EE P 567: Machine Learning for Cyber Security

Lecture 1: Introduction to Machine Learning (ML) for Cyber Security

Dept. of Electrical and Computer Engineering

University of Washington

Instructor: Prof. Radha Poovendran

E-mail: rp3@uw.edu

ELECTRICAL & COMPUTER ENGINEERING

UNIVERSITY of WASHINGTON

NSL

Machine Learning for Cybersecurity

- What you will learn:
  - **How to use machine learning algorithms to implement cybersecurity concepts**
  - How to implement machine learning algorithms such as k-means clustering, regression and ensemble methods
  - How to use Python libraries - NumPy, and Scikit-learn
  - Understand how to combat malware, detect spam, and cyber anomalies
  - How to use TensorFlow in the cybersecurity domain and implement real- world examples

- **Course Grade will be based upon three homework (45%) and a final team-project (55%)**

Lecture and Lab Schedule

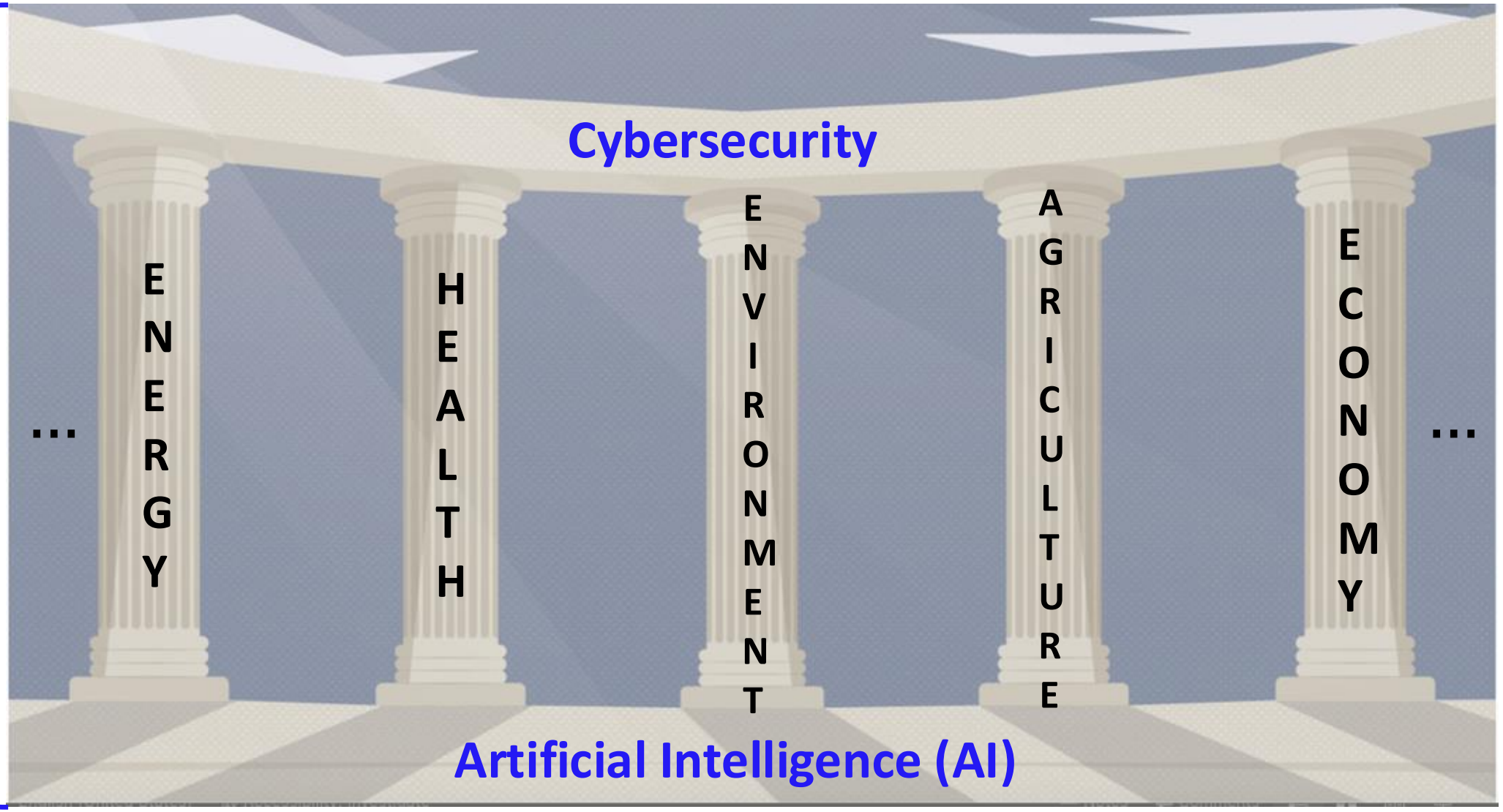
1. Introduction to Machine Learning (ML) for Cyber Security
2. Email Spam Detection using Supervised Learning **(Scheduled Release of HW #1)**
3. Machine Learning for Solving Completely Automated Public Turing Test to Tell Computers and Humans Apart (CAPTCHA)
4. Transformers in Cyber Defense **(Scheduled Release of HW #2)**
5. Data Dimensionality Reduction in Cyber Attack Data
6. Network Anomaly Detection Using Clustering Techniques **(Scheduled Release of HW #3)**
7. Use of NLP for Instruction Set Architecture Identification
8. Credit Card Fraud and Malicious Event Detection Using Decision Trees
9. Ensemble Learning for Online Ad blocking, Program Binary Analysis, and Credit Card Fraud Detection
10. Adversarial Machine Learning
11. Student presentations of Course Projects

