DEFENCE INSTITUTE OF ADVANCED TECHNOLOGY

DEEMED UNIVERSITY

शास्त्रेण शस्त्रं प्रकरोति रक्षाम्

Machine Learning for Cyber Security (CS-602)

L#01

Introduction

By

Dr Sunita Dhavale

Syllabus

- Data Analytics Foundations: R programming, Python Basics -Expressions and Variables, String Operations, Lists and Tuples, Sets, Dictionaries Conditions and Branching, Loops, Functions, Objects and Classes, Reading/Writing files, Hand ling data with Pandas, Scikit Library, Numpy Library, Matplotlib, scikit programming for data analysis, setting up lab environment, study of standard datasets. Introduction to Machine Learning- Applications of Machine Learning, Supervised, unsupervised classification and regression analysis
- Python libraries suitable for Machine Learning Feature Extraction. Data pre-processing, feature analysis etc., Dimensionality Reduction & Feature Selection Methods, Linear Discriminant Analysis and Principal Component Analysis, tackle data class imbalance problem

Syllabus

- Supervised and regression analysis, Regression, Linear Regression, Non-linear Regression, Model evaluation methods, Classification, K-Nearest Neighbor, Naïve Bayes, Decision Trees, Logistic Regression, Support Vector Machines, Artificial Neural Networks, Model Evaluation. Ensemble Learning, Convolutional Neural Networks, Spectral Embedding, Manifold detection and Anomaly Detection
- Unsupervised classification K-Means Clustering, Hierarchical Clustering, Density-Based Clustering, Recommender Systems-Content-based recommender systems, Collaborative Filtering, machine learning techniques for standard dataset, ML applications, Case studies on Cyber Security problems that can be solved using Machine learning like Malware Analysis, Intrusion Detection, Spam detection, Phishing detection, Financial Fraud detection, Denial of Service Detection.

Text/Reference Books

1. Building Machine Learning Systems with Python – Willi Richert, Luis Pedro Coelho

2. Alessandro Parisi, Hands-On Artificial Intelligence for Cybersecurity: Implement smart AI systems for preventing cyber attacks and detecting threats and network anomalies Publication date :Aug 2, 2019, Packt, ISBN-13, 9781789804027

3. Machine Learning: An Algorithmic Perspective – Stephen Marsland

4. Sunita Vikrant Dhavale, "Advanced Image-based Spam Detection and Filtering Techniques", IGI Global, 2017

5. Soma Halder , Sinan Ozdemir, Hands-On Machine Learning for Cybersecurity: Safeguard your system by making your machines intelligent using the Python ecosystem, By Publication date : Dec 31, 2018, Packt, ISBN-13 :9781788992282

1. Stuart Russell, Peter Norvig (2009), "Artificial Intelligence – A Modern Approach", Pearson Elaine Rich & Kevin Knight (1999), "Artificial Intelligence", TMH, 2nd Edition

2. NP Padhy (2010), "Artificial Intelligence & Intelligent System", Oxford

3. ZM Zurada (1992), "Introduction to Artificial Neural Systems", West Publishing Company

4. Research paper for study (if any) – White papers on multimedia from IEEE/ACM/Elsevier/Spinger/ Nvidia sources.

Lab assignments

1	**Python Programming part-1**
2	Python Programming part-2
3	Study and Implement Linear Regression Algorithm for any standard dataset like in cyber security domain
4	Study and Implement KMeans Algorithm for any standard dataset in cyber security domain
5	Study and Implement KNN for any standard dataset in cyber security domain
6	Study and Implement ANN for any standard dataset in cyber security domain
7	Study and Implement PCA for any standard dataset in cyber security domain
8	Case Study: Use of ML along with Fuzzy Logic/GA to solve real world Problem in cyber security domain
9	Mini assignment: Apply ML along with PSO/ACO to solve any real world problem in cyber security domain
10	ML Practice Test – 1 Quiz

Defence Institute of Advanced Technology

School of Computer Engineering & Mathematical Sciences

SEMESTER-I TIME TABLE (AUTUMN 2024)$

PROGRAMMES: (I) CS [M.TECH IN CYBER SECURITY] (II) AI [M.TECH CSE (ARTIFICIAL INTELLIGENCE)] BATCH: 2024-2026

Lecture Day	L1 0900-1000	L2 1000-1100	L3 1100-1200	L4 1200-1300		L4 1400-1500	L4 1500-1600	L4 1600-1700	L4 1700-1800
Monday	CE-602 (AI) CS-602 (CS)	CE-604 (AI) CS-603 (CS)	CE-601 (AI) CS-604 (CS)	CE-601 (AI)	Lunch Break 1300-1400	LAB CE-601 (AI) LAB CS-602 (CS)		AM607	
Tuesday	CE-603 (AI) LAB CS-603 (CS)	CE-602 (AI) CS-602 (CS)	CE-601 (AI) CS-605 (CS)	CE-604 (AI) CS-604 (CS)		PGC 601		AM607	LAB CS-603 (CS)
Wednesday	CE-604 (AI) CS-605 (CS)	CE-603 (AI) CS-602 (CS)	CE-602 (AI) CS-603 (CS)	LAB CE-604 (AI) CS-604 (CS)		CE-605(AI) LAB CS-605 (CS)	LAB CS-605 (CS)	AM607	LAB CE-604 (AI)
Thursday	CE-604 (AI) CS-603 (CS)	CS-605 (CS)	LAB CE-602 (AI) CS-601 (CS)	CE-603 (AI) CS-601 (CS)		PGC 601		AM607	
Friday	LAB CE-603 (AI) LAB CS-601 (CS)		LAB CE-602 (AI) CS-601 (CS)	LAB CS-604 (CS)		CE-605(AI) LAB CS-604 (CS)	CE-605(AI)	LAB CE-605(AI)	

COURSE CODE & COURSE NAME		FACULTY
Programme: CS [M.Tech in Cyber Security] Classroom: Arjun	Programme: AI [M.Tech CSE (Artificial Intelligence)] Classroom: Kaveri	
CS-601 Data Security & Privacy	CE-601 Responsible Artificial Intelligence;	MJN: Dr. Manisha J. Nene
CS-602 ML for Cyber Security	CE-604 Practical Machine Learning;	SVD: Dr. Sunita V. Dhavale
CS-605 Network and Cloud Security	CE-602 Intelligent Algorithms	CRS: Prof. CRS Kumar
CS-604 Advanced System Security	---------	DVV: Dr. Deepti V. Vidyarthi
CS-603 Applied Cryptography	---------	AM: Dr. Arun Mishra
---------	CE-603 Deep Neural Network;	US: Dr. Upasna Singh
---------	CE-605 Mathematics for ML;	Unit-2: Dr Upasna, Unit 4: Dr Sunita, Unit3:MJN, Unit 1: Faculty To be Nominated
AM-607 Mathematics for Engineers	AM-607 Mathematics for Engineers	OO/DS/DP: Dr Odellu O., Dr Dasari S., Dr. Debasis P.
PGC-601 Research Methodology	PGC-601 Research Methodology	Common Subject for All

$ TENTATIVE T.T. SUBJECT TO CHANGE

Program Coordinator,
M.Tech (CS & AI), Batch 2024-26

Director, SoCE&MS

Introduction

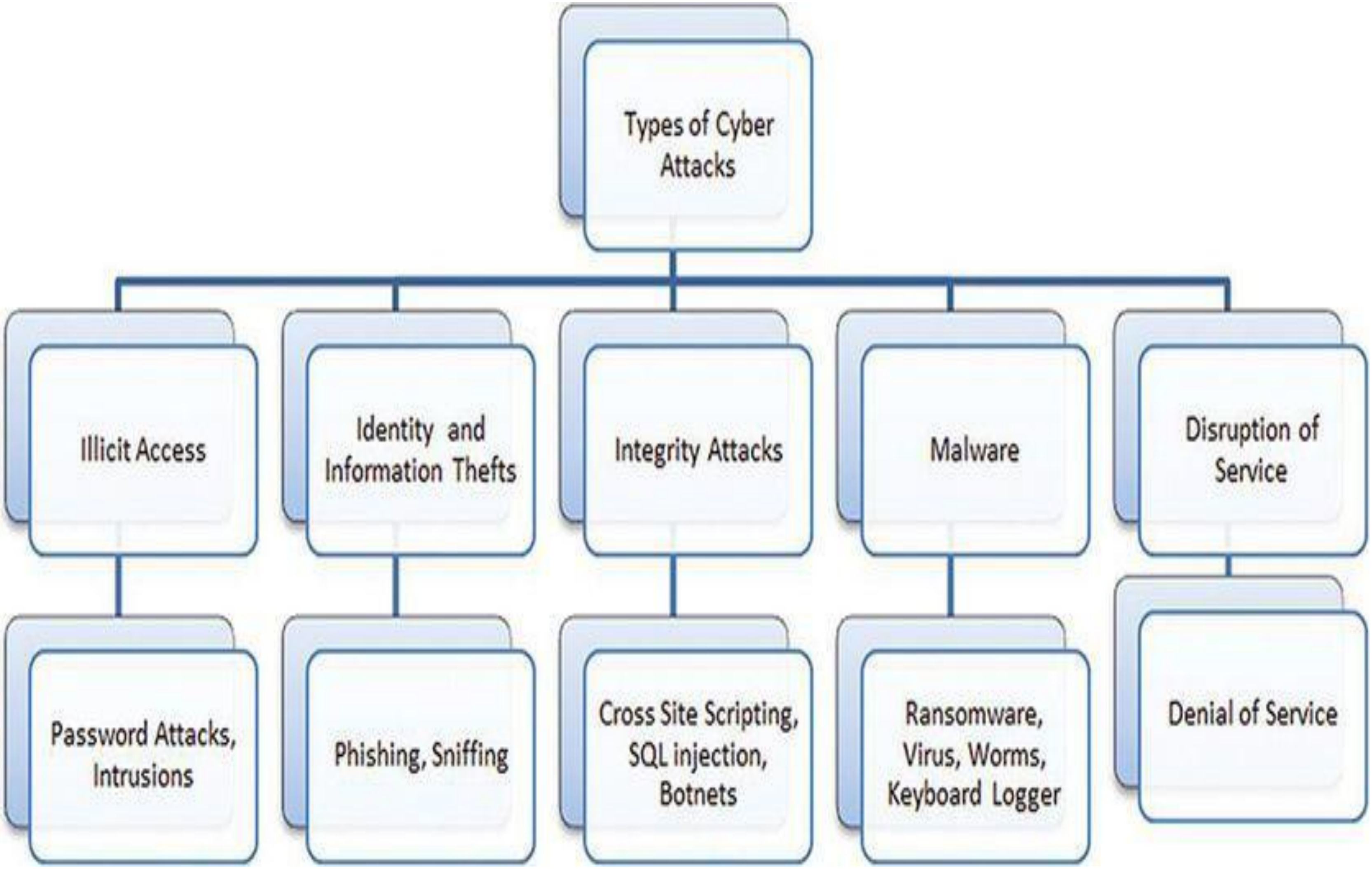
Introduction – Cyber Security Domain Requirements

- Cybersecurity - encompasses mechanisms for ensuring confidentiality, integrity, and availability of information during access, transit, and storage.
- cybersecurity represent techniques for information, network, Internet, and computer system security.
- Cybercrime - any unlawful act that can lead to failure, damage, error, accidents, harm, or any other undesirable event in cyberspace.
- To trade off the cost incurred in securing assets and attacks thwarted, some effective prioritization has to be done.
- One way to do this is to assess the context in which these assets are being accessed and utilized.

Introduction – Cyber Security Domain Requirements

- E.g, sudden heavy traffic on a website -> may be a DoS attack as well as legitimate surge in demand
- Need to find the context in which demand has increased.
- multilevel monitoring and logging -> a huge amount of data gets accumulated at a very high rate.
- A major challenge -> to process this data in real time to mine relevant threat intelligence in real time for immediate action as well as long-term system hardening.
- Machine learning (ML) techniques have proven to be great at mining information from heaps of data.
- Threat Intelligence -> evidence-based info. about cyber attacks that cyber security experts organize and analyze. Mechanisms of an attack/How to identify that an attack is happening/correlate various events and timely predict/detect onset of attack.

Introduction – Cyber Attacks



Types of Cyber Attacks
Illicit Access
Identity and Information Thefts
Integrity Attacks
Malware
Disruption of Service
Password Attacks, Intrusions
Phishing, Sniffing
Cross Site Scripting, SQL injection, Botnets
Ransomware, Virus, Worms, Keyboard Logger
Denial of Service

Introduction – Means of launching attacks

Attack vector	Type of attack
Social engineering, email spam	Identity and information thefts
Drive-by-download, P2P networks, free software sites	Spreading malware
Network based	Intrusions and disruption of service
Buffer overflow	Illicit access
System software (operating system, database management system, browser, etc.) flaws	Integrity attack

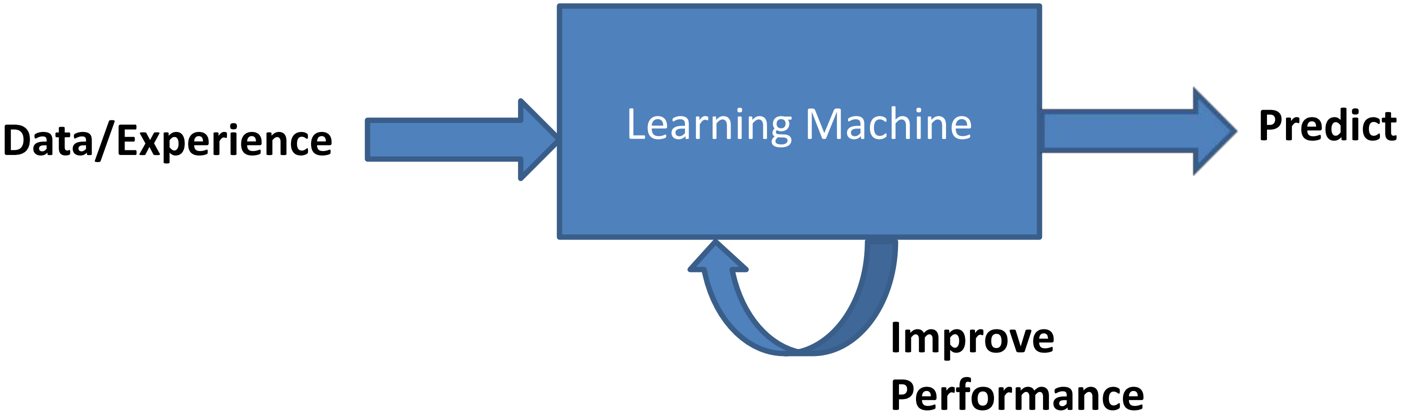
Introduction – Defense strategies

	Analysis (offline)	Prevention (online)	Detection and response (online)
Illicit access	Password strength indicator	Hashed password storage	Password compromise detection via user behavior detection
	Web application intrusion analytics	Anti-phishing	Script detection in Website code
	Breach and attack simulation (BAS)	Penetration testing	Insider threat/advanced persistent threat detection
	Botnet and DDoS characterization	Intrusion prevention system	Intrusion detection system
Identity/information theft Phishing	Algorithms for secure biometrics	Biometric identity verification	Replay attack and fake template detection
	masquerader profiling	Identity theft prevention	Identity theft/masquerade detection
	Characterization of side-channel leaks	Information leakage prevention	Man-in-the-middle and side-channel attack detection
	Featurization of malicious URLs and phishing pages	Real-time alerts	Phishing detection

