artificial Intelligence*. MIT Press

Zheng, A. (2015). *Evaluating Machine Learning Models*. O'Reilly

Uncertainty

Stracuzzi, D.J., Darling, M.C., Peterson, M.G., Chen, M.G. (2018). *Quantifying Uncertainty to Improve Decision Making in Machine Learning*. Sandia National Laboratories, SAND2018-11166.

Machine Learning, Domain Knowledge, & Interactions

Bauer, T. (in preparation). *Human Constrained Machine Learning: A Brief Survey and Ideas for Capability Development*. Sandia National Laboratories (SAND Report)

Karpatne, A., et al. (2017). Theory-Guided Data Science: A New Paradigm for Scientific Discovery from Data. *IEEE Transactions on Knowledge and Data Engineering, 29* (10).

Machine Learning Approaches and Data Considerations

Warren L. Davis IV (wldavis@sandia.gov)

September 9, 2019

Outline

- Factors in deciding upon a machine learning approach
- Classes of Machine Learning
  - Supervised Learning
  - Unsupervised Learning
  - Semi-Supervised Learning
  - Reinforcement Learning
- Information Representation

Deciding Upon a Machine Learning Approach

What problem are you trying to solve?

- Predict a category
- Predict a value
- Group data
- Find anomalies
- Find correlations
- Optimize parameters

What data is available?

- Numerical
- Categorical
- Images/Audio/Video
- Text
- …

Supervised Learning

Tasks

- Regression (continuous response)
- Classification (discrete response)
  - Binary (2 classes)
  - Multiclass (>2 classes)

Experience (data)

- Regression: input-output pairs
- Classification: feature-label pairs

Performance measures

- Many different methods

Iris Data (subset)

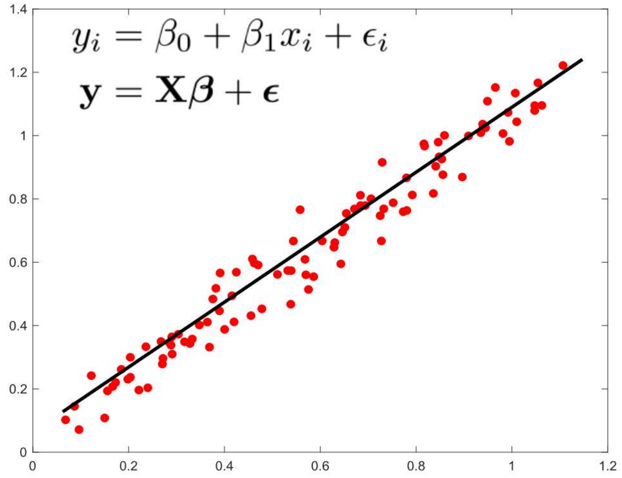
Sepal length	Sepal width	Petal length	Petal width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5	3.6	1.4	0.2	setosa
7	3.2	4.7	1.4	versicolor
6.4	3.2	4.5	1.5	versicolor
6.9	3.1	4.9	1.5	versicolor
5.5	2.3	4	1.3	versicolor
6.5	2.8	4.6	1.5	versicolor
6.3	3.3	6	2.5	virginica
5.8	2.7	5.1	1.9	virginica
7.1	3	5.9	2.1	virginica
6.3	2.9	5.6	1.8	virginica
6.5	3	5.8	2.2	virginica

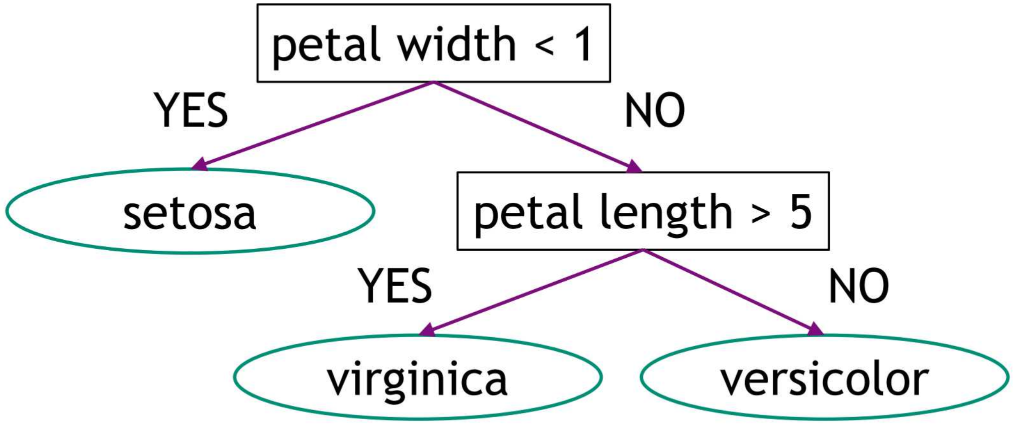
Features Label

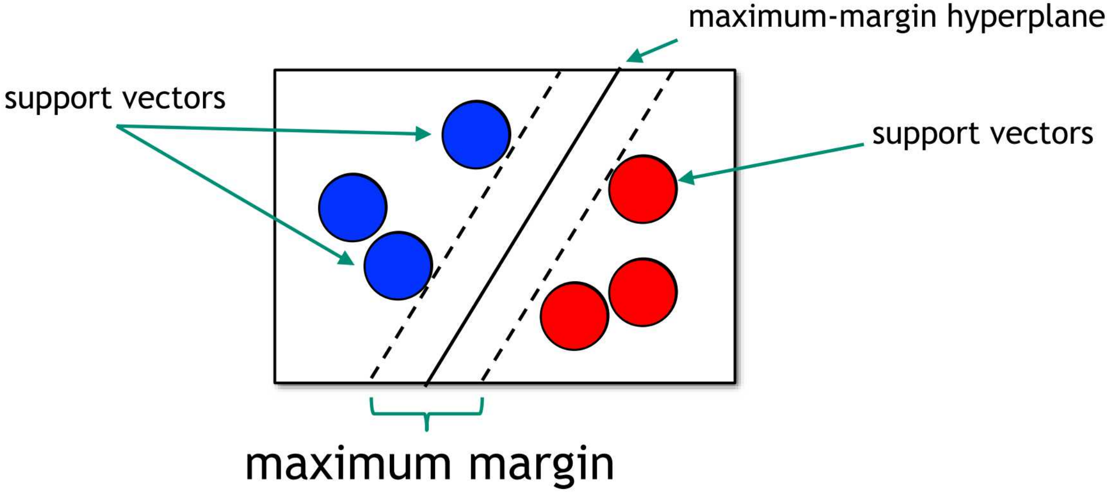
Fisher, 1936. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*. 7 (2): 179–188.
Anderson, 1936. The species problem in Iris. *Annals of the Missouri Botanical Garden*. 23 (3): 457–509.

Examples of Supervised Learning

- **Linear Regression**
- **Support Vector Machines**
- **Naïve Bayes**
- **Decision Trees / Random Forests**
- **Neural Networks**
- **k-Nearest Neighbor**







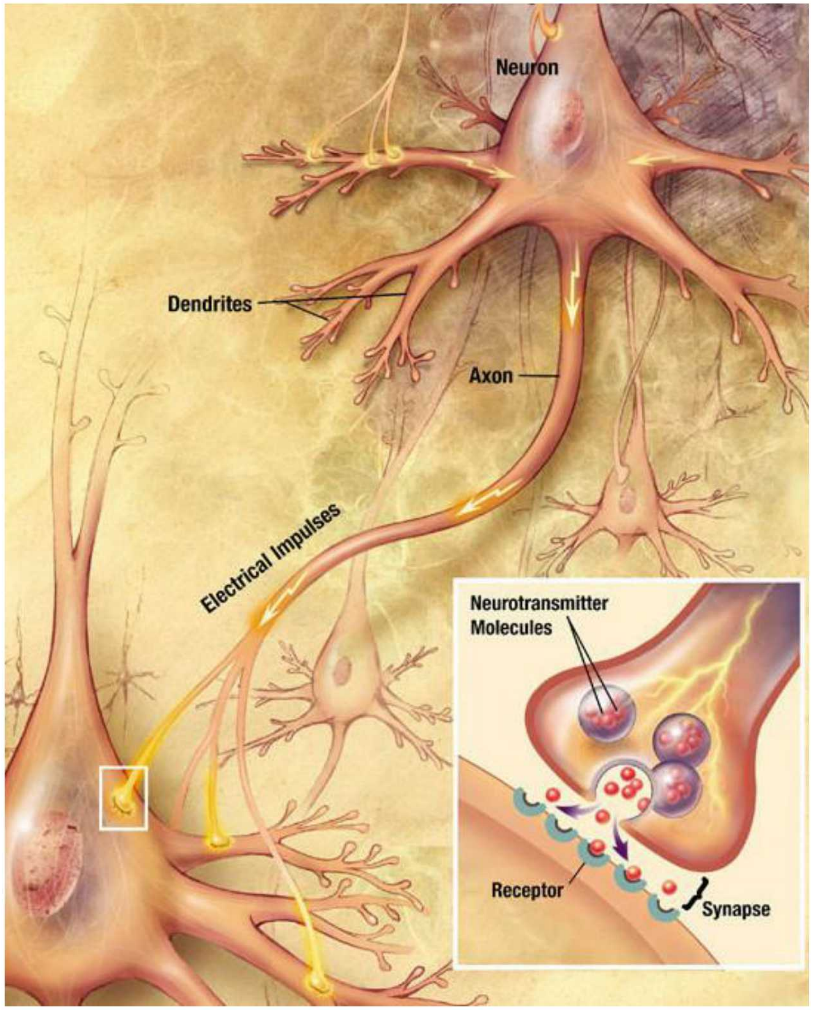
Neural Networks

Brain has neurons that communicate with other neurons through electrical impulses.

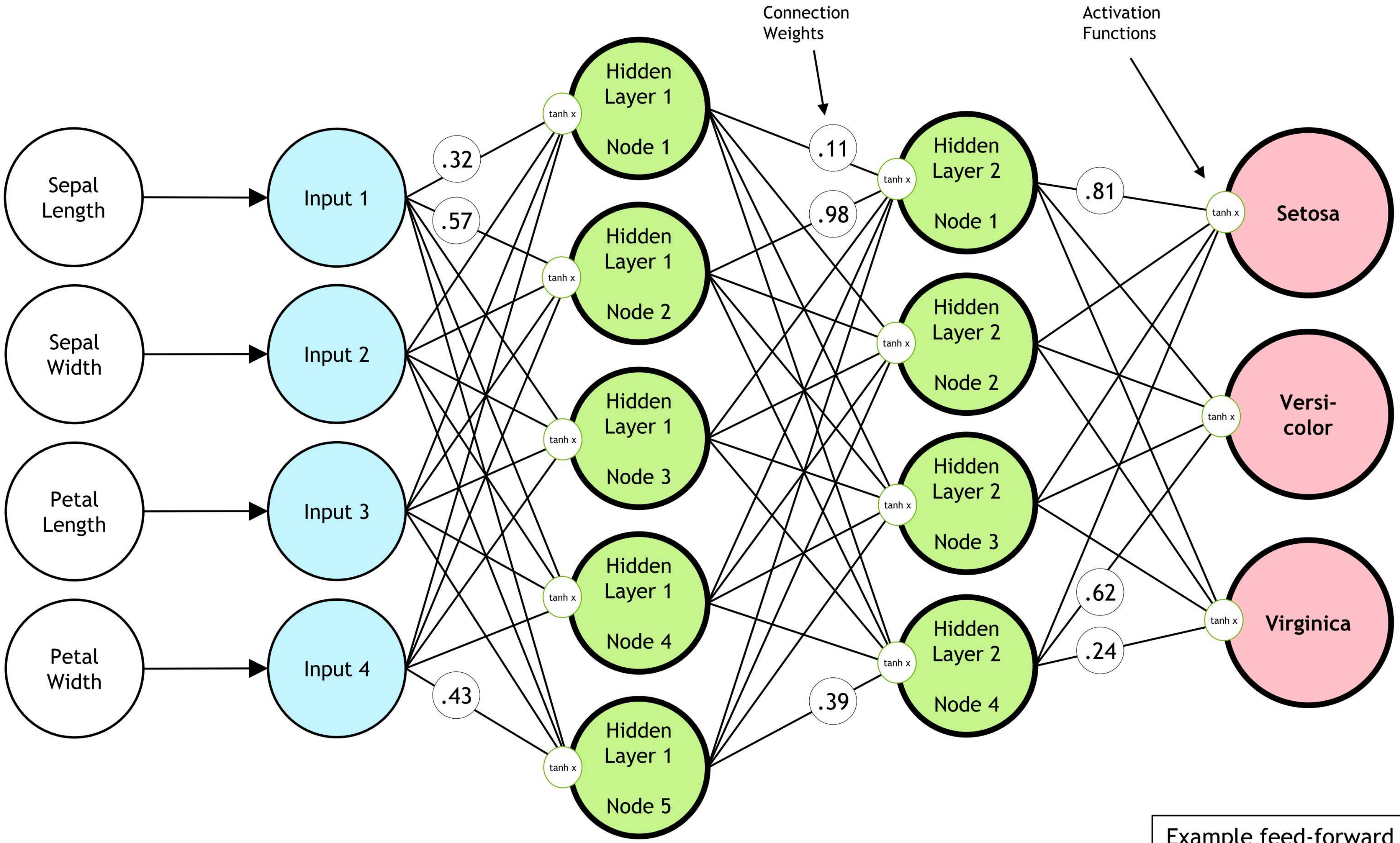
- Approximately 100 billion in human brain