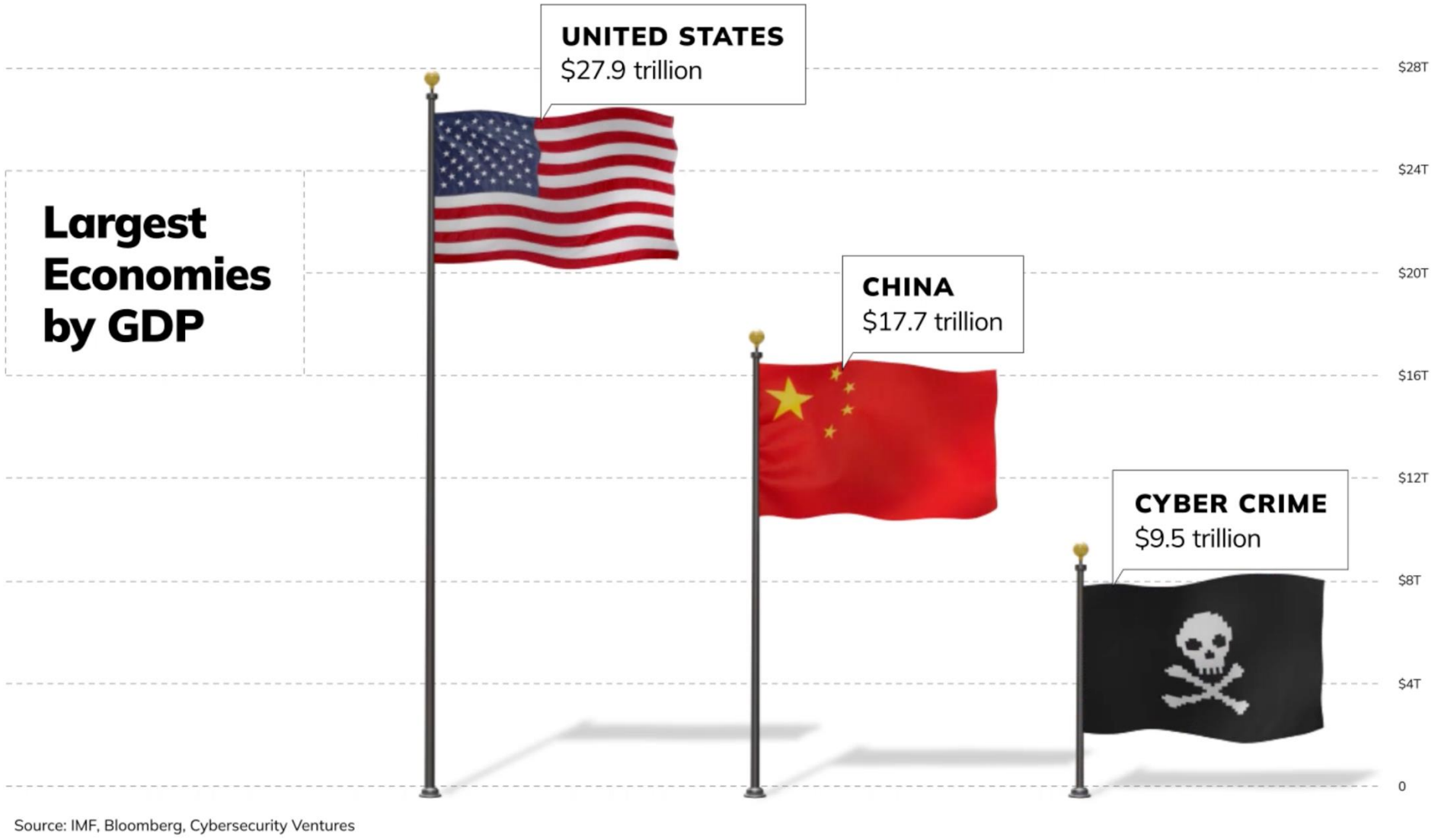
Topics Covered Today

- Cybersecurity Landscape Statistics
- Machine Learning in Cybersecurity
- Different Data Types
- Types of Machine Learning Models

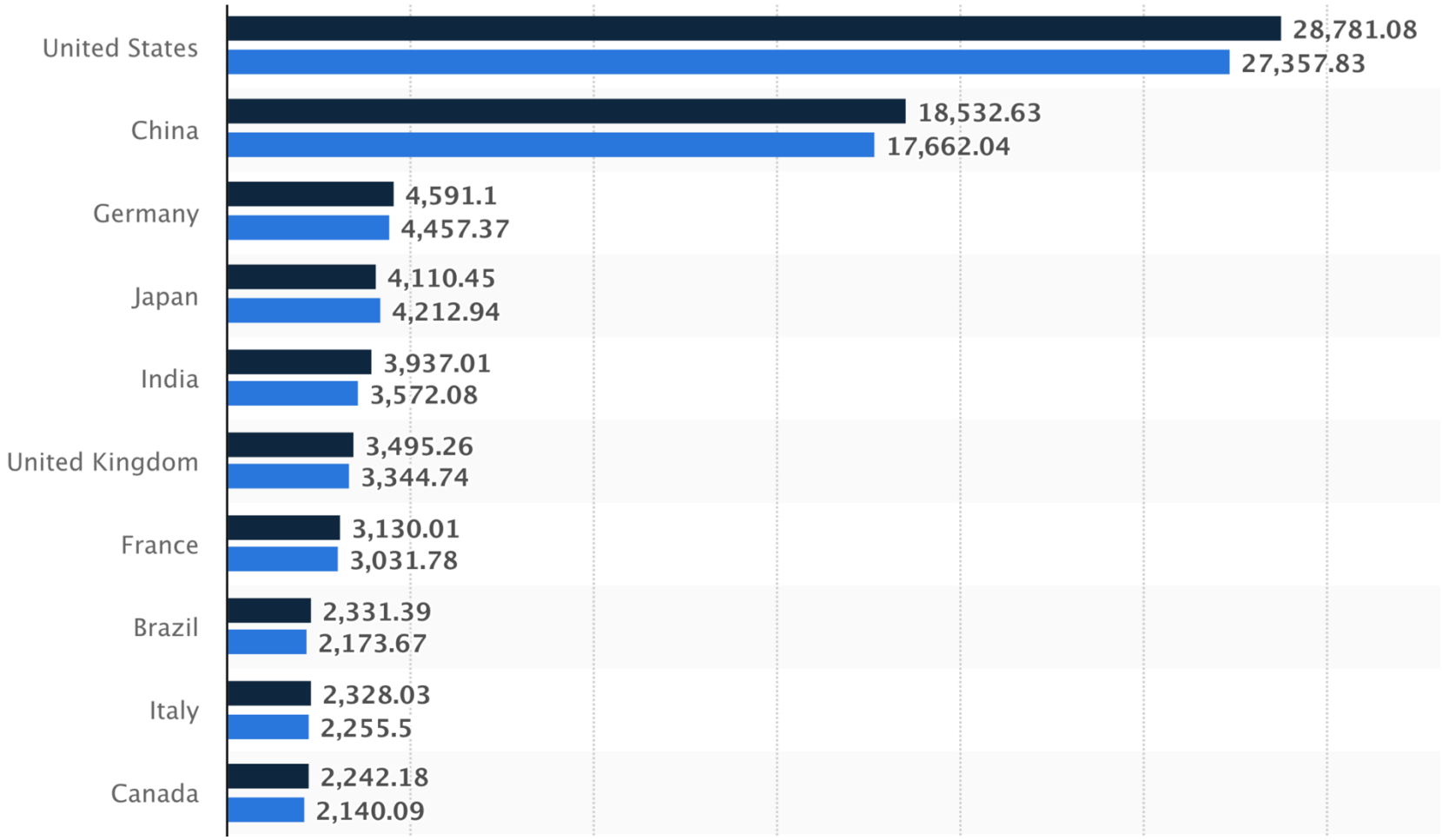
Why should we care about Cybersecurity?