Introduction – Defense strategies

	Analytics (offline)	Prevention (online)	Detection and response (online)
Integrity attack	**Software vulnerability enumerators**	**Fuzzing**	**Detection of undesired runtime behavior**
	Featurization for file-type distinction	File-type identification	Detection of masqueraded file from behavior
	Website vulnerability analytics	Detecting code and command injection vulnerabilities and remote code execution possibilities	Detecting unauthorized record updates
Malware	Malware analytics, anti-malware design	Malware scanners	Static and dynamic malware detection and quarantine
Disruption of service	Featurize intrusions and disruptions	Intrusion prevention systems	Network anomaly detection

Introduction to ML

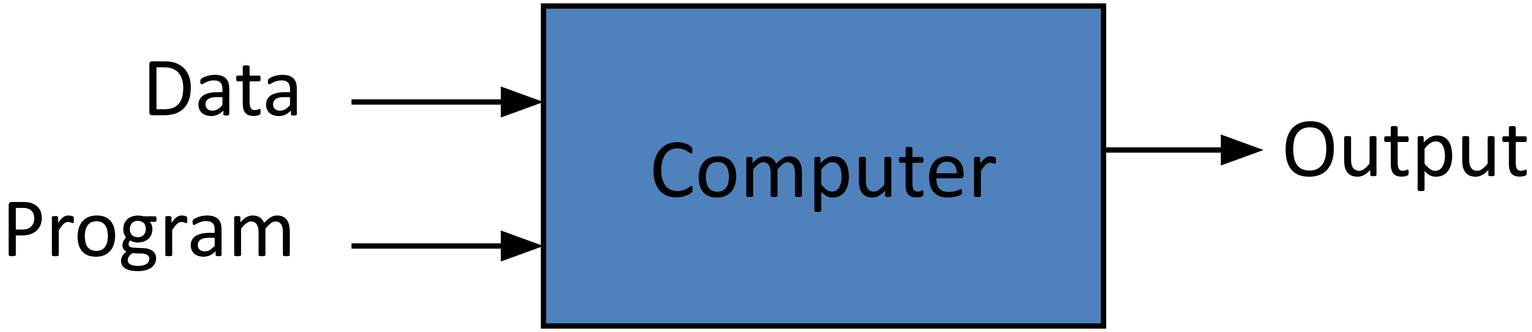


- *Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed.*
- *A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E. – Tom Mitchell, 1997*
- *These processes/algorithms for learning, prediction and improving performance are mathematical.*

Introduction

- *E.g. Email Program –watches your actions and learns to filter mails*
  - *E-watching you label emails as spam or not spam*
  - *T-classifying the mails as spam or not spam*
  - *P-number of emails correctly classified as spam or not spam*
- Aim - to uncover hidden patterns, unknown correlations, and find useful information from data.

Traditional Programming

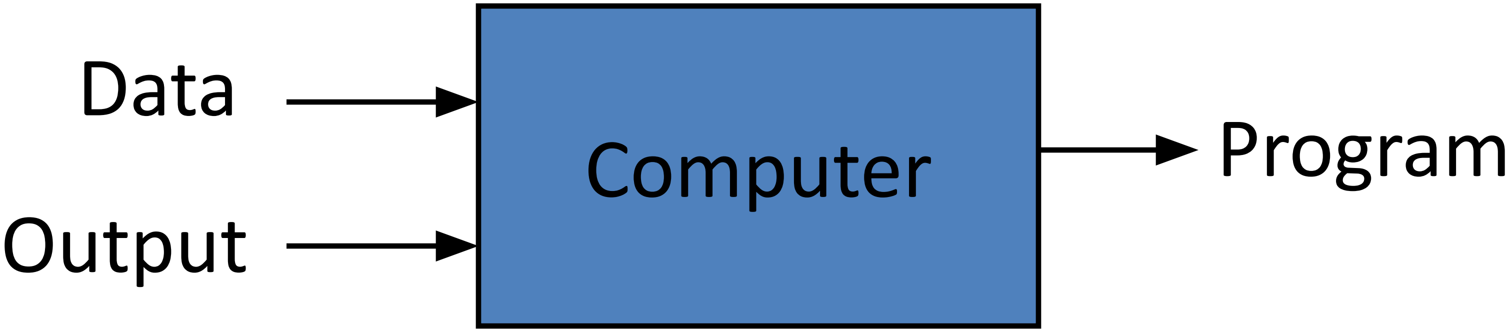

Data

Program

Computer

Output

Machine Learning


Data

Output

Computer

Program

Why Important?

- Learning is used when:
  - Human expertise does not exist (navigating on Mars),
  - Complex problems like speech recognition
  - Solution changes in time (routing on a computer network)
  - Solution needs to be adapted to particular cases (user biometrics)
- Data in many domains is huge
  - Thousands to billions of data samples
  - Hundreds to millions of attributes
  - Impossible for human analysts to see patterns across so much data
- Patterns in many domains are subtle, weak, buried in noise, or involve complex interactions of attributes
  - Often very difficult for human analysts to find
- In some domains discovery and use of patterns must happen in real time, e.g. in streaming data
  - Human analysts could never keep up

Machine Learning vs. Rule-Based Systems

- Spam Detection: discovered patterns and come up with following two simple rules to catch these messages:
  - If sender = promotions@online.com, then "spam"
  - If title contains "buy now 50% off" and sender domain is "online.com," then "spam"
  - Otherwise, "good email"
- At the beginning, the system works well. Later: The rules we have are no longer successful at marking these messages as spam.
- May discover a few patterns and modify the rules again
  - If body contains "deposit," then
    - If the sender's domain is "test.com," then spam
    - If description length is >= 100 words, then spam
- impossible to include new patterns in the code without breaking the existing logic.
- In the long run, it's quite difficult to maintain and adjust existing rules…

Machine Learning vs. Rule-Based Systems

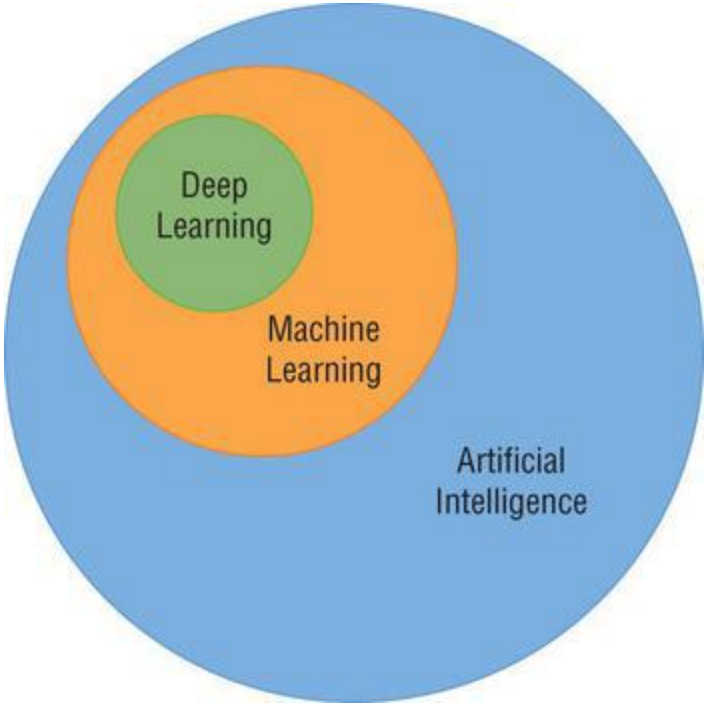
- ML -> do not extract these patterns manually.
- ML-> Provide labeled dataset e.g. email - spam or not spam with a set of its characteristics/features.
- When a new pattern emerges—if there's a new type of spam—we, instead of manually adjusting the existing set of rules, simply provide a machine learning algorithm with the new data. As a result, the algorithm picks up the new important patterns from the new data without damaging the old existing patterns
- For some simple tasks, rules and heuristics often work well.
- If no data is available, machine learning is not possible.

AI,ML and DL

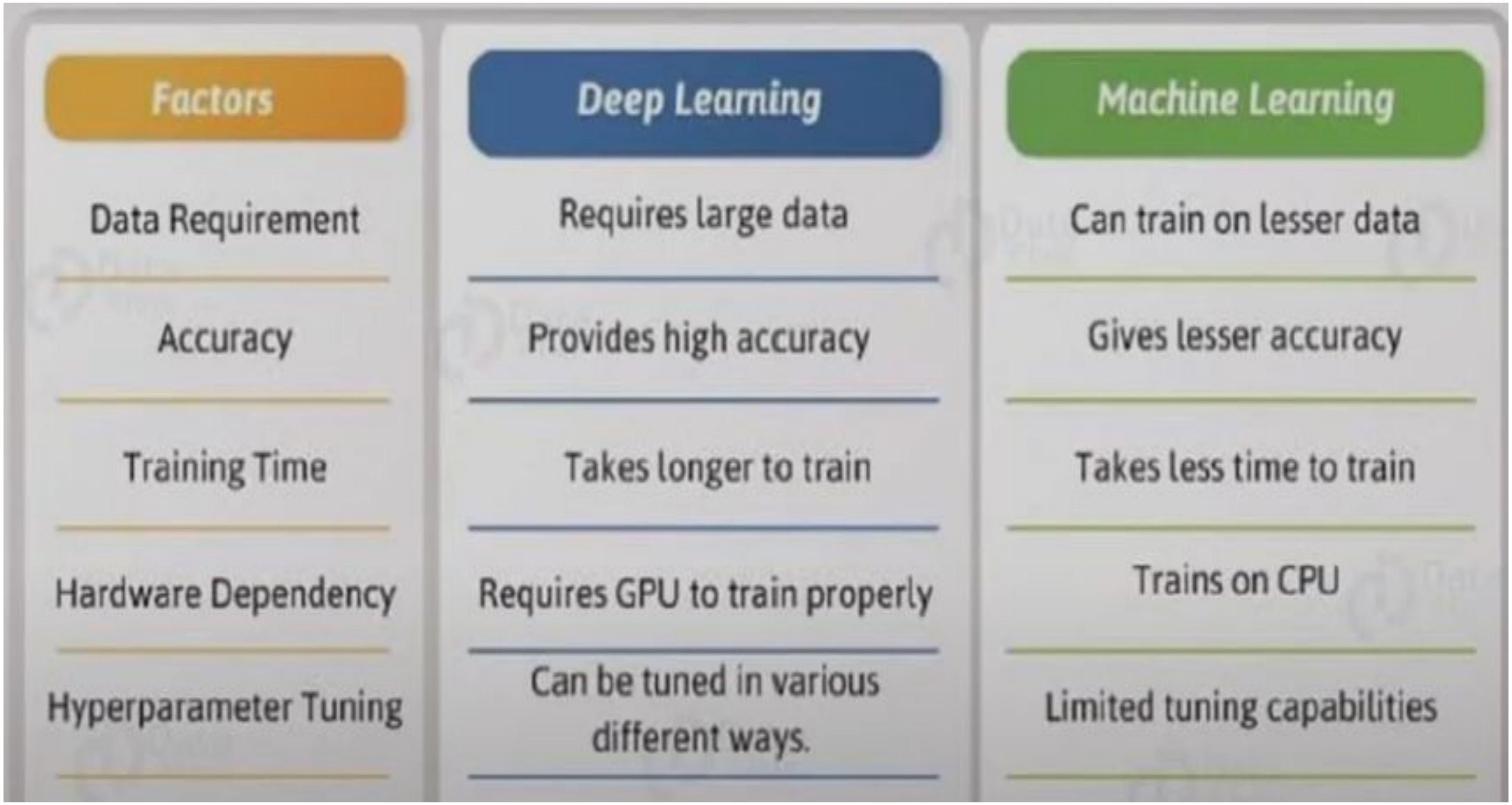
Artificial intelligence (AI) includes any type of technique where we are attempting to get a computer system to imitate human behavior. i.e. to ask computer systems to artificially behave as if they were intelligent.

Machine learning (ML) is a subset of AI techniques that attempt to apply statistics to data problems in an effort to discover new knowledge by generalizing from examples.

Deep learning (DL) is a further subdivision of machine learning that uses a set of complex techniques, known as neural networks, to discover knowledge in a particular way



ML and DL

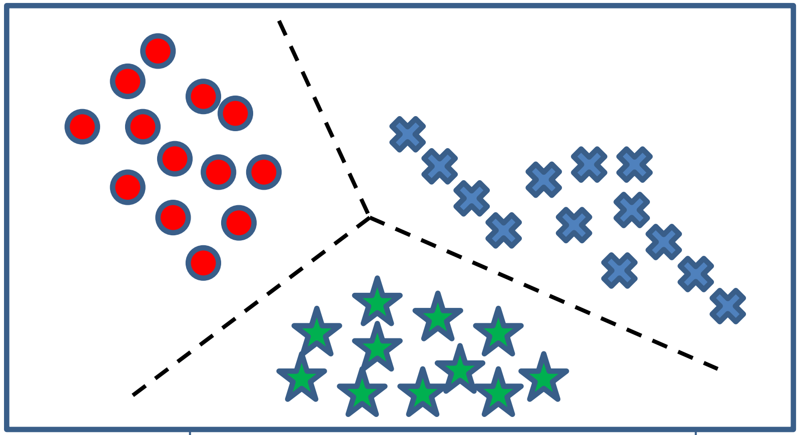


Factors	Deep Learning	Machine Learning
Data Requirement	Requires large data	Can train on lesser data
Accuracy	Provides high accuracy	Gives lesser accuracy
Training Time	Takes longer to train	Takes less time to train
Hardware Dependency	Requires GPU to train properly	Trains on CPU
Hyperparameter Tuning	Can be tuned in various different ways.	Limited tuning capabilities

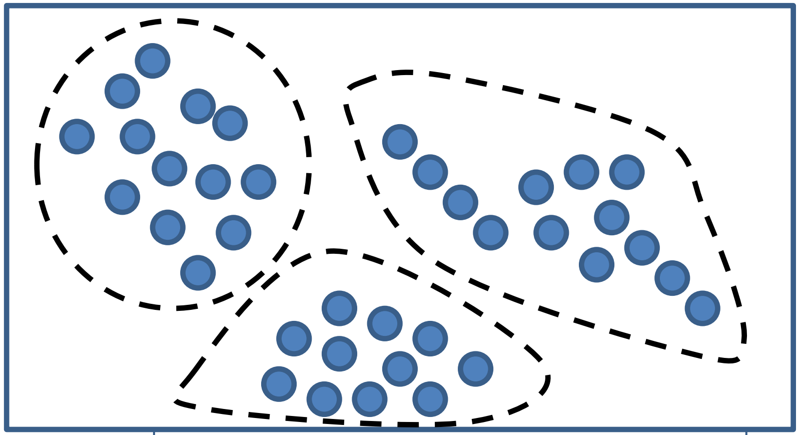
Types of Learning

- **Supervised (inductive) learning** $\{x_n \in R^d, y_n \in R\}_{n=1}^{N}$
  - Training data includes desired outputs
- **Unsupervised learning** $\{x_n \in R^d\}_{n=1}^{N}$
  - Training data does not include desired outputs, clustering, dimension reduction
- **Semi-supervised learning**
  - it can use available unlabeled data to improve supervised learning tasks when labeled data are scarce/expensive
- **Reinforcement learning**
  - Rewards/penalties from sequence of actions, trial error with feedback, agent learns policy that maximizes some performance from close interaction from stochastic/noisy env.

