# Applying artificial intelligence techniques for cyber defense

A. Ravishankar Rao
Ph.D
IEEE Fellow

AI and Cybersecurity
Organized by
Dr. Maksim Iavich and the team
Caucasus University and
Scientific Cyber Security Association


# Part 1: Background and motivation

The New York Times

# A.I. Is Not Sentient. Why Do People Say It Is?

Robots can't think or feel, despite what the researchers who build them want to believe.

Aug 5, 2022

Desdemona, a robot designed and built by Ben Goertzel, performs in Mr. Goertzel's band. He is the head of SingularityNET.

- Blake Lemoine, AI Researcher was fired by Google for claiming AI is sentient.
- There is no evidence this technology is sentient or conscious — two words that describe an awareness of the surrounding world.
- The problem is that the people closest to the technology — the people explaining it to the public — live with one foot in the future. They sometimes see what they believe will happen as much as they see what is happening now.

https://www.nytimes.com/2022/08/05/technology/ai-sentient-google.html

# Hype about AI

https://www.thelocal.dk/20220804/danish-ai-driven-political-party-eyes-parliament/

POLITICS

## Could next party in Danish parliament be led by AI?

**A new political party in Denmark whose policies are derived entirely from artificial intelligence (AI) hopes to stand in the country's next general election in June 2023.**

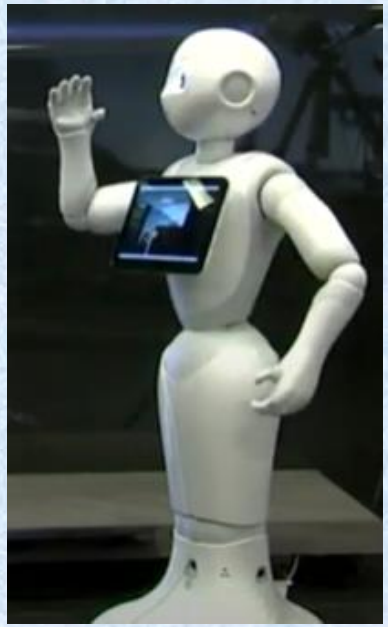
Published: 4 August 2022 14:50 CEST

By analysing all of Denmark's fringe parties' written publications since 1970, the Synthetic Party's AI has devised a programme that it believes represents "the political visions of the everyday person", one of the members of the collective, Asker Bryld Staunaes, told AFP.

# Interesting applications of current AI

**All You Need To Know About Tokyo's Robot Restaurant**

Klook Team
Last updated 14 Mar 2020





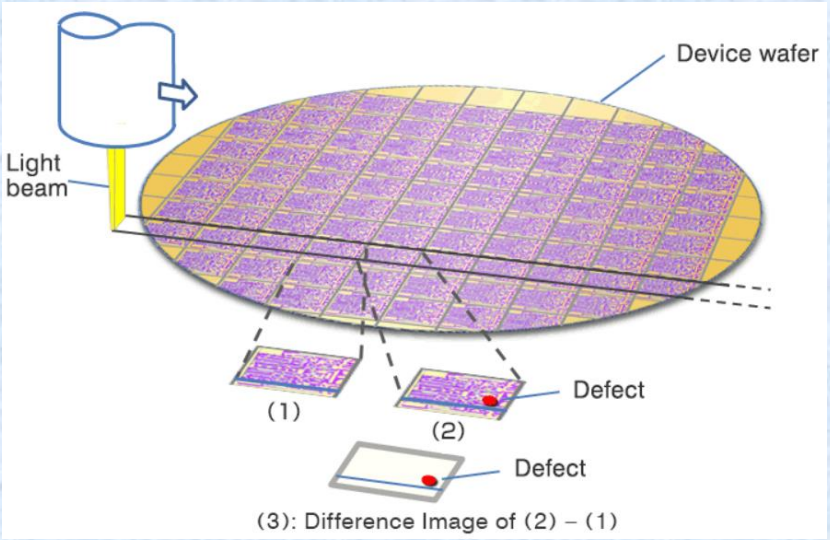
ჩემთვის სასიამოვნოა ამ ლექციის წაკითხვა

It is a pleasure for me to give this lecture →ჩემთვის სასიამოვნოა ამ ლექციის წაკითხვა

# My personal work in the field

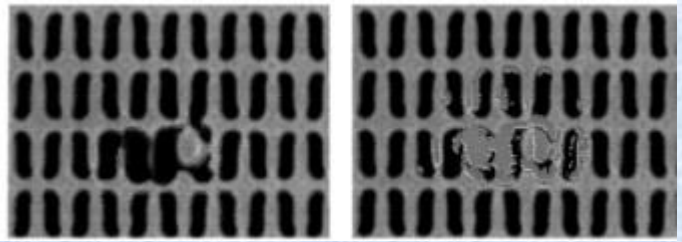
IBM Research Headquarters

Semiconductor manufacturing clean room




Defect detection

Machine Vision and Applications
© Springer-Verlag 1997



**Automatic defect classification for semiconductor manufacturing**

**Paul B. Chou, A. Ravishankar Rao, Martin C. Sturzenbecker, Frederick Y. Wu, Virginia H. Brecher**

I.B.M., T. J. Watson Research Center, Yorktown Heights, NY 10598, USA

Defect classification

https://www.hitachi-hightech.com/global/products/device/semiconductor/inspection.html

# My personal work in the field

IBM Infoprint 2000 series,
110 pages per minute


Need to remove Moire patterns
https://www.scantips.com/basics06.html

- Image segmentation and descreening solutions
- Used for printing books on demand
- Customers like Lightning Print (a division of Amazon)

**Segmentation and Automatic Descreening of Scanned Documents**

Alejandro Jaimes[a], Frederick Mintzer[b], A. Ravishankar Rao[b] and Gerhard Thompson[b]

[a]Columbia University
Department of Electrical Engineering
New York, NY 10027

[b]IBM T.J. Watson Research Center
Yorktown Heights, NY 10598

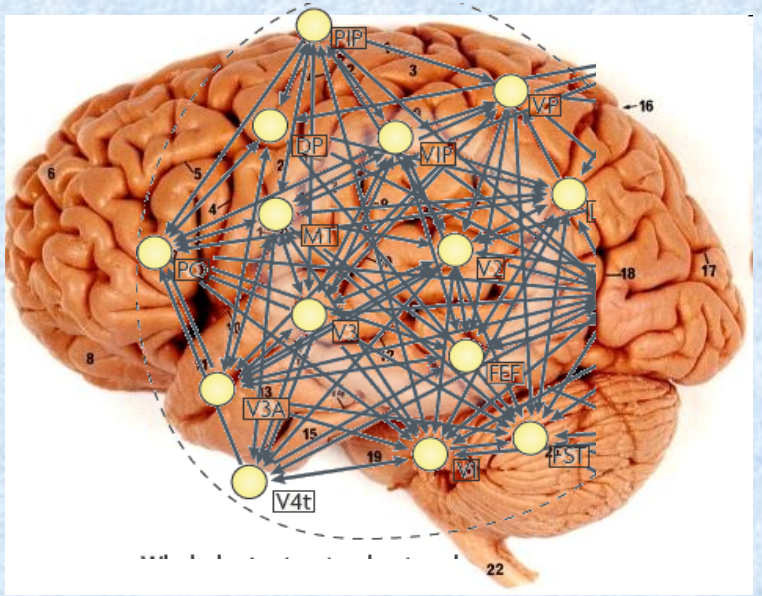
frontiers in
SYSTEMS NEUROSCIENCE

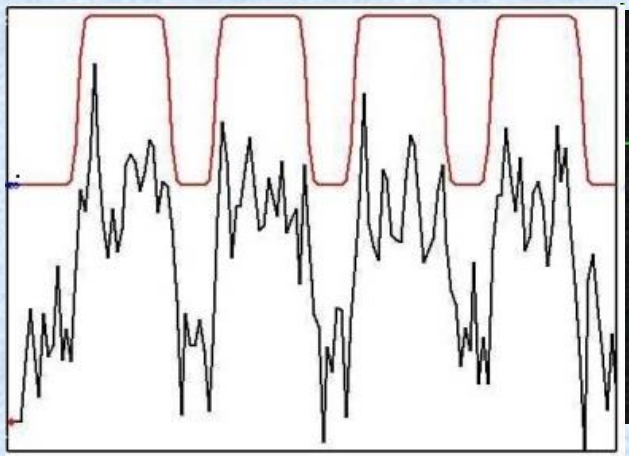
ORIGINAL RESEARCH ARTICLE
published: 16 November 2012
doi: 10.3389/fnsys.2012.00075

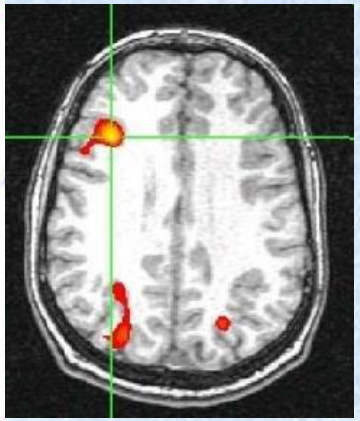
# Exploiting the brain's network structure in identifying ADHD subjects

***Soumyabrata Dey[1]*, A. Ravishankar Rao[2] and Mubarak Shah[1]***

[1] *Computer Vision Lab, Department of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL, USA*
[2] *IBM T.J. Watson Research Center, Yorktown Heights, NY, USA*





# Current work: Embedded systems, IoT, low-cost solutions

## Developing surveillance applications with Raspberry Pi, Django, and cloud services

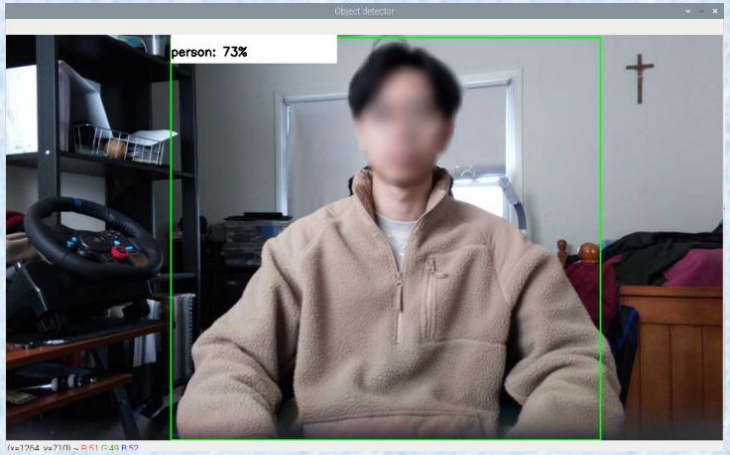
A. Ravishankar Rao, PhD,
Fellow, IEEE
Fairleigh Dickinson University, NJ, USA
raviraodr@gmail.com

Brennan Gebusion
Fairleigh Dickinson University, NJ, USA
brengeb314@gmail.com

Jared Porpora
Fairleigh Dickinson University, NJ, USA
porporajared.jp@gmail.com

IEEE STEM Education Conference, 2022


Face Recognition




Cheaper than Google Nest

# Application of AI in Education: Automated grading and large courses


THE WALL STREET JOURNAL.

English Edition ▾ Print Edition | Video | Podcasts | Latest Headlines

Home World U.S. Politics Economy Business Tech Markets Opinion Books & Arts Real Estate Life & Work Style Sports

A-HED

**Imagine Discovering That Your Teaching Assistant Really Is a Robot**

May 2016

- 300 students in a class at Georgia Tech on Knowledge-based AI
- Question answering robot used


WSJ | OPINION

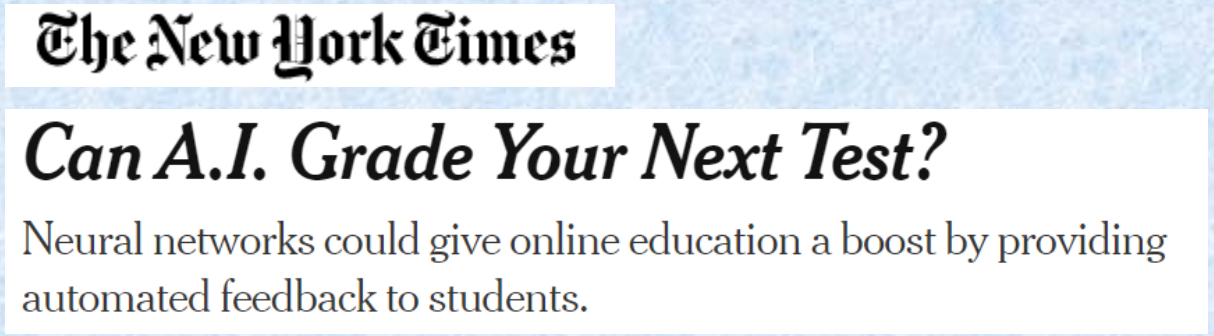
OPINION | THE WEEKEND INTERVIEW

*The Man Who Made Online College Work*

Years before Covid, Zvi Galil launched Georgia Tech's successful online master's in computer science. Is Zoom U. the future?

April 2021

- $7000 for online Masters degree in CS at Georgia Tech
- 11,000 students enrolled


The New York Times

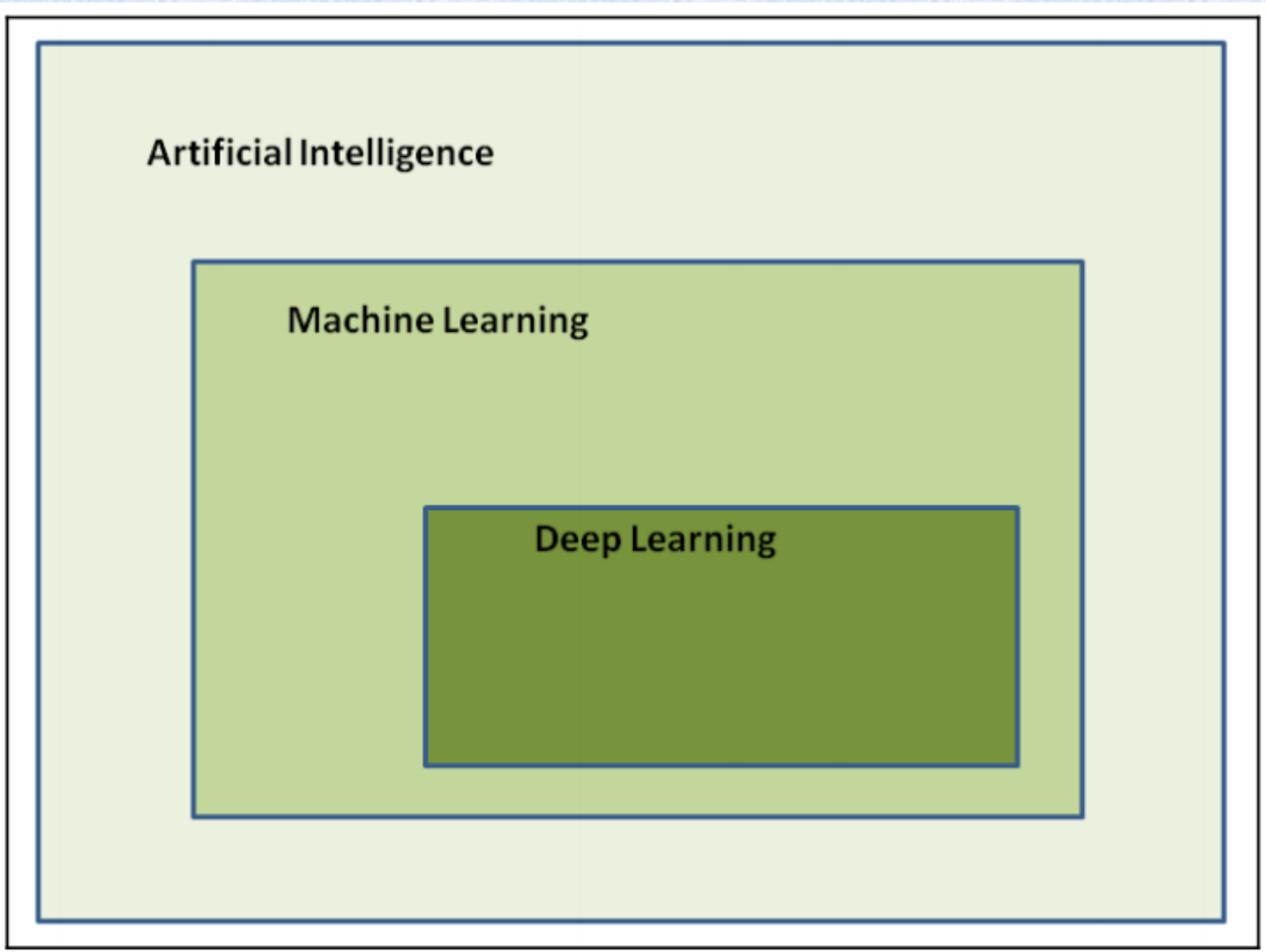
*Can A.I. Grade Your Next Test?*

Neural networks could give online education a boost by providing automated feedback to students.