Leading Areas of Interest



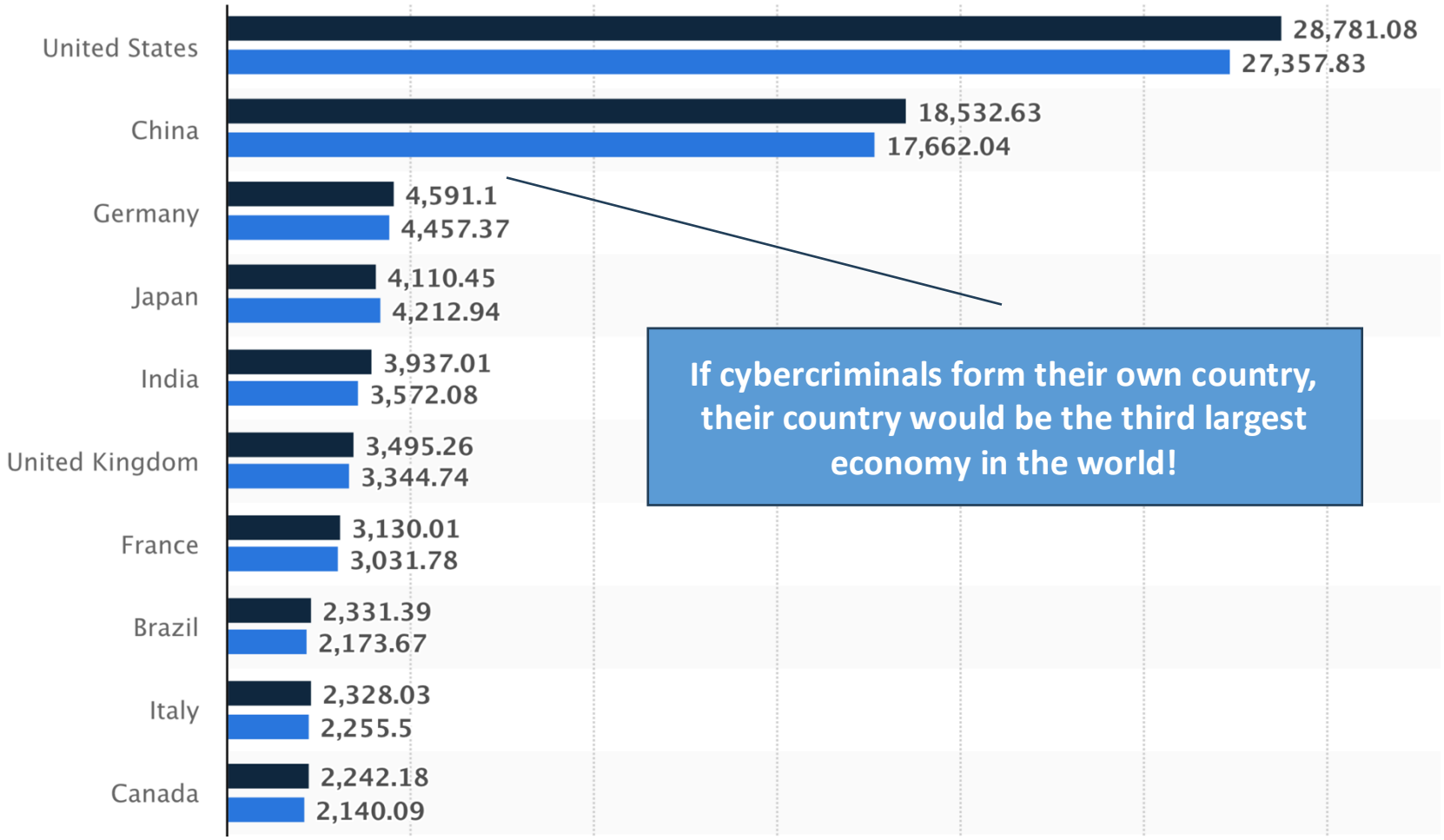

Largest Economies by GDP
UNITED STATES
$27.9 trillion
CHINA
$17.7 trillion
CYBER CRIME
$9.5 trillion
$28T
$24T
$20T
$16T
$12T
$8T
$4T
0
Source: IMF, Bloomberg, Cybersecurity Ventures

World Economies in 2024 (Measured by GDP)



How would money made by cybercriminals compare to GDPs of leading economies in 2024 (dark blue)?

GDP Vs. Cybercrime in 2024



In Year 2024:

- **Money made by Cybercriminals** [Cybercrime]:
  $9.5 trillion in Year 2024.
  $792 billion per Month.
  $183 billion per Week.
  $26 billion per Day.
  $1.08 billion per Hour.
  $18 million per Minute.

- **Drug Trafficking** [UN report]:
  $23 billion USD

- **Cost of global natural disasters** [Atlas]:
  $108 billion USD
  Cost of man-made disasters [Atlas]:
  $8 billion USD

Cybersecurity Landscape Stats in 2021-2025

- Cybercrime damage costs are expected to **grow by 15% per year**, **reaching $10.5 trillion annually by 2025** [*Sources: Varonis, Norton, Cybersecurity Ventures*]
  Here are some vital statistics about data breaches, hacking, industry-specific statistics, spending & costs.

The Big Five Stats

- **Global spending on cybersecurity** products and services is predicted to **exceed $1.75 trillion** cumulatively **from 2021 to 2025,** with a **growth of 15%** year-over-year
- The world will have **3.5 Million unfilled Cybersecurity Jobs by 2024-2025.**
- The world will need to cyber **protect 200 Zettabytes of data By 2025**.
- Global **ransomware damage costs** are predicted **to exceed $265 Billion by 2031.**
- **Phishing** remains a prevalent method of cyberattack, with **94% of malware being delivered by email**

Sources: https://cybersecurityventures.com/stats/; https://financesonline.com/cybersecurity-trends/

Cybersecurity Statistics 2024—Industry Spending

- The **global AI in cybersecurity market size in 2024** is estimated at **USD 25.35 billion**, with a compound annual growth rate **(CAGR) of 24.4% from 2025 to 2030**. [Grand View Research]
- For **information security and risk management**, **in 2024**, the estimated end-user spending reached **$215 billion**. [The Cyber Express]
- The **average cost of a data breach in 2024** has increased to **$4.88 million**, which is the **highest average on record**. [IBM]
- Regarding **remote work**, when it's a factor in causing a data breach, the **average cost per breach is $173,074 higher**. [IBM]
- In terms of cybercrime, the average **ransomware payout** has increased from **$1.5 million in 2023 to $2.7 million in 2024**. [Security Intelligence]

Industry budgets for cybersecurity and cyber insurance is growing significantly.

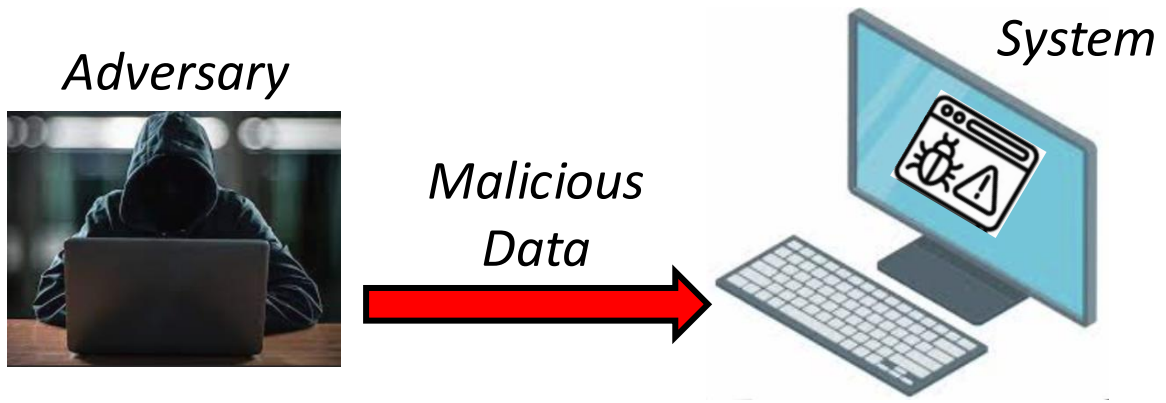
Where are the Entry Points of an Attack?