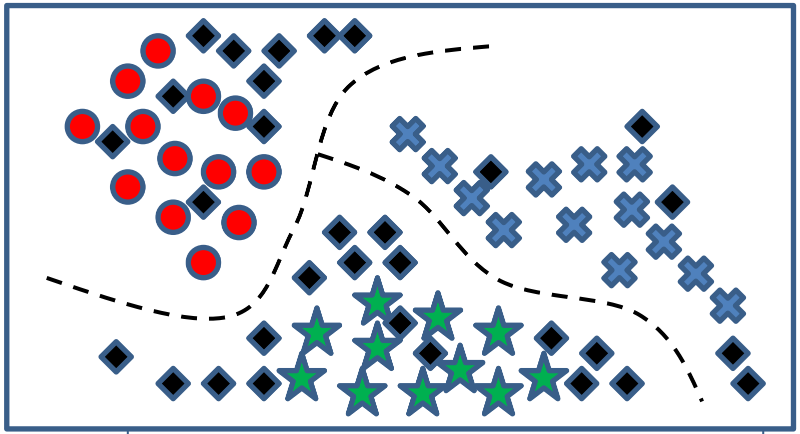
Types of Learning



Supervised learning

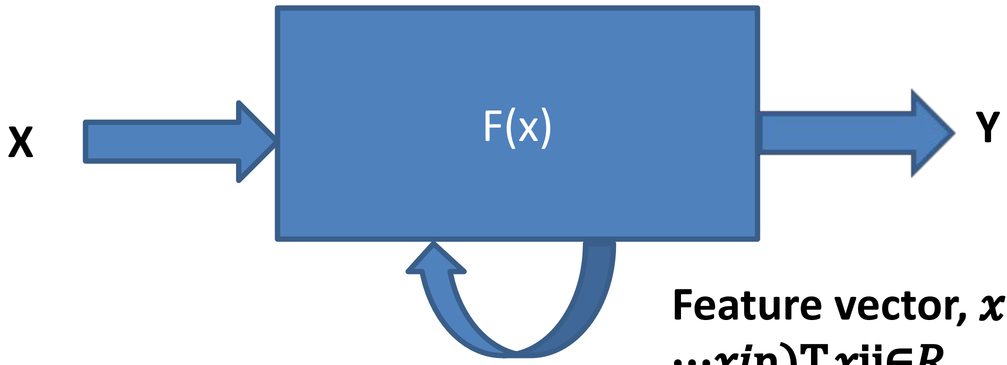


Unsupervised learning



Semi-supervised learning

Supervised Learning



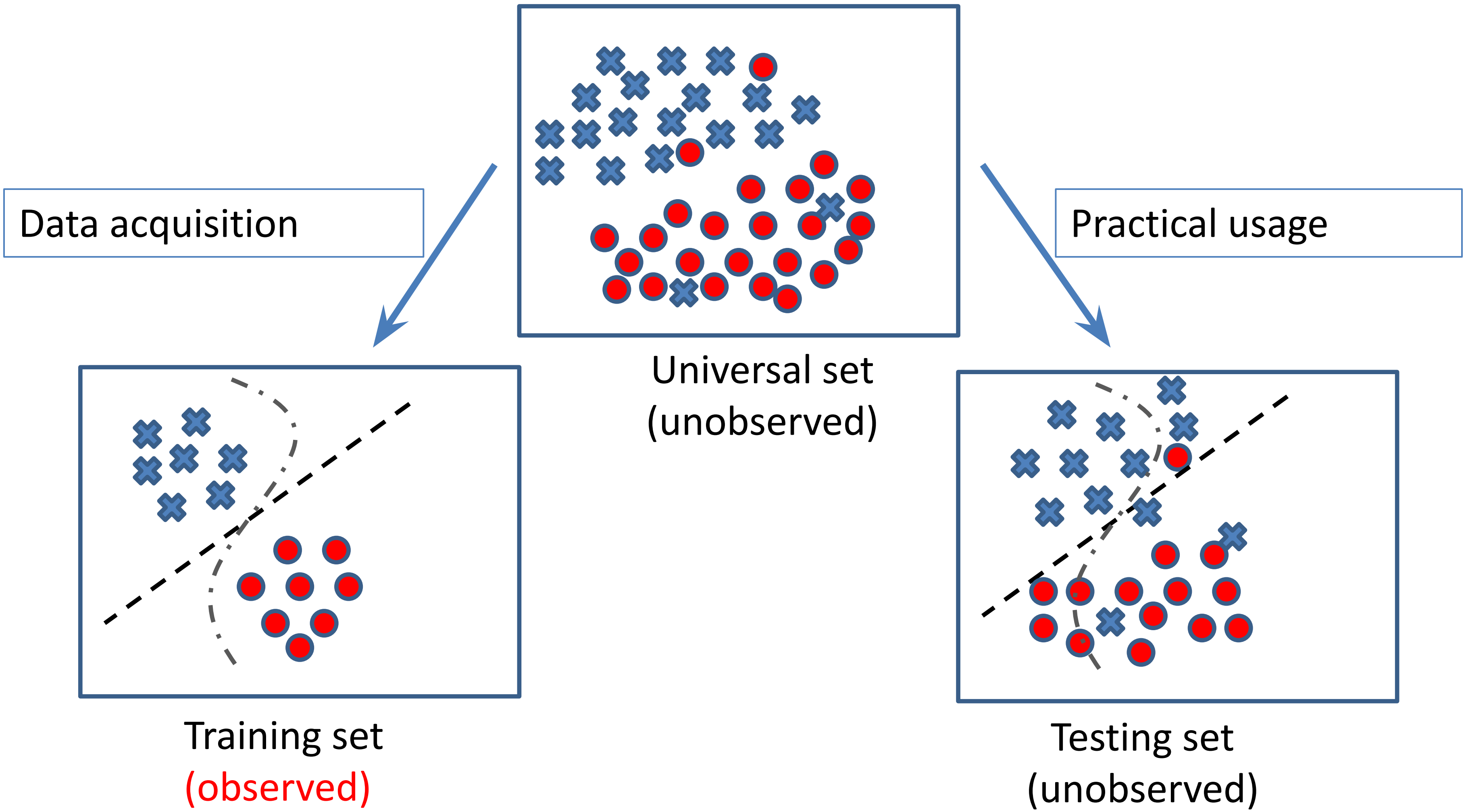
Feature vector, $xi=xi1,xi2,$
$\cdots xin)\mathrm{T},xij\in R$
Outcome, $y\in\{-1,1\}$ or $\{0,1\}$

- *What is f(x)?.*

Supervised Learning

- **Given** examples of a function *(X, F(X))*
- **Predict** function *F(X)* for new examples *X*
  - Discrete *F(X)*: Classification – person is buying computer or not based on income/age?, (1) *binary classification*, which has only two possible outcomes, such as spam or not spam, and (2) *multiclass classification*, which has more than two possible outcomes, such as a car make (Toyota, Ford, Volkswagen, and so on).
  - Continuous *F(X)*: Regression – how much person will spend if age/income?
  - *F(X)* = Probability(*X*): Probability estimation, given an observation of an input, a probability distribution over a set of classes, rather than only outputting the most likely class that the observation should belong to. E.g Naïve Bayes.
  - Ranking: the target variable *y* is an ordering of elements within a group, such as the order of pages in a search-result page. E.g. search and recommendations system

Training and testing



Data acquisition

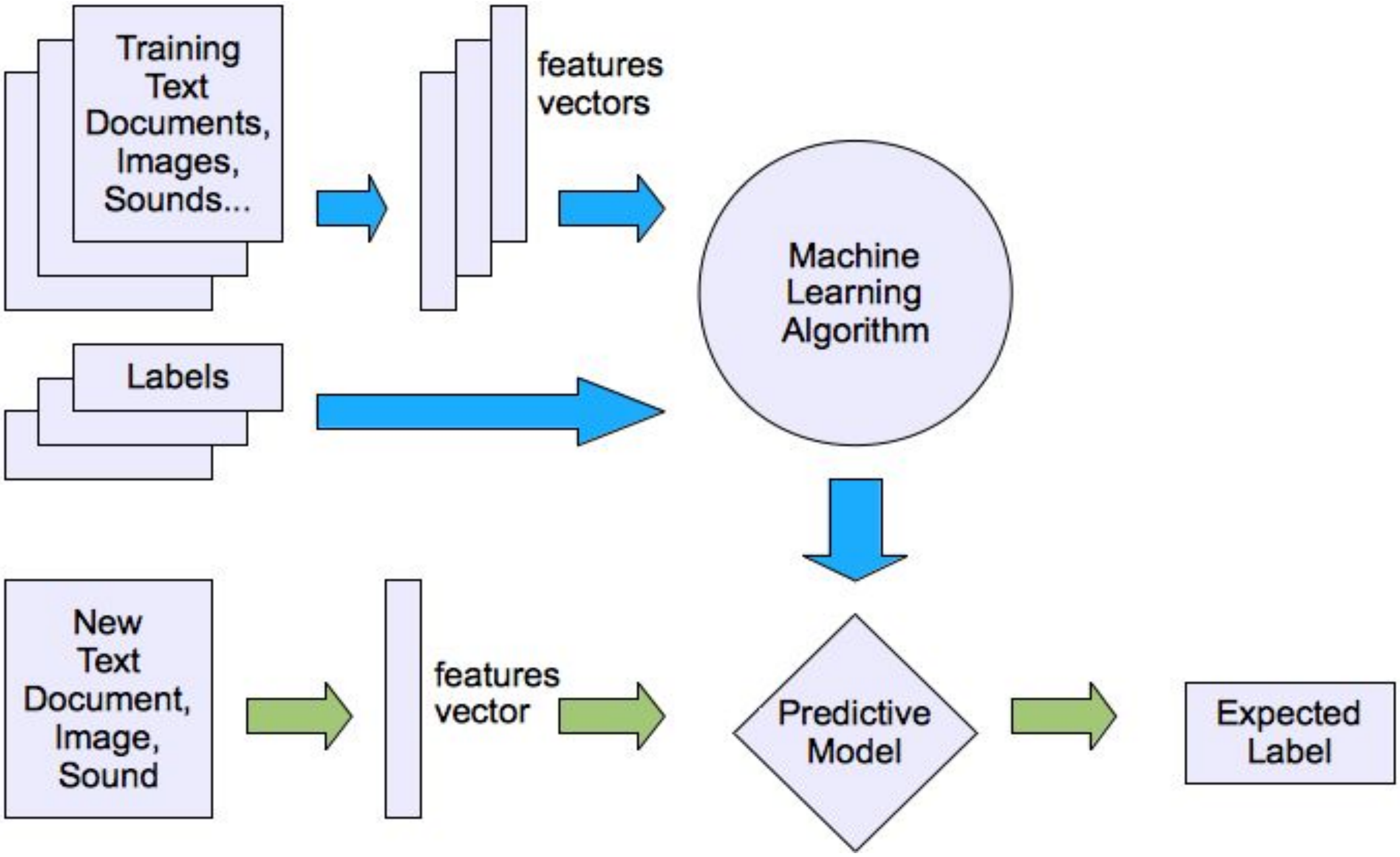
Practical usage

Universal set
(unobserved)

Training set
(observed)

Testing set
(unobserved)

Supervised learning

- 
Training Text Documents, Images, Sounds...
features vectors
Machine Learning Algorithm
Labels
New Text Document, Image, Sound
features vector
Predictive Model
Expected Label

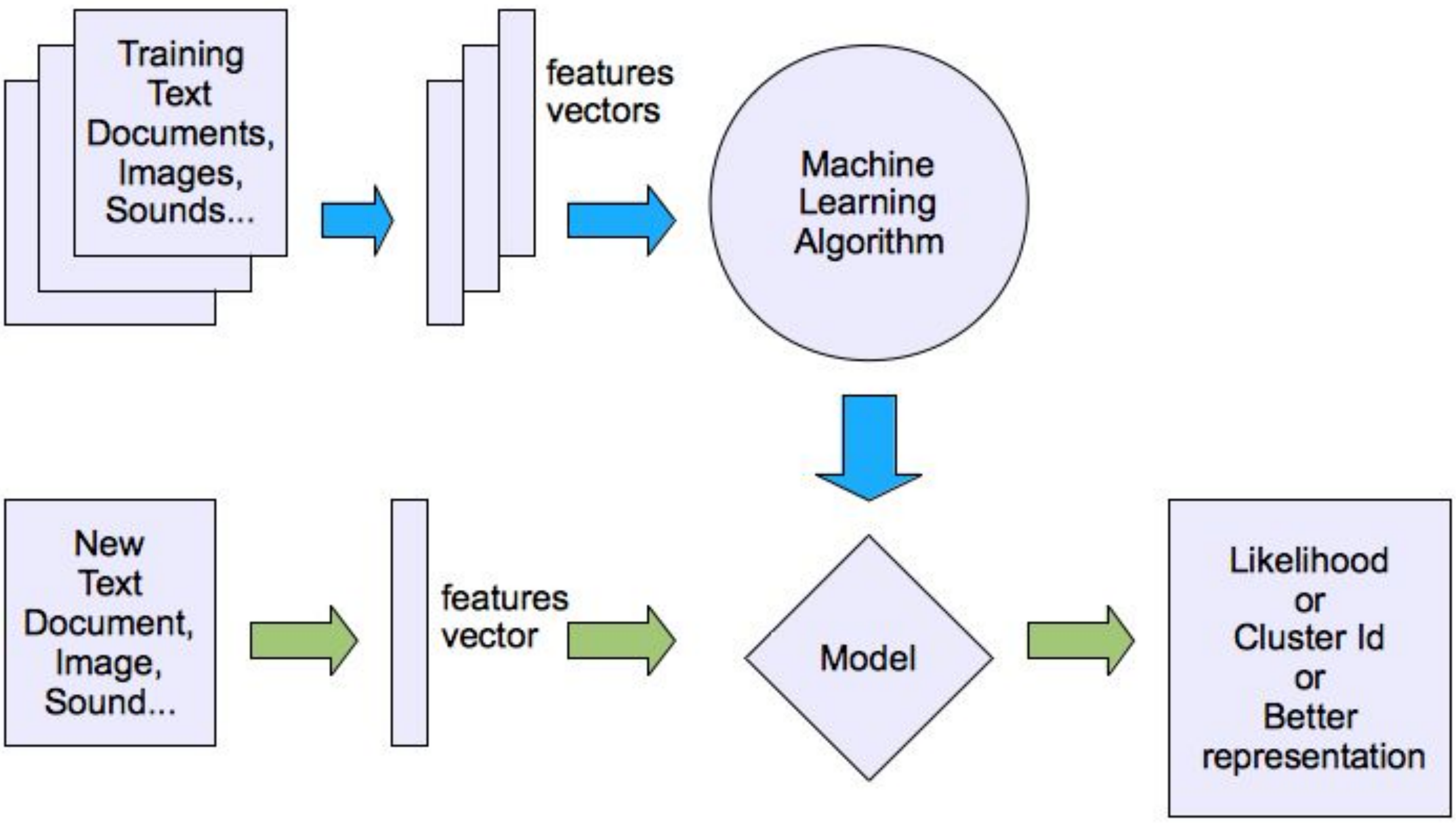
Un-Supervised Learning

- **Given** Data *(X)* , **Identify Patterns in data sets containing data points that are neither classified nor labeled**.
- clustering, association, and dimensionality reduction.
- Clustering is a data mining technique which groups unlabeled data based on their similarities or differences.
- Exclusive clustering/ “hard” clustering is a form of grouping that stipulates a data point can exist only in one cluster. E.g.. K-means clustering algorithm.
- Overlapping clusters differs from exclusive clustering in that it allows data points to belong to multiple clusters with separate degrees of membership. E.g. “Soft” or fuzzy k-means clustering.
- Density estimation- Want to estimate probability of feature vectors, simplest method->histogram