July 20, 2021

- 12000 students in Stanford University course "Code in Place"
- Feedback produced by AI software

# Part 2:How does the technology work? A brief introduction to AI and machine learning




Hands on artificial intelligence for cybersecurity, Packt Publishing, Parisi

# Three basic types of techniques

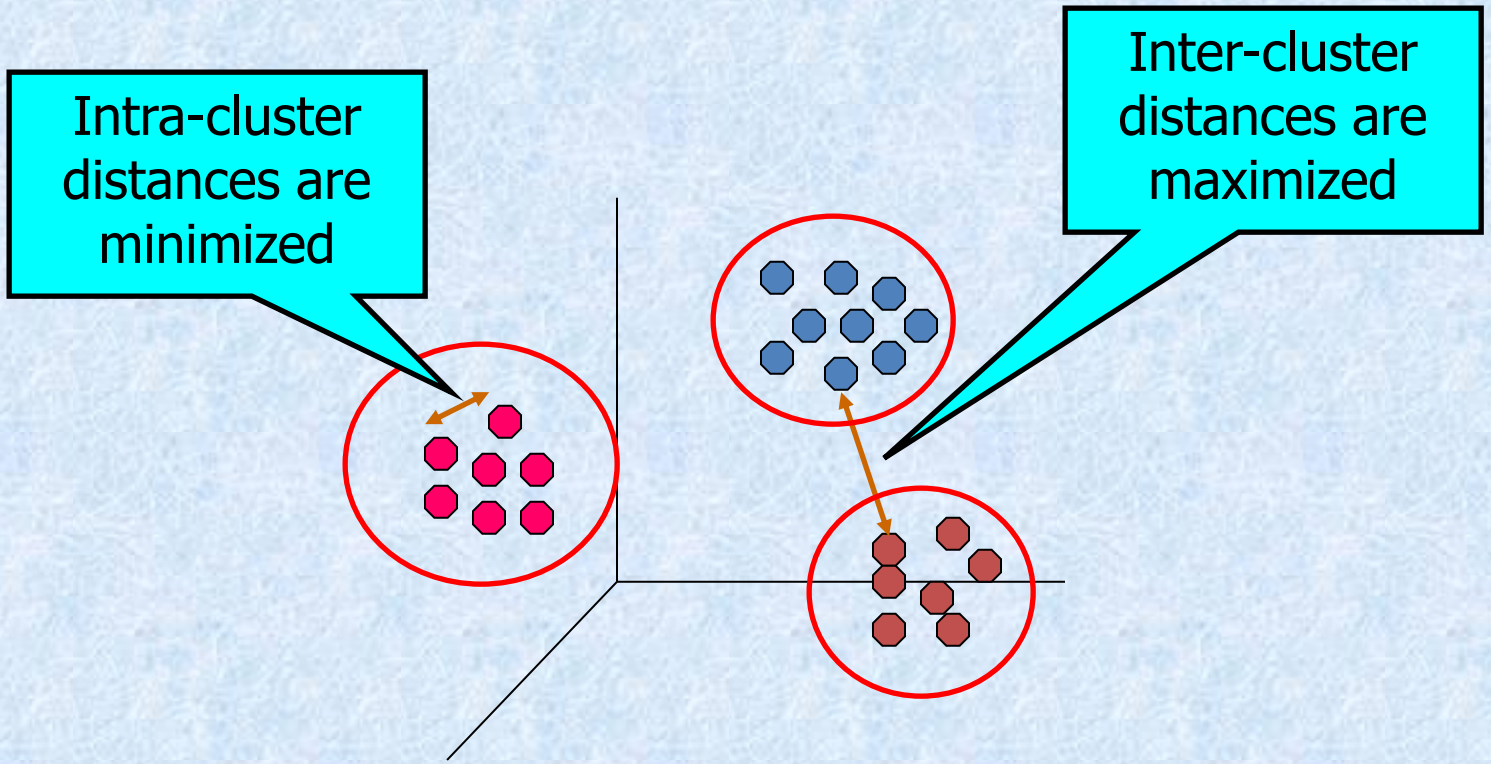
- Unsupervised learning (there is no teacher)
- Supervised learning (use labeled training examples)
- Reinforcement learning (use a reward function)

We will cover the following methods:

- Unsupervised learning: K-means clustering
- Supervised learning: Decision trees, Bayesian methods, neural networks

# What is Cluster Analysis?

- Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups

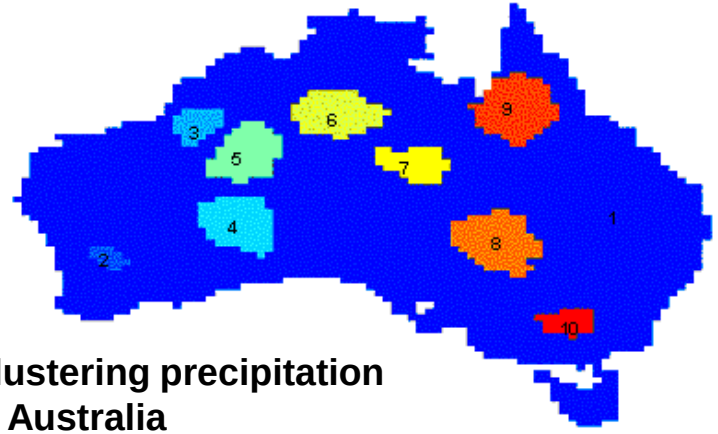


# Applications of Cluster Analysis

- **Understanding**
  - Group stocks with similar price fluctuations

- **Summarization**
  - Reduce the size of large data sets

| | *Discovered Clusters* | *Industry Group* |
|---|---|---|
| 1 | Applied-Matl-DOWN,Bay-Network-Down,3-COM-DOWN, Cabletron-Sys-DOWN,CISCO-DOWN,HP-DOWN, DSC-Comm-DOWN,INTEL-DOWN,LSI-Logic-DOWN, Micron-Tech-DOWN,Texas-Inst-Down,Tellabs-Inc-Down, Natl-Semiconduct-DOWN,Oracl-DOWN,SGI-DOWN, Sun-DOWN | Technology1-DOWN |
| 2 | Apple-Comp-DOWN,Autodesk-DOWN,DEC-DOWN, ADV-Micro-Device-DOWN,Andrew-Corp-DOWN, Computer-Assoc-DOWN,Circuit-City-DOWN, Compaq-DOWN, EMC-Corp-DOWN, Gen-Inst-DOWN, Motorola-DOWN,Microsoft-DOWN,Scientific-Atl-DOWN | Technology2-DOWN |
| 3 | Fannie-Mae-DOWN,Fed-Home-Loan-DOWN, MBNA-Corp-DOWN,Morgan-Stanley-DOWN | Financial-DOWN |
| 4 | Baker-Hughes-UP,Dresser-Inds-UP,Halliburton-HLD-UP, Louisiana-Land-UP,Phillips-Petro-UP,Unocal-UP, Schlumberger-UP | Oil-UP |




**Clustering precipitation in Australia**

# Notion of a Cluster can be Ambiguous

How many clusters?

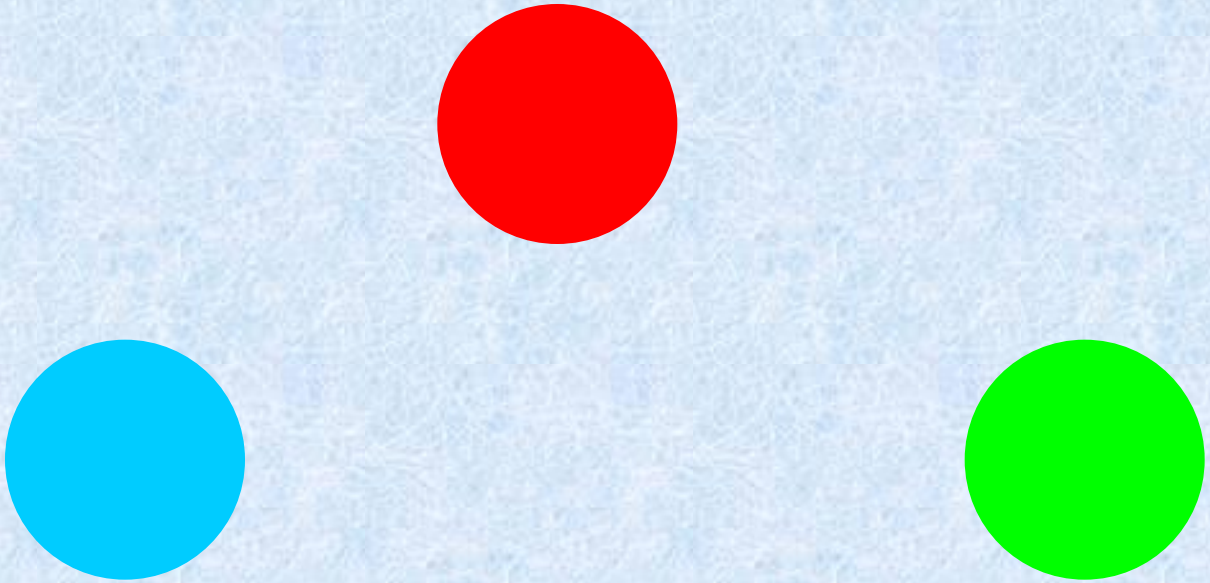
Six Clusters

Two Clusters

Four Clusters

# Types of Clusters: Well-Separated

- Well-Separated Clusters:
  - A cluster is a set of points such that any point in a cluster is closer (or more similar) to every other point in the cluster than to any point not in the cluster.



**3 well-separated clusters**

# K-means Clustering – Details

- Initial centroids are often chosen randomly.
  - Clusters produced vary from one run to another.
- The centroid is (typically) the mean of the points in the cluster.
- ‘Closeness’ is measured by Euclidean distance, cosine similarity, correlation, etc.
- K-means will converge for common similarity measures mentioned above.
- Most of the convergence happens in the first few iterations.
  - Often the stopping condition is changed to ‘Until relatively few points change clusters’
- Complexity is O( n * K * I * d )
  - n = number of points, K = number of clusters,
    I = number of iterations, d = number of attributes

# Simple 1-D example

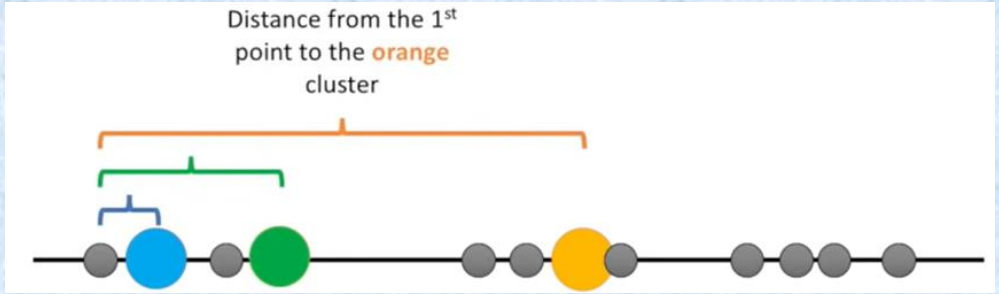
Start with the input data. We want to assign these points to different clusters.

Input

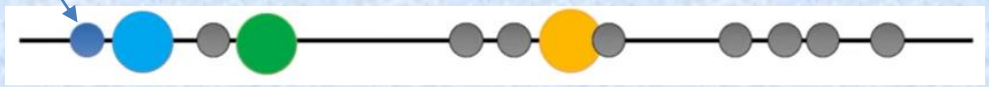
Desired answer

Step 1: select k=3,and choose 3 random points as cluster “centroids”

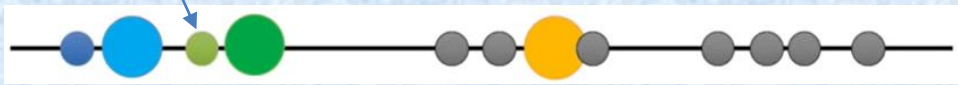
Step 2: measure distance of the first point to each cluster centroid




Step 3: Assign first point to the nearest cluster, ie ‘Blue’



Step 4: Assign second point to the nearest cluster, ie ‘Green’



Step 5: Continue assigning all the points

Step 6: Recalculate centroids

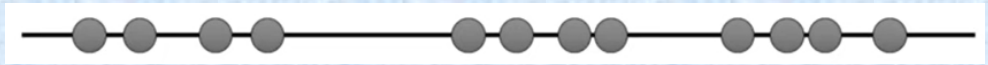
Step 7: Repeat until there is no change

Statquest k-means clustering: https://www.youtube.com/watch?v=4b5d3muPQmA

# Simple 1-D example

Start with the input data. We want to assign these points to different clusters.

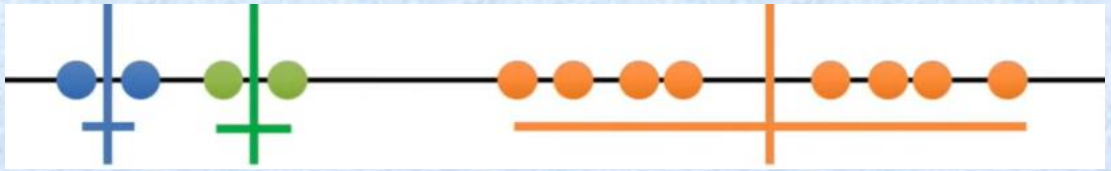
Input



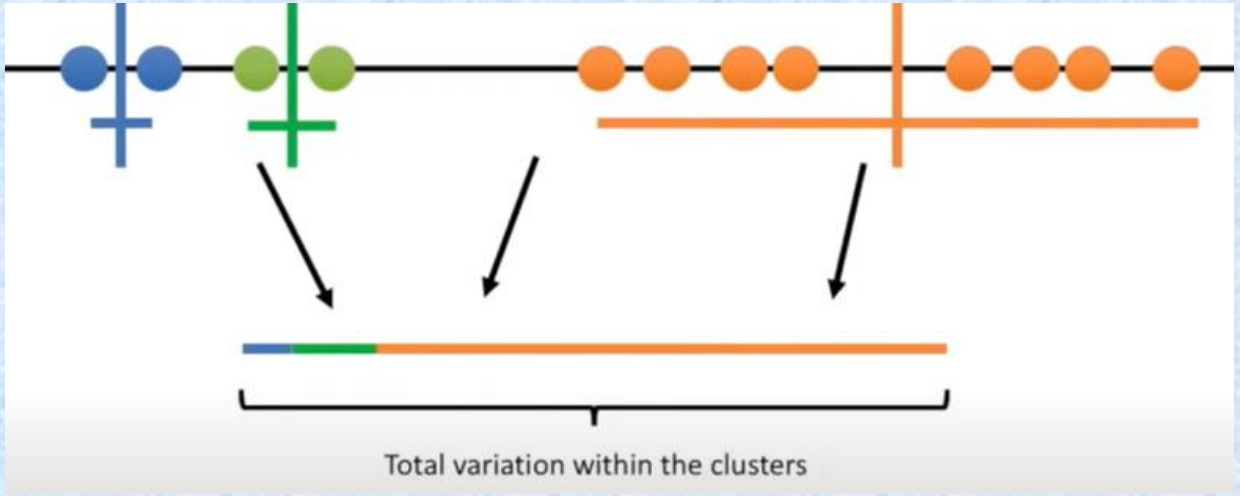
Desired answer

Actual answer

Quantify the result: compute the variation with each cluster



Add all the variations




- Rerun the whole algorithm again with different guesses for the centroids

Statquest k-means clustering:
https://www.youtube.com/watch?v=4b5d3muPQmA

# Evaluating K-means Clusters

- Most common measure is Sum of Squared Error (SSE)
  - For each point, the error is the distance to the nearest cluster
  - To get SSE, we square these errors and sum them.