Connections strengthen with experience

Neural networks are mathematical models inspired by the connectionist model of the brain



Artificial Neural Networks



Example feed-forward neural network

Advanced Neural Networks

Convolutional/Deep Networks
- Convolutional networks take advantage of local dependencies
- Deep networks capitalize on the power of deeper networks to encode/represent higher level, latent features
- Deep convolutional networks revolutionized the processing of images, sounds, and video
- Applicable to other modalities

Recurrent Neural Networks
- Takes data of varying length
- Useful for temporal and sequential data (e.g., text, signal processing)

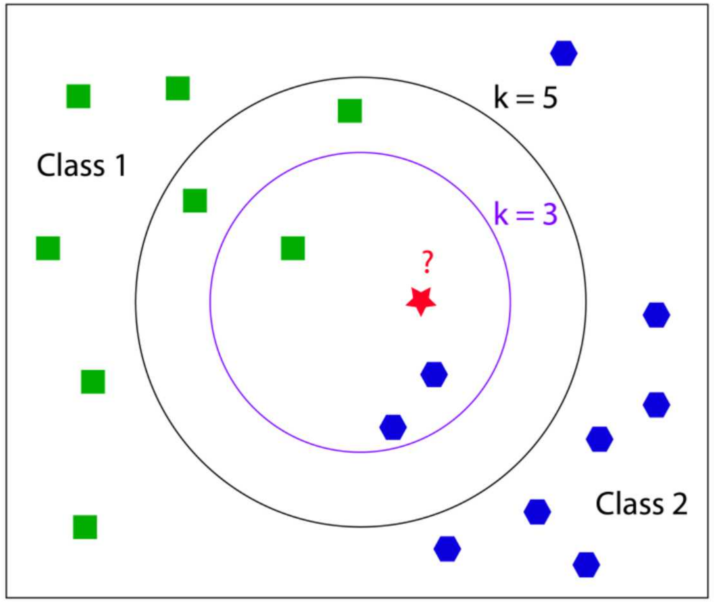
Autoencoders/Generative Adversarial Networks
- Autoencoders create compressed representations of the original data
  - Useful in anomaly detection, compression, domain feedback
- Variational autoencoders can generate new data
- Generative Adversarial Networks pit two models (usually neural networks) against each other
  - Generator creates new samples
  - Discriminator learns to tell original samples from generated samples
  - Generator and Discriminator co-evolve
  - ”Battle-tested” generator produces high quality new samples

k-Nearest Neighbor

Input: *k* closest instances (nearest neighbors) in feature space

Output

- Regression: average values of *k* nearest neighbors
- Classification: majority class of *k* nearest neighbors



https://www.quora.com/How-is-the-k-nearest-neighbor-algorithm-different-from-k-means-clustering

kNN: example of **instance-based learning**

- Function only approximated locally
- Computation deferred until prediction

Unsupervised Learning

Tasks

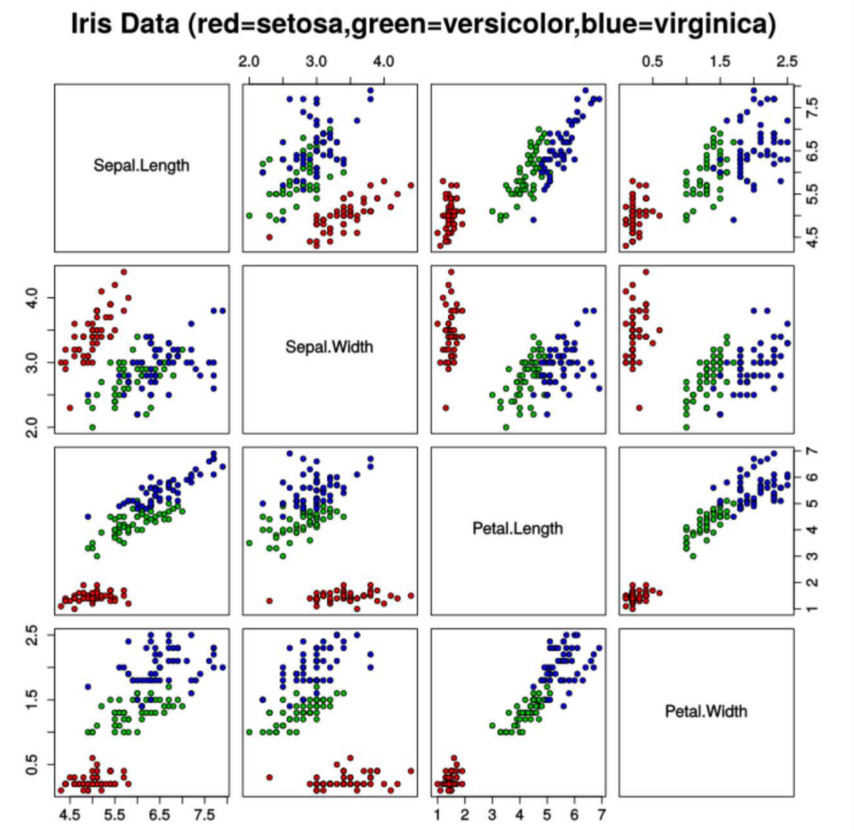
- Clustering (grouping)
- Dimensionality reduction
- Anomaly detection
- Association
- Generative modeling

Experience (data)

- Instances are unlabeled

Performance measures

- Challenging due to lack of labels/known solutions
- Validation often leverages labeled data sets (labels only used in testing)



Fisher, 1936. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*. 7 (2): 179–188.
Anderson, 1936. The species problem in Iris. *Annals of the Missouri Botanical Garden*. 23 (3): 457–509.

K-means Clustering

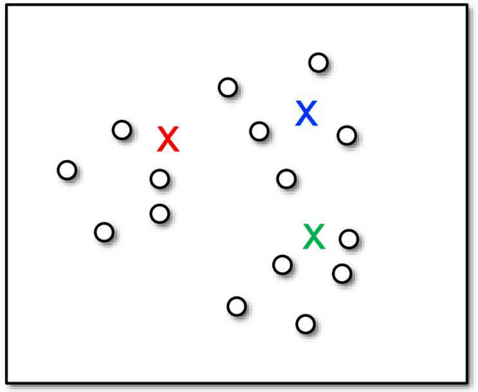
Task

- Group data instances by distance into *K* groups
- Data instances are points in a multidimensional feature vector space

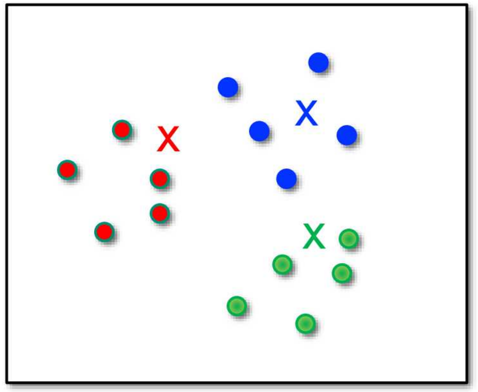
Standard Algorithm

1. Initialize cluster centroids randomly
2. Iterate until convergence
   a) **Assign** each instance to the cluster whose centroid is "closest"
   b) **Update** the centroids given the current cluster assignments

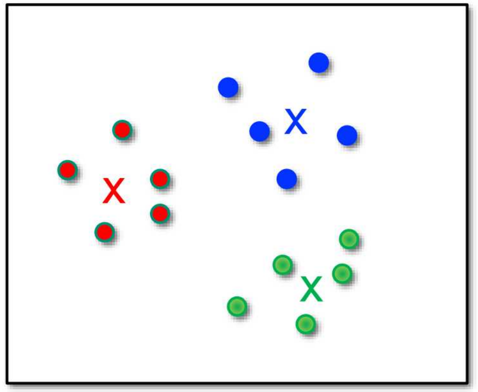
cluster centroid = arithmetic mean of the points in the cluster



Centroids (x) and cluster assignments (color) at **start of iteration**



Assignment of instances to cluster with closest centroid



Update centroids based on new cluster assignments

K-means Clustering

Task

- Group data instances by distance into *K* groups
- Data instances are points in a multidimensional feature vector space

Challenges

- What value to use for *K*?
  - Most often chosen by the user/analyst/subject matter expert
- How to initialize the centroids?
  - Random instances as centroids vs. random cluster assignments
- How to compute distances?
  - Euclidean distance often used
  - Often data- and problem-dependent
- When to stop iterating?
  - Assignment stagnation often used
  - *K*-means clustering is equivalent to local minimization

Other Partitional Clustering Methods

_K_-medoids

- *K*-means like algorithm using medoids (median values of cluster points) instead of means for assignments

Fuzzy K-means

- Fuzzy set membership for observations

DBSCAN

- density-based clustering with outlier detection and no predetermined number of clusters

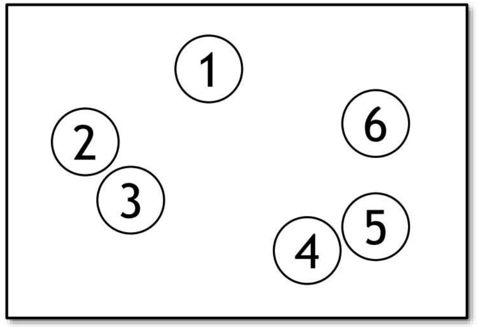
Gaussian Mixture Models

- *K*-means like algorithm with Gaussian distribution assumptions & probabilistic assignment

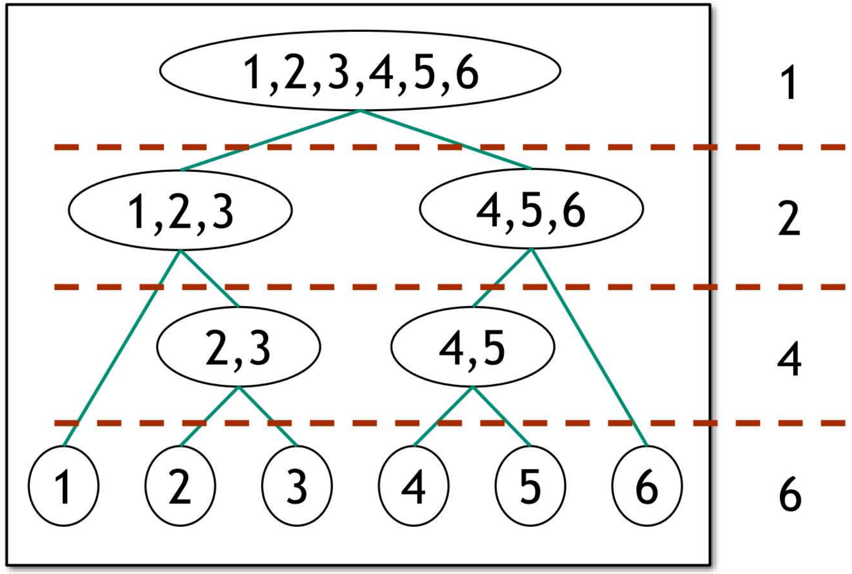
Spectral Clustering

- Useful for exploiting affinities (e.g., connections, similarities), in data points, regardless of Cartesian proximity

Hierarchical Clustering



Data in 2D Feature Space



Dendrogram

Clusters

Clustering Approaches

- Agglomerative
  - Merging from bottom to top
- Divisive
  - Splitting from top to bottom

Metric

- Distance between data points

Linkage Criteria

- Distance between sets
  - Single: minimum
  - Complete: maximum
  - Average

Number of clusters

- Choose a level to cut dendrogram

Gan, *et al.*, Data Clustering: Theory, Algorithms, and Applications. SIAM, 2007.

Semi-Supervised Learning

Tasks

- Supervised Learning Tasks

Experience (data)

- Small amount of labeled data
- Mostly unlabeled data

Performance measures

- Supervised Learning measures

Training model

- Train a model using labeled data
- Use model to predict labels for unlabeled data
- Add (some) unlabeled data and predicted labels to labeled data
- Repeat

Co-training

- Multiple classifiers working in tandem
- Requires independence between classifiers

Reinforcement Learning

Tasks

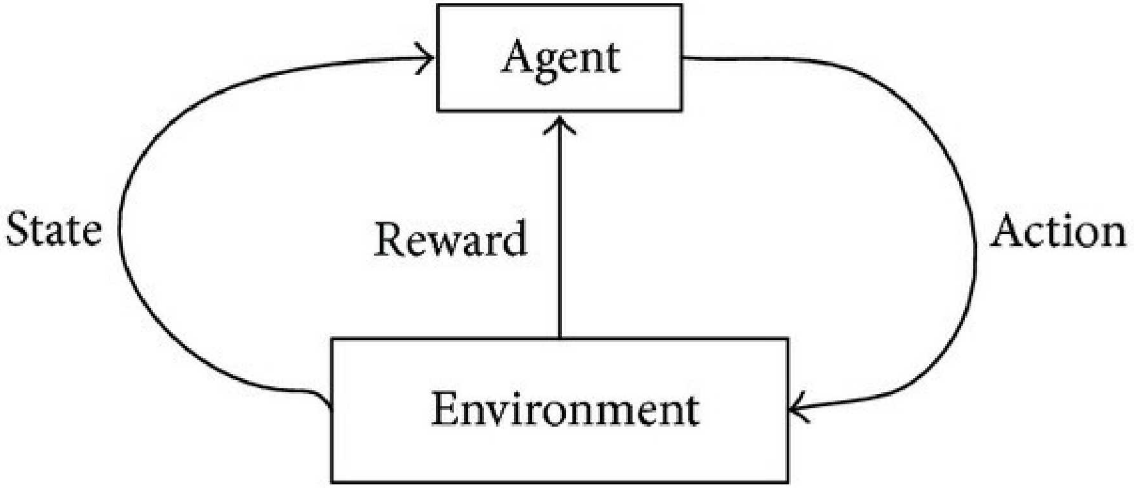
- Take the best action based on current state (i.e., information available)

Experience (data)

- Interactions with the environment/system
- State of environment/system

Performance measures

- Maximize reward
- Minimize risk



Reinforcement Learning: State-of-the-Art. Eds. Wiering and van Otterlo, Springer-Verlag, 2012.