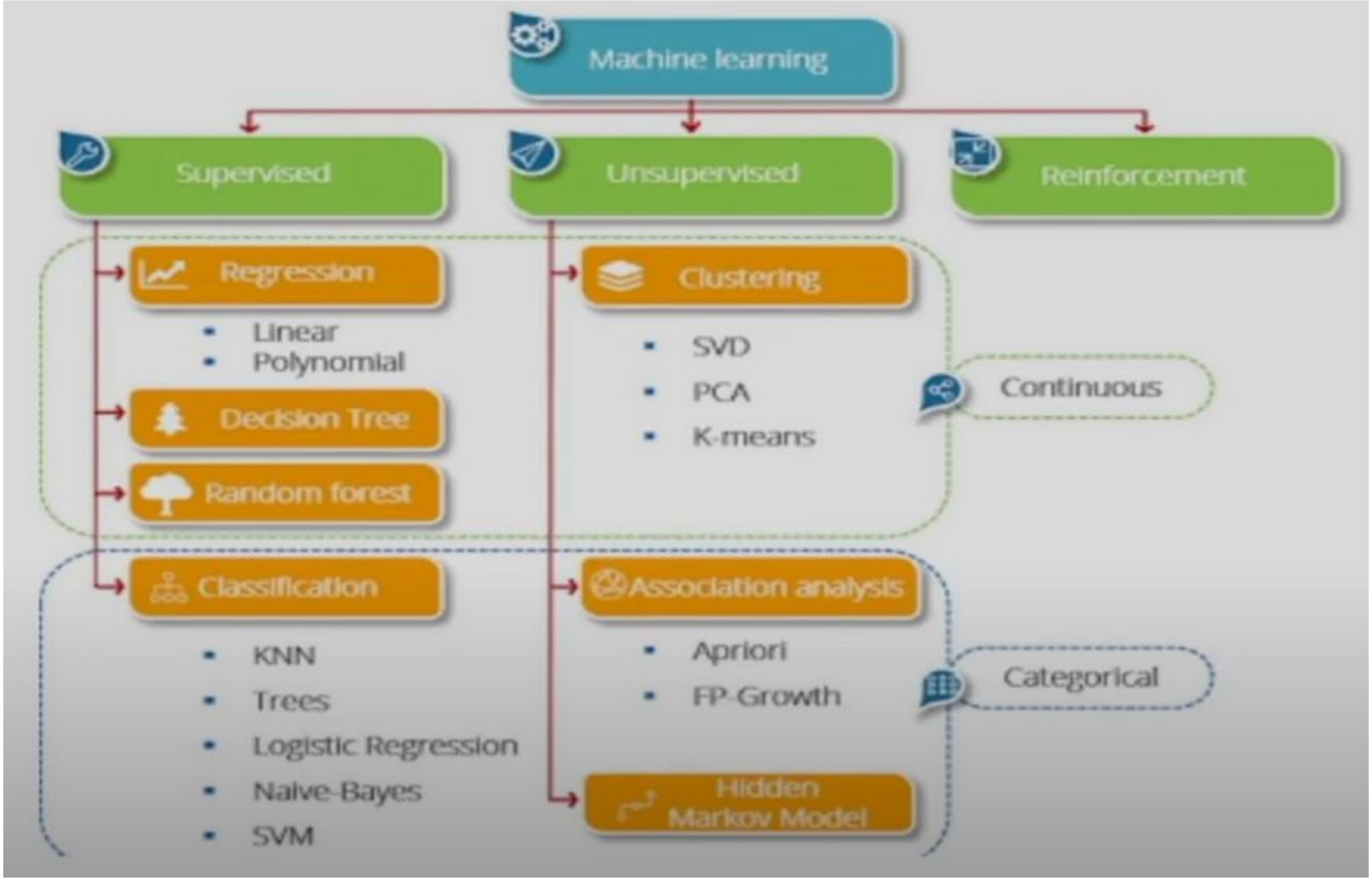
Un-Supervised Learning

- An association rule
  - rule-based method for finding relationships between variables in a given dataset.
  - used for market basket analysis, allowing companies to better understand relationships between different products.
  - Understanding consumption habits of customers enables businesses to develop better cross-selling strategies and recommendation engines.
  - E.g. Amazon’s “Customers Who Bought This Item Also Bought” or Spotify’s "Discover Weekly" playlist.
  - Algorithms used to generate association rules, such as Apriori, FP-Growth, the Apriori algorithm.
- Dimensionality reduction
  - used when the number of features, or dimensions, in a given dataset is too high.
  - It reduces the number of data inputs to a manageable size while also preserving the integrity of the dataset as much as possible.
  - It is commonly used in the preprocessing data stage. E.g. Principal component analysis (PCA).

Unsupervised learning


Training Text Documents, Images, Sounds...
features vectors
Machine Learning Algorithm
New Text Document, Image, Sound...
features vector
Model
Likelihood or Cluster Id or Better representation

Machine Learning Methods



Machine learning

Supervised

Unsupervised

Reinforcement

Regression

- Linear
- Polynomial

Decision Tree

Random forest

Classification

- KNN
- Trees
- Logistic Regression
- Naive-Bayes
- SVM

Clustering

- SVD
- PCA
- K-means

Association analysis

- Apriori
- FP-Growth

Hidden Markov Model

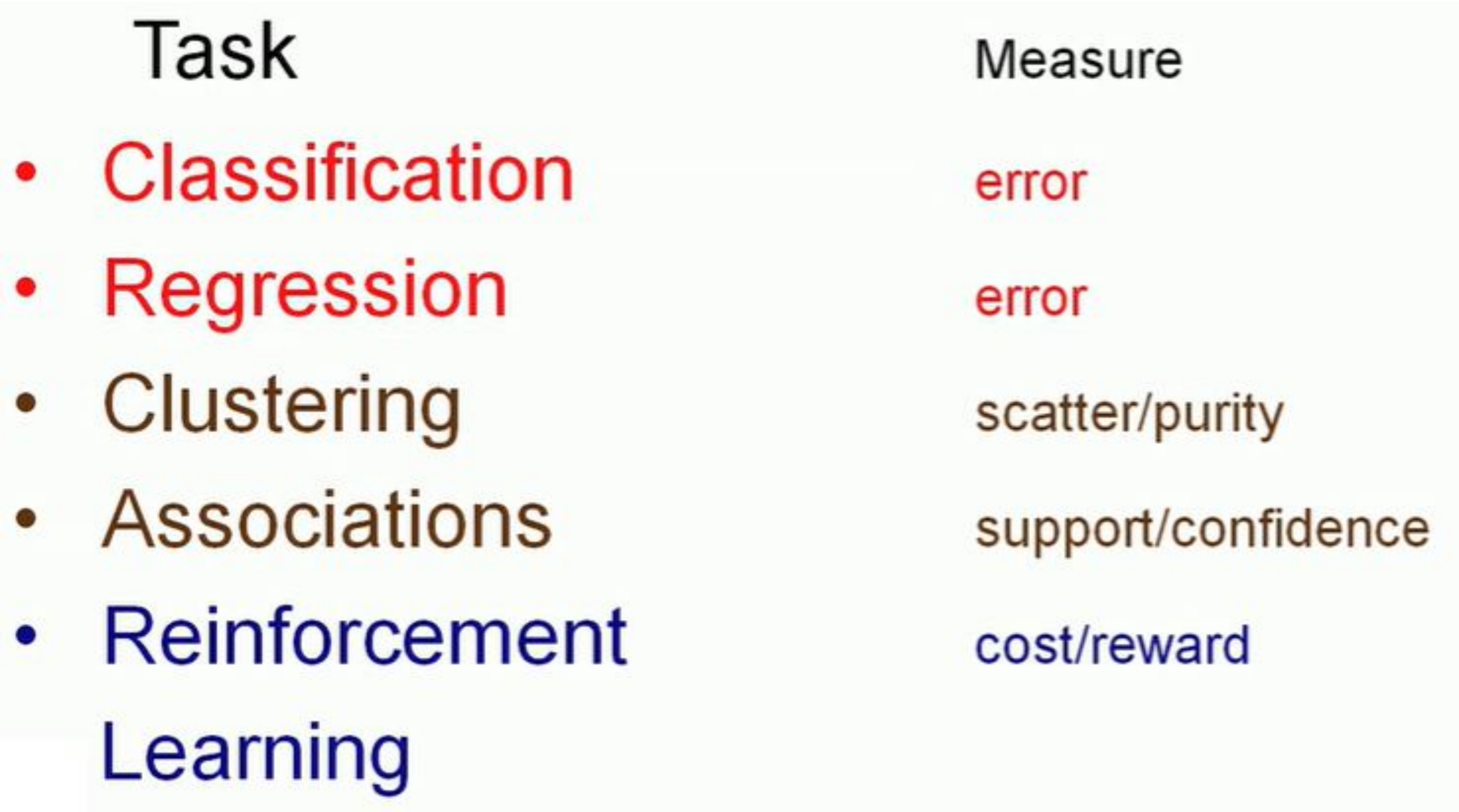
Continuous

Categorical

Types of Data

- X-> vectors or matrices
- Numerical
- Categorical
- Time series –data collected over time like stocks or weather
- Text – textual format like emails, social media posts, chat logs
- Image – represented as a matrix of pixel values, where each pixel's intensity is represented by a numerical value
- Audio –data from audio signals represented as a series of numbers. each representing amplitude of the audio waveform at a specific point in time
- Video –consists of sequence of frames, where each frame is a still image captured at a specific point in time
- Geo-spatial –associated with a specific location on the Earth's surface etc.
- Type conversion (or encoding) is possible through mathematics!
- For e.g.: Character level encoding by representing words with individual characters or n-grams (sequences of characters) numerically.

Types of Learning – performance measures



Task	Measure
Classification	error
Regression	error
Clustering	scatter/purity
Associations	support/confidence
Reinforcement Learning	cost/reward

Types of Learning Tasks

- **Supervised Learning:**
  - Tasks: Regression, classification.
  - **minimizing a cost function, such as mean squared error (MSE), to find the best-fitting line or curve to the data.**
  - **gradient descent for optimization, minimization of cross-entropy loss**
- **Unsupervised Learning:**
  - Tasks: Clustering, dimensionality reduction.
  - **involves distance metrics, matrix theory**
- **Reinforcement Learning:**
  - Tasks: Game playing, robotics.
  - **Dynamic programming, Markov decision process, Probability theory to capture uncertainty etc.**

Applications

- Classification: Determine which discrete category the example is.
- Prediction: Regression
- Recognizing patterns: Speech Recognition, facial identity, etc.
- Recommender Systems: Noisy data, commercial pay-off (e.g., Amazon, Netflix).
- Information retrieval: Find documents or images with similar content.
- Computer vision: detection, segmentation, depth estimation, optical flow, etc.
- Robotics: perception, planning, etc.
- Learning to play games
- Recognizing anomalies: Unusual sequences of credit card transactions, panic situation at an airport.
- Spam filtering, fraud detection
- Many more….

EXERCISES

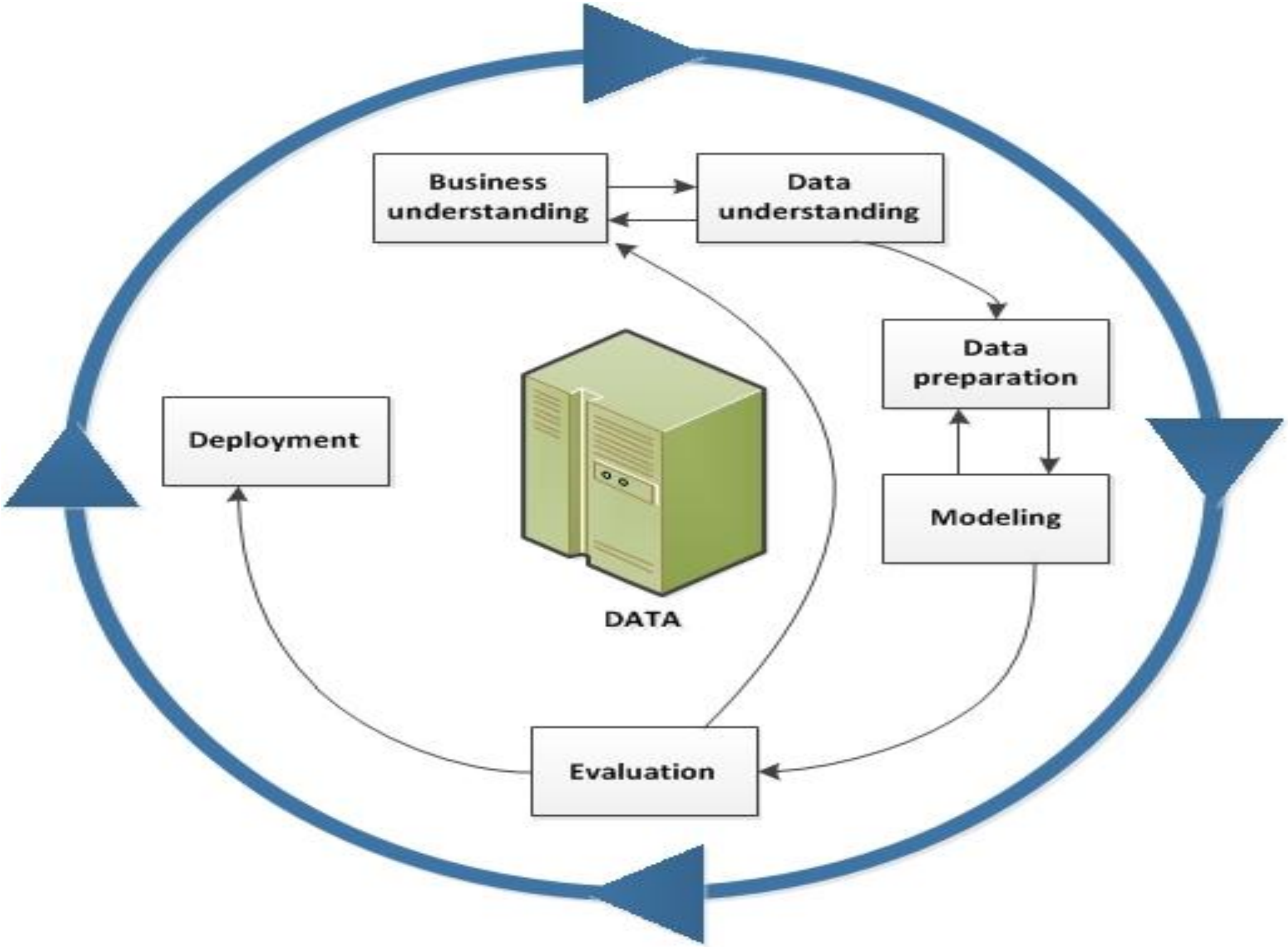
- Consider each of the following machine learning problems. Would the problem be best approached as a classification problem or a regression problem?
  - Predicting the number of fish caught on a commercial fishing voyage
  - Finding protest activities from images
  - Using weather and population data to predict bicycle rental rates

Machine learning project

- starts with understanding the problem and then moves into data preparation, training the model, and evaluating the results. Finally, the model goes to deployment.
- This process is iterative.
  - Get the Data
  - Process the data
  - Split into training set; test set
  - Determine representation of input features; output
  - Choose form of model
  - Decide how to evaluate the system's performance: objective function
  - Set model parameters to optimize performance
  - Evaluate on test set