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$

  - (wikipedia)

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2$$

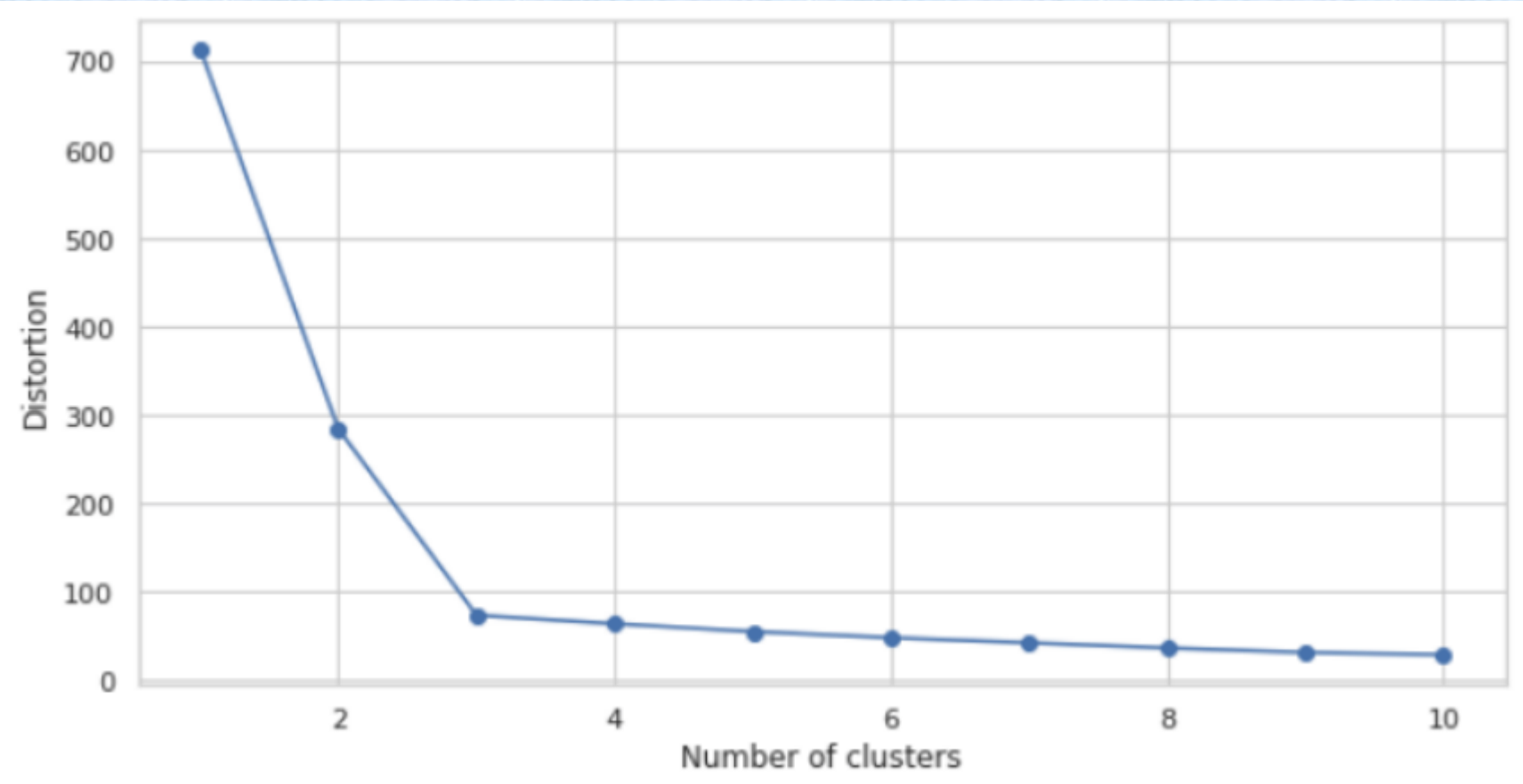
  - $x$ is a data point in cluster $C_i$ and $m_i$ is the representative point for cluster $C_i$
    - can show that $m_i$ corresponds to the center (mean) of the cluster
  - Given two clusters, we can choose the one with the smallest error
  - One easy way to reduce SSE is to increase K, the number of clusters
    - A good clustering with smaller K can have a lower SSE than a poor clustering with higher K

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

*class* `sklearn.cluster.`**`KMeans`**(*n_clusters=8, *, init='kmeans++', n_init=10, max_iter=300, tol=0.0001, verbose=0, random_state=None, copy_x=True, algorithm='lloyd'*)

- 'k-means++' : selects initial cluster centroids using sampling to ensure these centroids are far away from each other
- n_init: refers to the number of times you will re-run the k-means algorithm with different initial random centroids

How do we select k in the first place?
Use an elbow plot as shown




https://towardsdatascience.com/k-means-clustering-with-scikit-learn-6b47a369a83c

# Application to outlier detection

SN Computer Science (2021) 2:477
https://doi.org/10.1007/s42979-021-00871-7

Check for updates

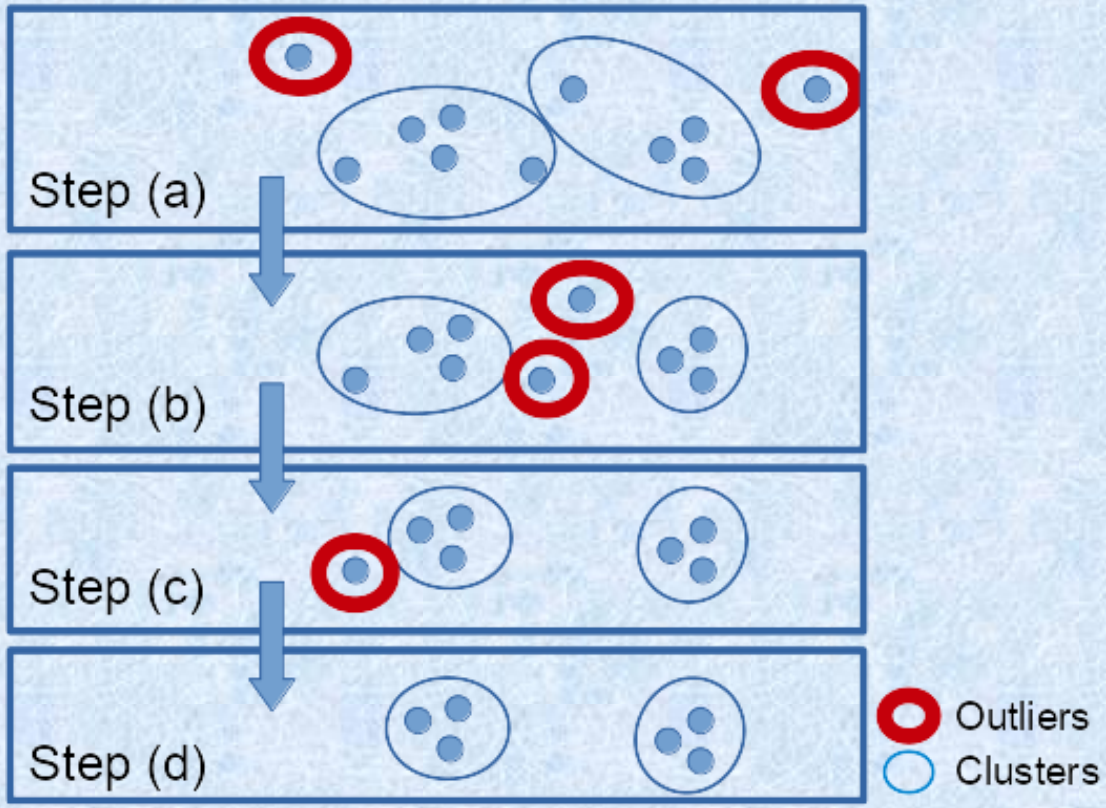
ORIGINAL RESEARCH

SN

**PIKS: A Technique to Identify Actionable Trends for Policy-Makers Through Open Healthcare Data**

A. Ravishankar Rao[1] · Subrata Garai[1,2] · Soumyabrata Dey[2] · Hang Peng[1,2]

Pruned Iterative k-means searchlight



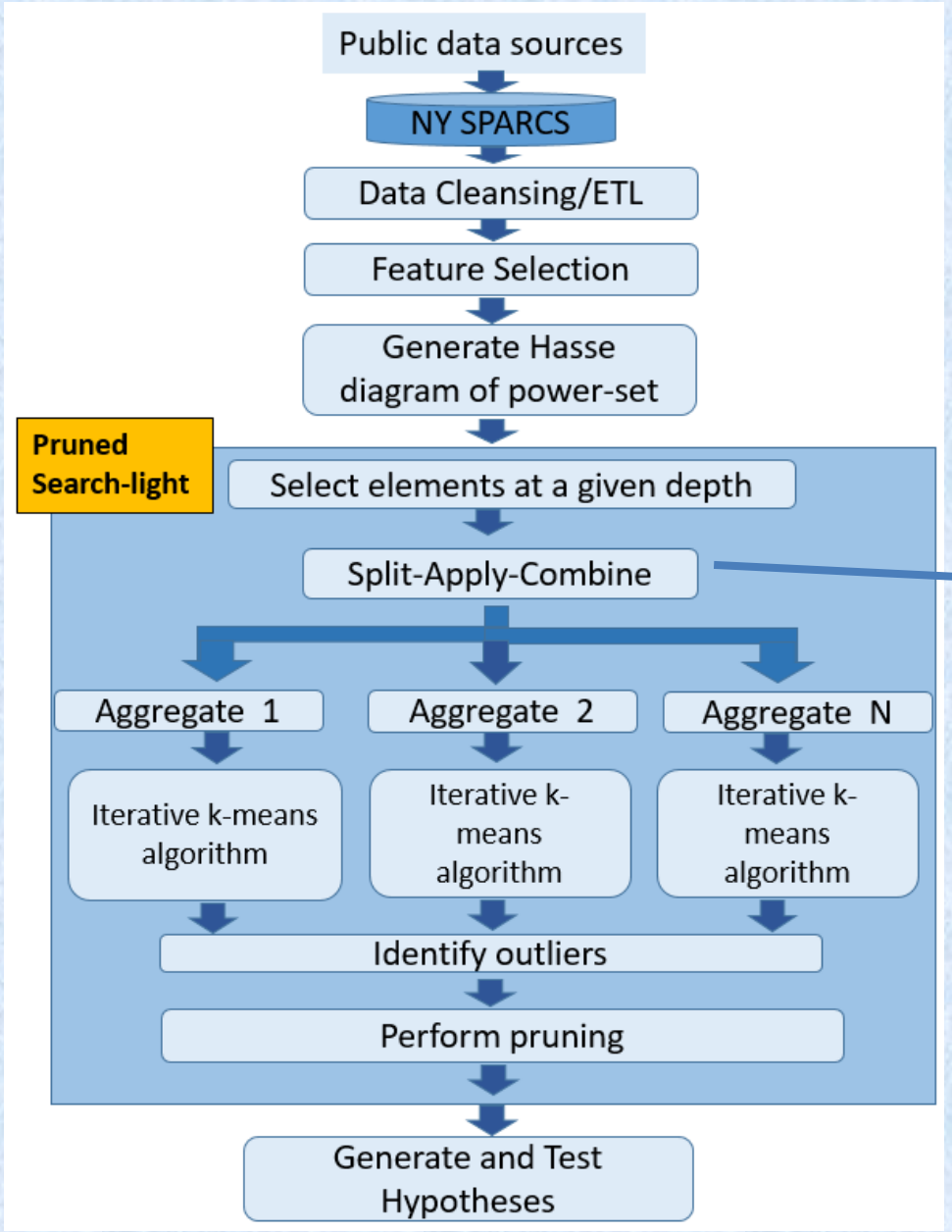

1)Apply the k-means clustering algorithm.
2)If single-element or substantially small clusters (e.g. size < 2 or 3) exist, these are treated as outliers. These outliers are removed and we continue with the rest of the data.
3)If there are no substantially small clusters, we terminate the iterative k-means algorithm.

# Data

| Facility Name | Age Group | CCS Diagnosis Des | APR DRG Description | Payment Typology 1 | Total Charges | Total Costs |
|---|---|---|---|---|---|---|
| NewYork-Presbyterian/Hudso | 50 to 69 | OSTEOARTHRITIS | HIP JOINT REPLACEMENT | Blue Cross/Blue Shield | $118,606.63 | $47,403.96 |
| NewYork-Presbyterian/Hudso | 50 to 69 | OSTEOARTHRITIS | HIP JOINT REPLACEMENT | Blue Cross/Blue Shield | $134,606.18 | $53,063.33 |
| NewYork-Presbyterian/Hudso | 50 to 69 | OSTEOARTHRITIS | HIP JOINT REPLACEMENT | Private Health Insurance | $129,815.21 | $51,773.89 |
| White Plains Hospital Center | 50 to 69 | OSTEOARTHRITIS | HIP JOINT REPLACEMENT | Medicaid | $45,407.00 | $23,703.32 |
| White Plains Hospital Center | 50 to 69 | OSTEOARTHRITIS | HIP JOINT REPLACEMENT | Medicare | $46,572.00 | $19,795.44 |
| White Plains Hospital Center | 50 to 69 | OSTEOARTHRITIS | HIP JOINT REPLACEMENT | Private Health Insurance | $30,774.00 | $14,475.87 |

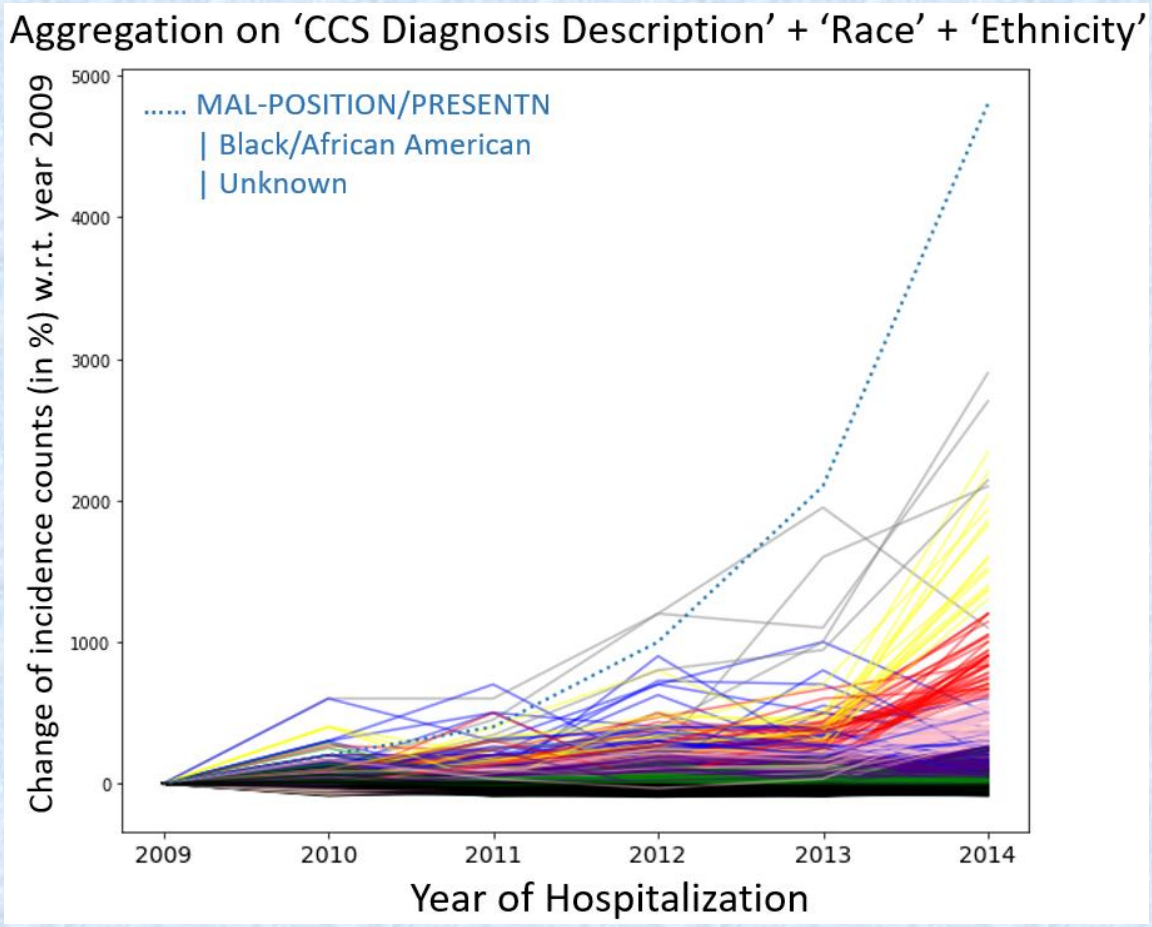
- 2 million patient records per year
- Available for 10 years
- Source: New York State SPARCS repository