At Data Level

- **SQL Injection:** This attack involves inserting or "injecting" a SQL query via the input data from the client to the application.
- **Cross-site Scripting (XSS):** An attacker uses web applications to send malicious scripts, in the form of a browser side script, to a different end user.
- **Buffer Overflow:** This attack occurs when the volume of data exceeds the storage capacity of the memory buffer.

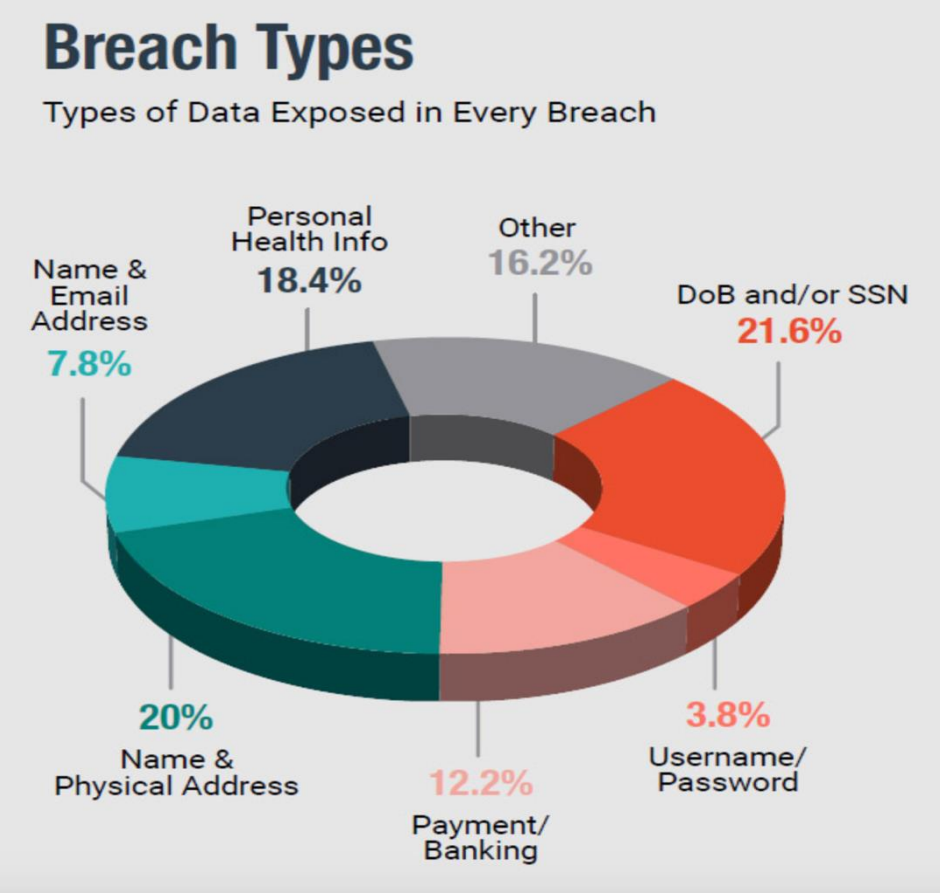
At System Level:

- **Rootkits:** Once a rootkit has been installed, attacker can gain root or privileged access to the system.
- **Ransomware:** This malicious software is designed to block access to a computer system until a sum of money is paid.
- **Trojan**: Short for Trojan horse, is a type of malware that disguises itself as legitimate software.



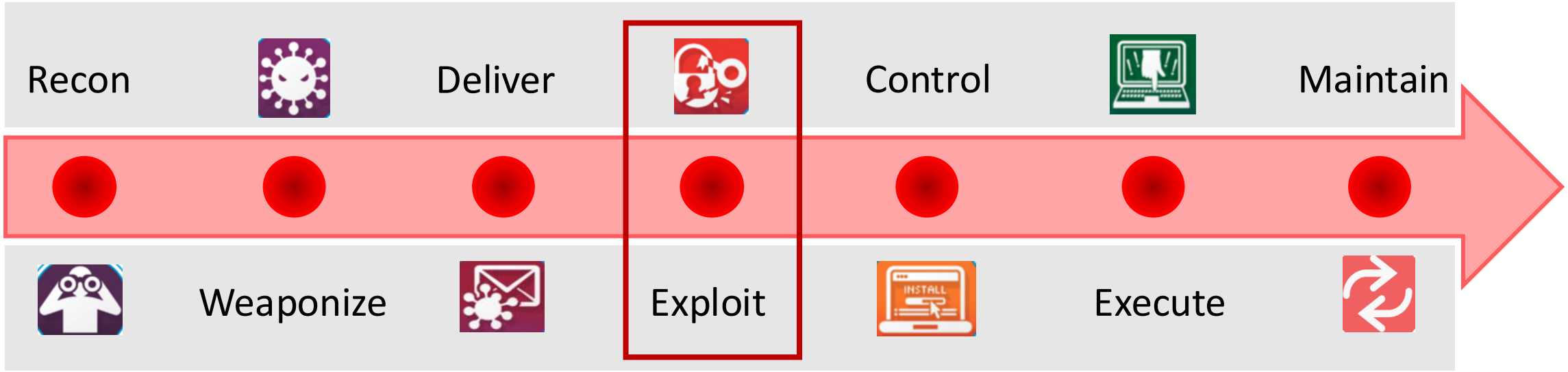
Components of Data Breach (as of 2024)

- Data breach puts a business reputation, customers and partners at high risk
- **9 Most Common Causes of Data Breach**
  - Weak and Stolen Credentials, a.k.a. Passwords
  - Back Doors, Application Vulnerabilities
  - Malware
  - Social Engineering
  - Too Many Permissions
  - Insider Threats
  - Improper Configuration and User Error
  - Ransomware