CRISP-DM process: Cross-Industry Standard Process for Data Mining

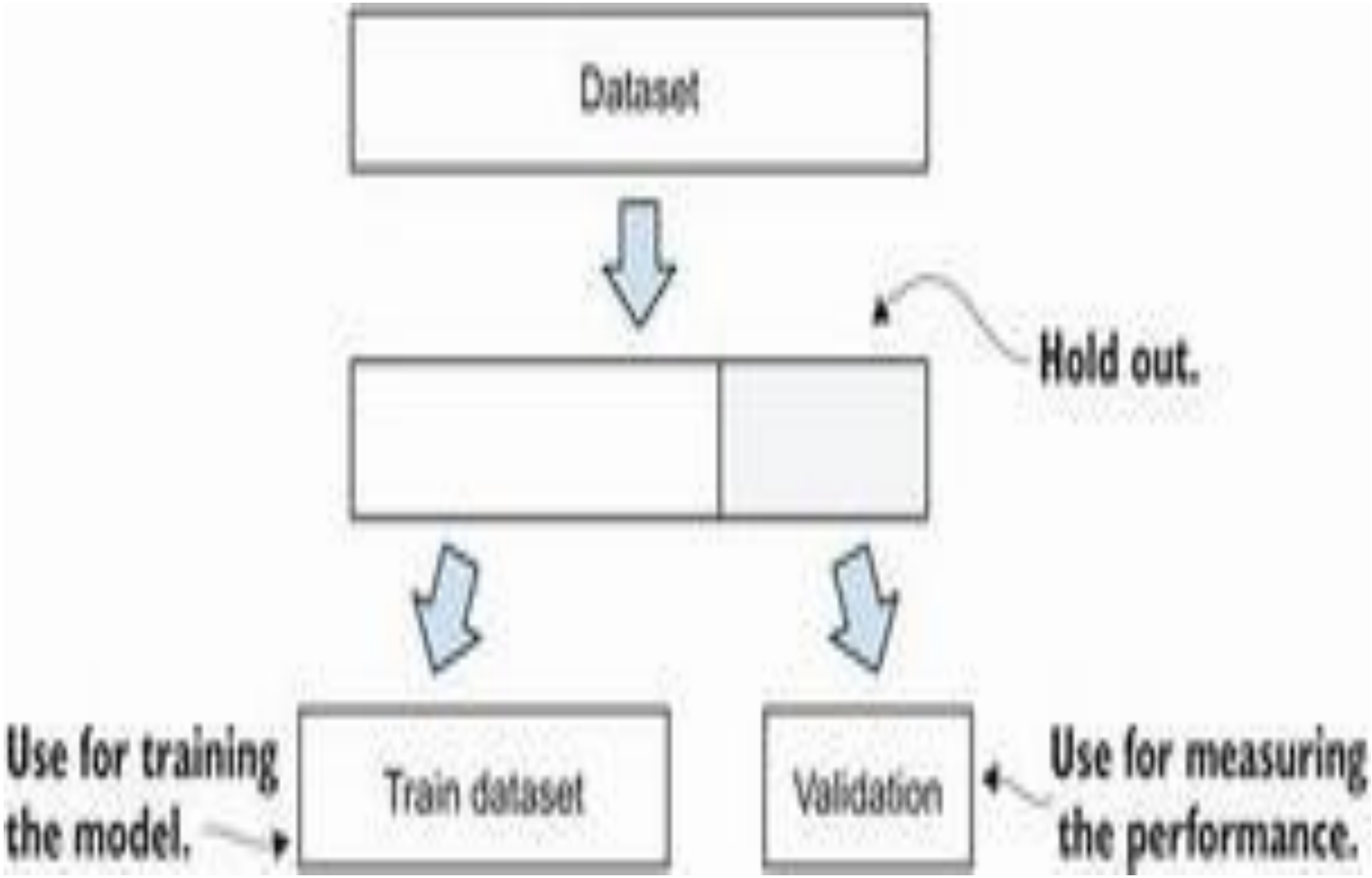
is an industry-proven process model with six phases that describes the data science life cycle.



Steps

- business understanding step: Analyze the problem and the existing solution and try to determine if adding machine learning to that system will help us stop spam messages. Define the goal and how to measure it.
- data understanding: analyze available datasets and decide if need to collect more data, dataset is too small /too noisy
- data preparation: transform the data into a tabular form that can be used as input for a machine learning model
- Modeling: decide which machine learning model to use and how to make sure that we get the best out of it, validate, choose, go back and process data/add new features

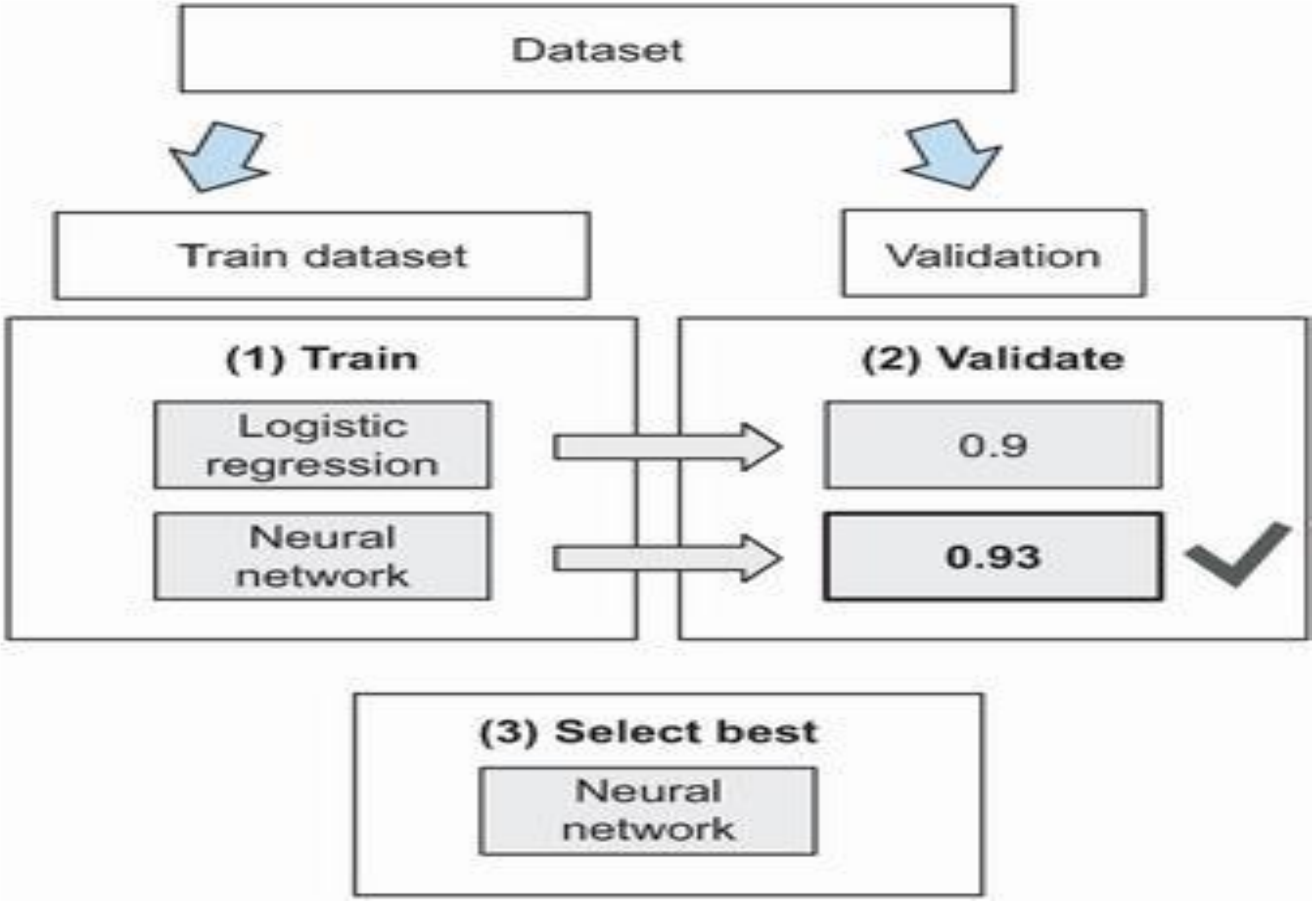
Steps- Training and Evaluation


Dataset
Hold out.
Use for training the model.
Train dataset
Validation
Use for measuring the performance.

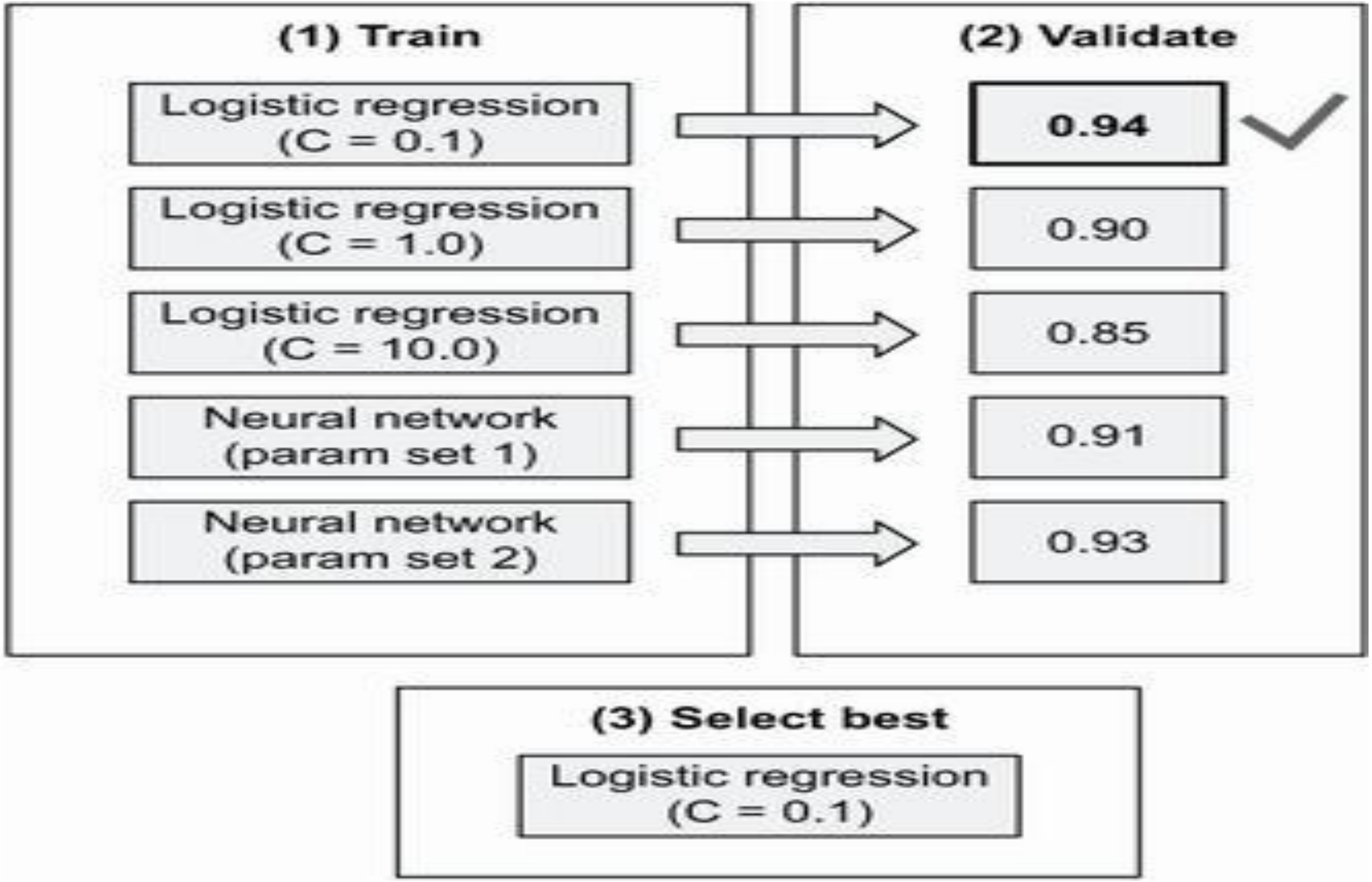
Steps

- Evaluation: check whether the model lives up to expectations, choose metrics, validation
- deployment: best way to evaluate a model is to roll it out to a fraction of users and then check whether our business metric changes for these users. Then deploy the model to the production environment.
- Iterate: refine the original problem, and change it based on the learned information. Rethink the problem and see what can be done better in the next iteration –add/remove features/more data.

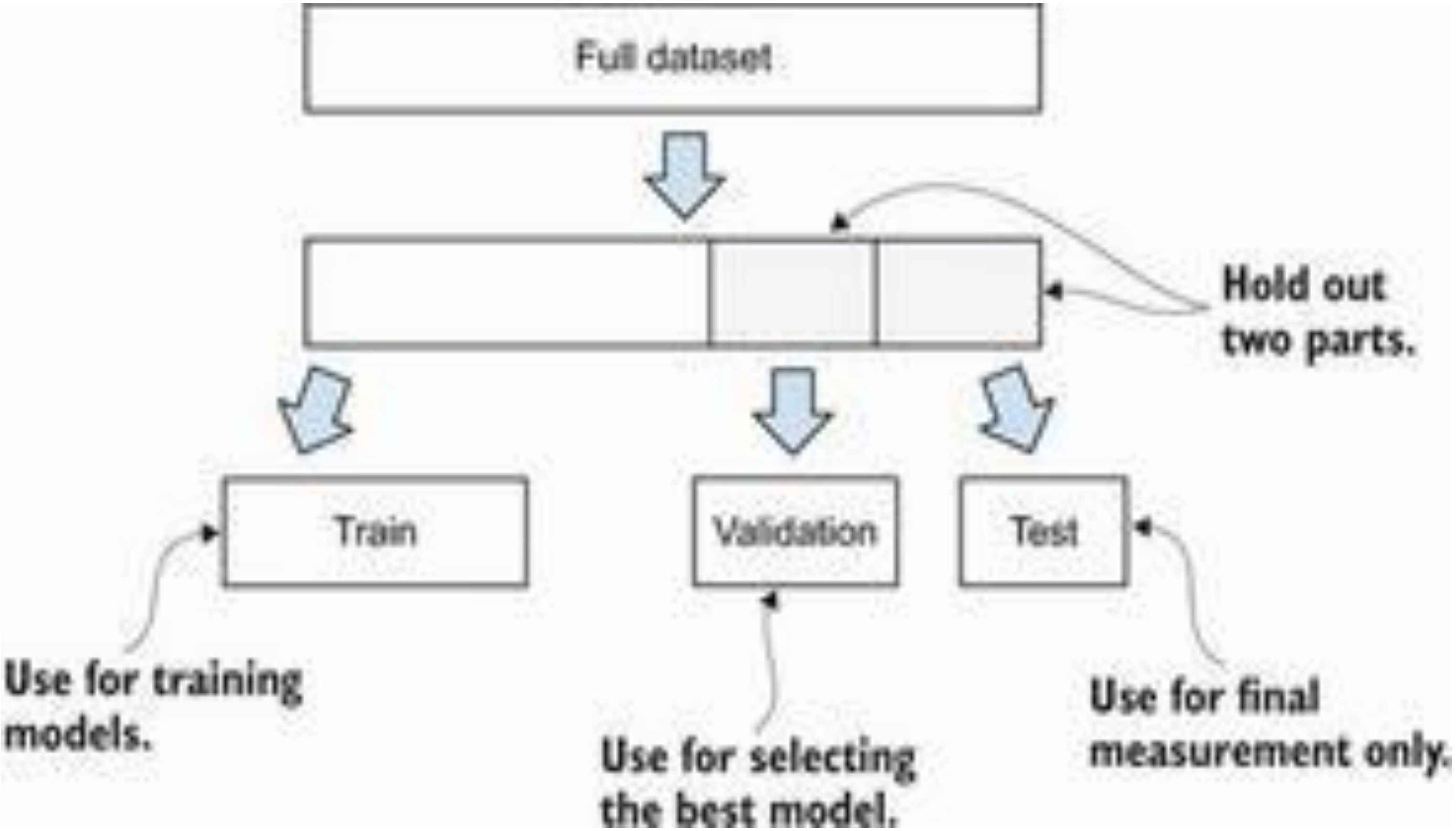
Validation-Example



Validation-Example


(1) Train
Logistic regression (C = 0.1)
Logistic regression (C = 1.0)
Logistic regression (C = 10.0)
Neural network (param set 1)
Neural network (param set 2)
(2) Validate
0.94
✓
0.90
0.85
0.91
0.93
(3) Select best
Logistic regression (C = 0.1)