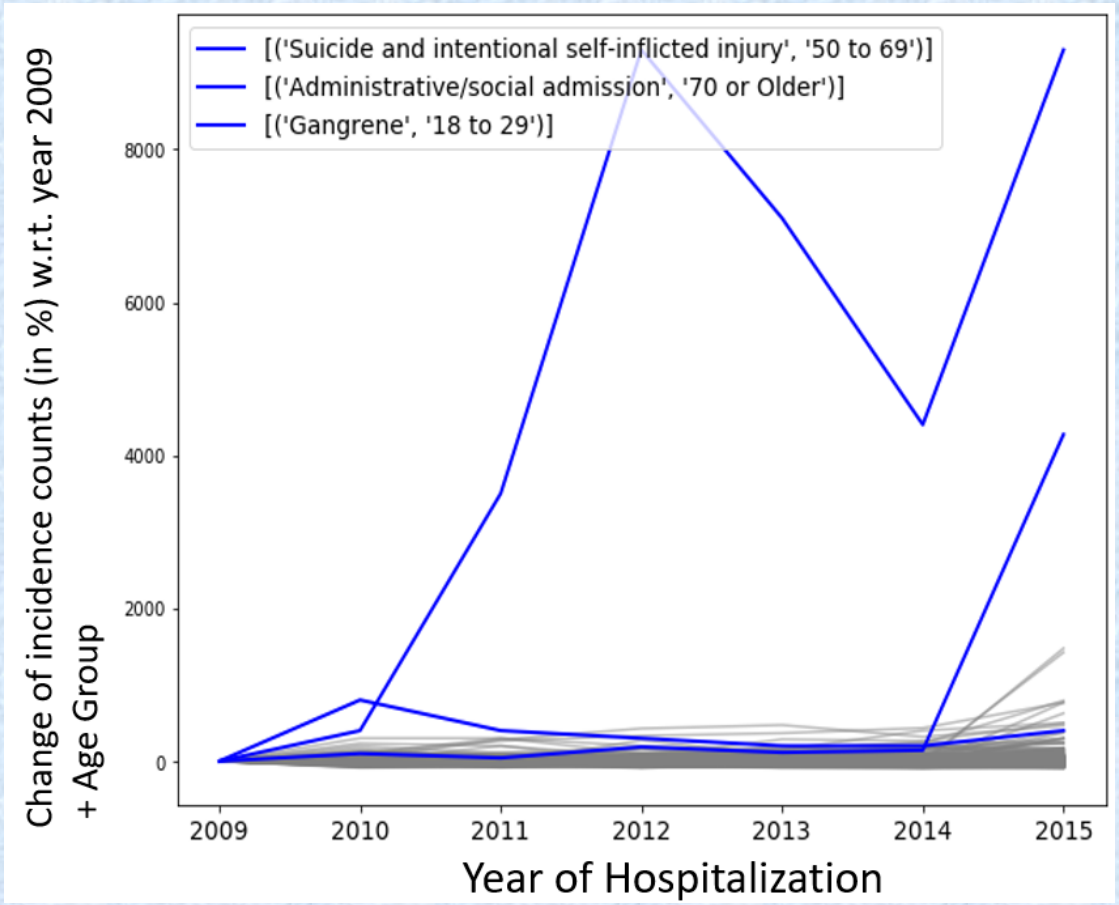
# Processing pipeline



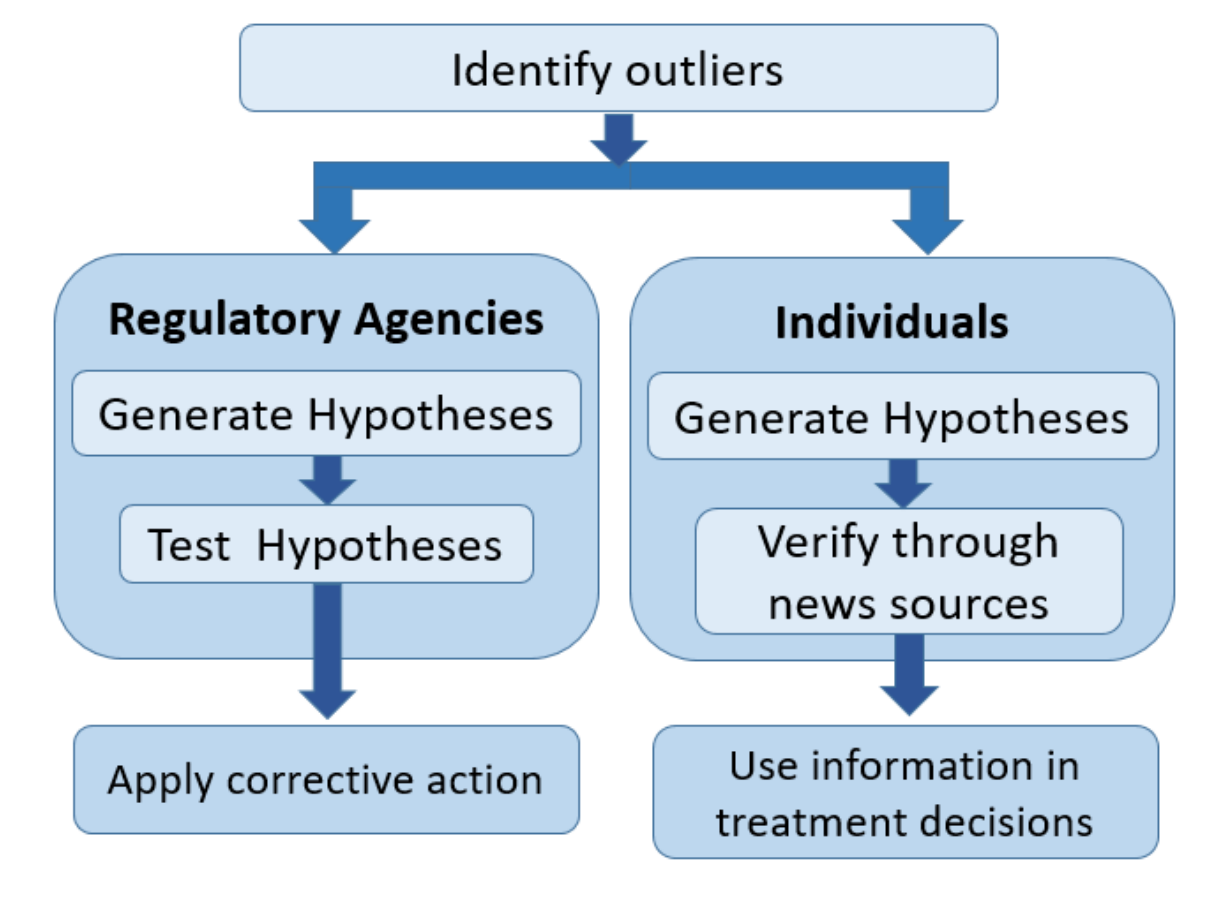

We address the high dimensionality of potential datasets by first applying the split-apply-combine paradigm from the data analysis literature

One or two slides from my own research



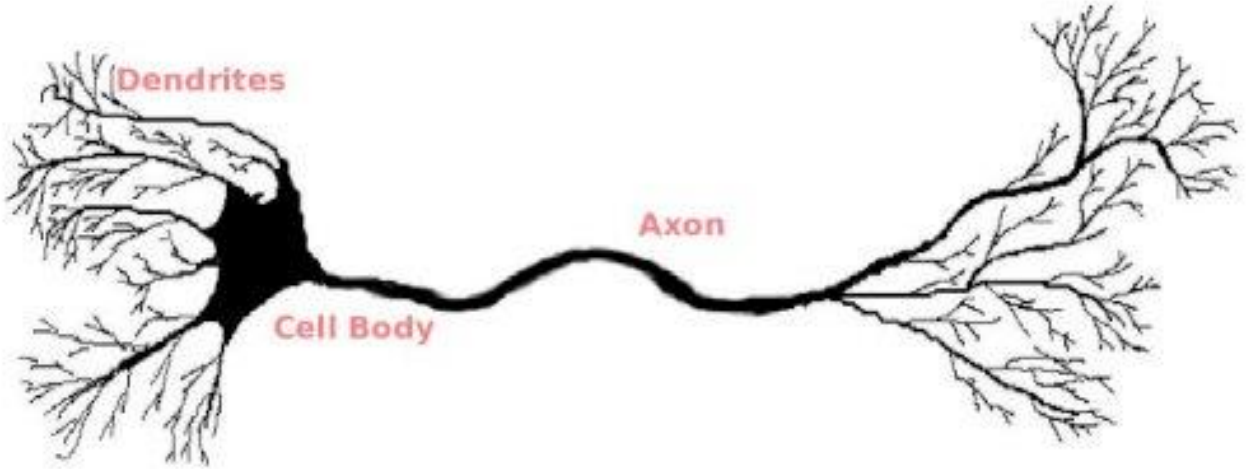



# Use case scenarios enabled by the outlier detection algorithm



# Neural Networks

- Complex learning systems recognized in animal brains
- Single neuron has simple structure
- Interconnected sets of neurons perform complex learning tasks
- Human brain has $10^{15}$ synaptic connections
- Artificial Neural Networks attempt to replicate non-linear learning found in nature—(artificial usually dropped)

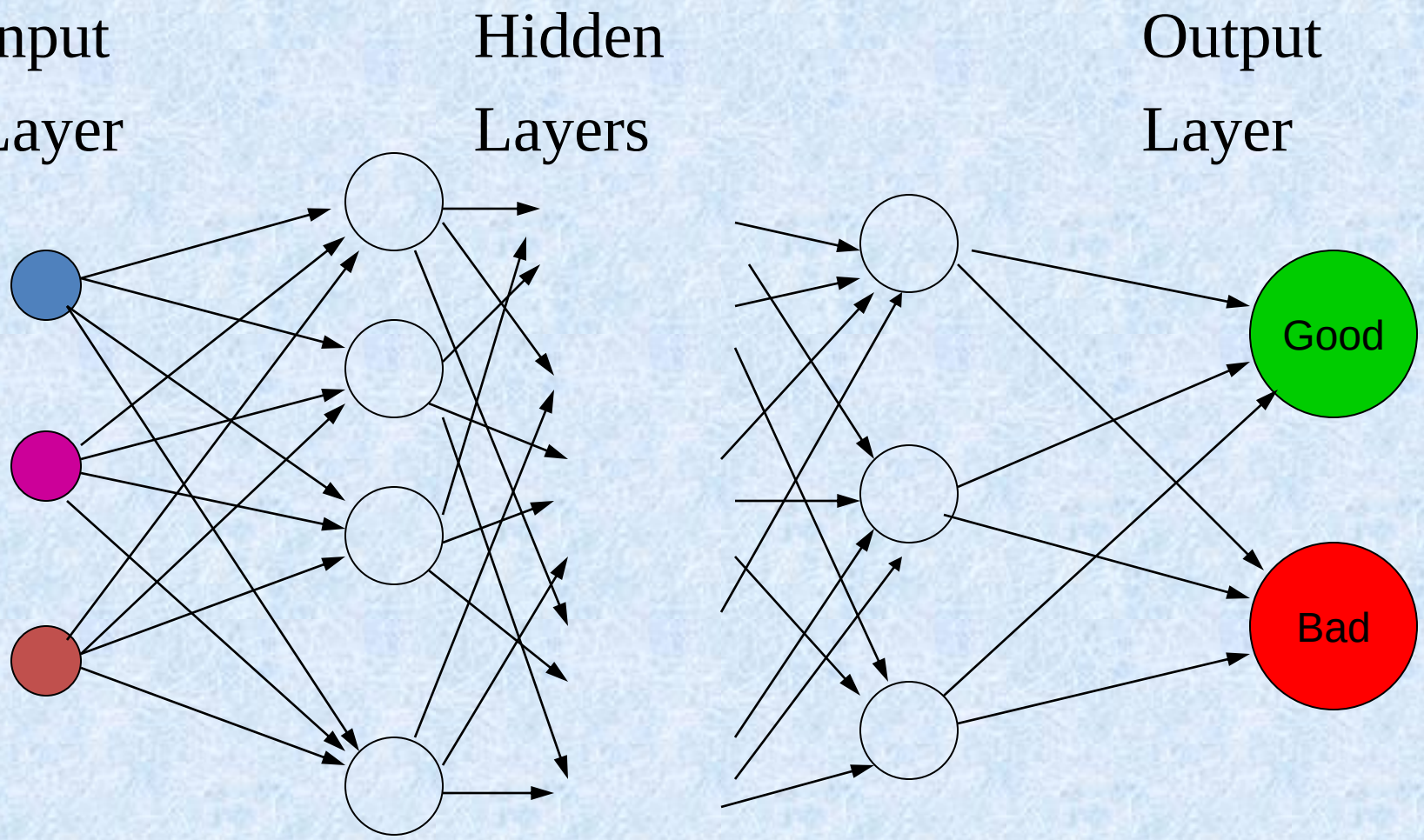



Adapted from Larose

# Neural Networks (cont)

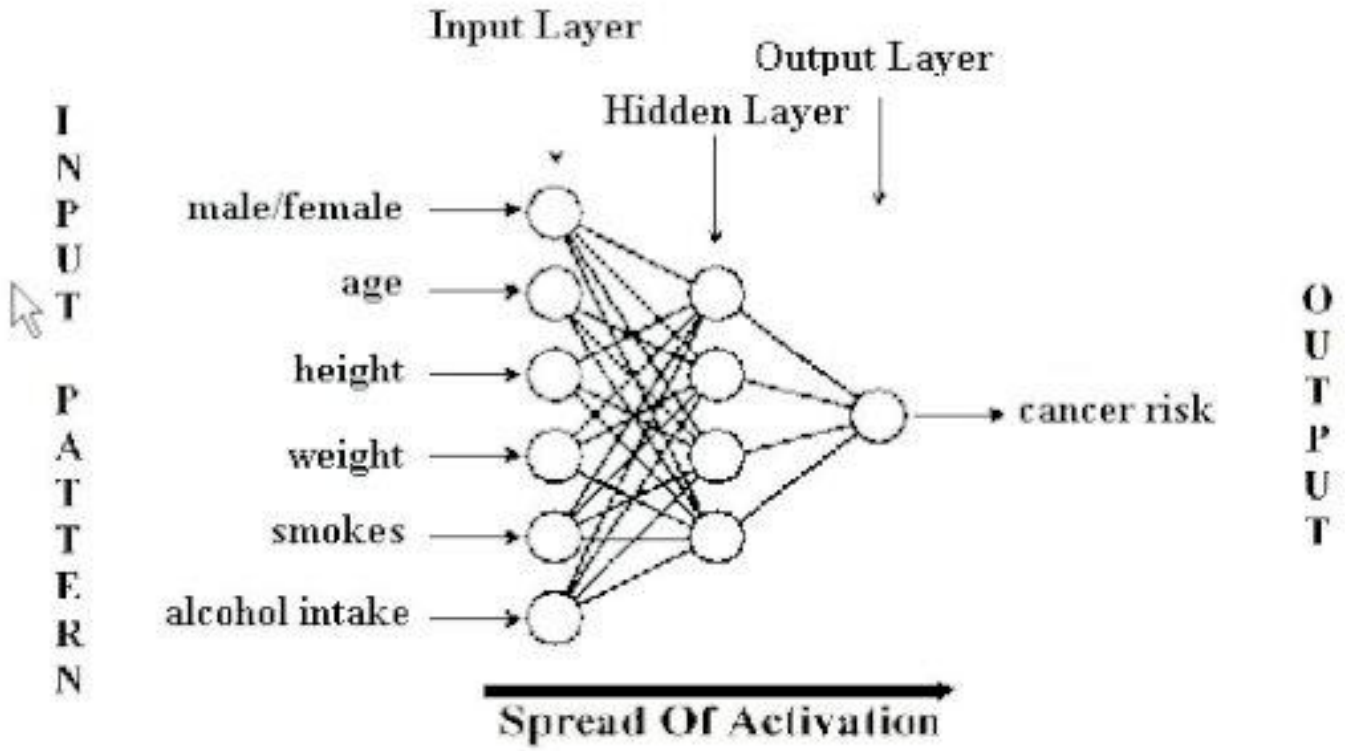
- Terms
  - Layers
    - Input, hidden, output
  - Feed forward
  - Fully connected
  - Back propagation
  - Learning rate
  - Momentum
  - Optimization / sub optimization