Information Representation is Key to Machine Learning Success

Aforementioned examples assume that the data is already in the correct form to solve the problem

Knowledge Elicitation

- Gaining knowledge from Subject Matter Experts

Feature engineering / Data wrangling

- Getting the data in a form useful for answering the pertinent questions
- Often an iterative process

Feature selection

- Some features may be irrelevant
- Many algorithms are robust to this, but irrelevant features can degrade performance or cause machine learning methods to take longer than desired

Data properties

- Are the relevant features included?
- Is there enough of the data?
- Is the data drawn from the correct distribution?

Technical References

Tools

- **Scikit-learn**
  - https://scikit-learn.org/
- **PyTorch**
  - https://pytorch.org
- **Tensorflow**
  - https://www.tensorflow.org

Data

- **UCI Machine Learning Repository:**
  - https://archive.ics.uci.edu/ml/index.php
- **Kaggle:**
  - https://www.kaggle.com/datasets

Optimization with Application to Machine Learning and Power Systems

Jean-Paul Watson (jwatson@sandia.gov)

September 9, 2019

What Do We Mean By "Optimization"?

Linear programming (LP)

$$
\begin{aligned}
\arg\min_x \; & c^T x \\
s.t. \quad & Ax \le b \\
& x \in Q^n
\end{aligned}
$$

Classic example: Linear Assignment Problem (LAP)

$$
\begin{aligned}
\arg\min_x \; & \sum_{j \in N} \sum_{i \in N} c_{ij} x_{ij} \\
s.t. \quad & \sum_{i \in N} x_{ij} = 1 \quad \forall j \in N \\
& \sum_{j \in N} x_{ij} = 1 \quad \forall i \in N \\
& x_{ij} \ge 0 \qquad \forall i \in N, j \in N
\end{aligned}
$$

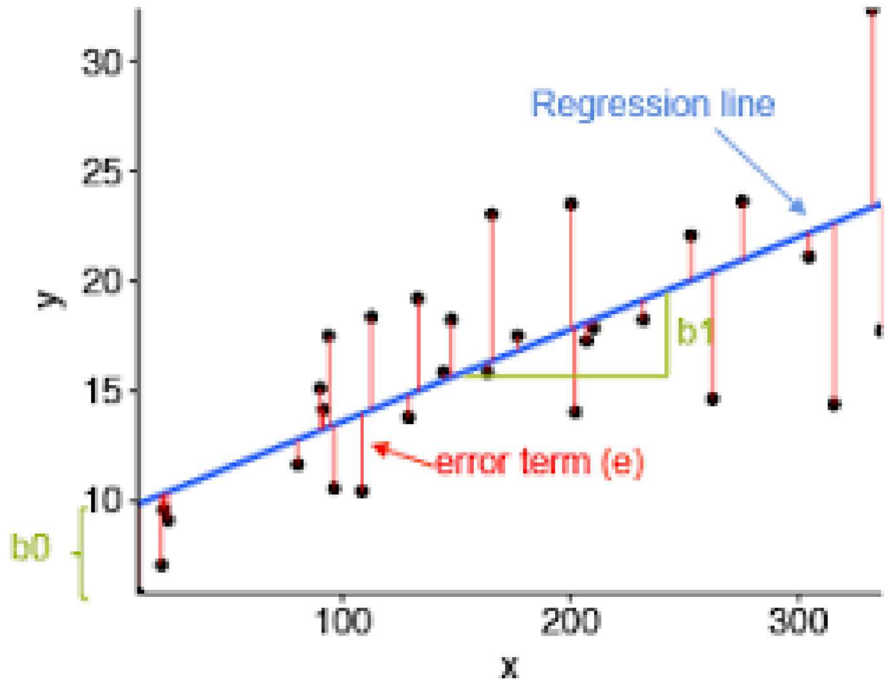
"Standard" form:

$$
\begin{aligned}
\arg\min_x \; & c^T x \\
s.t. \quad & Ax = b \\
& x \ge 0 \\
& x \in Q^n
\end{aligned}
$$

We generally assume that an algebraic description of the underlying problem is available

Popular extensions:
- Mixed-integer programming
- Non-linear programming
- Stochastic programming
- Robust optimization

Machine Learning and Optimization (I)

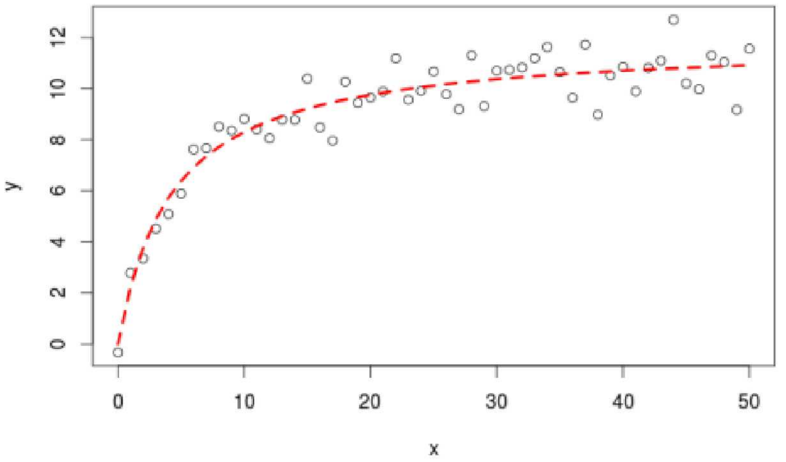


Linear regression **is** an optimization problem

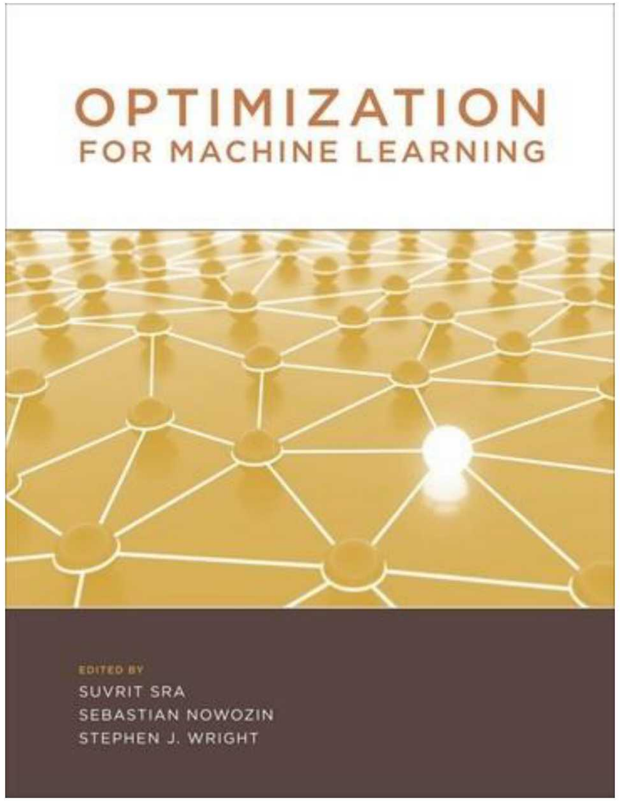
$$\text{Find } \min_{\alpha,\,\beta} Q(\alpha,\beta), \quad \text{for } Q(\alpha,\beta) = \sum_{i=1}^{n} \hat{\varepsilon}_i^{\,2} = \sum_{i=1}^{n} (y_i - \alpha - \beta x_i)^2$$

Slope-intercept parameters of a line

Non-linear regression is still an optimization problem – you just shift from linear programming to non-linear programming models and methods

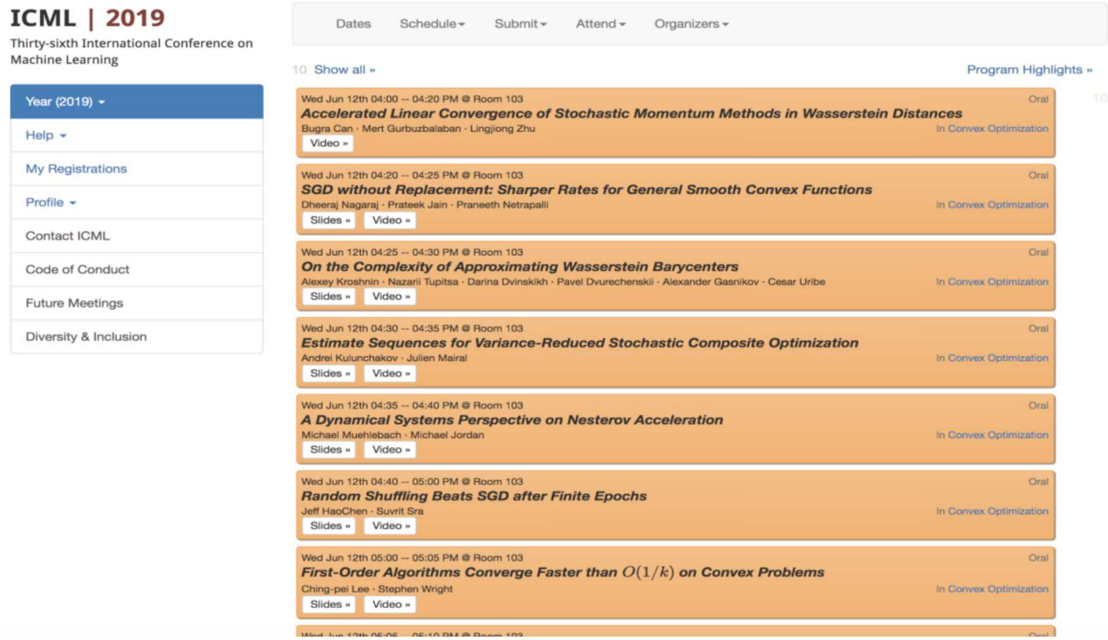


Machine Learning and Optimization (2)



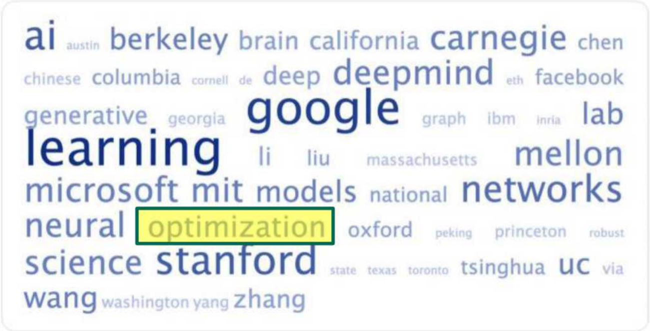
MIT Press

"The interplay between optimization and machine learning is one of the most important developments in modern computational science. Optimization formulations and methods are proving to be vital in designing algorithms to extract essential knowledge from huge volumes of data."



Many talk sessions at major machine learning conferences would be at home at optimization conferences

A word doodle of accepted papers at @NeurIPSConf -- Learning is more than deep.