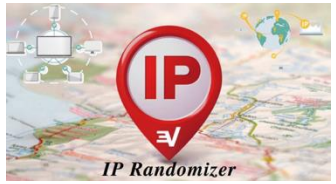
Source: https://www.helpnetsecurity.com

MITRE Cyber Attack Lifecycle



Predictive Defense Approaches

Reactive Defense Approaches

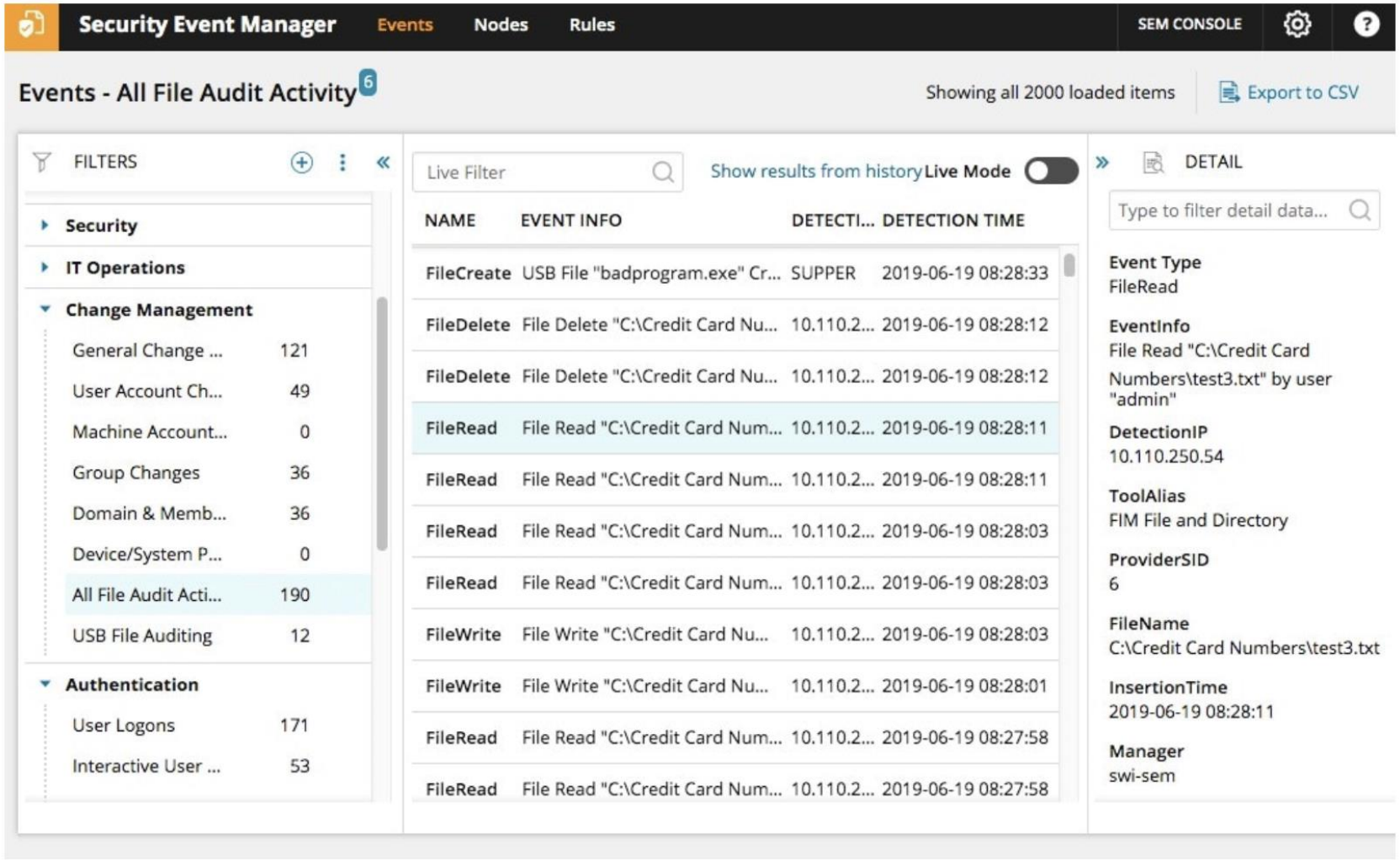


e.g., Moving Target Defense, IP randomization, defense in depth

e.g., Anti-virus software, Anomaly detectors, dynamic defense

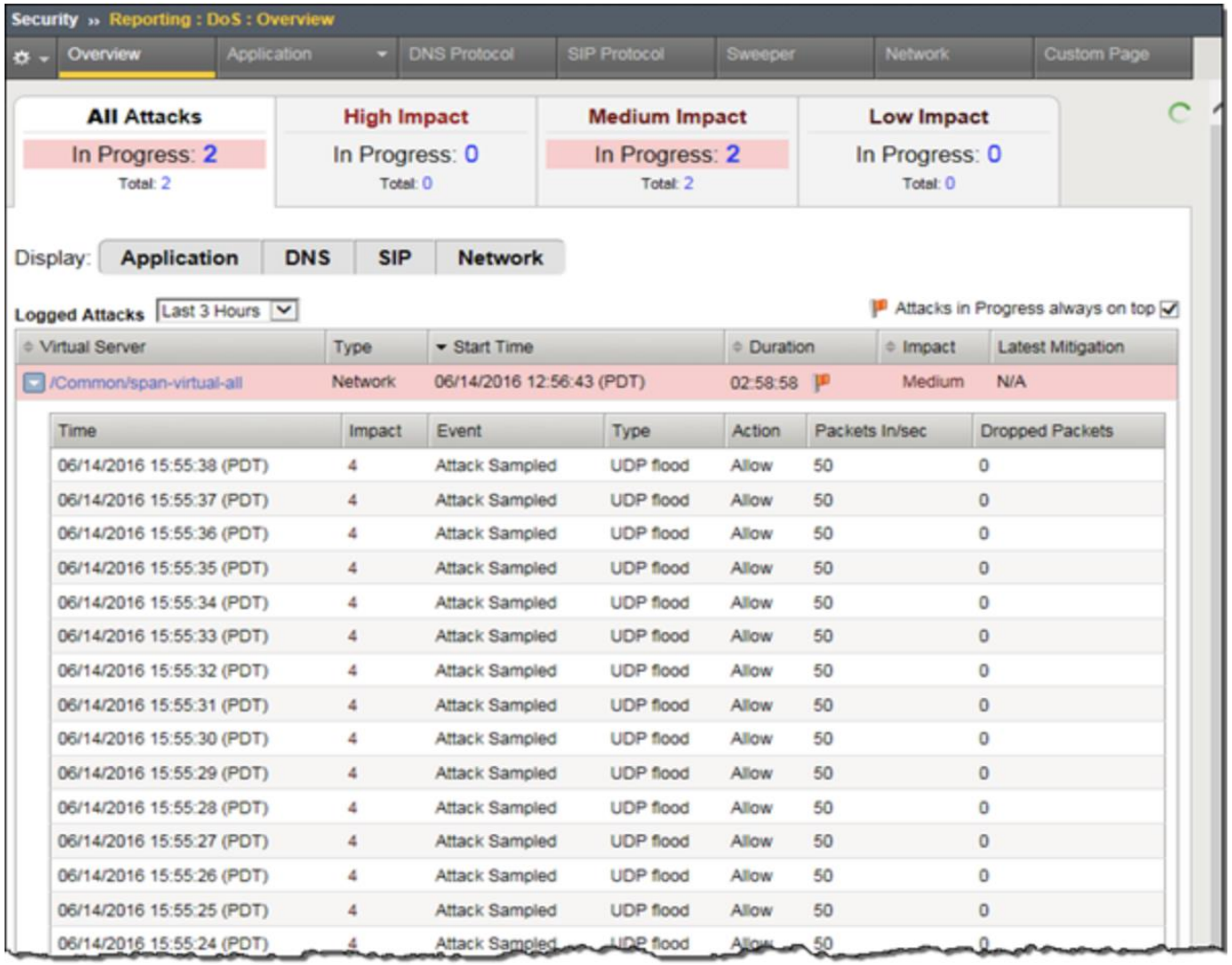
How to Identify Attacks on a System?