# Network



# Neural Networks (cont)

- Structure of a neural network



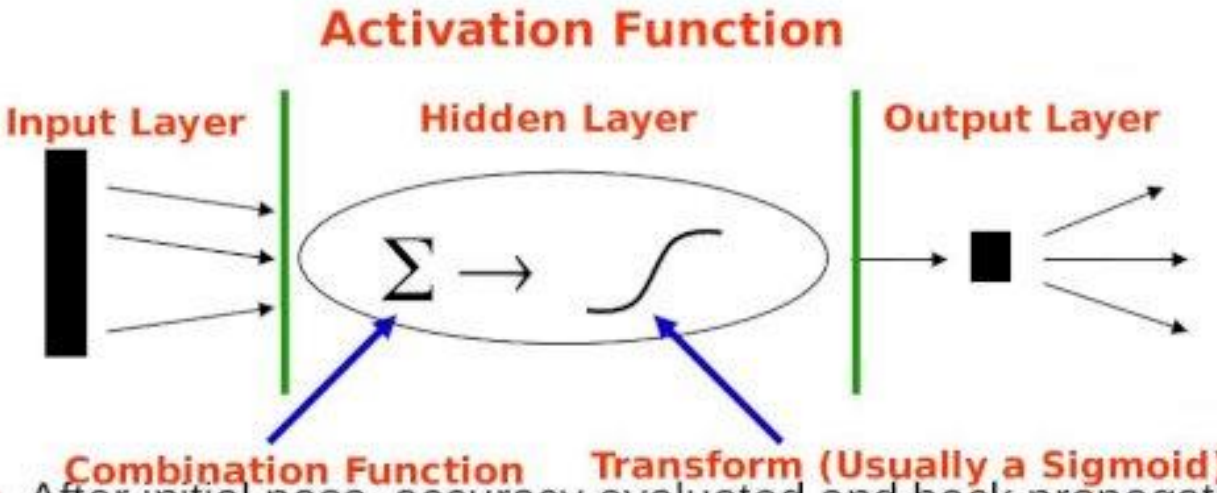

Adapted from Barry & Linoff

# Neural Network

- For classification neural network models, the output layer has on node for each classification category (true or false).
- Each node is connected by an arc to nodes in the next layer. These arcs have weights, which are multiplied by the value of incoming nodes and summed.
- Middle layer node values are the sum of incoming node values multiplied by the arc weights.
- ANN learn through feedback loops. Output is compared to target values, and the difference between attained and target output is fed back to the system to adjust the weights on arcs.
- Measure fit
  - fine tune around best fit

# Neural Networks (Cont)

- Inputs uses weights and a combination function to obtain a value for each neuron in the hidden layer
- Then a non-linear response is generated from each neuron in the hidden layer to the output




- After initial pass, accuracy evaluated and back propagation through the network changing weights for next pass
- Repeated until apparent answers (delta) are small—beware, this could be sub optimal solution

Adapted from Larose

# Neural Networks (Cont)

- Neural network algorithms require inputs to be within a small numeric range. This is easy to do for numeric variables using the min-max range approach as follows (values between 0 and 1)

$$X = \frac{\lfloor x - \min(x) \rfloor}{Range(x)}$$

- Other methods can be applied
- ***Neural Networks, as with Logistic Regression, do not handle missing values whereas Decision Trees do. Many data mining software packages automatically patches up for missing values but I recommend the modeler know the software is handling the missing values***

Adapted from Larose

# Neural Networks (Cont)

- **Categorical**
  - Indicator Variables (sometimes referred to as 1 of n) used when number of category values small
  - Categorical variable with *k* classes translated to *k* – 1 indicator variables
  - For example, *Gender* attribute values are "Male", "Female", and "Unknown"
  - Classes *k* = 3
  - Create *k* – 1 = 2 indicator variables named *Male_I* and *Female_I*
  - *Male* records have values *Male_I* = 1, *Female_I* = 0
  - *Female* records have values *Male_I* = 0, *Female_I* = 1
  - *Unknown* records have values *Male_I* = 0, *Female_I* = 0

Adapted from Larose

# Neural Networks (Cont)

- **Categorical**
  - Be very careful when working with categorical variables in neural networks when mapping the variables to numbers. The mapping introduces an ordering of the variables, which the neural network takes into account. 1 of n solves this problem but is cumbersome for a large number of categories.
  - Codes for marital status ("single," "divorced," "married," "widowed," and "unknown") could be coded
    - Single 0
    - Divorced .2
    - Married .4
    - Separated .6
    - Widowed .8
    - Unknown 1.0
    - Note the implied ordering

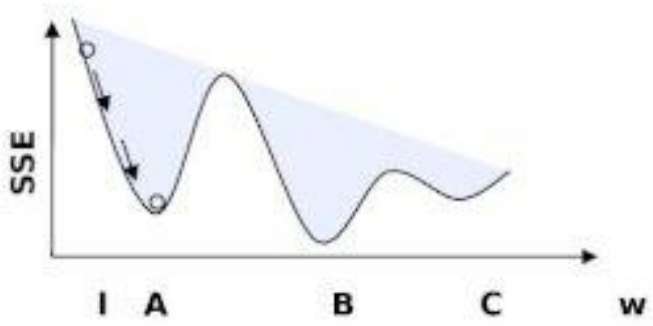
Adapted from Barry & Linoff

# Neural Networks (Cont)

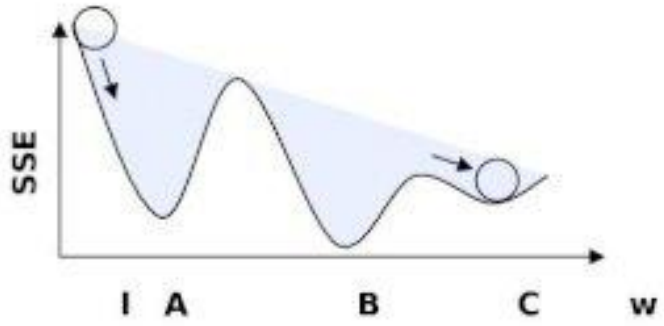
- **Data Mining Software**
  - Note that most modern data mining software takes care of these issues for you. But you need to be aware that it is happening and what default setting are being used.
  - For example, the following was taken from the PASW Modeler 13 Help topics describing binary set encoding—an advanced topic
  - **Use binary set encoding**. If this option is selected, a compressed binary encoding scheme for set fields is used. This option allows you to more easily build neural net models using set fields with large numbers of values as inputs. However, if you use this option, you may need to increase the complexity of the network architecture (by adding more hidden units or more hidden layers) to allow the network to properly use the compressed information in binary encoded set fields. Note: The simplemax and softmax scoring methods, SQL generation, and export to PMML are not supported for models that use binary set encoding

# Learning rate and Momentum

- The learning rate, eta, determines the magnitude of changes to the weights
- Momentum, alpha, is analogous to the mass of a rolling object as shown below. The mass of the smaller object may not have enough momentum to roll over the top to find the true optimum.



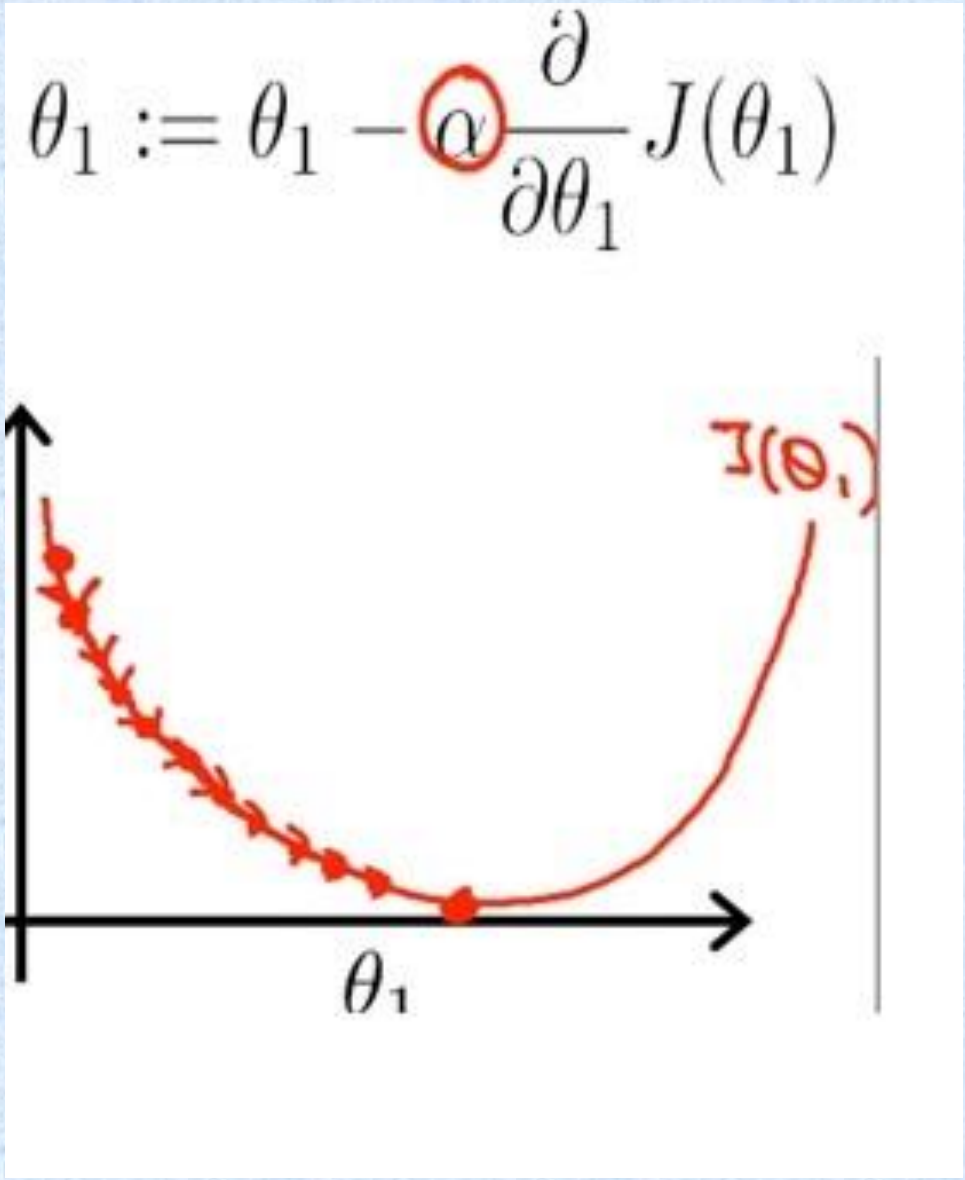

Small Momentum




Large Momentum

Adapted from Larose

Use gradient descent algorithms




https://upload.wikimedia.org/wikipedia/commons/6/68/Gradient_descent.jpg

# Neural Network Testing

- Usually train on part of available data
  - package tries weights until it successfully categorizes a selected proportion of the training data
- When trained, test model on part of data
  - if given proportion successfully categorized, quits
  - if not, works some more to get better fit
- The “model” is internal to the package
- Model can be applied to new data

# Neural Network Process

1. Collect data
2. Separate into training, test sets
3. Transform data to appropriate units
   - Categorical works better, but not necessary
4. Select, train, & test the network
   - Can set number of hidden layers
   - Can set number of nodes per layer
   - A number of algorithmic options
5. Apply (need to use system on which built)

# Lessons Learned

- Versatile data mining tool
- Proven
- Based on biological models of how the brain works
- Feed-forward is most common type
- Back propagation for training sets has been replaced with other methods, notable conjugate gradient
- Drawbacks
  - Work best with only a few input variables and it does not help on selecting the input variables
  - No guarantee that weights are optimal—build several and take the best one
  - Biggest problem is that it does not explain what it is doing—no rules

# Decision Trees

- A decision tree is a structure that can be used to divide a large collection of records into successively smaller sets of records by applying a sequence of simple decisions rules.

  —Berry and Linoff.

- It consists of a set of rules for dividing a large heterogeneous population into smaller and smaller homogeneous groups based on a target variable.

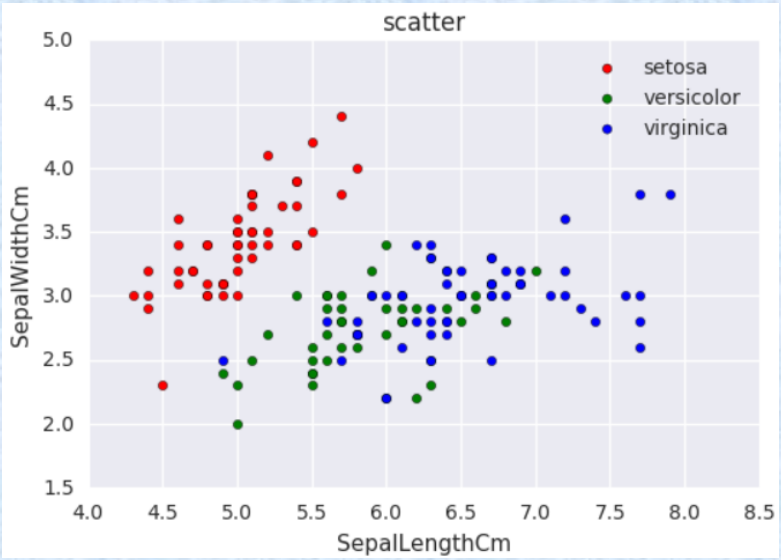
- A decision tree is a tree-structured plan of a set of attributes to test in order to predict the output.

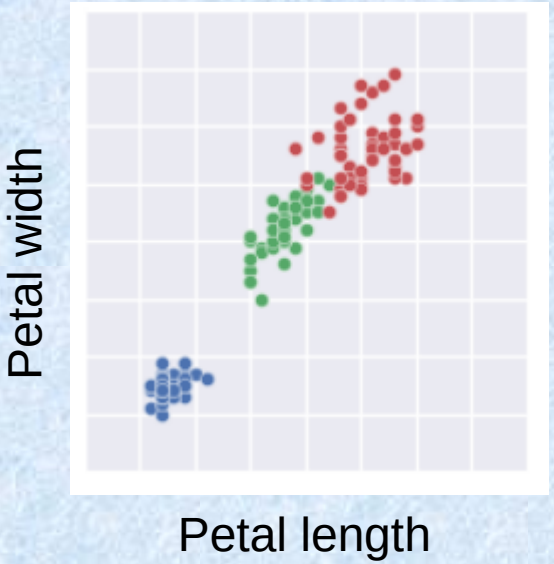
  —Andrew Moore.

- Target variable is usually categorical.

# Example

| sepal-length | sepal-width | petal-length | petal-width | class |
|---|---|---|---|---|
| 6.3 | 3.3 | 6 | 2.5 | Iris-virginica |
| 5.8 | 2.7 | 5.1 | 1.9 | Iris-virginica |
| 7.1 | 3 | 5.9 | 2.1 | Iris-virginica |
| 6.2 | 2.9 | 4.3 | 1.3 | Iris-versicolor |
| 5.1 | 2.5 | 3 | 1.1 | Iris-versicolor |
| 5.7 | 2.8 | 4.1 | 1.3 | Iris-versicolor |
| 5.1 | 3.8 | 1.9 | 0.4 | Iris-setosa |
| 4.8 | 3 | 1.4 | 0.3 | Iris-setosa |
| 5.1 | 3.8 | 1.6 | 0.2 | Iris-setosa |




https://datahub.io/machine-learning/iris

https://www.kaggle.com/code/xuhewen/iris-dataset-visualization-and-machine-learning/notebook

Different visualizations possible ..