Machine Learning and Optimization (3)

Journal of Machine Learning Research 7 (2006) 1265–1281 Submitted 7/06; Published 7/06

The Interplay of Optimization and Machine Learning Research

Kristin P. Bennett BENNEK@RPI.EDU
Department of Mathematical Sciences
Rensselaer Polytechnic Institute
Troy, NY 12018, USA

Emilio Parrado-Hernández EMIPAR@TSC.UC3M.ES
Department of Signal Processing and Communications
University Carlos III de Madrid
Leganés (Madrid), 28911, Spain

Editors: Kristin P. Bennett and Emilio Parrado-Hernández

Abstract

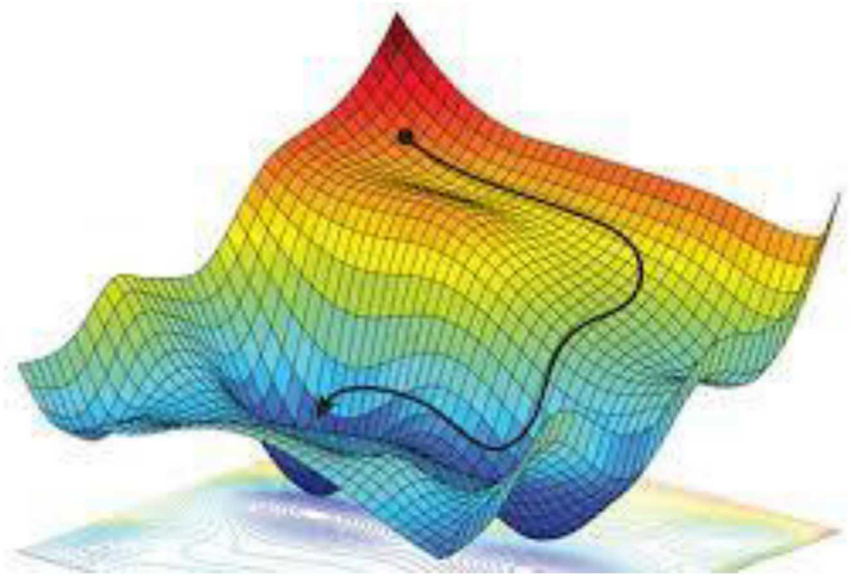
The fields of machine learning and mathematical programming are increasingly intertwined. Optimization problems lie at the heart of most machine learning approaches. The Special Topic on Machine Learning and Large Scale Optimization examines this interplay. Machine learning researchers have embraced the advances in mathematical programming allowing new types of models to be pursued. The special topic includes models using quadratic, linear, second-order cone, semi-definite, and semi-infinite programs. We observe that the qualities of good optimization algorithms from the machine learning and optimization perspectives can be quite different. Mathematical programming puts a premium on accuracy, speed, and robustness. Since generalization is the bottom line in machine learning and training is normally done off-line, accuracy and small speed improvements are of little concern in machine learning. Machine learning prefers simpler algorithms that work in reasonable computational time for specific classes of problems. Reducing machine learning problems to well-explored mathematical programming classes with robust general purpose optimization codes allows machine learning researchers to rapidly develop new techniques. In turn, machine learning presents new challenges to mathematical programming. The special issue include papers from two primary themes: novel machine learning models and novel optimization approaches for existing models. Many papers blend both themes, making small changes in the underlying core mathematical program that enable the develop of effective new algorithms.

This is even before the deep learning revolution…

”Optimization problems lie at the heart of most machine learning problems”

Machine Learning and Optimization (4)

Stochastic gradient descent (SGD) – a now standard optimization method – is at the center of the deep learning revolution



Training of deep (autoencoder) neural networks is a non-linear optimization problem to minimize reconstruction errors

But: SGD is a *local* method for solving a non-linear optimization model

- A heuristic – not a rigorous, complete solution method
- Absolutely no guarantee of optimality
- Nor any indication of how far you are from a global optimum

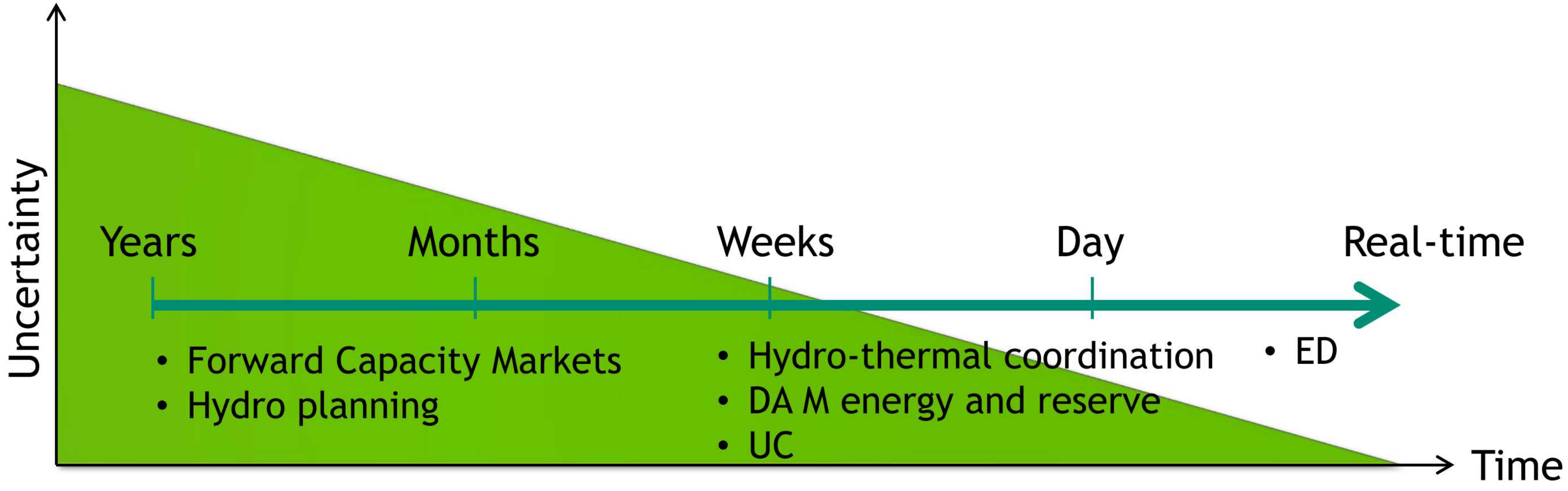
There is still much more that optimization can do for machine learning, e.g.,

- Rigorous proofs of global optimality
- Basis for adversarial machine learning
- From neural net training to architecture design

Most of Power Systems Operations and Planning is Optimization…

Decision making in power systems looks at processes ranging from very large time constants to near real-time:

Years, Seasons, Months, Weeks: Resource adequacy, transmission and hydro resource planning
Days: Hydro-thermal coordination, day-ahead UC of energy and reserves, intra-day UC
Hours: intra-day look-ahead processes, dynamic economic dispatch
Minutes: Economic Dispatch (ED)
Seconds: Automatic Generation Control (AGC)



<u>Every</u> **problem at the five minute and larger time scales is formulated and solved as an optimization problem**

ML for Power Systems Optimization: Warm Starting

The time required to solve operations problems such as commitment and dispatch can be significantly lowered by up to 80% via “warm starting” - use historical data to fit a ML model that predicts what are likely to be high-quality solutions for a given

Learning to Solve Large-Scale Security-Constrained Unit Commitment Problems

Álinson S. Xavier[1], Feng Qiu[1], and Shabbir Ahmed[2]

[1] Energy Systems Division, Argonne National Laboratory, Argonne, IL, USA. {axavier,fqiu}@anl.gov
[2] School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA. sahmed@isye.gatech.edu

Abstract. Security-Constrained Unit Commitment (SCUC) is a fundamental problem in power systems and electricity markets. In practical settings, SCUC is repeatedly solved via Mixed-Integer Linear Programming, sometimes multiple times per day, with only minor changes in input data. In this work, we propose a number of machine learning (ML) techniques to effectively extract information from previously solved instances in order to significantly improve the computational performance of MIP solvers when solving similar instances in the future. Based on statistical data, we predict redundant constraints in the formulation, good initial feasible solutions and affine subspaces where the optimal solution is likely to lie, leading to significant reduction in problem size. Computational results on a diverse set of realistic and large-scale instances show that, using the proposed techniques, SCUC can

A Distributed Framework for Solving and Benchmarking Security Constrained Unit Commitment with Warm Start

Publisher: IEEE

4 Author(s) Yonghong Chen ; Fengyu Wang ; Yaming Ma ; Yiyun Yao **View All Authors**

Abstract
Authors
Keywords
Metrics

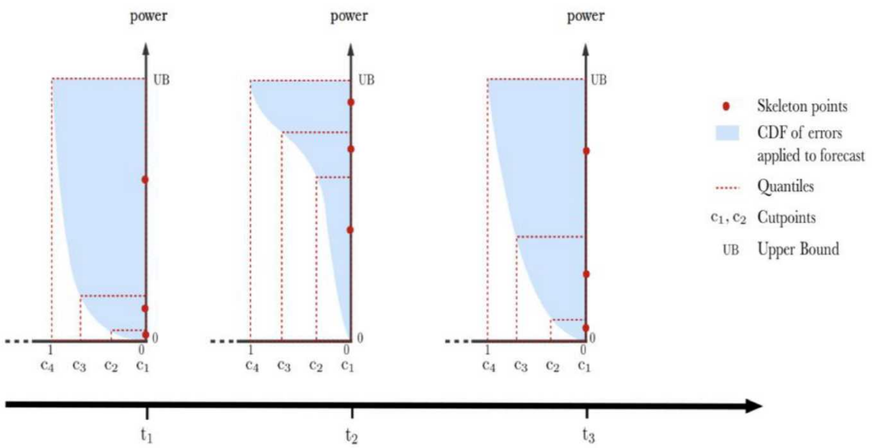
Abstract:
This paper discusses several methods to improve commercial optimization solver performance on day ahead security constrained unit commitment through warm start and lazy constraint settings. Data analytics is performed to greatly improve the quality of the initial commitment solution and lazy constraint setting. A distributed optimization framework is proposed to take advantage of the diversity from prevalent solvers (GUROBI and CPLEX) and different warm start strategies. A systematic distribution profile based benchmarking method is also proposed.

Published in: IEEE Transactions on Power Systems (Early Access)

Related techniques hold even more promise in the context of stochastic power systems operations problems, which are significantly more difficult in practice

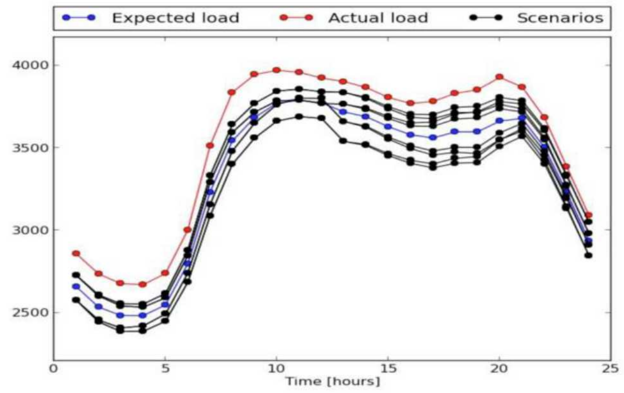
ML for Power Systems Optimization: Scenario Construction

Historical forecasts and corresponding actuals are fed into ML algorithms to characterize error distributions…

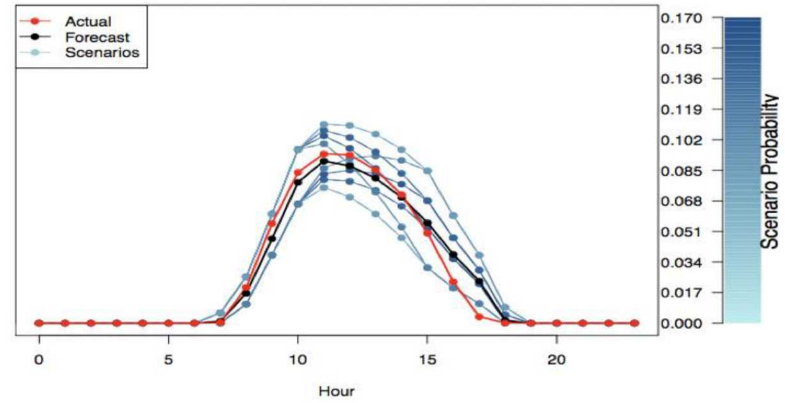


.. which are then used to construct probabilistic scenarios for operations

Day-Ahead Scenarios for Bulk Load



Day-Ahead Scenarios for Bulk Solar



Probabilistic scenarios form the basis for stochastic power systems operations and planning problems – and they are provided by ML

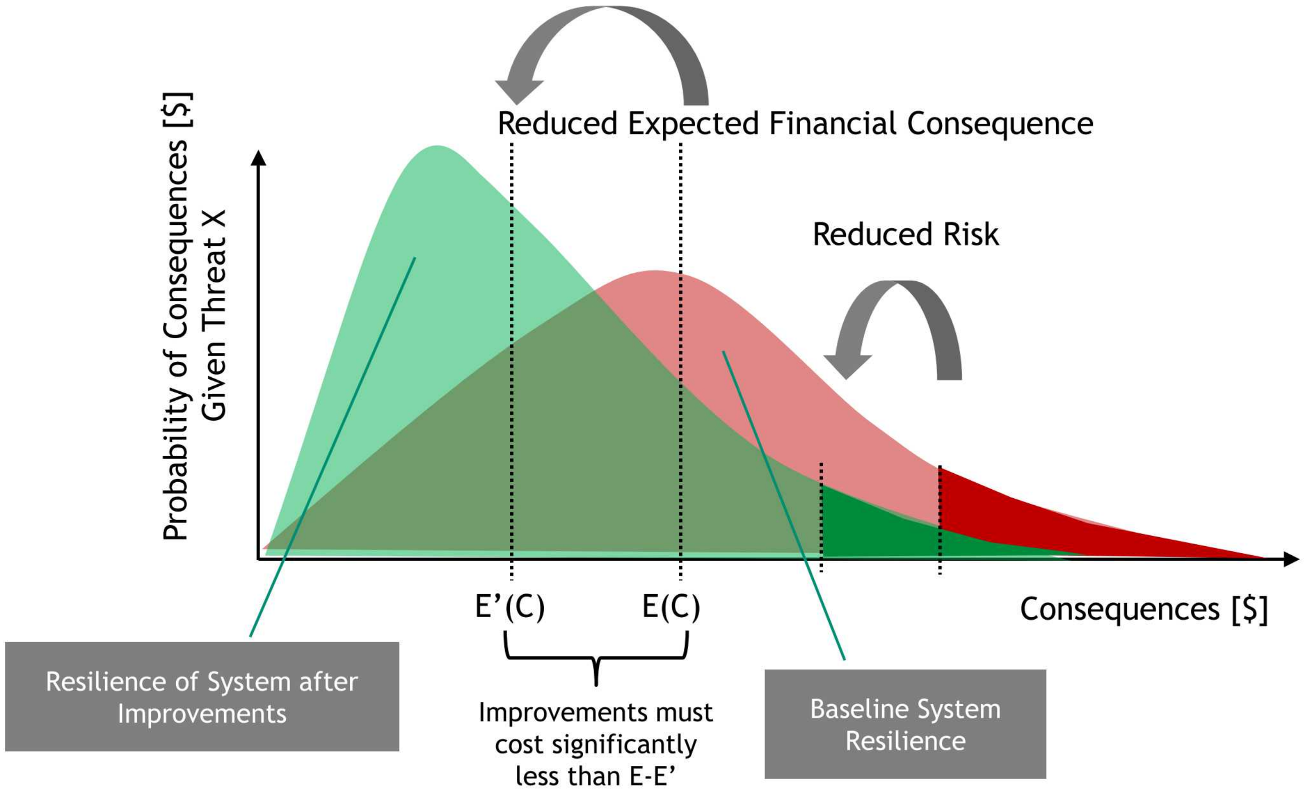
(Examples of) Machine Learning for the North American Energy Resilience Model (NAERM)