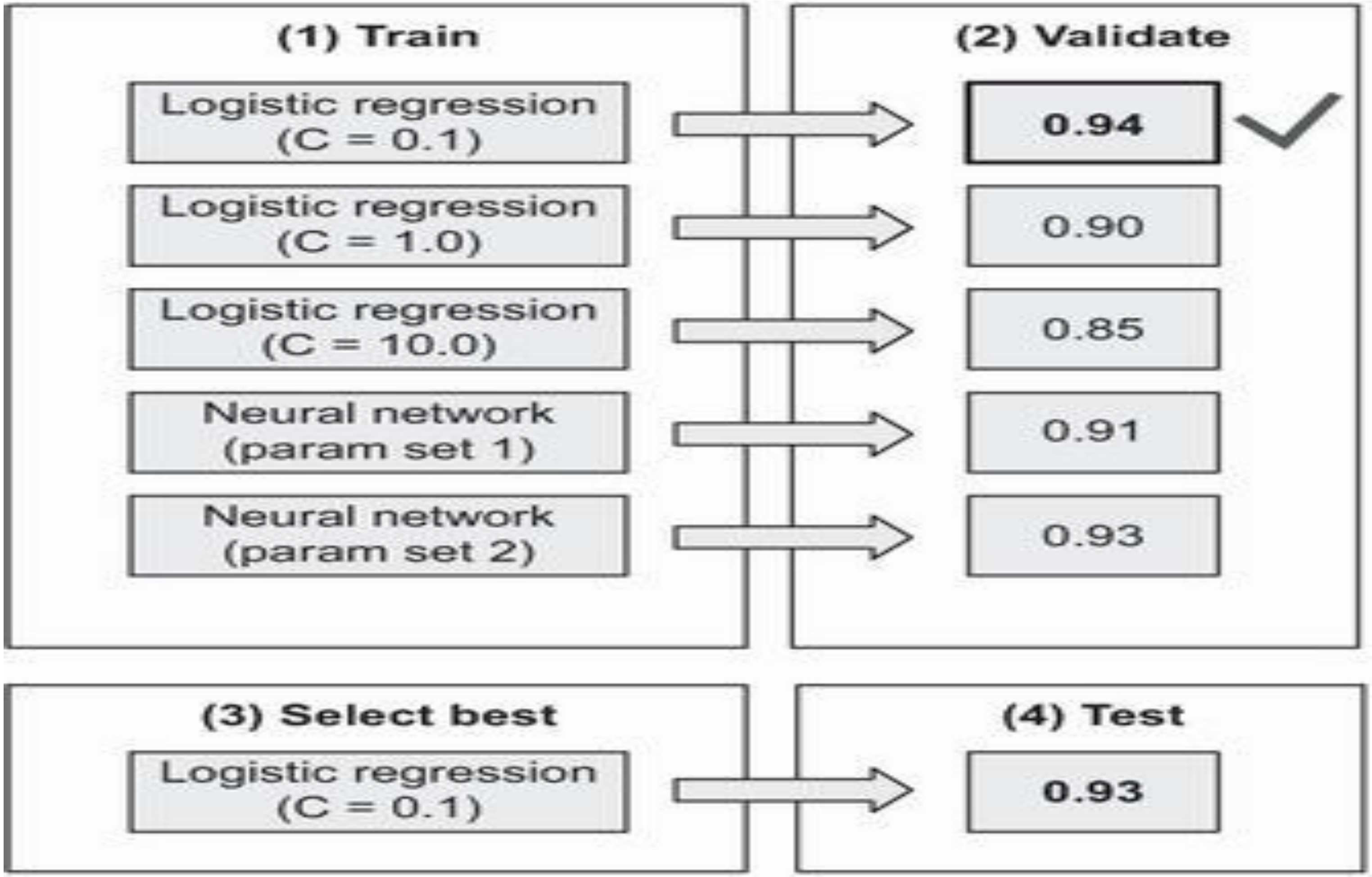
validation data-problem???



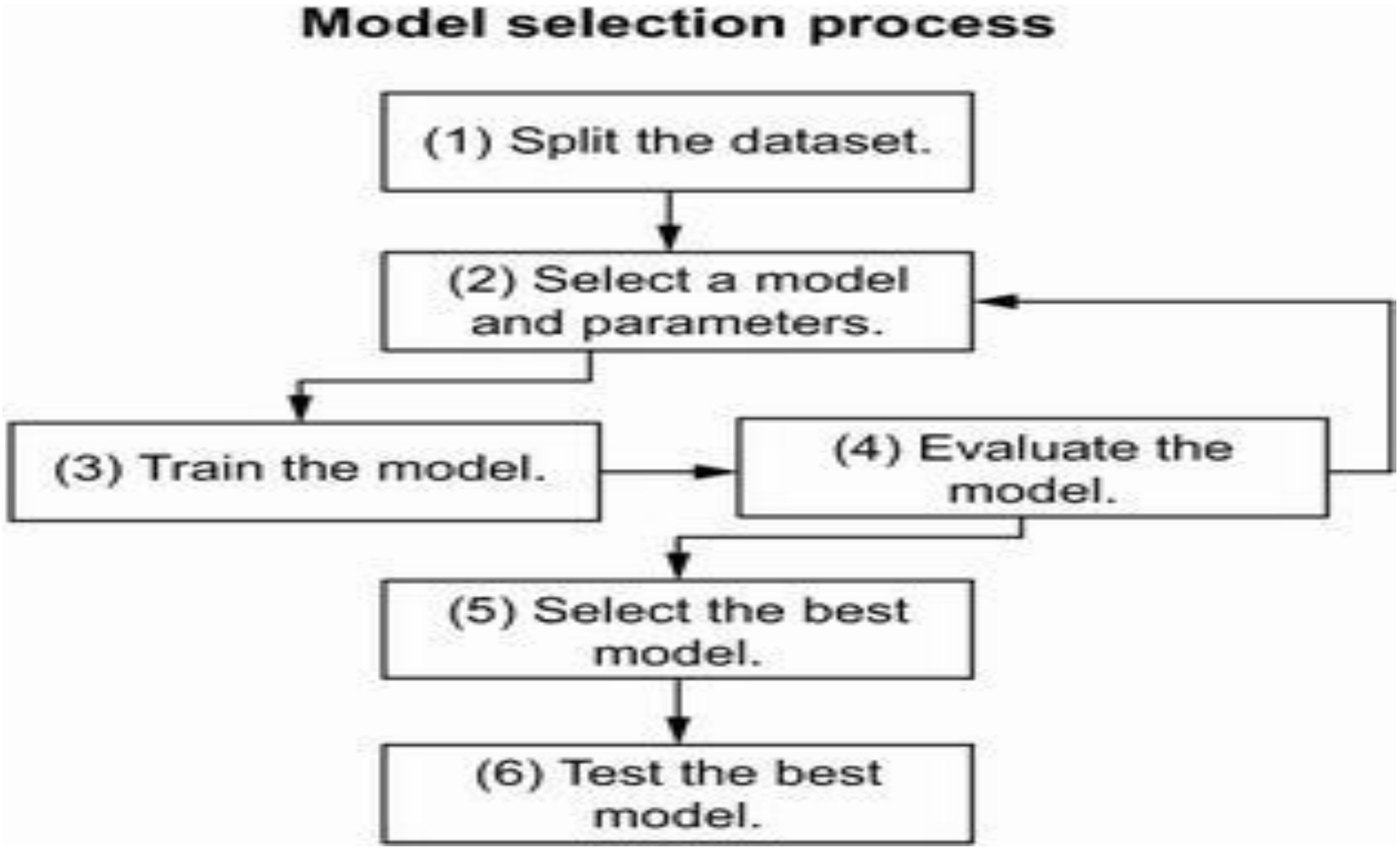
validation data-problem???

- If we repeat the process of model evaluation over and over again and use the same validation dataset for that purpose, the good numbers we observe in the validation dataset may appear just by chance.
- the "best" model may simply get lucky in predicting the outcomes for this particular dataset.
- Solution- hold out part of the data as the *test* dataset. use only for testing the model that we selected as the best

determine which model is the best and test it on the test dataset


(1) Train
Logistic regression (C = 0.1)
Logistic regression (C = 1.0)
Logistic regression (C = 10.0)
Neural network (param set 1)
Neural network (param set 2)
(2) Validate
0.94
✓
0.90
0.85
0.91
0.93
(3) Select best
Logistic regression (C = 0.1)
(4) Test
0.93

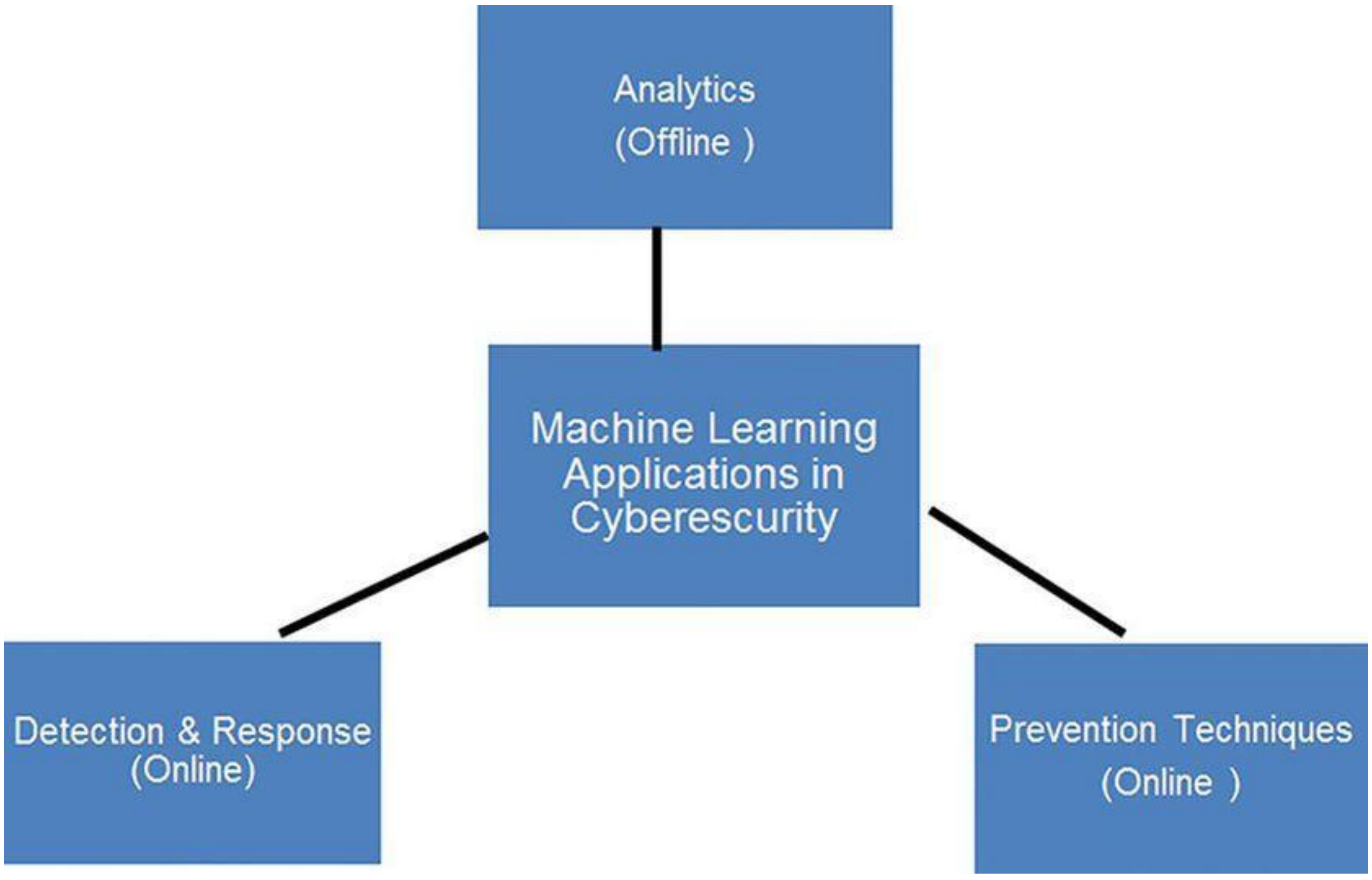
Process of selecting the best model and the best parameters for the model



Summarize - model selection

- split the data into training, validation, and testing parts.
- train each model first on the training part and then evaluate it on validation.
- Each time while train a different model, record the evaluation results using the validation part.
- At the end, determine which model is the best and test it on the test dataset.

Scope of machine learning in cybersecurity defense mechanisms



Analytics
(Offline)

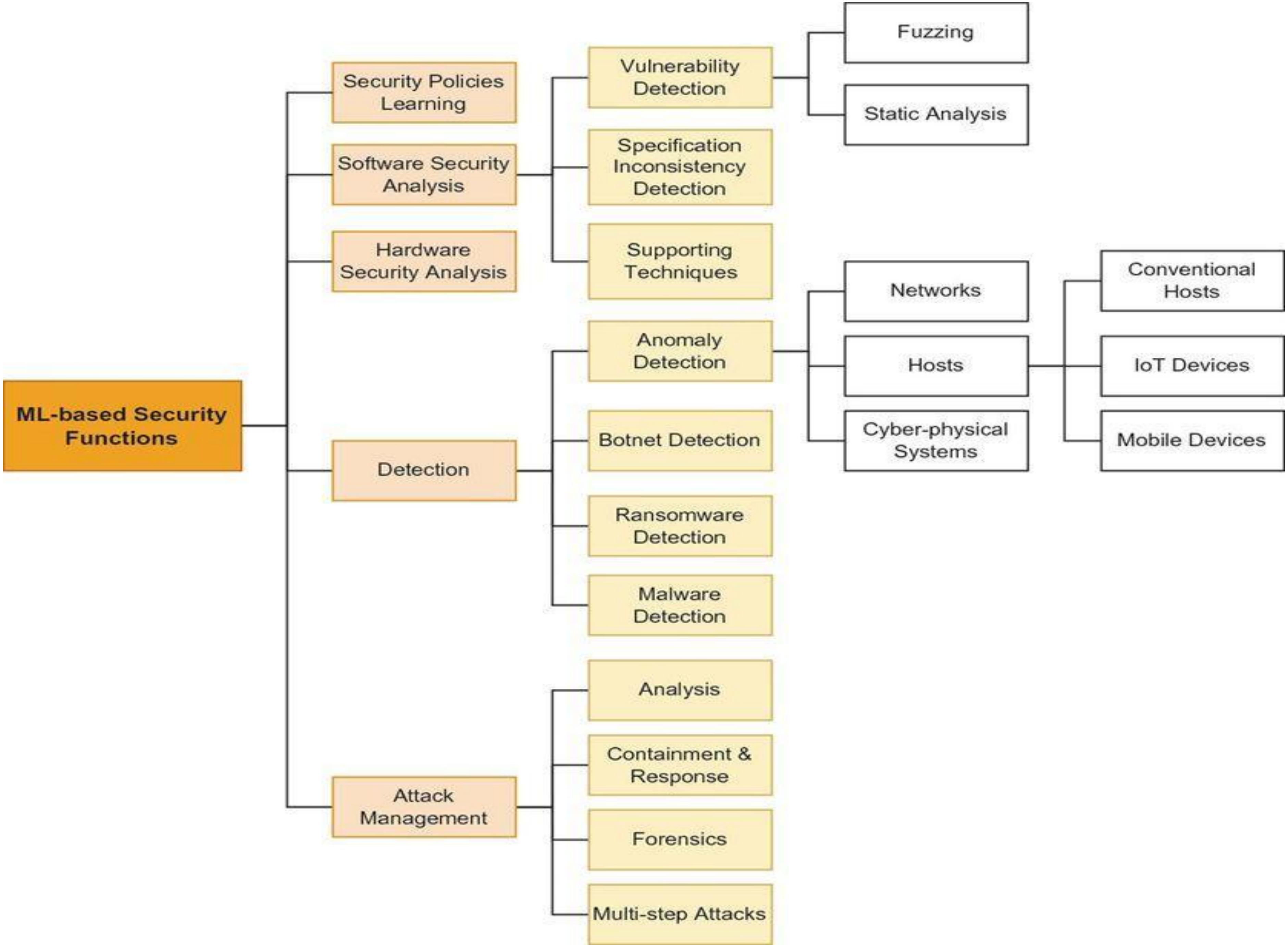
Machine Learning Applications in Cyberescurity