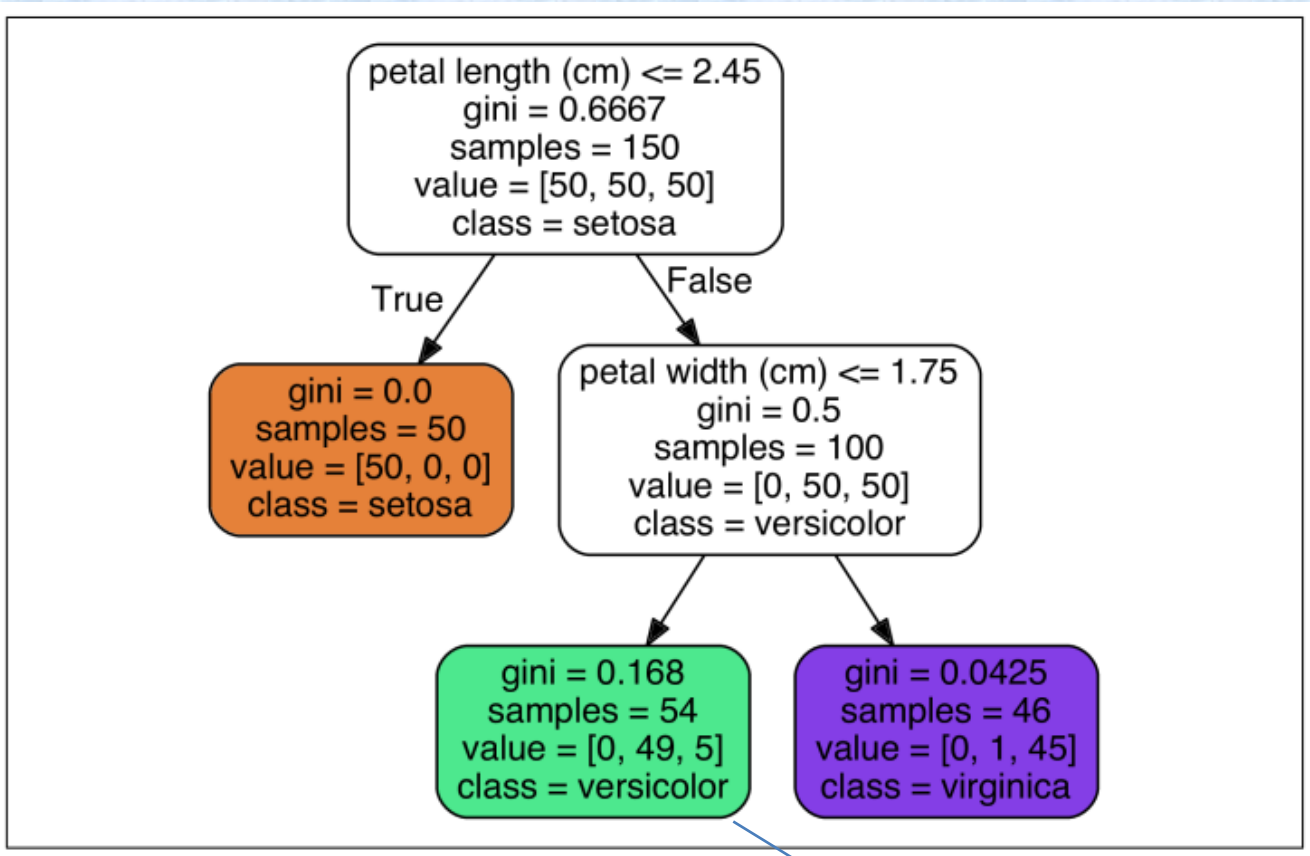



*Figure 6-1. Iris Decision Tree*

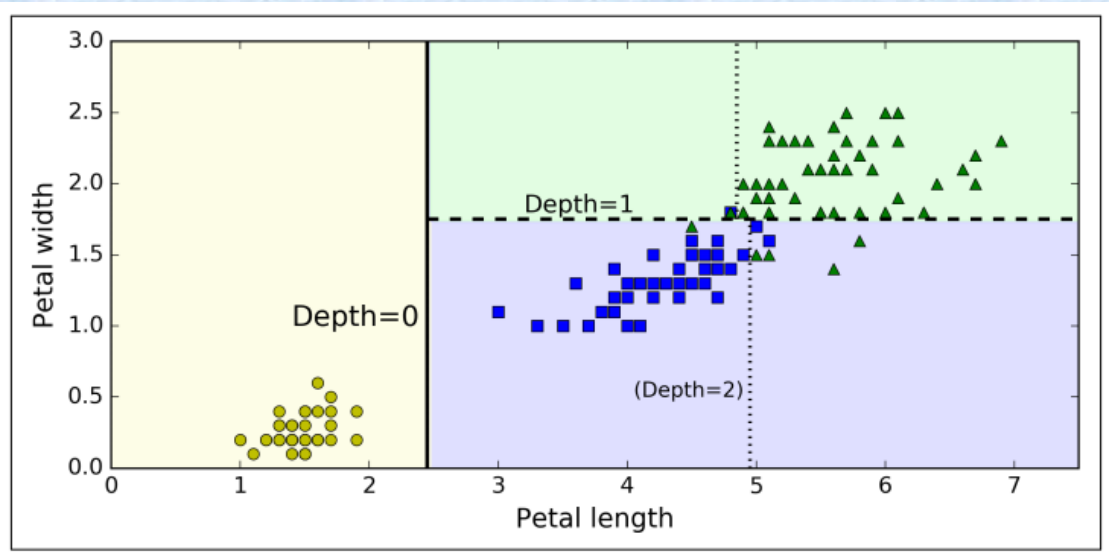



*Figure 6-2. Decision Tree decision boundaries*

Hands-on machine learning with scikit learn, Aurelion Geron

# Uses of Decision Trees

- Decision trees are popular for both classification and prediction (Supervised/Directed).
- Attractive largely due to the fact that decision trees represent rules—expressed in both English and SQL.
- Can also be used for data exploration—thus a powerful first step in model building.

# Finding the Splits

- A decision tree is built by splitting records at each node based on a **single input field**—thus there has to be a way to identify the input field that makes the best split in terms of the target variable.

- Measure to evaluate the split is **purity** (Gini, Entropy, Information Gain, Chi-square for categorical target variables and variance reduction and F test for continuous target variables)

- **Tree building algorithms** are exhaustive—try each variable to determine best one on which to split (increase in purity)—not recursive because it repeats itself on the children.

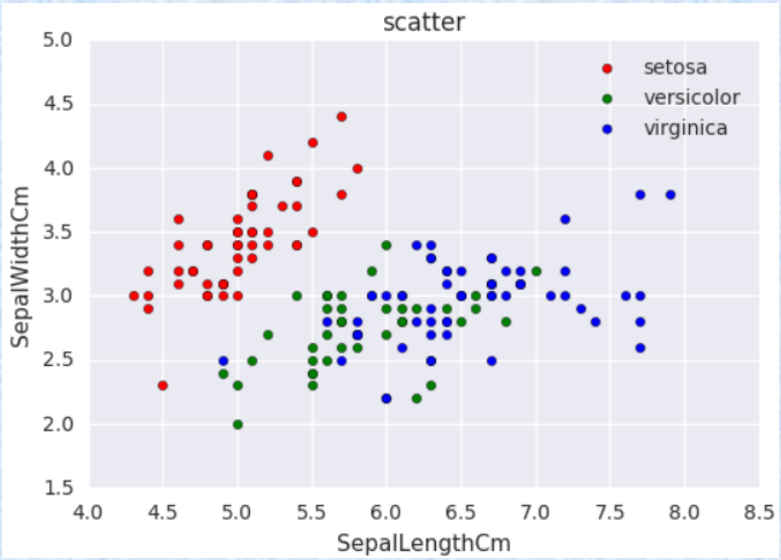
# Building Decision Trees

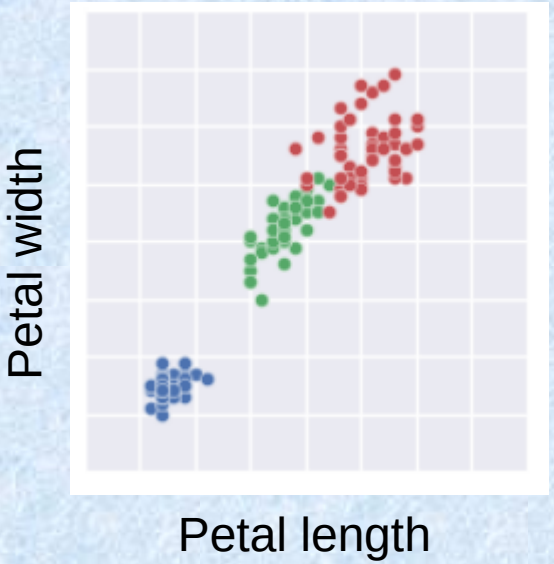
Key points in building a decision tree

- **Purity**→ the idea is to split attributes in such a way as move from heterogeneous to homogenous based on target variable
- **Splitting algorithm** (criterion)
  - Repeat for each node. At a node, all attributes analyzed to determine the best variable on which to split (How to measure?)
  - There are a number of algorithms and various implementations of the algorithms.
- **Stopping**
  - When a node is pure →leaf
  - No more splits are possible.
  - User defined parameters such as maximum depth or minimum number in a node.

# Example

| sepal-length | sepal-width | petal-length | petal-width | class |
|---|---|---|---|---|
| 6.3 | 3.3 | 6 | 2.5 | Iris-virginica |
| 5.8 | 2.7 | 5.1 | 1.9 | Iris-virginica |
| 7.1 | 3 | 5.9 | 2.1 | Iris-virginica |
| 6.2 | 2.9 | 4.3 | 1.3 | Iris-versicolor |
| 5.1 | 2.5 | 3 | 1.1 | Iris-versicolor |
| 5.7 | 2.8 | 4.1 | 1.3 | Iris-versicolor |
| 5.1 | 3.8 | 1.9 | 0.4 | Iris-setosa |
| 4.8 | 3 | 1.4 | 0.3 | Iris-setosa |
| 5.1 | 3.8 | 1.6 | 0.2 | Iris-setosa |




https://datahub.io/machine-learning/iris

https://www.kaggle.com/code/xuhewen/iris-dataset-visualization-and-machine-learning/notebook

Different visualizations possible ..

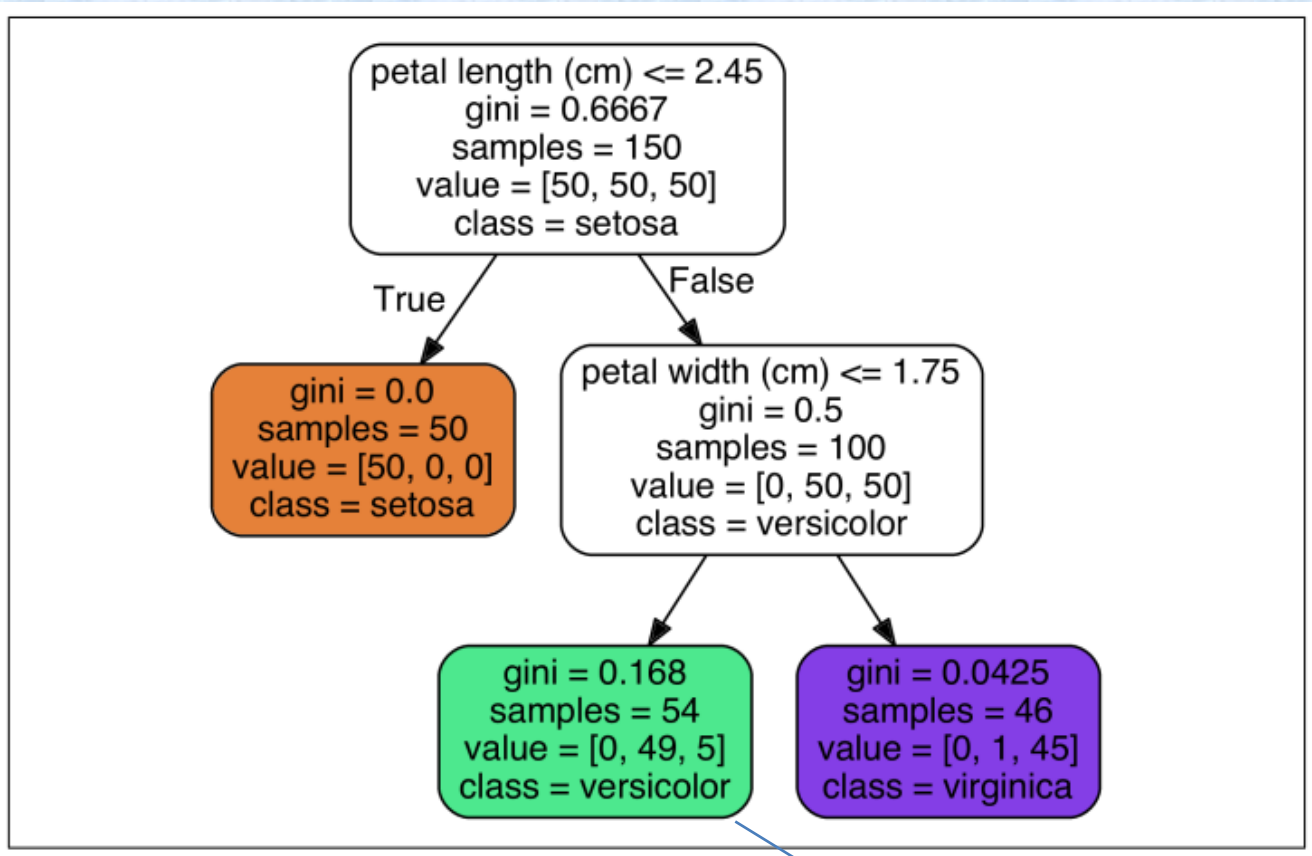



*Figure 6-1. Iris Decision Tree*

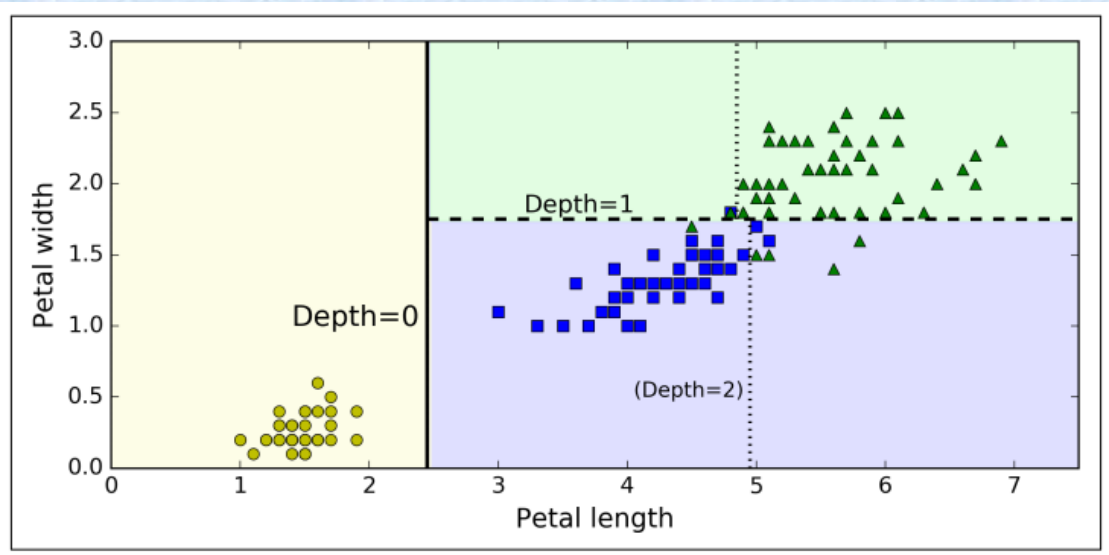



*Figure 6-2. Decision Tree decision boundaries*

Gini index

$$G_i = 1 - \sum_{k=1}^{n} p_{i,k}^{2}$$

Classes are in the order [Setosa, Versicolor, Virginica]
So, value = [0, 49, 5]
The Gini index = 1 – [ (0/0)^2 + (49/54)^2 + (5/54)^2] = 0.168
Note that there are 54 samples at this node

Hands-on machine learning with scikit learn, Aurelion Geron

# Extracting Rules from Trees

- Fewer leafs is better for generating rules.
- Easy to develop English rules.
- Easy to develop SQL rules that can be used on a database of new records that need classifying.
- Rules can be explored by domain experts to see if rules are usable or perhaps a rule is simply echoing a procedural policy.

# Decision Trees in Practice

- Data exploration tool.
- Predict future states of important variables in an industrial process.
- To form directed clusters of customers for a recommendation system.

# Decision Trees: Conclusion

- Decision Trees are the single most popular data mining tool.
  - Easy to understand
  - Easy to implement
  - Easy to use
  - Computationally cheap
- It is possible to get into trouble with overfitting.
- Mostly, decision trees predict a categorical output from categorical or numeric input variables.