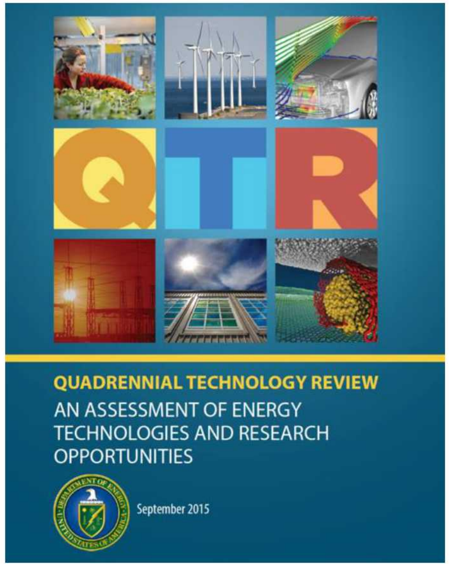
Jean-Paul Watson (jwatson@sandia.gov)

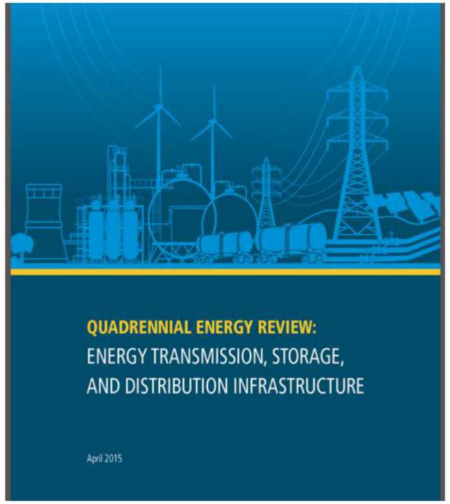
September 9, 2019

Resilience Quantification: Stochastic Models are Critical

ML is central to developing probabilistic models of threats – which are critical inputs to resilience analysis

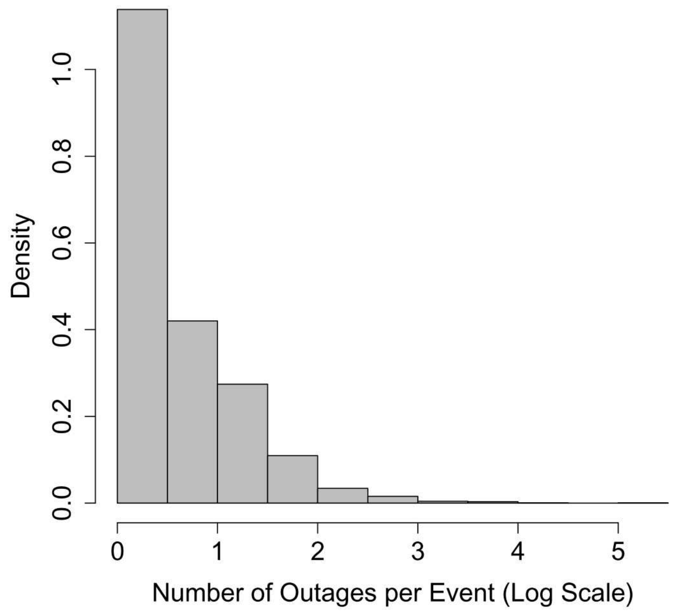




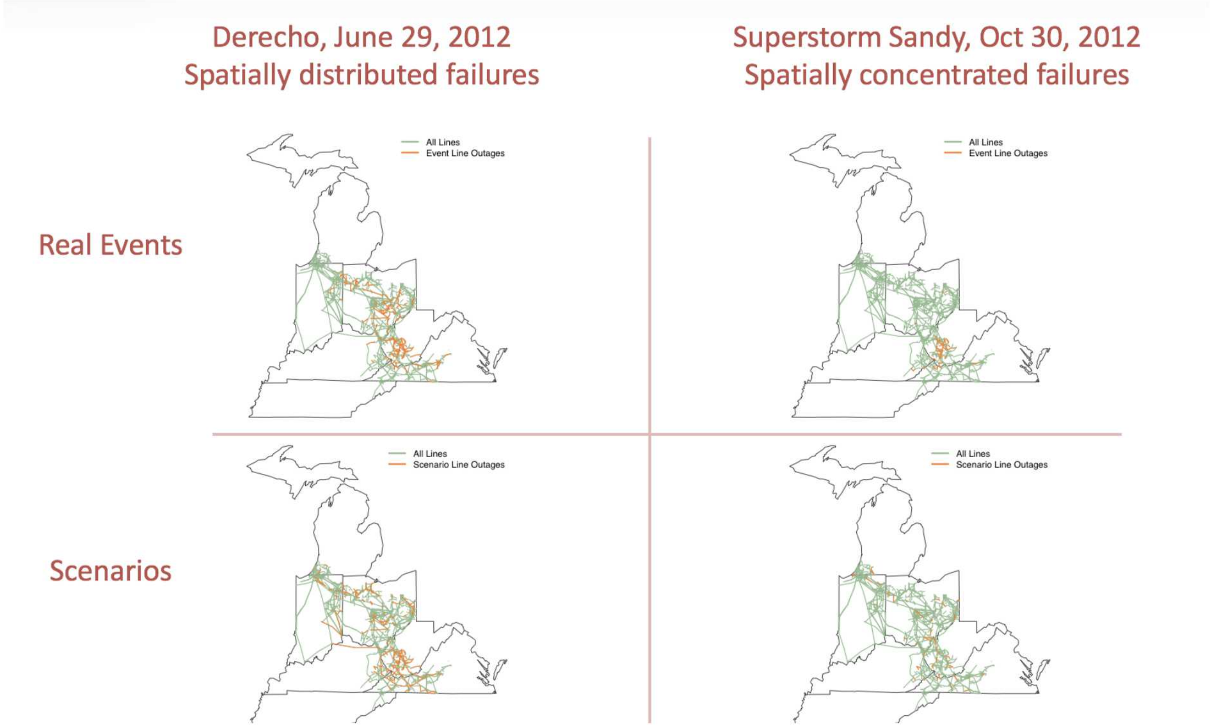


Resilience Analysis: Probabilistic Outage Scenarios

Historical transmission outage data associated with extreme weather events



Probabilistic ML models calibrated using historical outage data

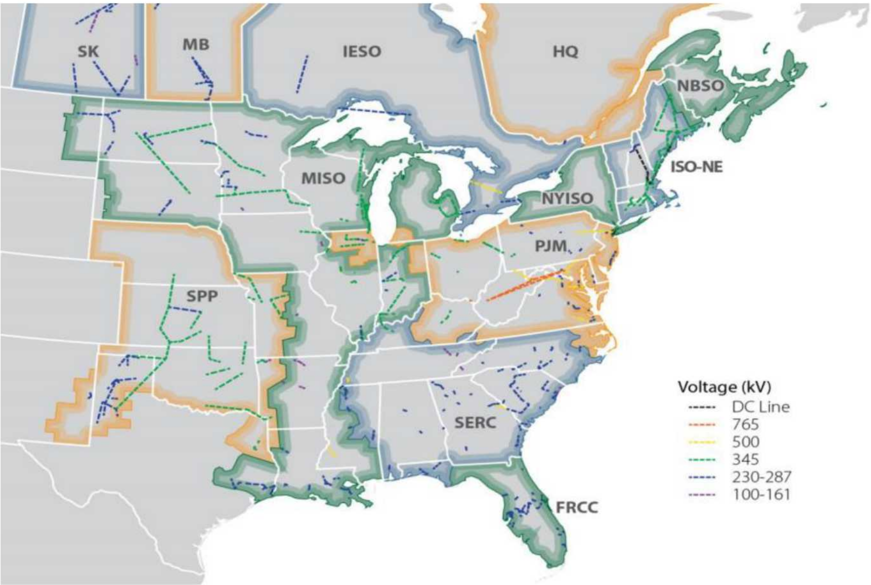


Probabilistic outage scenarios are a pre-requisite for proactive resilience operations and investment strategies..

… and are equally applicable in planning and real-time contexts

ML for Accelerating National-Scale Grid Computation

Future EI Case (ERGIS, from NREL)



Significant technology development efforts required to execute ERGIS cases in tractable run times

~11K generators in entire system

- Includes two very large ISOs
  - Difficult to solve in isolation, let alone in a coordinated manner
- Major challenges for solving core operations simulations such as commitment and dispatch

ML methods for accelerating commitment and dispatch optimization model solves can potentially yield order-of-magnitude reductions in run times

ML-Based Grid Situational Awareness and Control

Significant emerging efforts in the realm of ML for proactive power grid operations via deep ML

From Grid Eye to Grid Mind

-A Data-driven Autonomous Grid Dispatch Robot Based on PMU Measurements

Di Shi, Ruisheng Diao, Jiajun Duan, Bei Zhang, Zhe Yu, Zhiwei Wang, Xiao Lu*,

Haifeng Li*, Chunlei Xu*, Ya

GEIRI North America (GEIRINA

*State Grid Jiangsu Electric Power (

April 15-17, 2

@ NASPI April Work G

L2RPN Challenge

- Learning to Run a Power Network throu

Di Shi

Team: Tu Lan, Jiajun Duan, Bei Zhang, Zhiwei

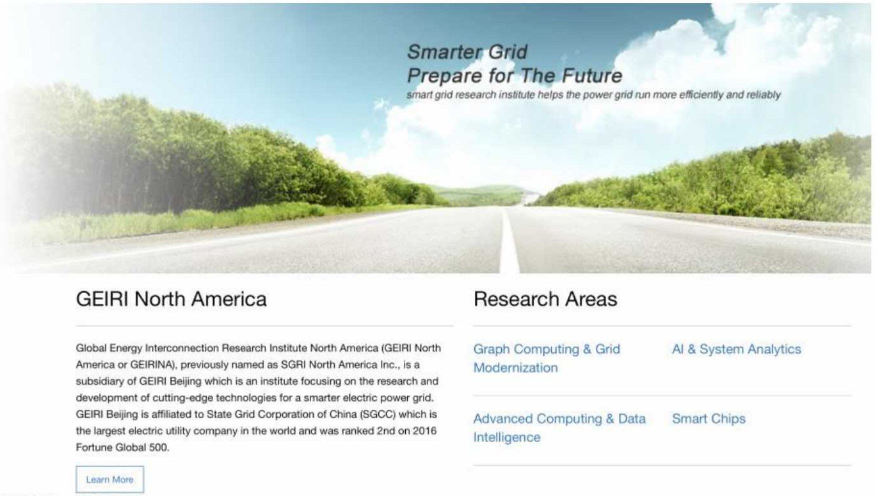
Zhang, Ruisheng Diao, Yan Zan

AI & System Analytics

GEIRI North America (GEIRINA)

@PSERC Summer Workshop

July 16, 2019



Key question is whether such methods can be extended from reliability to resilience contexts, and beyond minute-scale look-ahead

Highlights of Artificial Intelligence/Machine Learning in Power Systems

Matthew Reno (mjreno@sandia.gov)

Logan Blakely (lblakel@sandia.gov)

September 9, 2019

Power System Applications of Artificial Intelligence

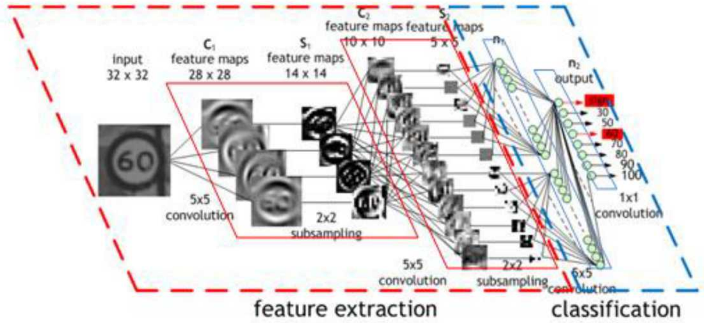
Power Systems is an ideal field for applications of Artificial Intelligence due to the complex systems and large amounts of data.