Detection & Response (Online)

Prevention Techniques (Online)

Scope of machine learning in cybersecurity defense mechanisms

- ML for analytics in cybersecurity
  - Source of data for analytics is very diverse and can range from network traffic, endpoint systems, user behavior data, applications, identity and access management systems, and threat intelligence sources.
- ML for prevention in cybersecurity:
  - leverage the intelligence gained from offline analytics to prevent any untoward incident in real time.
  - include firewalls, intrusion prevention systems, identity threat prevention methods, antimalware tools, penetration testing of software, encryption, detecting vulnerabilities, honeypots, firewalls, and intrusion detection systems and employee awareness.
- ML for detection and response in cybersecurity
  - detecting attacks at the earliest, correctly and quickly estimate the scope, scale and impact of the attack, identify the exact technical cause of the attack and eliminate it, and find all possible ways to contain the attack and stop it from spreading further.
  - requires combining information from usage logs, packet information, network behavior, user behavior, identity, and fraud data from across various domains

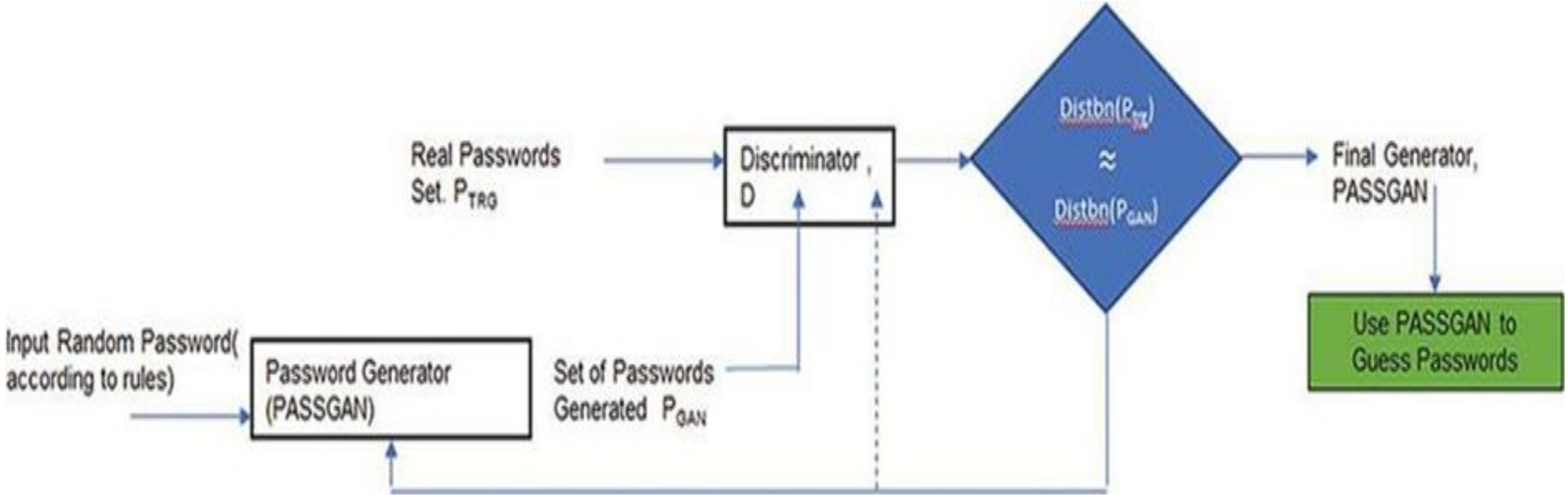

ML-based Security Functions
Security Policies Learning
Software Security Analysis
Hardware Security Analysis
Detection
Attack Management
Vulnerability Detection
Specification Inconsistency Detection
Supporting Techniques
Anomaly Detection
Botnet Detection
Ransomware Detection
Malware Detection
Analysis
Containment & Response
Forensics
Multi-step Attacks
Fuzzing
Static Analysis
Networks
Hosts
Cyber-physical Systems
Conventional Hosts
IoT Devices
Mobile Devices

Attack-wise ML-based defense strategies

- Illicit access to a resource
- Phishing URL detection, extracted URL-based features, fake login detection, hyperlinks, copied Cascaded Style Sheets (CSS), and fake web identity features (from copyright, favicon, and identity keywords)
- Ebbu2017 Phishing dataset
- Spam mails
- Web vulnerability: SQL injection vulnerability in web pages
- Insider threat detection -illicit data transfer and damaging or misuse of organizational resources, logs, CERT insider threat datasets
- Identity and information theft –
  - Digital ID may get stolen via phishing, vishing, credit card skimming, public Wi-Fi hotspots, and unintentional username and password exposure,
  - Equifax data breach in the USA and Domino’s data breach in India are recent examples, Many cracking tools check passwords by searching heuristically in password space.
  - Apart from dictionaries, various password cracking tools also create and generate more probable passwords by using contextual information like age, date of birth, and family member name.

Example: Deep learning-based PassGAN

- makes use of character distribution and placement in actual passwords to train a generative adversarial network (GAN) and as a result output realistic guesses to crack passwords with higher chance
- Discriminator processes passwords in training dataset and those produced by generators through residual networks.
- Based on the discriminator’s feedback, generator adjusts its parameters to output samples in distribution similar to actual passwords.
- After many iterations and epochs of training, the generator converges and can be used to generate real password guesses.
- Network administrators can run password files through PassGAN and all users whose passwords are guessed correctly can be prompted for a password change

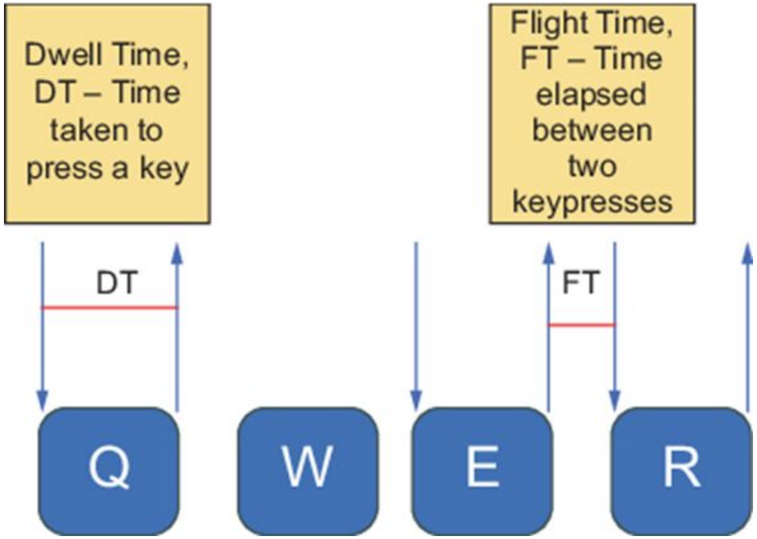

Real Passwords Set. P_{TRG}
Discriminator , D
Distbn(P_{TRG})
≈
Distbn(P_{GAN})
Final Generator, PASSGAN
Use PASSGAN to Guess Passwords
Input Random Password(according to rules)
Password Generator (PASSGAN)
Set of Passwords Generated P_{GAN}

Attack-wise ML-based defense strategies example

- Tor network
  - largest networks on the unsearchable web.
  - Tor traffic can be traced by using right contextual information.
  - De-anonymizing Tor has been done using ML techniques by collecting features and information from individual sessions to be able to identify the activity of anonymous users.
  - To uncover activities on the dark web, conmarap/website-fingerprinting repositories are very helpful
- Penetration testing:
  - Metasploit is a popular tool for penetration testing.
  - it has been updated to its deep learning-based version, DeepExploit by the use of reinforcement learning while performing automated deep penetration testing.
  - uncover all vulnerabilities in individual systems and networks.
- ML solutions used to detect sophisticated targeted attacks known as APT
  - In APT, multiple vectors of attack like social engineering for basic info, spear phishing for targeting high authorities, and drive by download for spreading malware are used.

Preventing identity theft by modeling keystroke dynamics

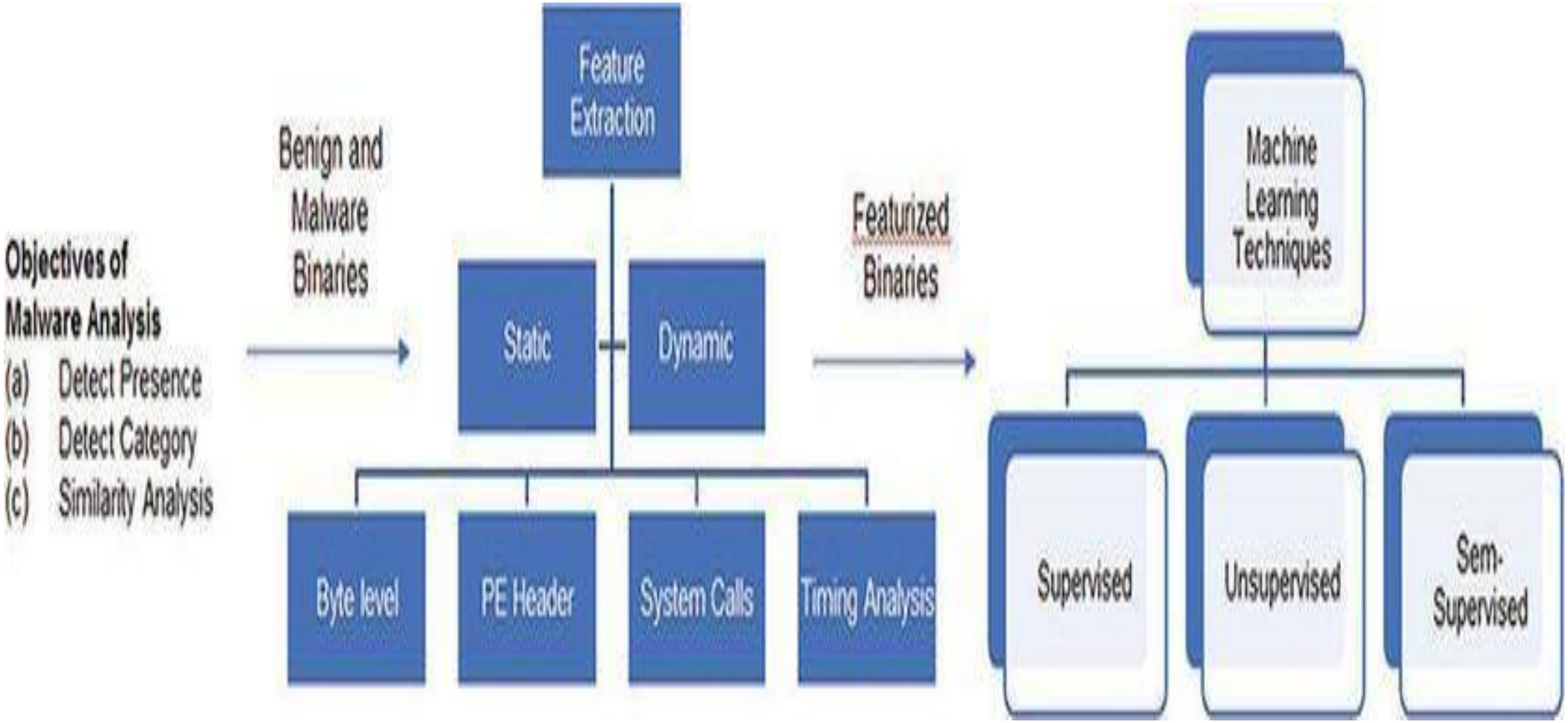
- speed and rhythm of typing can be explicitly profiled for an individual to such an extent that it can become the digital identity of each user.
- A robust keystroke analyzer based on ML can prove to be a good authentication system.
- It can enable password-less access as well as detect identity theft quickly.
- two time-based features are mainly used for capturing keystroke dynamics.
- First parameter called dwelling time is actually the keypress duration.
- The time elapsed between two consecutive keypress and key-release is the second feature and is termed as flight time.


Dwell Time, DT – Time taken to press a key
Flight Time, FT – Time elapsed between two keypresses
DT
FT
Q
W
E
R

Integrity attacks

- access of data illegally but also modifying the content and in some cases encrypting it for ransom, Ransomware detection and prevention
- Software vulnerability analysis is generally done by techniques such as software penetration testing, fuzz testing, and tainted flow testing
- ML -> anomaly detection and pattern recognition on the underlying code.
- Unintentional software errors, like buffer overflow, -> source of major cyberattacks.
- Deep learning-based system has been proposed to automatically detect such issues in the underlying legacy code especially for C and C++. This is cheaper and more effective than a software tester manually looking for these unintentional flaws.
- VulDeePecker is a deep learning-based tool to detect buffer overflow vulnerabilities.
- DeepFake recognition is an emerging technology for detecting videos tampered using DL
- fake news detection
- Software testing
  - highly manual process, and thus human expertise and knowledge of software becomes a limiting factor. Fuzzing is an important tool to automate the software testing process in which a large number of inputs are automatically generated to test if any of these cause the program to crash.
  - ML-based fuzzing methods reduce the probability of retesting similar inputs. NEUZZ, based on neural networks, to find inputs that cause programs to crash with greater probability

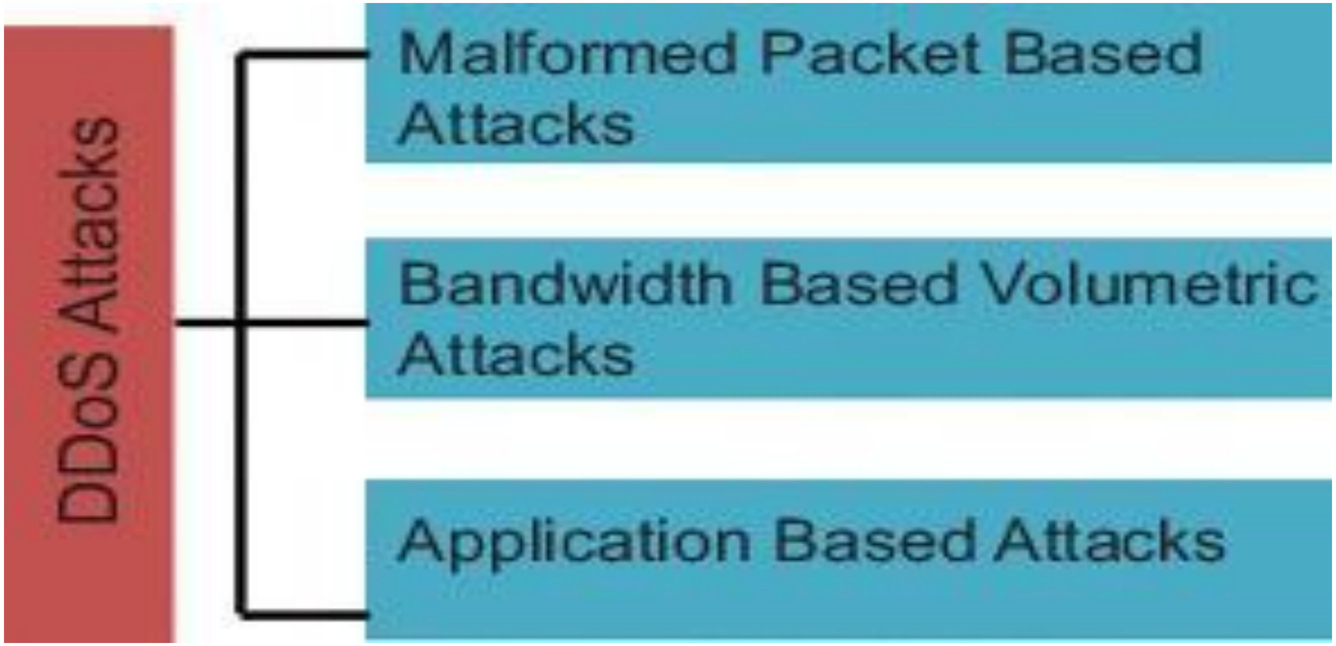
ML techniques for malware analysis



Features such as CPU usage, memory usage, network traffic, and battery usage can be used to detect mobile phone malware.