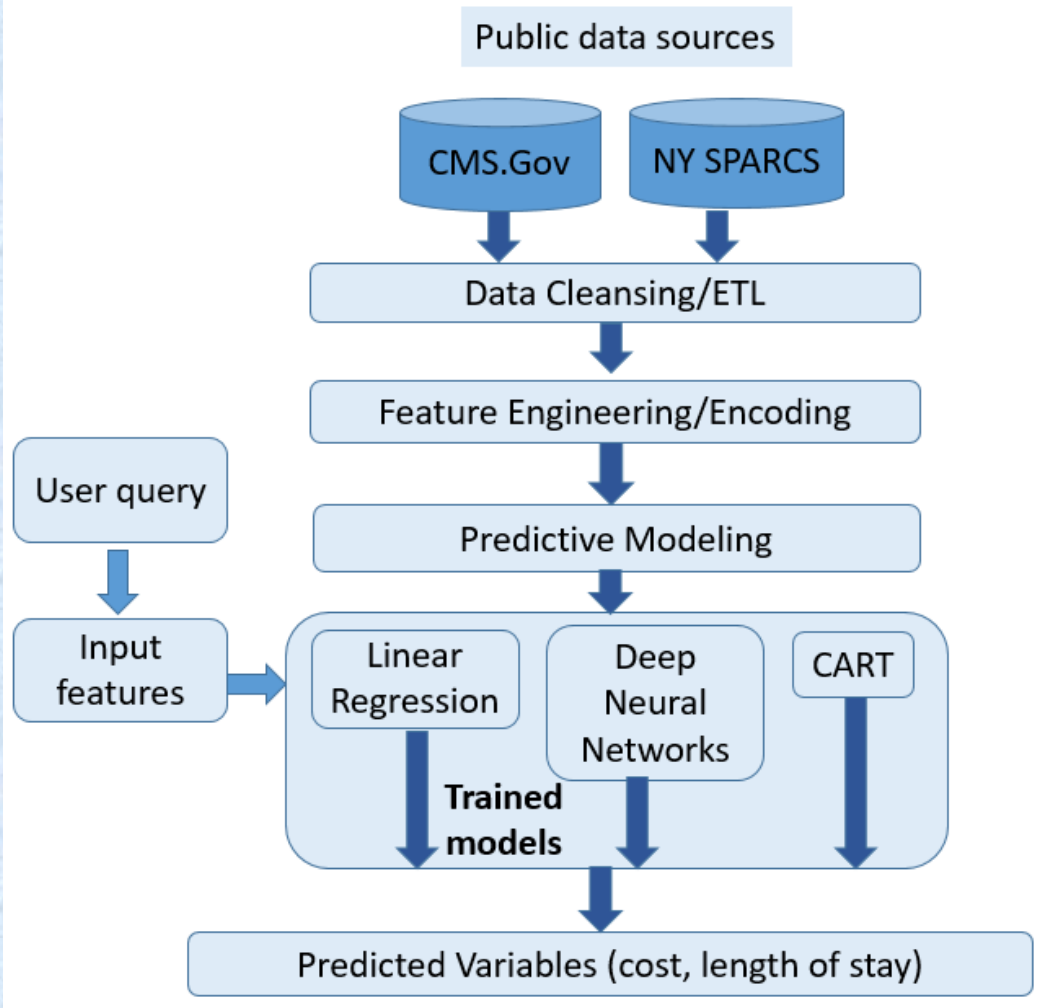
**Note:** Overfitting is when the model fits noise (i.e. pays attention to parts of the data that are irrelevant)—Another way of saying this is it memorizes the data and may not generalize.

# A comparison of models to predict medical procedure costs from open public healthcare data

A. Ravishankar Rao, PhD, Daniel Clarke

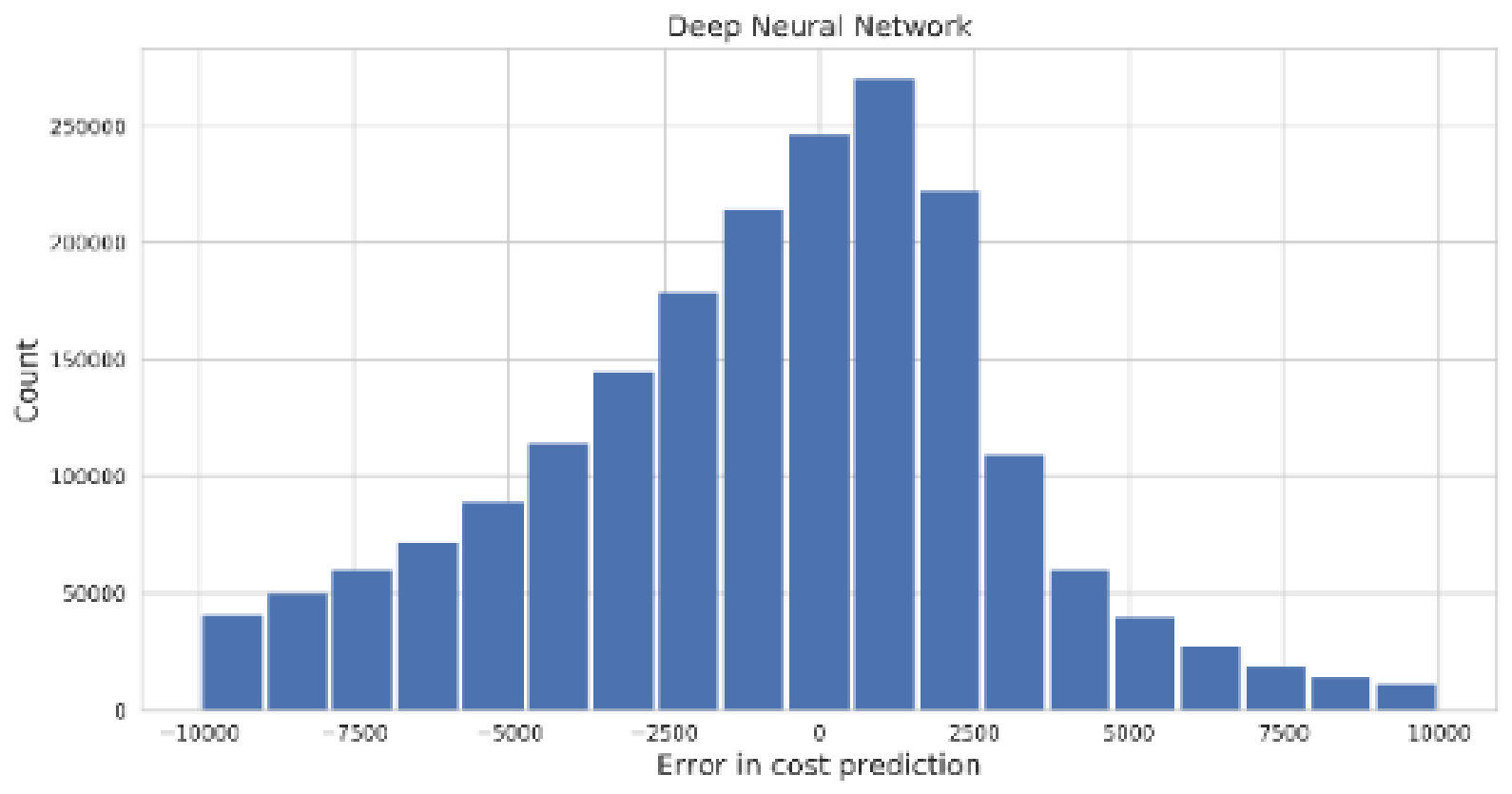
IEEE IJCNN 2018 Conference

| Hospital County | Albany |
|---|---|
| Facility Name | Albany Med. Center |
| Age Group | 18 to 29 |
| Zip Code - 3 digits | 124 |
| Gender | F |
| Race | Other Race |
| Ethnicity | Unknown |
| CCS Diagnosis Description | OTH COMP BIRTH |
| CCS Procedure Description | CESAREAN SECTION |
| Payment Typology 1 | Managed Care |
| Total Costs | $12,068.11 |



# Results

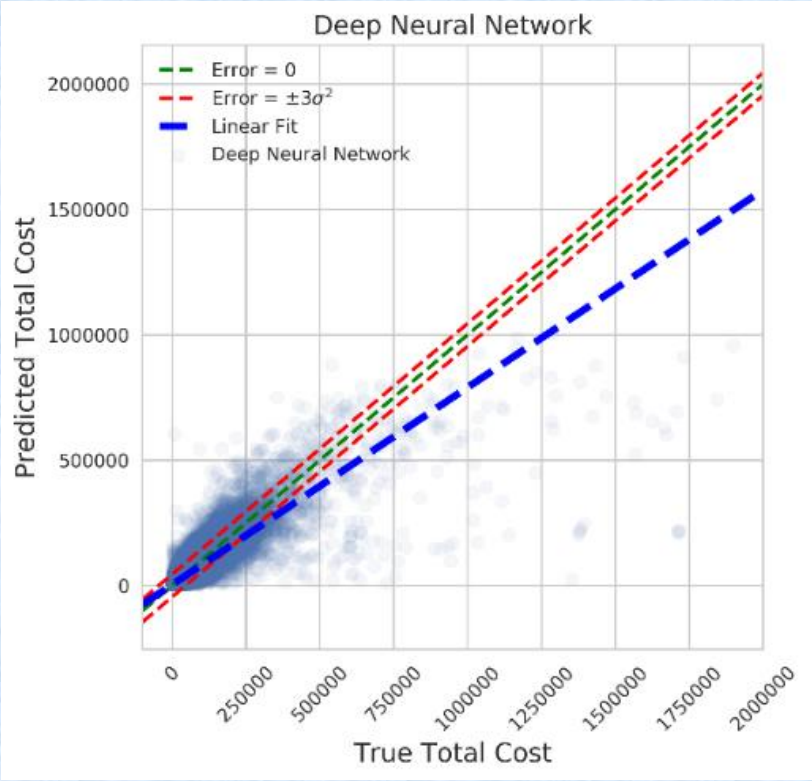
Histogram of prediction errors
for the best performing DNN.




**8 layered DNN with size 5x5x10x25x25x10x5x5 using an Adam optimizer with learning rate of 0.01. We obtained an $R^2$ value of 0.71**

| Model type | RMS Error | $R^2$ |
|---|---|---|
| Linear regression | $19,092 | 0.64 |
| Regression Tree | $18,132 | 0.68 |
| DNN | $16,841 | 0.71 |

Scatter plot:
True versus predicted costs



# Applications to cybersecurity

NCYTE CENTER

# Machine Learning and Cybersecurity Applications

J. Philip Craiger, Ph.D., CISSP

Professor of Cybersecurity

Embry-Riddle Aeronautical University

Whatcom COMMUNITY COLLEGE

# The Need for AI in Cybersecurity

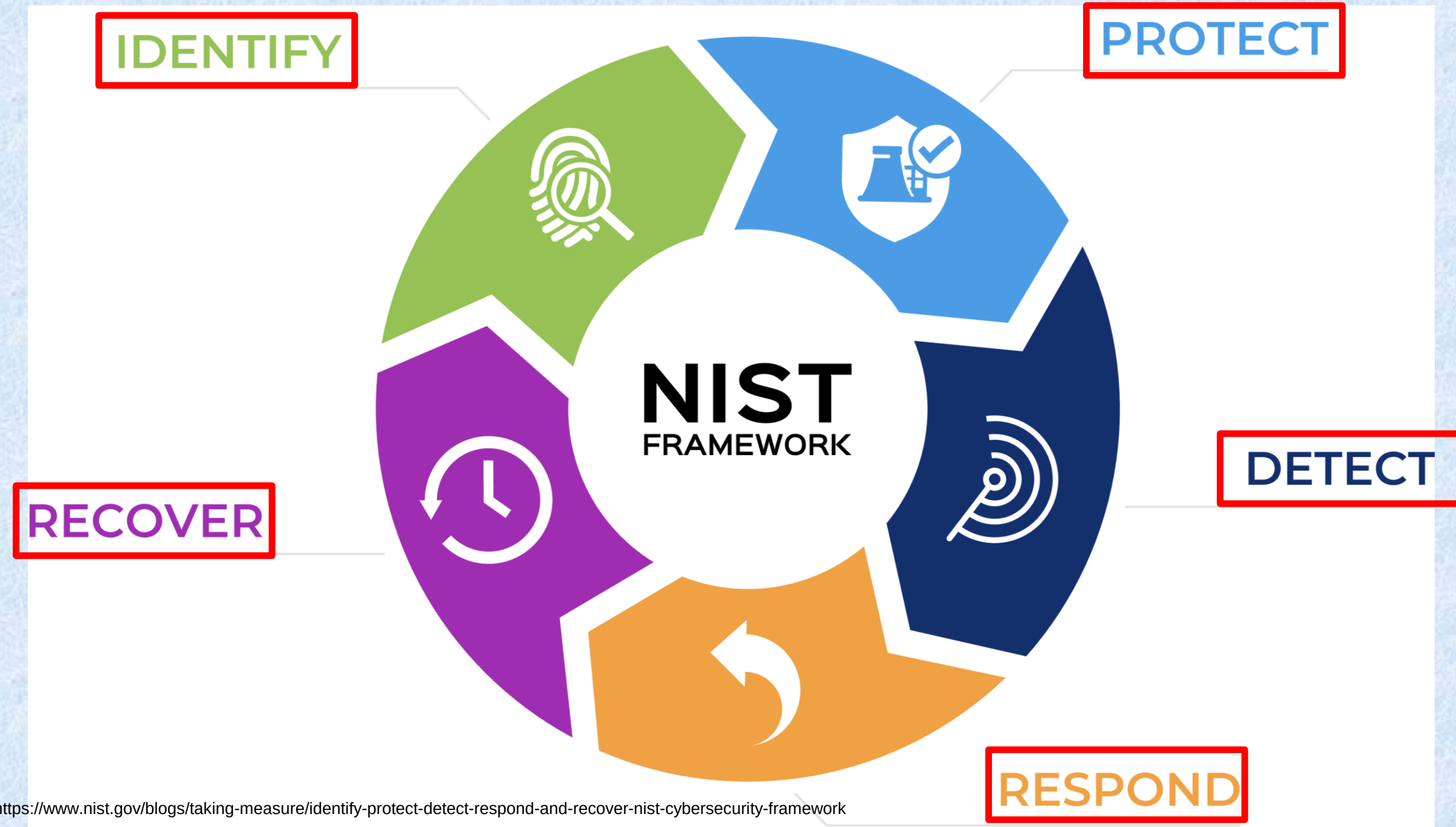
- 80% of companies in the US experienced an increase in cyberattacks in 2020
- Ransomware attacks rose 148%
- Monetary losses are projected to hit $6 trillion by 2021 (annually)
- Cloud-based attacks rose 630% between January and April 2020

https://yashasvi-kumar.medium.com/applications-of-ai-in-cyber-security-6f82abae7146


*Cost of a data Breach REPO*. https://www.ibm.com/security/data-breach.

As the number of cyberattacks increase, advanced, automated methods for detecting,  identifying, protecting, responding and recovering will be crucial


*ORT 2021*. IBM. (n. ecurity/data-




https://www.nist.gov/blogs/taking-measure/identify-protect-detect-respond-and-recover-nist-cybersecurity-framework

# Uses of AI/ML in Cybersecurity

- **Identify** new network vulnerabilities and threats
  - AI systems can predict how and where you are most likely to be compromised so that you can plan and allocate resources towards the most vulnerable areas
- Immediate **detection** at start of attacks
  - AI can respond to the attacks in real-time
- AI can help build an understanding of website traffic and distinguish (**identify**) between good bots (e.g., search engine crawlers), bad bots, and humans
- Used for proactive threat hunting or in reactive incident investigations
- AI can **identify, respond** to, and **protect** against anomalous behaviors for end-point protection
  - AI-driven malware detection or network anomaly detection

https://www.computer.org/publications/tech-news/trends/the-use-of-artificial-intelligence-in-cybersecurity

# Malware Detection

# Identifying New Malware and ML

- Current malware protection uses various methods to identify ***existing*** malware
  - Hashes (i.e., a digital fingerprint of a file)
  - Specific segments of internal code
  - Behavioral characteristics when running
- These methods provide little protection from ***zero-day*** attacks
  - “Zero-day” means that the attack or code has never been seen before “in the wild” so hashes, code, or behavior may have never been observed
  - Current methods may deem a file “safe” when it’s actually malware – an error that is called a “false negative”

## 1. There are now more than 1 billion malware programs out there.

(AV-Test Institute)

Since 2013, malware has been spreading exponentially. The initial boom doubled the number of malicious files and programs infecting the web. In the following years, the growth might have slowed down, but it definitely hasn't stopped. Even with built-in antivirus software protecting the newest operating systems, there's more malware online than ever before.

---

## 2. 560,000 new pieces of malware are detected every day.

(AV-Test Institute)

The rate at which malware spreads is terrifying. Anti-malware institutes include every new malicious program they find in their malware database. Hundreds of thousands of files become infected by malware on computers and websites every day. These are mostly the result of existing infections that keep spreading like actual diseases. According to the latest statistics, more than 17 million new malware instances are registered each month.

https://dataprot.net/statistics/malware-statistics/

# Identifying New Malware and ML

- Current malware protection uses various methods to identify ***existing*** malware
  - Hashes (i.e., a digital fingerprint of a file)
  - Specific segments of internal code
  - Behavioral characteristics when running
- These methods provide little protection from ***zero-day*** attacks
  - “Zero-day” means that the attack or code has never been seen before “in the wild” so hashes, code, or behavior may have never been observed
  - Current methods may deem a file “safe” when it’s actually malware – an error that is called a “**false negative**”

What is the procedure for ML to learn which files are malware and which are not?