- **System Logs (Syslogs)**



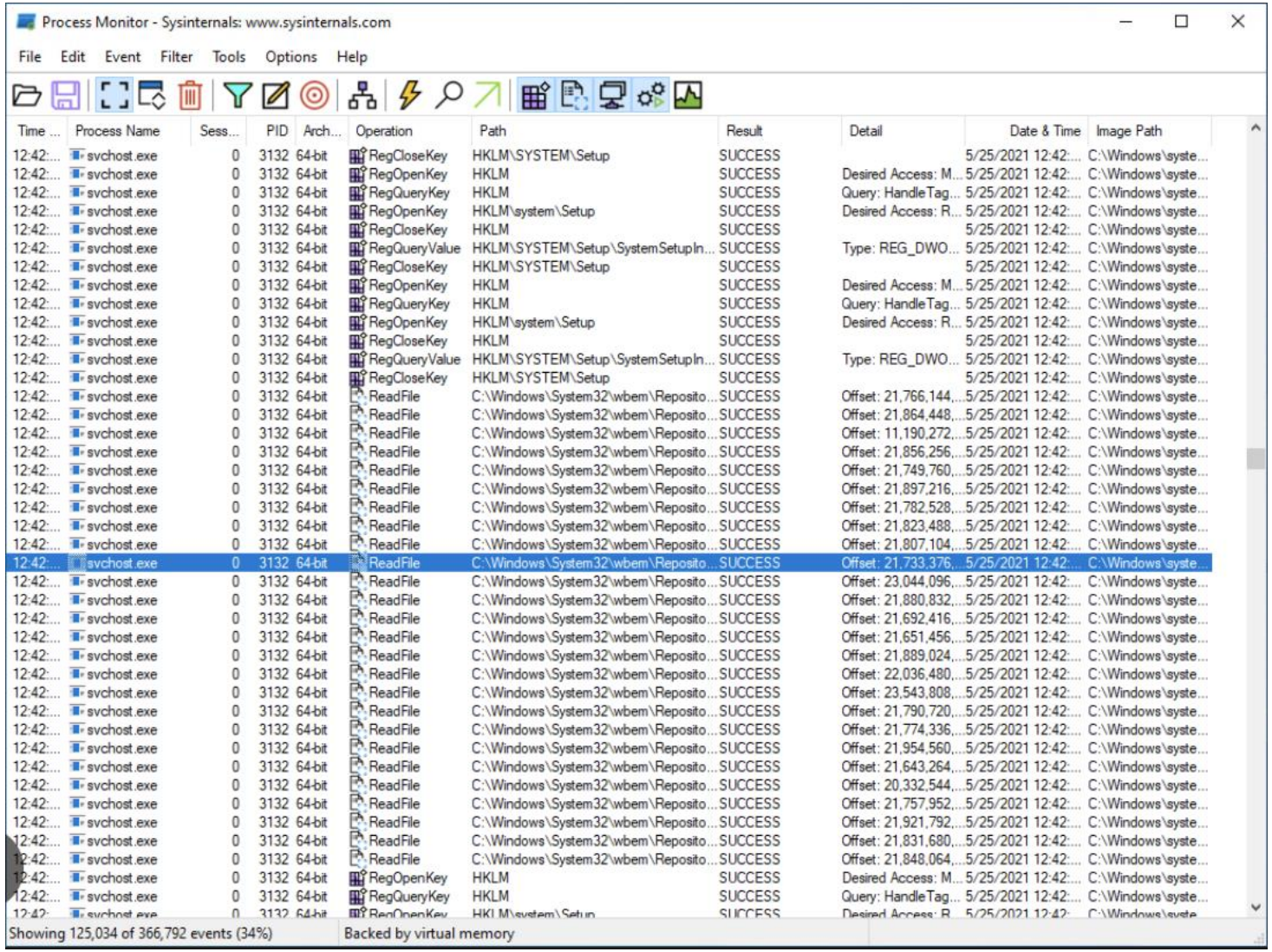
How to Identify Attacks on a System?

- **Network Logs**



How to Identify Attacks on a System?

- **Process Logs**

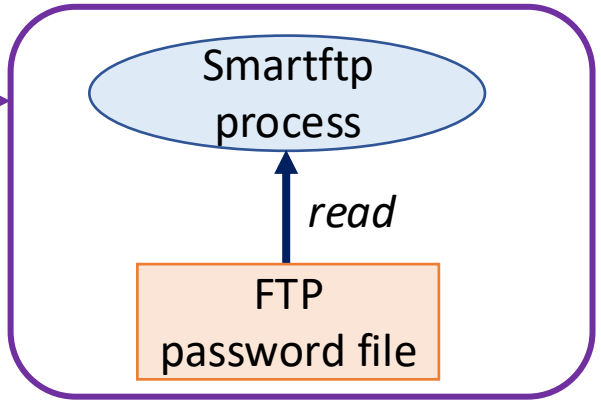


Using Syslog for Cyber Security

- Traditional threat detection systems require a human expert to inspect a large amount of data logs and identify heuristics and static signatures (rules) to detect threat and anomalies
  - **MITRE ATTACK** framework (https://attack.mitre.org/) provides a large database of static signatures