AI/ML algorithms were recently made possible due to:

- Advances in computing power for real-time learning and decision making
  - Massive computing power even at the grid edge in advanced inverters and Real-Time Automation Controllers (RTAC)

- Additions of new sensing equipment such as smart meters and PMU
  - 2000 PMU and 70 million smart meters in the U.S.
  - Synchronized and real-time communication

- New Artificial Intelligence algorithms to handle large datasets, the advent of transferable learning, and physics-based algorithms

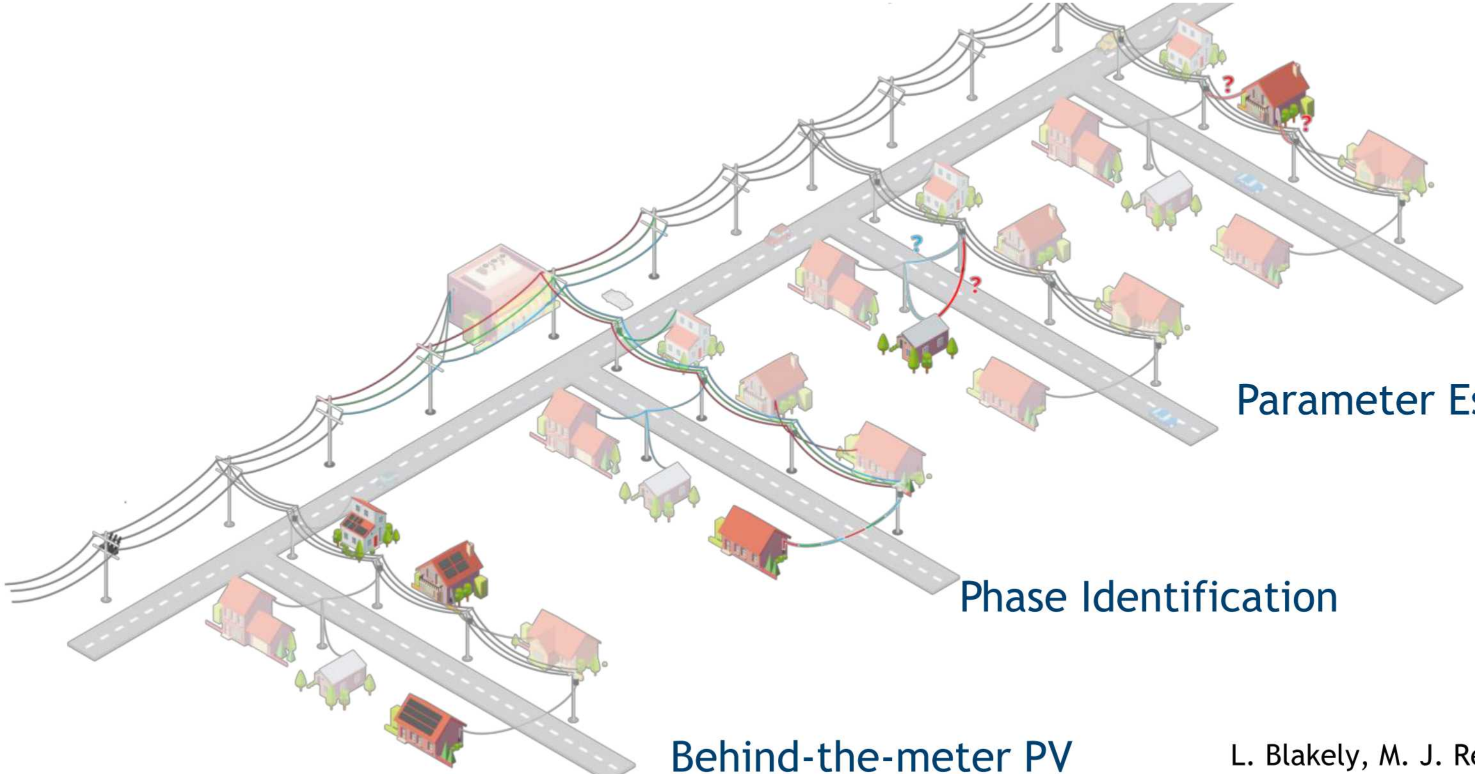


These slides include several examples of AI successes in Power Systems and future research directions.

AI for Model Calibration

Use measured data to estimate distribution system parameters and state

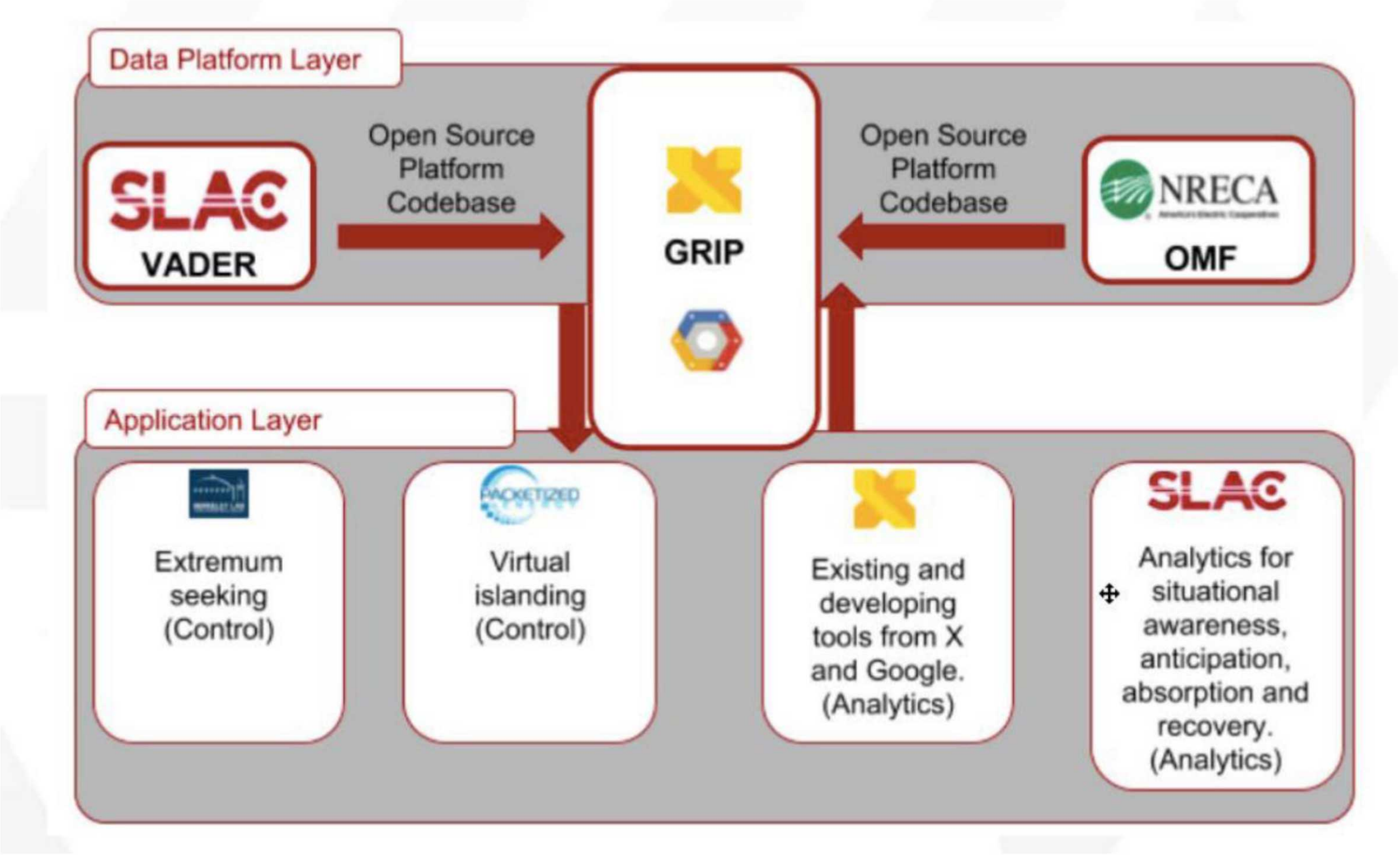
- AI algorithms can provide insight into distribution systems and distributed energy resources by leveraging and integrating high-fidelity sensors and multiple data sources
- Ingest data from AMI, SCADA, μPMU, etc. and use data analytics and machine learning methods to estimate system parameters (phase, meter-transformer pairing, line lengths, etc.) and do state estimation



L. Blakely, M. J. Reno, and W. Feng, "Spectral Clustering for Customer Phase Identification Using AMI Voltage Timeseries," Power and Energy Conference at Illinois, 2019.

AI for Resilient Response

Grid Resilience and Intelligence Platform (GRIP) aggregates data, anticipates disruptions, validates control options, and reduces recovery time from extreme events

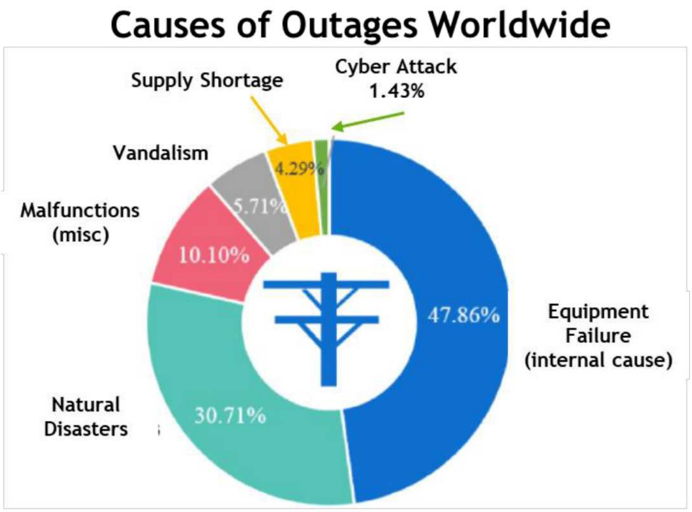


https://gmlc.doe.gov/sites/default/files/resources/1.5.01_GRIP_Fact%20Sheet_8-30-18.pdf

AI for Protection Applications

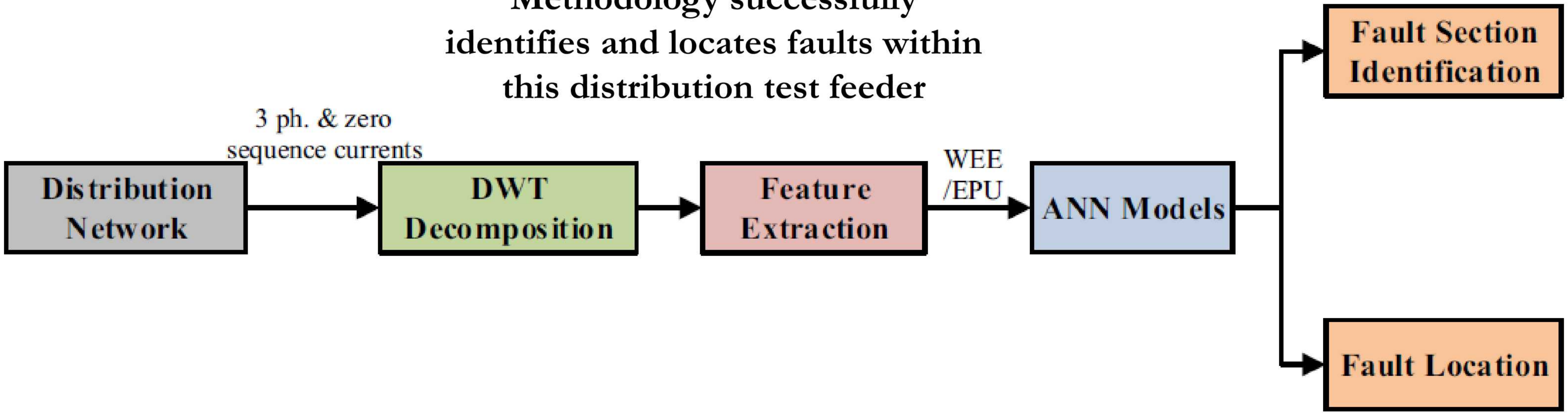
Neural Networks for Fault Identification and Fault Location

- AI and Machine Learning can improve resilience by aiding grid operators during fault events
- Using AI for relay-less protection schemes



Z. Bie, Y. Lin, G. Li, and Li Furong, "Battling the Extreme: A Study on the Power System Resilience," *Proc. IEEE*, vol. 105, no. 7, pp. 1253–1266.