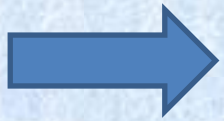
Malware training cases

ML Algorithms

ML training cases may include the file’s code (executable or otherwise) and whether it is malware or not

1,000,000 malware training cases

ML Algorithms
(after learning)



Trained to identify relationships between code and prediction


Each
new file
ML
Algorithms
(after learning)
Assessment
(malware or not)

# How to convert **vulnerable code** to **effective features**?

## Source code

```
/* ssl/d1_both.c */
// [...]
int dtls1_process_heartbeat(SSL *s)
{
  unsigned char *p = &s->s3->rrec.data[0], *pl;
  unsigned short hbtype;
  unsigned int payload;
  unsigned int padding = 16; /* Use minimum padding */
  /* Read type and payload length first */
  hbtype = *p++;
  n2s(p, payload);
  if (1 + 2 + payload + 16 > s->s3->rrec.length)
    return 0; /* silently discard per RFC 6520 sec.4*/
  pl = p;
  // [...]
  if (hbtype == TLS1_HB_REQUEST){
   unsigned char *buffer, *bp;
   int r;
   // [...]
   buffer = OPENSSL_malloc(1 + 2 + payload + padding);
   bp = buffer;
   /* Enter response type, length and copy payload */
   *bp++ = TLS1_HB_RESPONSE;
   s2n(payload, bp);
   memcpy(bp, pl, payload);
   bp += payload;
   /* Random padding */
   RAND_pseudo_bytes(bp, padding);
   r = dtls1_write_bytes(s, TLS1_RT_HEARTBEAT,buffer,
                         3 + payload + padding);
   // [...]
   if (r < 0) return r;
  }
  // [...]
  return 0;
}
```

**Code Properties**

| | |
|---|---|
| Count Metrics | Blank Lines of Code |
| | Lines of Code |
| | Lines with Comments |
| | Statements |
| | Physical Lines of Code |
| | Declarative Lines of Code |
| | Executable Lines of Code |
| | Lines with Comments |
| | Inactive Lines |
| | Preprocessor Lines |
| | FanIn [11] |
| | FanOut [11] |
| | Path |
| Complexity Metrics | Cyclomatic Complexity (CC) [14] |
| | Modified CC [14] |
| | Strict CC [14] |
| | Essential Complexity [14] |
| | Knots [36] |
| | Nesting [10] |

Prof. Yang Xiang,
Swinburne Univ. Australia

# Feature Engineering

## How do we convert **vulnerable code** to **effective features**?

### Binary

```
94 06 4A 4A 18 4F 9A D3 AF 03 67 69 91 3D 24 88  ..JJ.O....gi.=$.
93 9D 5B C6 DD F0 72 01 43 6F 5F DC 9F 17 17 A0  ..[...r.Co_.....
84 87 27 1C 43 45 1E 92 12 DE 0E 3E 69 6C 21 9B  ..'.CE.....>il!.
C6 3C F8 60 5A 38 A1 C8 20 9C EF DA B8 A9 95 F9  .<.`Z8.. .......
CA AC 57 1C 4C 27 7B 17 FF B7 C1 98 61 7D E0 E5  ..W.L'{.....a}..
12 0A DE 53 3D C7 6C DE 1B DF 8F 22 68 77 EC 5F  ...S=.l....."hw._
57 98 69 5A 70 0E B6 06 CD 32 20 93 82 32 25 D1  W.iZp....2 ..2%.
7D 58 94 B0 CD 6D 15 37 A2 B9 F5 85 98 D9 65 F7  }X...m.7......e.
59 14 B4 93 8C 41 AD 3E 59 D5 26 53 C0 C5 8D 72  Y....A.>Y.&S...r
08 5A B4 03 19 17 A0 70 BE BE 0C 87 87 55 7B 04  .Z.....p.....U{.
57 2F 74 7F B5 54 73 8C 9F F8 6E DE 0B 28 36 18  W/t..Ts...n..(6.
81 73 3B C9 FF D0 8D 7A FD E8 3D 5B 7C 03 96 EA  .s;....z..=[|...
3D 31 51 D9 FD 46 1C F0 9A 3F C8 3C 18 7E 07 02  =1Q..F...?.<.~..
29 23 97 EE F8 64 58 3E 80 EB 84 99 82 3E 92 F5  )#...dX>.....>..
94 EB F0 6B 0A C8 CF D2 71 A2 27 41 73 0C 71 74  ...k....q.'As.qt
39 7C 59 DB A8 28 1B 3F D6 21 10 6A 68 4C 2A 05  9|Y..(.?.!.jhL*.
```

```
C01E 8D F0     INHEX   BSR     INCH      GET A CHAR
C020 81 30             CMP A   #'0       ZERO
C022 2B 11             BMI     HEXERR    NOT HEX
C024 81 39             CMP A   #'9       NINE
C026 2F 0A             BLE     HEXRTS    GOOD HEX
C028 81 41             CMP A   #'A
C02A 2B 09             BMI     HEXERR    NOT HEX
```

Pre-process

Features that are:

- Representative
- Invariant
- Distinctive

Classifiers

Prof. Yang Xiang,
Swinburne Univ. Australia

ML can learn complex relationships between training cases and outcomes …

ML algorithms applied to new cases (new files) that then ***automatically*** and ***quickly*** distinguish the good from the bad

# Sources for malware data

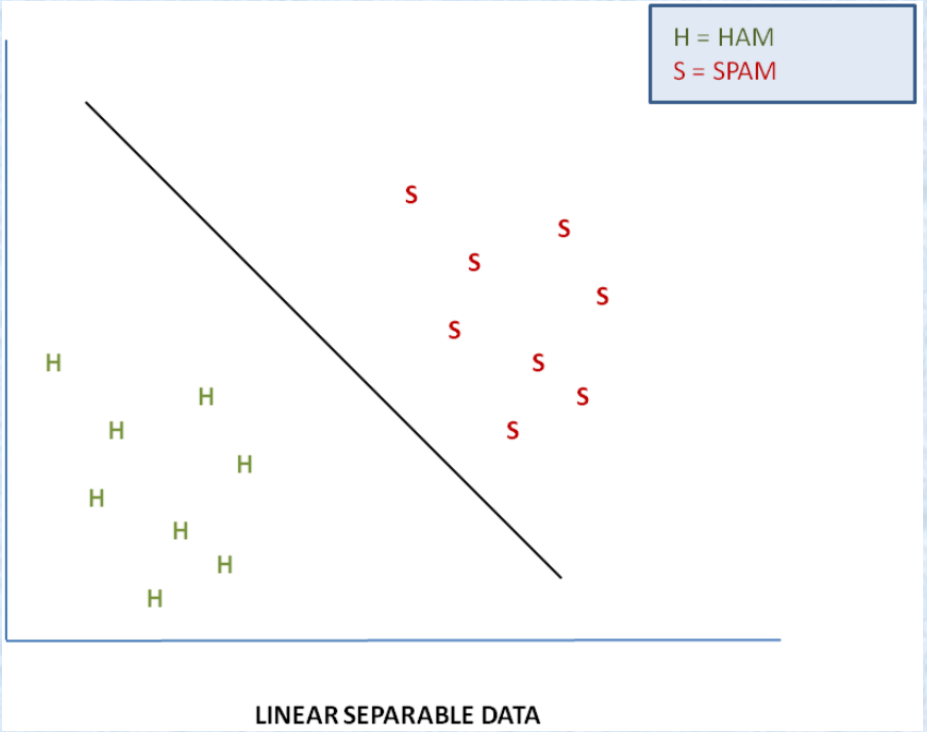
*MALWARE-TRAFFIC-ANALYSIS.NET*:
https://www.malware-trafficanalysis.net/
*VIRUSTOTAL*: https://www.virustotal.com
*VirusShare*: https://virusshare.com
*theZoo*: https://github.com/ytisf/theZoo (defined by the authors as *a repository of live malware for your own joy and pleasure*)

# SPAM filtering

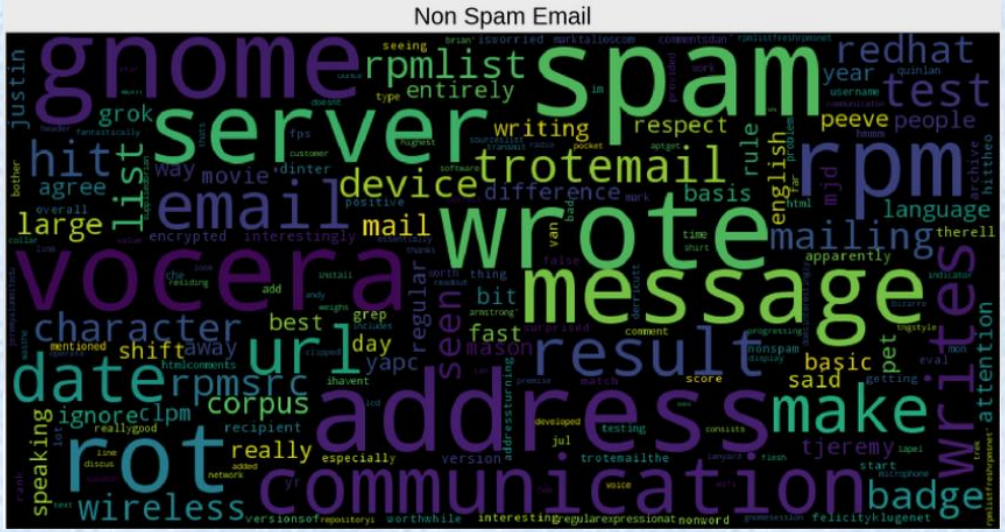
| Email | Buy | Sex | Spam or Ham? |
|---|---|---|---|
| 1 | 1 | 0 | H |
| 2 | 0 | 1 | H |
| 3 | 0 | 0 | H |
| 4 | 1 | 1 | S |

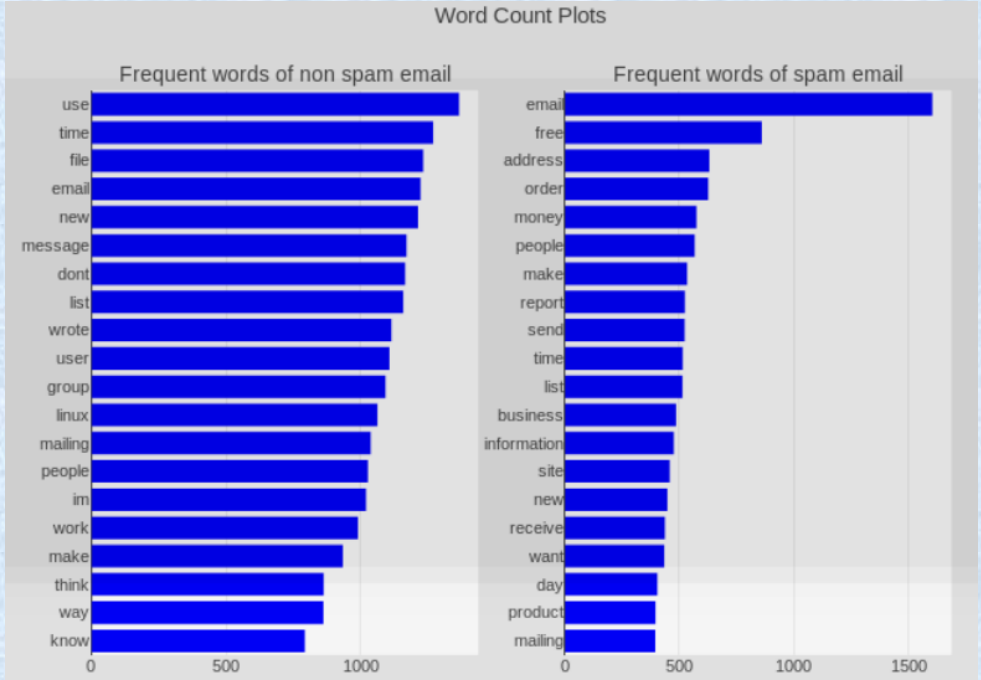
Feature engineering

| Email | B | S | B+S | Spam or Ham? |
|---|---|---|---|---|
| 1 | 1 | 0 | 1 | H |
| 2 | 0 | 1 | 1 | H |
| 3 | 0 | 0 | 1 | H |
| 4 | 1 | 1 | 2 | S |



# Word cloud for spam versus non-spam (ham) email


https://towardsdatascience.com/email-spam-detection-1-2-b0e06a5c0472

# Convert data to numeric form

Use CountVectorizer in scikit-learn

```
doc=["One Cent, Two Cents, Old Cent, New Cent: All About Money"]
```

This text is transformed to a sparse matrix as shown in Figure 1(b) below:

| | about | all | cent | cents | money | new | old | one | two |
|---|---|---|---|---|---|---|---|---|---|
| doc | 1 | 1 | 3 | 1 | 1 | 1 | 1 | 1 | 1 |

In theory (a)

| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|---|
| doc | 1 | 1 | 3 | 1 | 1 | 1 | 1 | 1 | 1 |

In practice (b)

https://kavita-ganesan.com/how-to-use-countvectorizer/

```
0  text = ['Hello my name is james',
1  'james this is my python notebook',
2  'james trying to create a big dataset',
3  'james of words to try differnt',
4  'features of count vectorizer']
```

| | big | count | create | dataset | differnt | features | hello | james | name | notebook | of | python | this | try | trying | vectorizer | words |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **1** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| **2** | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| **3** | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| **4** | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |

If we have a target label for each string, we can feed this into a machine learning algorithm. E.g. the target can be Spam or Ham.

https://towardsdatascience.com/basics-of-countvectorizer-e26677900f9c

# There are many other approaches possible

- Use different feature spaces
  - E.g. TfidfVectorizer (term frequency, inverse term frequency)
- Calculate probabilities of different word patterns (N-grams)
  - Use Bayesian probabilistic methods

# Potential Problems/Issues with AI

- AI requires knowledgeable workers to build the AI correctly
- AI requires BIG data sets of training cases
- AI requires a lot of processing power (training and running real-time)
- AI algorithms must be protected from interference, compromise, or misuse (ethical issues and interference from malign actors)
- AI doesn’t replace competent cybersecurity personnel
- Malign actors can use AI against its adversaries (us!)

# What if malign actors have access to AI?

MUST READ: Facebook is the AOL of 2021

# AI, quantum computing and 5G could make criminals more dangerous than ever, warn police

Law enforcement needs to be innovative and act now in order to keep face with near future criminal threats, warns 'Do criminals dream of electric sheep' paper.

By Danny Palmer | July 19, 2019 -- 11:32 GMT (04:32 PDT) | Topic: Security

MORE FROM DANNY PALMER

Security
**Ransomware: It's only a matter of**

# AI to create smarter cybersecurity

## ANGLE 1 | Smarter Cybersecurity

**Smarter cybersecurity controls**

- Better biometric controls
- Network Intrusion Detection and Prevention systems
- Malware detection
- Email filters (spam, phishing, etc.)

**Automatization in labour intensive tasks**

- Automatic discovery of vulnerabilities
- Automatic exploitation of vulnerabilities
- AI applied to malware analysis
- Security logs and event correlation
- Automatization of security operations (e.g. incident response)
- Cybersecurity exercises (attack/defense) and training
- Awareness raising

# Summary

- Need for AI in cybersecurity
- Some uses of AI in cybersecurity
- How machine learning (ML) works
- Example of ML for identifying malware
- Potential issues with AI

# Selected Readings and References


- EUROPOL. (2019). *Do criminals dream of electric sheep?* Cyber Security Intelligence. https://www.cybersecurityintelligence.com/blog/do-criminals-dream-of-electric-sheep-4428.html .
- Walch, K. (2019, September 30). *Rethinking weak vs. strong ai*. Forbes. https://www.forbes.com/sites/cognitiveworld/2019/10/04/rethinking-weak-vs-strong-ai/.
- Adams, R. L. (2017, November 6). *10 powerful examples of artificial intelligence in use today*. Forbes. https://www.forbes.com/sites/robertadams/2017/01/10/10-powerful-examples-of-artificial-intelligence-in-use-today/
- Segal, E. (n.d.). *The impact of AI On Cybersecurity: IEEE Computer Society*. IEEE Computer Society The Impact of AI on Cybersecurity Comments. https://www.computer.org/publications/tech-news/trends/the-impact-of-ai-on-cybersecurity.
- Parisi, A. (2019). Hands-on artificial intelligence for cybersecurity implement smart AI systems for preventing cyber attacks and detecting threats and network anomalies. Packt Publishing.
- *Belani, G. (n.d.). The use of artificial intelligence in CYBERSECURITY: A Reviewieee Computer Society. IEEE Computer Society The Use of Artificial Intelligence in Cybersecurity A Review Comments. https://www.computer.org/publications/tech-news/trends/the-use-of-artificial-intelligence-in-cybersecurity.*
- *Wolff, J. (2020, June 8). How to improve cybersecurity for artificial intelligence. Brookings. https://www.brookings.edu/research/how-to-improve-cybersecurity-for-artificial-intelligence/.*