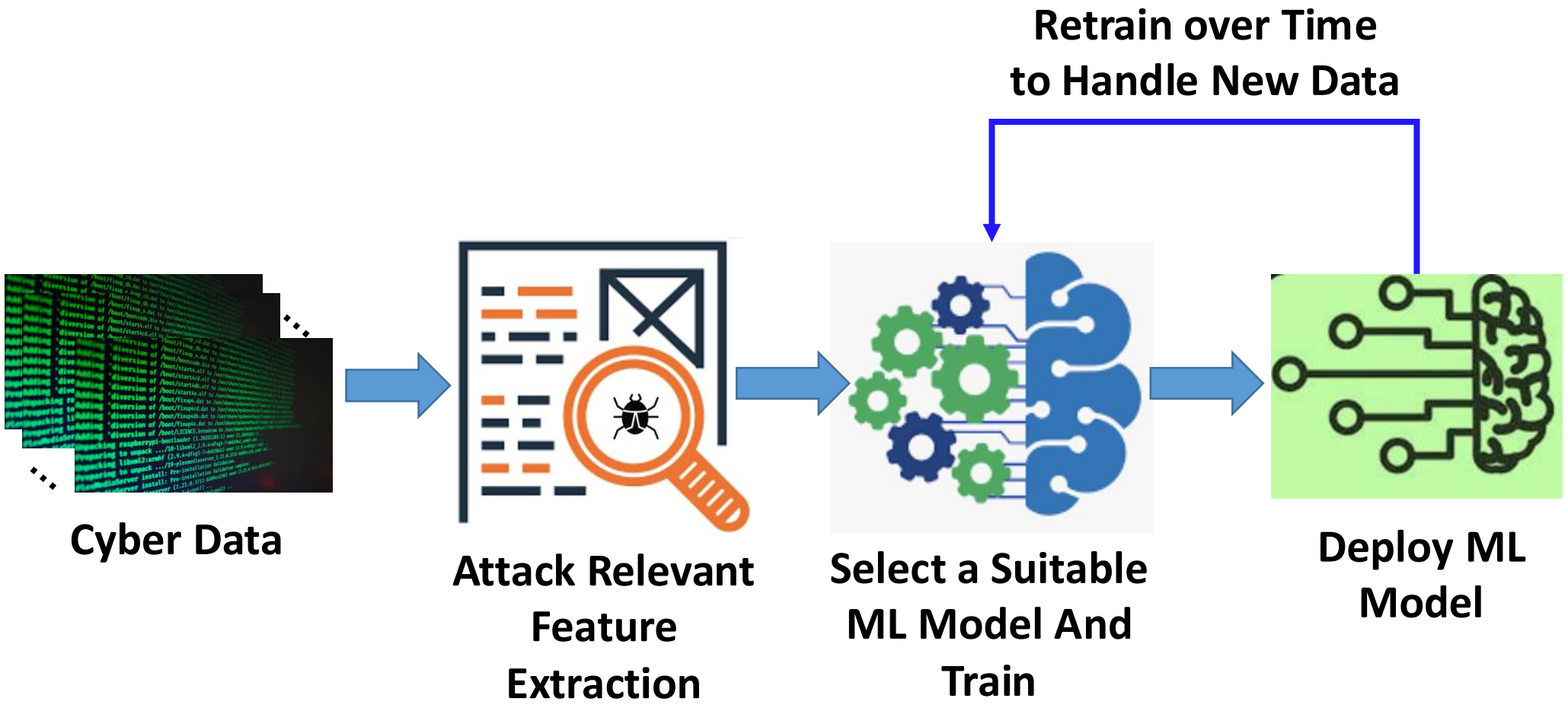
Threat name	Static signatures generated when attempting to gather credentials from Password Stores
Agent Tesla	Agent Tesla has the ability to steal credentials from FTP clients and wireless profiles.[1]
APT33	APT33 has used a variety of publicly available tools like LaZagne to gather credentials.[2][3]
APT39	APT39 has used the Smartftp Password Decryptor tool to decrypt FTP passwords.[4]
Astaroth	Astaroth uses an external software known as NetPass to recover passwords. [5]
Carberp	Carberp's passw.plug plugin can gather account information from multiple instant messaging, email, and social media services, as well as FTP, VNC, and VPN clients.[6]
CosmicDuke	CosmicDuke collects user credentials, including passwords, for various programs including popular instant messaging applications and email clients as well as WLAN keys.[7]

Visualization of the signature in system log data



- As the cyber threat evolves Machine Learning can predict and prescribe mitigation against the dynamic cyber threats via processing large amount of security data logs
- Challenge is to ensure the low false alarm rate (false positives) and low probability of misdetection (false negatives) corresponding to Machine Learning based cyber defenses