Methodology successfully identifies and locates faults within this distribution test feeder

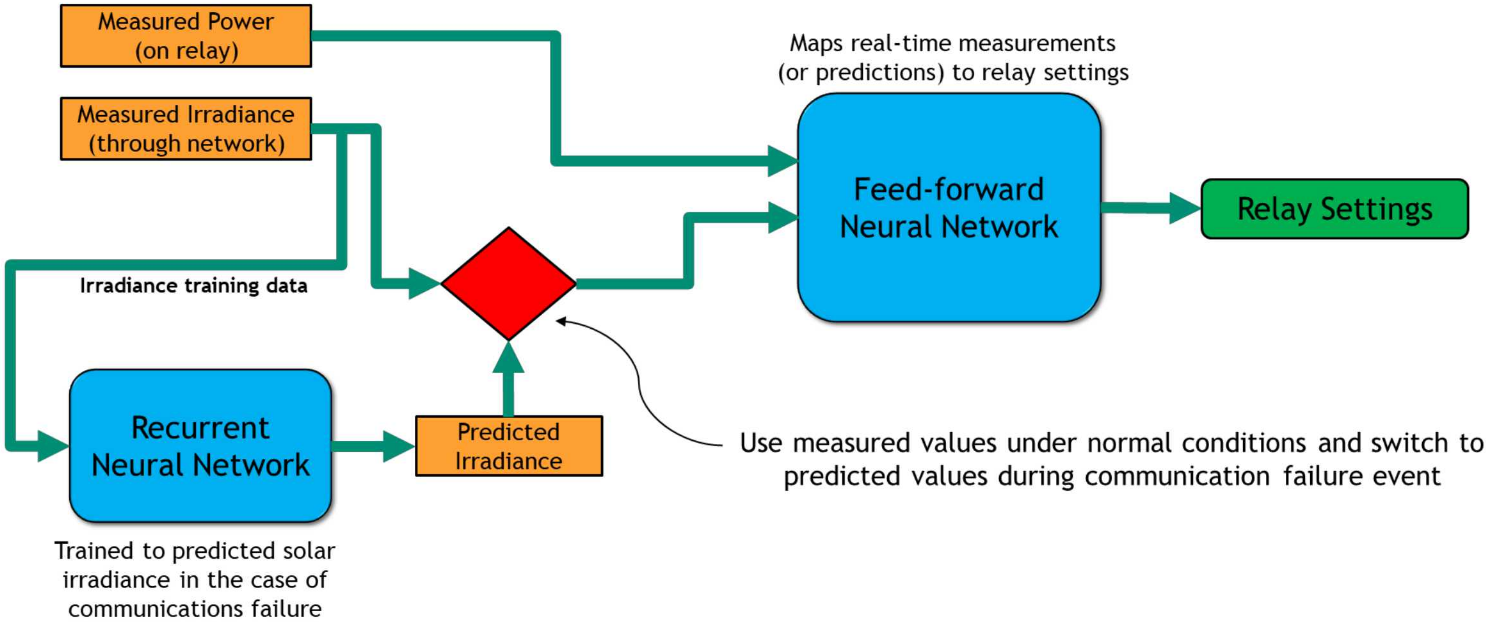


A. C. Adewole, R. Tzoneva, and S. Behardien, "Distribution Network Fault Section Identification and Fault Location Using Wavelet Entropy and Neural Networks," *Appl. Soft Comput.*, vol. 46, pp. 296–306, 2016.

AI in Protective Relays

Adaptive Model Driven Protective Relay with AI/Machine Learning

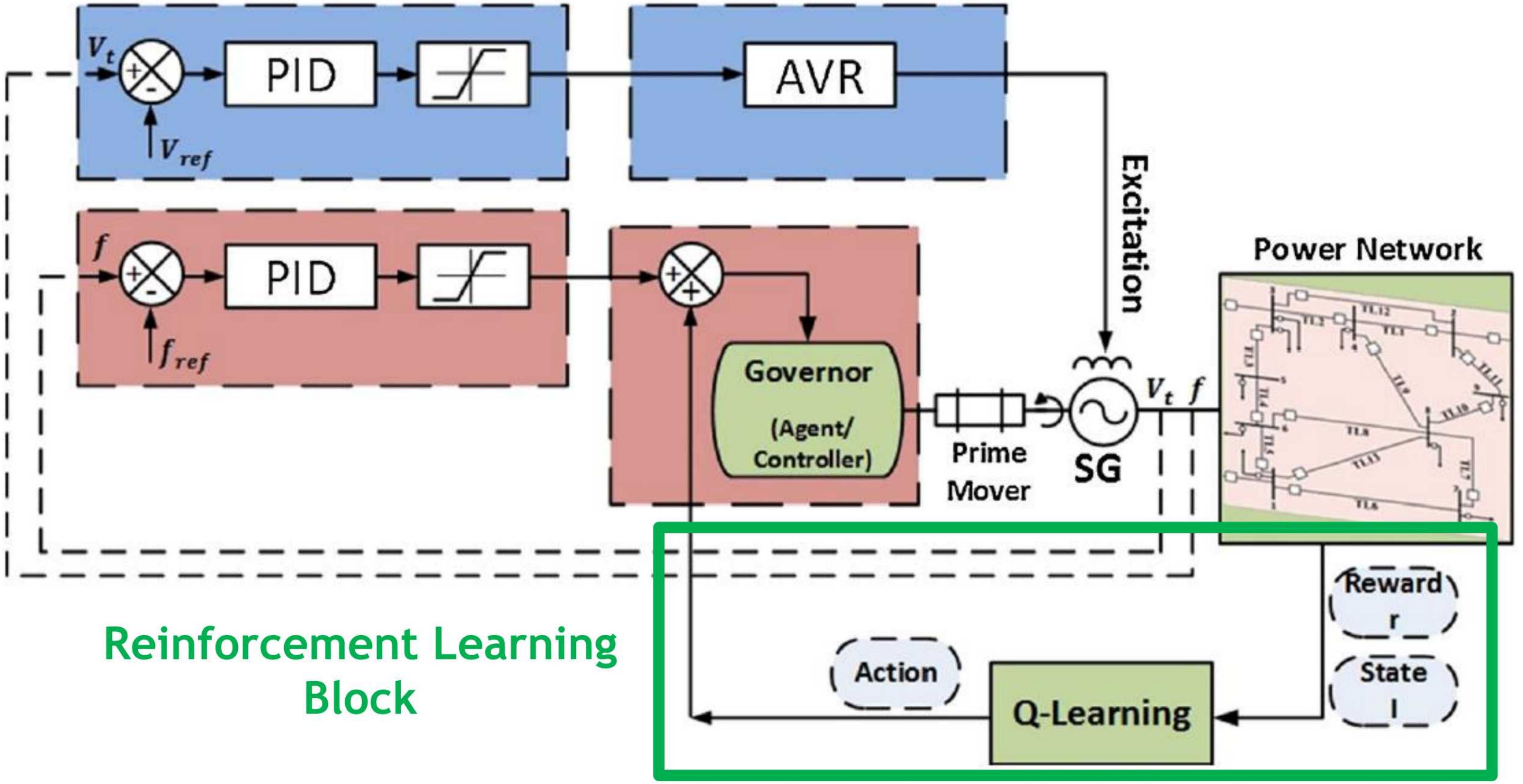
- AI can provide grid resilience with adaptive, smart relays using multiple data streams and autonomous model-based analysis
- Integrate ML into relays for resilience to loss of communication or cyber attacks
- Identify bad settings or miscoordination between devices
- Learn appropriate settings, reclose patterns, system events, and backup protection



AI for Controls Applications

Reinforcement learning can operate in real-time using rewards to develop new controls applications

- Control of devices, such as Megawatt Scale Grid Storage for frequency regulation
- Grid control for Resilience - avoid instability, respond to failures, and prevent cascading outages



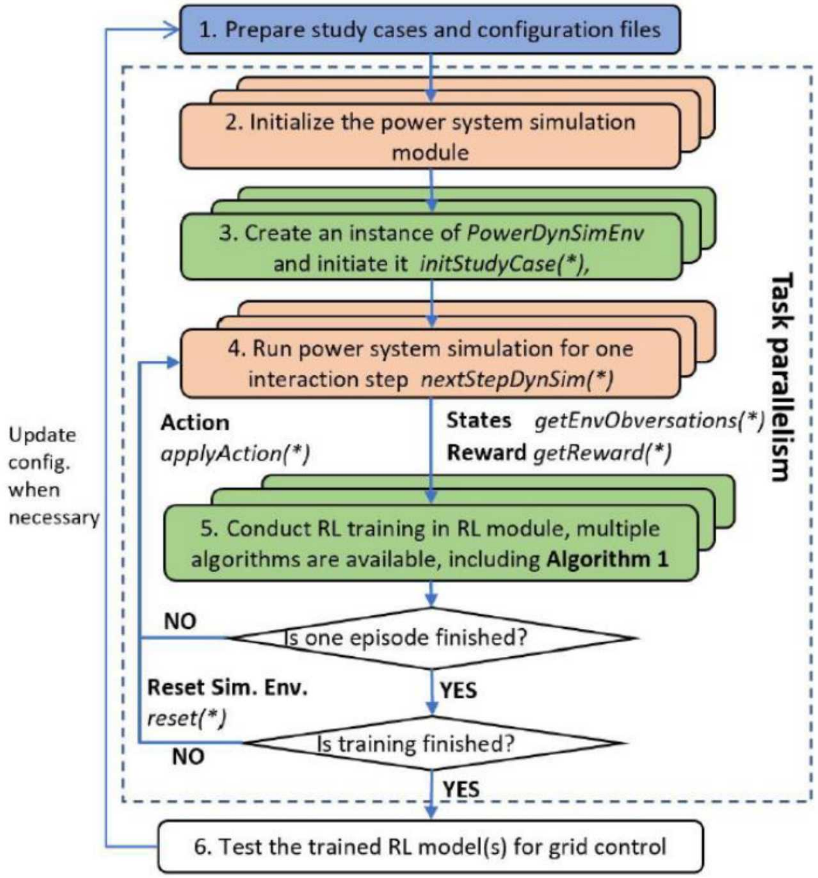
S. Zarrabian, R. Belkacemi, and A. A. Babalola, "Reinforcement Learning Approach for Congestion Management and Cascading Failure Prevention with Experimental Application," *Electr. Power Syst. Res.*, vol. 141, pp. 179–190, 2016.

AI for Controls Applications

Deep Reinforcement Learning for Emergency Scenarios

- AI can improve grid resiliency during extreme events by providing rapid controls such as dynamic generator brake and under-voltage load shedding

Deep Reinforcement learning algorithm successfully learns the dynamic generator brake task as well as the under-voltage load shedding task, outperforming conventional methods



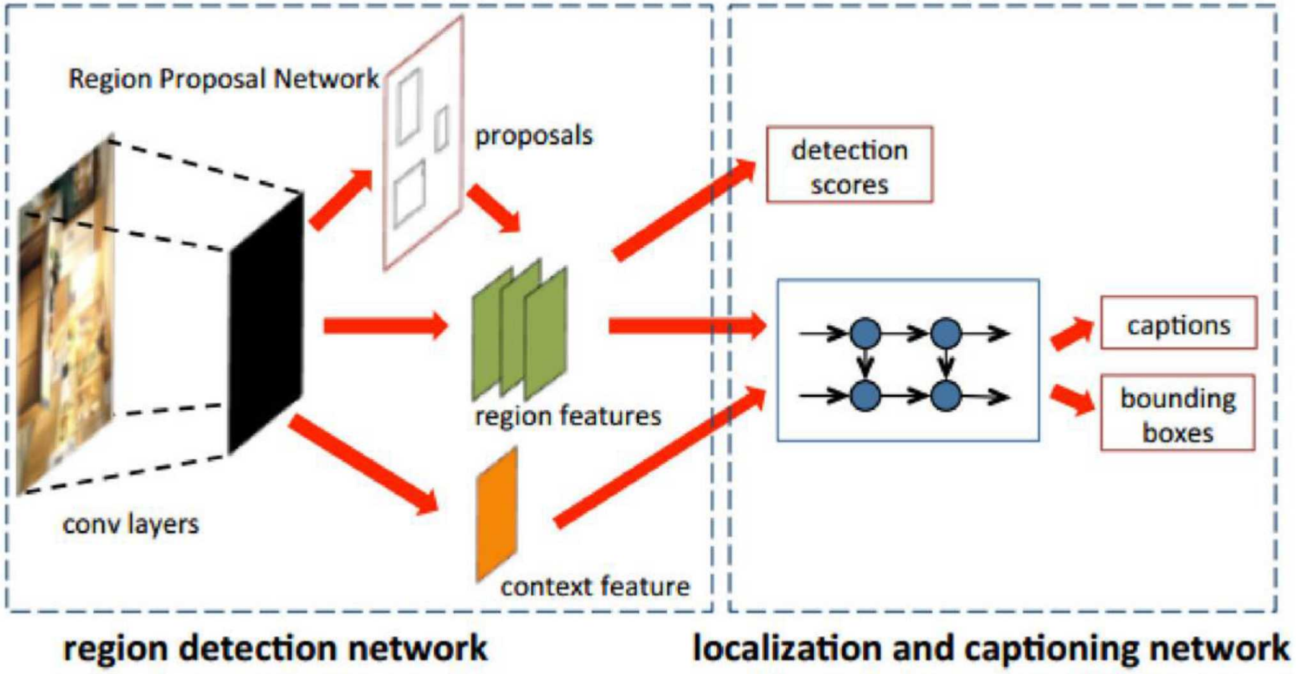
Q. Huang, R. Huang, W. Hao, J. Tan, R. Fan, and H. Zhenyu, “Adaptive Power System Emergency Control Using Deep Reinforcement Learning,” *IEEE Transacation Smart Grid*, 2019.