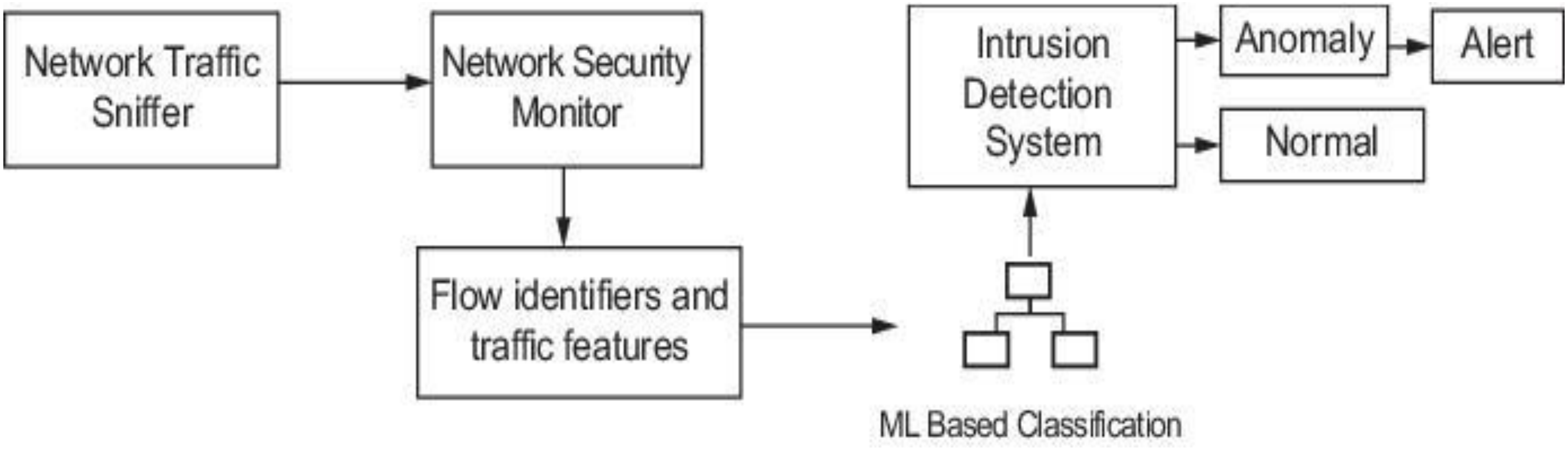
Android malware detection can also utilize permissions sought by apps as features followed by application of ML algorithms for binary classification to goodware and malware as well as multiclassifier to the type of malware

Classification of DDoS attacks


DDoS Attacks
Malformed Packet Based Attacks
Bandwidth Based Volumetric Attacks
Application Based Attacks

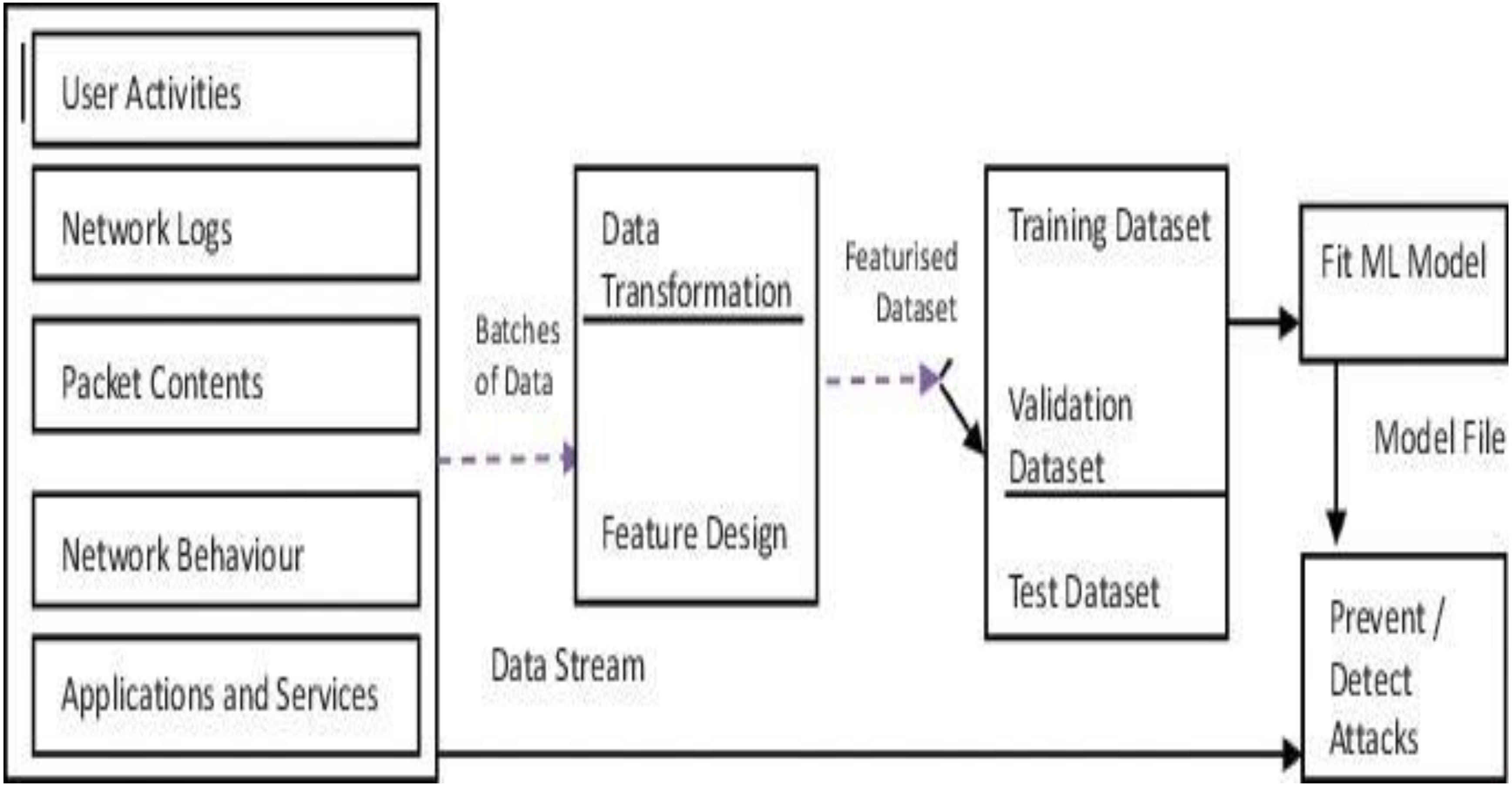
UNSW-NB15 dataset
DDoS attack involves the use of many compromised remote machines known as “bots” in targeting a server.
finding optimal window sizes so as to detect the botnet as soon as it starts compromising the system
checking ingress/Egress traffic

ML-based network behavior anomaly detection



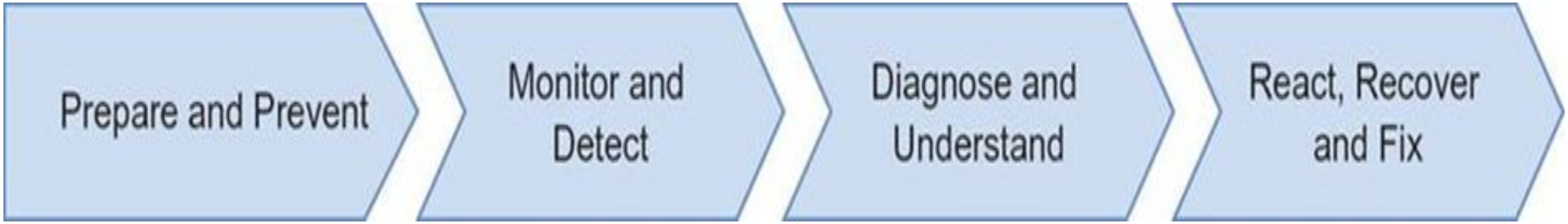
Signature-based and anomalous behavior-based detection are two main methods of detecting known and unknown types of attacks, respectively. Signature-based methods suffer from the drawback of detecting zero-day attacks, and anomaly-based methods find them sometimes difficult to differentiate smart attack traffic like low-rate DDoS from normal traffic.
hybrid techniques
KDD CUP99 benchmark dataset

Case study: machine learning-based defense strategy



spam detection, malware detection, and botnet detection

Security life cycle



- preparation phase -> ML techniques can be used in learning access control policies from data and context; guiding fuzzing techniques to discover application vulnerabilities, ensuring high coverage, and enhancing application testing; identifying where to allocate security resources to maximize security and minimize its cost.
- monitoring activities -> ML techniques can be used to build anomaly detection systems
- diagnostic activities -> approaches based on federated ML techniques and causality reasoning techniques are critical
- response activities -> ML reinforcement learning techniques are critical to determining the response actions that best contain the attack and minimize the attack damages

Data problems

- ML system is as good as the data it gets
- relevance and quality of its training dataset
- On real traffic data, general-purpose classifiers are not very accurate as compared to specific-attack ML-based detectors
- Lack of standard benchmarking datasets
- Lab versus real traffic
- Controlled, repeatable, and efficient processes for capturing datasets from the application domain for updated learning at regular intervals are difficult in practical.
- After a period of running, ML systems may become irrelevant (also known as “model rot”), as their performance may deteriorate with time. This may happen due to the evolving input data owing to some external effects

Model problems

- Vulnerability to adversarial attacks:
  - Adversaries are using novel strategies to evade detectors based on ML algorithms.
  - They may target to attack integrity, availability, and privacy of ML-based system. Integrity violations lead to misclassifications.
  - Availability violations may make the model look rogue by raising false alarms at a high rate.
  - Privacy attacks are about acquiring information not supposed to be public.
- Lack of new features design-> distinguishability for various types of threats/hand-engineered.
- Problems like polymorphism have been studied from malware point of view but not for other security applications like spam detection, phishing detection, and intrusion detection
- resistant to data quality problems like bias in data and label inaccuracy.

Model problems

- Model extraction attack: With access only to test samples and results from the black-box classifier, labeled training dataset can be generated with which we can train a local substitute model. This local substitute model can be analyzed offline to search for samples that belong to adversarial space.
- Warm start problems: ML models may have to be updated on the arrival of significant new data.
- Lack of an integrated model for holistic cybersecurity solutions. Currently, there are multiple models for multiple problems.
- Scaling: ML solutions are proposed on toy problems and getting it working in large live networks is not very smooth.
- None of the ML solutions guarantee threshold false positives (FPs) and true negatives (TNs). A high number of FPs are annoying and false negatives lead to compromise.
- Lack of studies on integration across models and domains for creating collective intelligence.

Model problems

- Lack of semantics: Semantic gap is a problem with ML. Compared to static rulesets or heuristics, it is not directly explainable as to why an event was flagged as an anomaly. This can lead to reduced confidence in predictions. A human-readable explanation for alerts generated by ML systems is a practical requirement that is missing in current systems.
- Increased time to detection in stealthy attacks that hide between normal traffic is yet to be addressed sufficiently.
- Scope of connecting cross-domain information: For example, if there is an increase in the number of DGA domains being detected, this input must be provided to malware detectors. Reduced attacks from some specific adversary also need attention to find out if it is attacking from some other unknown front or has worked out good detection evasion method.

Emerging New ML paradigms

- *Adversarial training*:
  - A well-performing model may misclassify frequently when poisoned with adversarial examples.
  - To counter adversarial attacks, proactive adversarial training can be used.
  - It involves augmenting the training dataset of supervised models with adversarial examples and adding their correct labels.
  - Methods used to poison models can be regenerated in a similar way and preadded to the training models.
- *Reinforcement learning*:
  - In this paradigm of ML, labeled data are not available. Thus, the algorithms learn to make predictions on their own.
  - There are agents who know the start and end states of goal but not the path.
  - A reward-based path finding will every time the agent finds solutions in the most optimal way.
  - In this manner, the agent learns that own.
  - This branch of ML would be very helpful in applying cybersecurity in areas where labeled dataset is not available.
  - For example, processing CCTV camera feeds for real-time detection of suspicious activities can be done by reinforcement learning.
  - intrusion detection, IoT security, network penetration testing, and malware analysis

Emerging New ML paradigms

- *Active learning*:
  - where the learner chooses the dataset for training.
  - desired accuracy can be achieved by a much smaller training size.
  - An active learner may find important data instances and ask queries to obtain their labels.
  - Used where most of the datasets are imbalanced and very few attack instances are labeled.
- *Distributed machine learning (DML)*:
  - network logs, malware, and all other security data emanate at a huge pace.
  - There is a dearth of processing power at some of organization to process all data or train the ML models in a centralized way with all the available data .
  - distributed ML facilitates distributing the ML workload across multiple machines
  - efficient parallelization of the training process and integrating models are addressed

Emerging New ML paradigms

- *Federated machine learning*:
  - While in DML, data available at one location can be distributed to multiple computing power facilities to have parallel learning, in federated learning, datasets at various sites are used to learn small in-place models, and then the learnt models are federated to obtain a robust integrated ML model based on huge variety of training datasets.
  - Organizations can enter into agreements to share models developed on their own datasets and contribute to a federated more robust learner.
  - organizations are wary to share their raw data but may be willing to share models once secure federated learning becomes realizable

Conclusions

- **Mathematics is the foundation of machine learning algorithms, providing the theoretical framework for understanding and developing models.**
- **Aim of Machine Learning Algorithms to extract knowledge from data and help to take an optimistic (feasible) decision with precision.**
- **To adapt in new circumstances it is trying to create intelligence within it.**

Thank you

- ???

References

- https://courses.washington.edu/css490/2012.Winter/lecture_slides/02_math_essentials.pdf
- Christopher Bishop: ”Pattern Recognition and Machine Learning” , 2006
- Kevin Murphy: ”Machine Learning: a Probabilistic Perspective”
- David Mackay: ”Information Theory, Inference, and Learning Algorithms”
- Ethem Alpaydin: ”Introduction to Machine Learning” , 2nd edition, 2010.
- R. Duda, P. Hart & D. Stork, ***Pattern Classification*** (2nd ed.), Wiley T. Mitchell, ***Machine Learning,*** McGraw-Hill