Future Research Directions

Many innovations in AI and machine learning have not yet been applied to the power systems domain

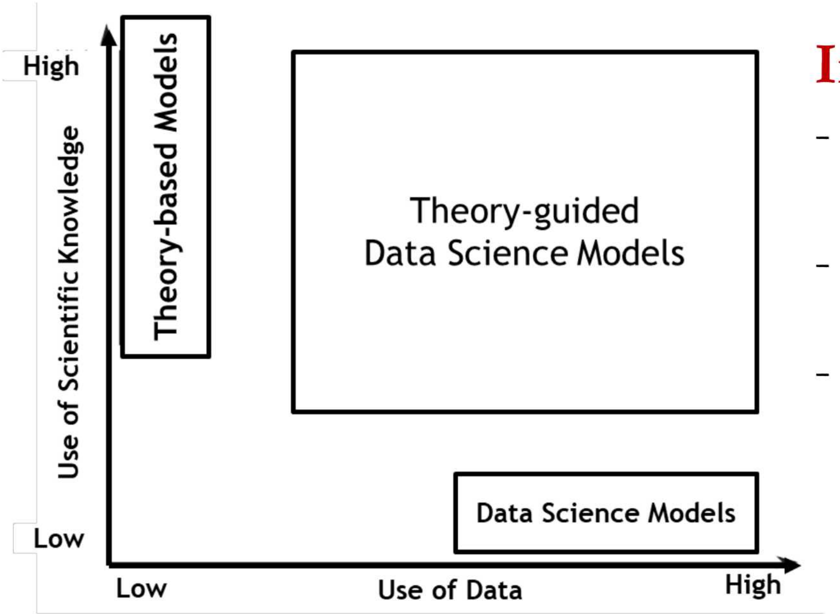
- As improvements and breakthroughs happen in other domains, those concepts can be adjusted and applied to solve power systems problems
- Similarly, lessons learned from other domains can be used to avoid similar situations

- Image Processing
  - Recognition
  - Captioning
  - Generation
  - Style Transfer
- Natural Language Processing
  - Translation
  - Summarization
  - Generation
- Autonomous Vehicles
- Game Theory



X. Liu, Q. Xu, and N. Wang, "A Survey on Deep Neural Network-based Image Captioning," *Vis. Comput.*, vol. 35, no. 3, pp. 445–470, Mar. 2019.

Future Research Directions



A. Karpatne *et al.*, "Theory-guided Data Science: A New Paradigm for Scientific Discovery from Data," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 10, pp. 2318–2331, 2017.

Integration of Physics-based Constraints into AI

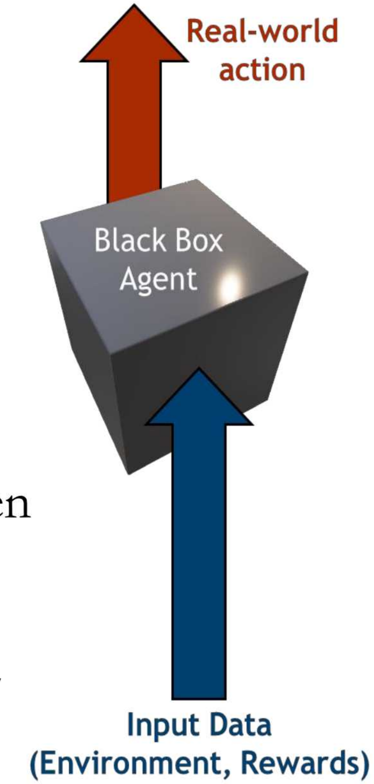
- Leverage existing knowledge (physical laws, power flow, etc) in AI-based algorithms
- Achieve more accurate results and faster training

- *SNL LDRD – "Integrating Physics Knowledge in Multi-Sensor Machine Learning Models"*

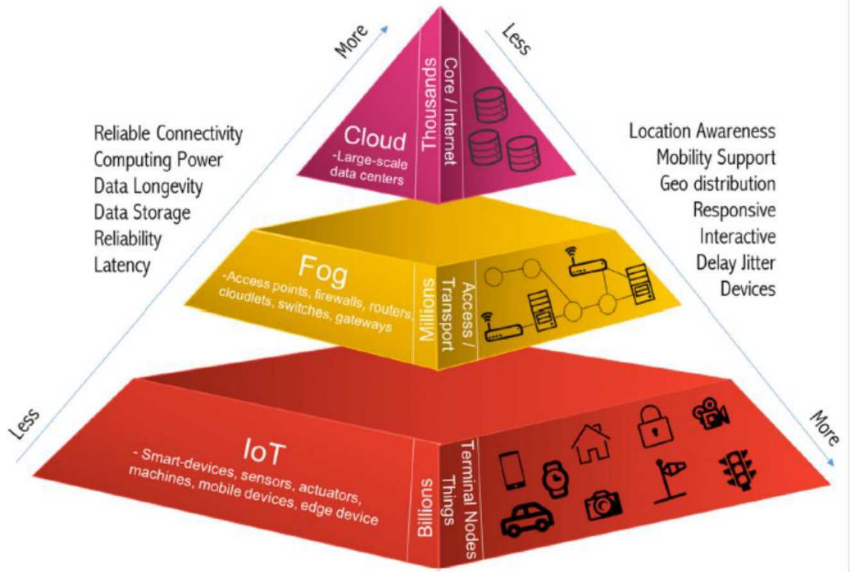
Explainable AI and Uncertainty Quantification

- Understand why a particular prediction/decision was given
- Understand the error bounds on predictions/decisions

- *SNL LDRD on "Opening the 'Black Box': An Experimentally-Validated Explainable Machine Learning Framework"*



Future Research Directions



A. Yousefpour *et al.*, "All One Needs to Know About Fog Computing and Related Edge Computing Paradigms: A Complete Survey," *J. Syst. Archit.*, vol. 98, pp. 289–330, Sep. 2019.

Distributed, AI-based Controls using Fog Computing

- Create resilient systems in the event of communication loss
- Accelerate systems with low latency because processing happens physically close to sensors

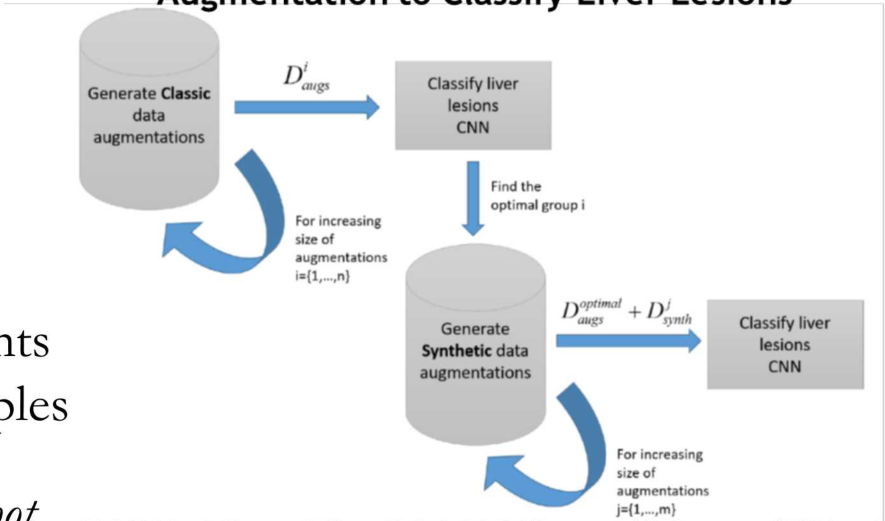
- *SNL LDRD – "HEDGES: High-Security Edge Computing for Smart Sensor Systems"*

Semi-Supervised, Few-Shot Learning, or Synthetically-Generated Training Data

- Learn with few or no examples of critical events
- Generate realistic new data from existing samples

- *SNL LDRD – "Semi-Supervised Bayesian Low-Shot Learning for Explosive Device Characterization"*

Generative Adversarial Networks (GAN) for Data Augmentation to Classify Liver Lesions



M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "GAN-based Synthetic Medical Image Augmentation for Increased CNN Performance in Liver Lesion Classification," *Neurocomputing*, vol. 321, pp. 321–331, 2018

References

1. A. Mosavi, M. Salimi, S. F. Ardabili, T. Rabczuk, S. Shamshirband, and A. R. Varkonyi-Koczy, "State of the Art of Machine Learning Models in Energy Systems, a Systematic Review," *Energies*, vol. 12, no. 7, Apr. 2019.
2. R. A. Sevlian *et al.*, "VADER: Visualization and Analytics for Distributed Energy Resources," *ArXiv 170809473 Cs CY*, 2017.
3. M. Lave, M. J. Reno, R. J. Broderick, and J. Peppanen, "Full-Scale Demonstration of Distribution System Parameter Estimation to Improve Low-Voltage Circuit Models," presented at the IEEE Photovoltaic Specialists Conference (PVSC), 2017.
4. L. Blakely, M. J. Reno, and W. Feng, "Spectral Clustering for Customer Phase Identification Using AMI Voltage Timeseries," *Power Energy Conf. Ill. PECI*, Feb. 2019.
5. K. Ashok, M. J. Reno, D. Divan, and L. Blakely, "Systematic Study of Data Requirements and AMI Capabilities for Smart Meter Connectivity Analytics," *Smart Energy Grid Eng. SEGE*, 2019
6. B. Foggo and N. Yu, "A Comprehensive Evaluation of Supervised Machine Learning for the Phase Identification Problem," *World Acad. Sci. Eng. Technol. Int. J. Comput. Syst. Eng.*, vol. 12, no. 6, 2018.
7. A. Karpatne *et al.*, "Theory-guided Data Science: A New Paradigm for Scientific Discovery from Data," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 10, pp. 2318–2331, 2017.
8. M. Lutter, C. Ritter, and J. Peters, "Deep Lagrangian Networks: Using Physics as Model Prior for Deep Learning," *Int. Conf. Learn. Represent. ICLR*, 2019.
9. L. Wu *et al.*, "Evaluating Machine Learning for Improving Power Grid Reliability," *ICML Workshop Mach. Learn. Gobal Chall.*, 2011.
10. A. C. Adewole, R. Tzoneva, and S. Behardien, "Distribution Network Fault Section Identification and Fault Location Using Wavelet Entropy and Neural Networks," *Appl. Soft Comput.*, vol. 46, pp. 296–306, 2016.
11. A. Prasad, J. B. Edward, and K. Ravi, "A Review of Fault Classification Methodologies in Power Transmission Systems: Part-I," *J. Electr. Syst. Informaton Technol.*, vol. 5, no. 1, pp. 48–60, May 2018.
12. C. Rosenberg, M. Hebert, and H. Schneiderman, "Semi-Supervised Self-Training of Object Detection Models," *Seventh IEEE Workshop Appl. Comput. Vis. WACVMOTION*, vol. 1, 2005.
13. A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap, "One-shot Learning with Memory-Augmented Neural Networks," *Proceeedings 33rd Int. Conf. Mach. Learn.*, 2016.
14. I. J. Goodfellow and et al., "Generative Adversarial Networks," *Adv. Neural Inf. Process. Syst.*, 2014.
15. M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "GAN-based Synthetic Medical Image Augmentation for Increased CNN Performance in Liver Lesion Classification," *Neurocomputing*, vol. 321, pp. 321–331, 2018
16. Z. Bie, Y. Lin, G. Li, and Li Furong, "Battling the Extreme: A Study on the Power System Resilience," *Proc. IEEE*, vol. 105, no. 7, pp. 1253–1266.
17. S. Zarrabian, R. Belkacemi, and A. A. Babalola, "Reinforcement Learning Approach for Congestion Management and Cascading Failure Prevention with Experimental Application," Electr. Power Syst. Res., vol. 141, pp. 179–190, 2016.
18. M. Glavic, R. Fonteneau, and D. Ernst, "Reinforcement Learning for Electric Power System Decision and Control: Past Considerations and Perspectives," Sci. Direct, vol. 50, no. 1, pp. 6918–6927, Jul. 2017.
19. Q. Huang, R. Huang, W. Hao, J. Tan, R. Fan, and H. Zhenyu, "Adaptive Power System Emergency Control Using Deep Reinforcement Learning," IEEE Transacation Smart Grid, 2019.
20. D. Ernst, M. Glavic, and L. Wehenkel, "Power Systems Stability Control: Reinforcement Learning Framework," IEEE Trans. Power Syst., vol. 19, no. 1, Feb. 2004.
21. A. Mosavi, M. Salimi, S. F. Ardabili, T. Rabczuk, S. Shamshirband, and A. R. Varkonyi-Koczy, "State of the Art of Machine Learning Models in Energy Systems, a Systematic Review," Energies, vol. 12, no. 7, Apr. 2019.
22. A. Yousefpour *et al.*, "All One Needs to Know About Fog Computing and Related Edge Computing Paradigms: A Complete Survey," *J. Syst. Archit.*, vol. 98, pp. 289–330, Sep. 2019.

Conclusions

There are many promising applications of AI/ML in power systems.

- It is an exciting time to be at this intersection – new algorithms, large datasets, computing power

There are many challenging problems yet to be solved with some fascinating future research directions in ML:

- Integration of Physics-based Constraints into AI
- Explainable AI and Uncertainty Quantification
- Distributed AI-based Controls using Fog Computing
- Semi-supervised, Few-shot learning, or Synthetically Generated Training Data

Best results require integration between ML experts and power system experts

See the included references for further reading.

Contact Information

Electric Power Systems Research

Ray Byrne
Manager, R&D Science and Engineering
rhbyrne@sandia.gov

Matthew Reno
R&D S&E, Electrical Engineering
mjreno@sandia.gov

Logan Blakely
R&D S&E, Computer Science
lblakel@sandia.gov

Machine Intelligence

David Stracuzzi
R&D S&E, Cognitive Systems
djstrac@sandia.gov

Scalable Analysis & Visualization

Warren Davis
R&D S&E, Computer Science
wldavis@sandia.gov

Data Science & Optimization

JP Watson
R&D S&E, Computer Science
jwatson@sandia.gov

Please feel free to contact us with any questions or follow-up discussions