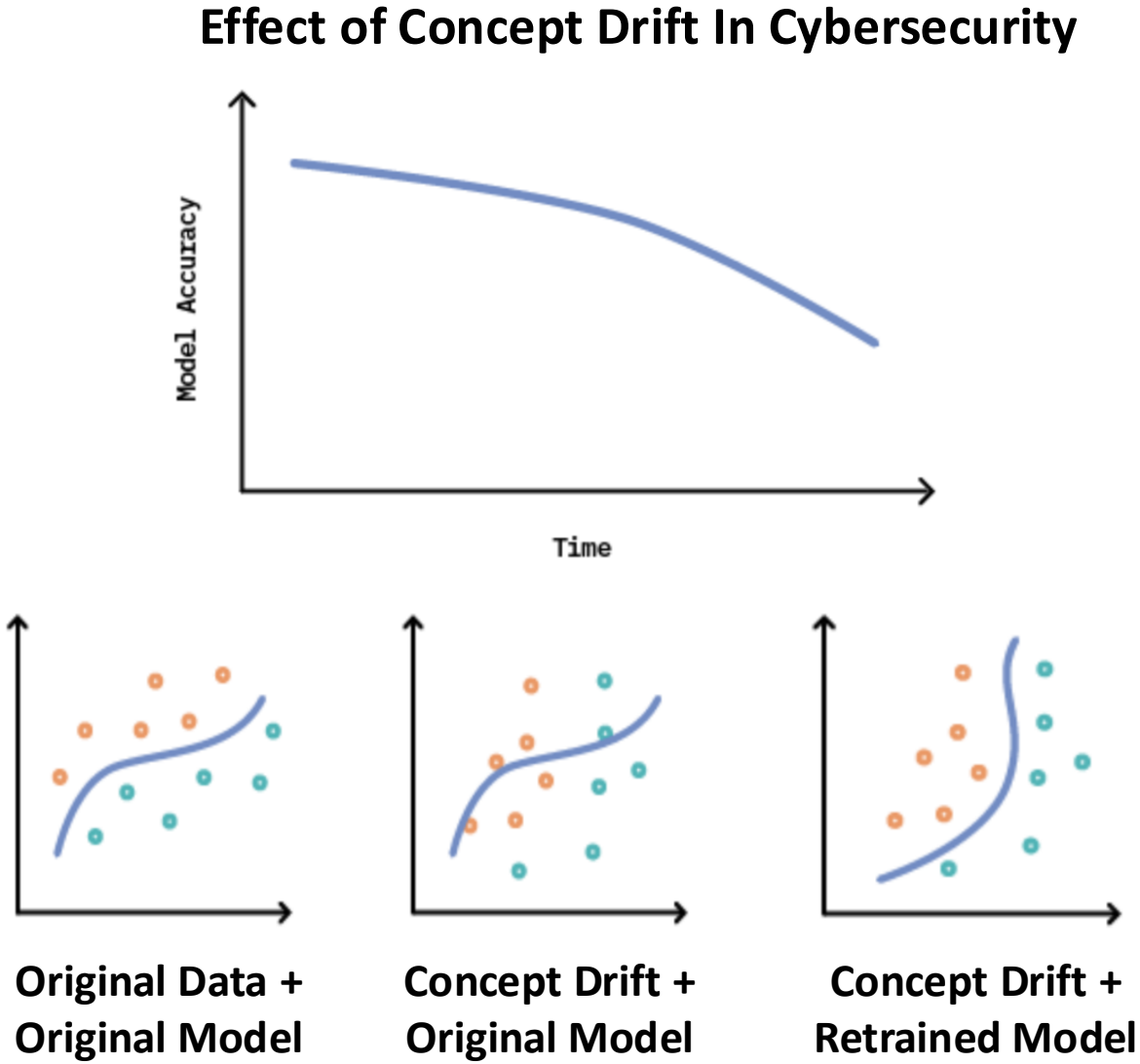
ML pipeline for cybersecurity



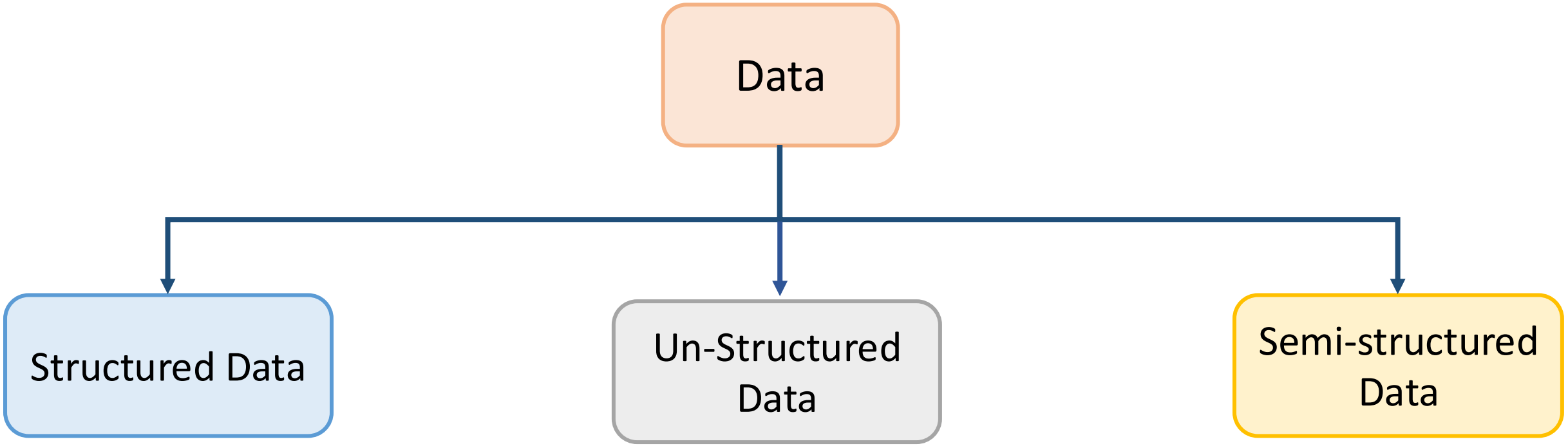
Components of ML4Cyber Pipeline

1. Collecting Data (Benign/Malicious)
2. Extracting Attack Relavent Features
3. Selecting a Suitable ML Model
4. Training the ML Model
5. Deployment of ML Model
6. Retraining or Active Learning to

   Tackle the **Concept Drift Over Time**



Categorization of Data - I



Structured Data

- Data that can be easily mapped to identifiable pre-defined column headers
- Usually stored in relational database as a table with rows corresponding to each data entry in the dataset and columns corresponding to features of data (e.g., names, dates, addresses, phone numbers, social security numbers)

Examples:

Route	Airline	Type	Plane	Points	Cash	Retail
SJU-EWR	JetBlue	Peasant	737	12,200	$0	$186
EWR-BRU	United	Polaris	787	125,000	$1,022	$7,812
BRU-MUC	Brussels Airlines	Business	A320	x	x	$497
MUC-OLB	Lufthansa	Business	A320	x	x	$435
OLB-BCN	Iberia	Peasant	A320	x	$111	$111
BCN-IST	Turkish	Business	A330	x	x	$1,090
IST-BEY	Turkish	Business	A321	x	x	$399
BEY-IST	Turkish	Business	A330	x	x	$476
IST-PVG	Turkish	Business	777	x	x	$2,765
CTU-KTM	Air China	Business	A319	x	x	$984.90
PVG-TPE	Air China	Business	A330	x	x	$450
TPE-JFK	EVA	Business	777	x	x	$3,312

Airline reservation systems data

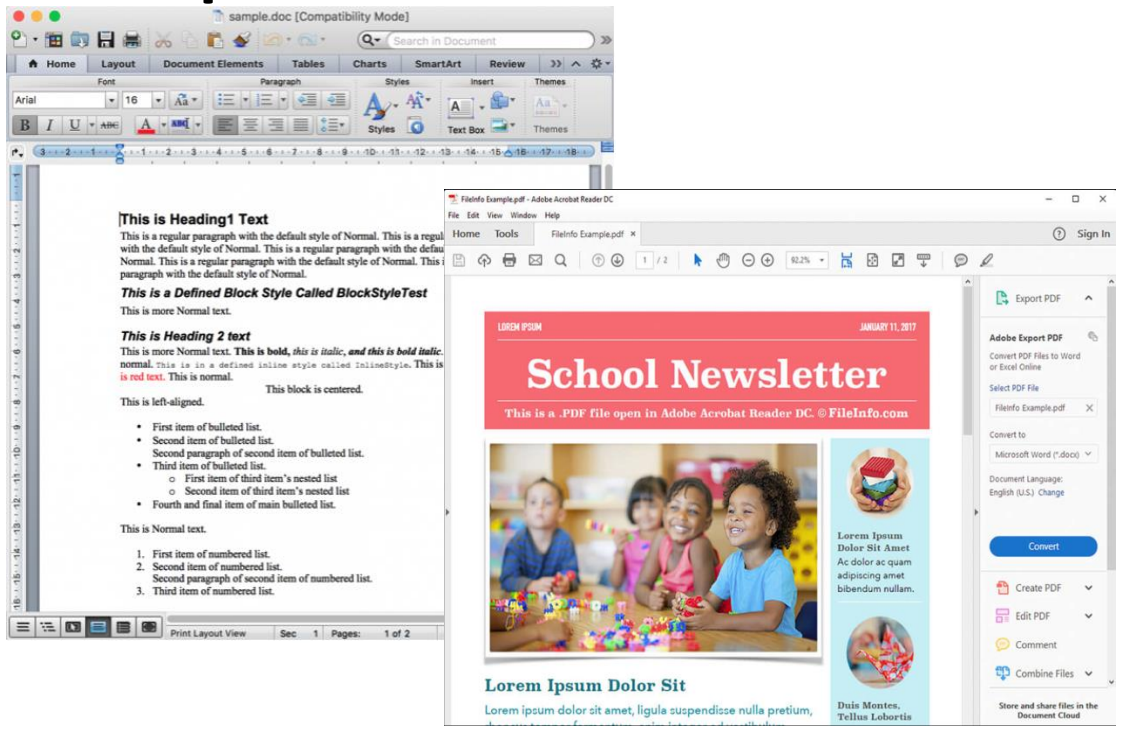
S No.	Date	Product ID	Product Name	Product Category	Unit Price	Units Received	Total Value
1	24-01-2019	Prd001	Detergents	Toiletories	10	250	2,500
2	09-02-2019	Prd002	Tooth Paste	Toiletories	25	300	7,500
3	28-01-2019	Prd003	Sun Flower Oil	Kitchen Items	85	50	4,250
4	22-01-2019	Prd004	Broom	Home Care	125	50	6,250
5	22-12-2018	Prd005	Bucket	Home Care	500	25	12,500
6	25-02-2019	Prd006	Vaccum Cleaner	Home Care	4500	10	45,000
7	13-01-2019	Prd007	Knife	Kitchen Items	60	150	9,000
8	22-12-2018	Prd008	Body Soap	Toiletories	40	250	10,000
9	07-01-2019	Prd009	Wheet Powder	Kitchen Items	80	35	2,800
10	05-02-2019	Prd010	Rice	Kitchen Items	50	25	1,250

Inventory control data

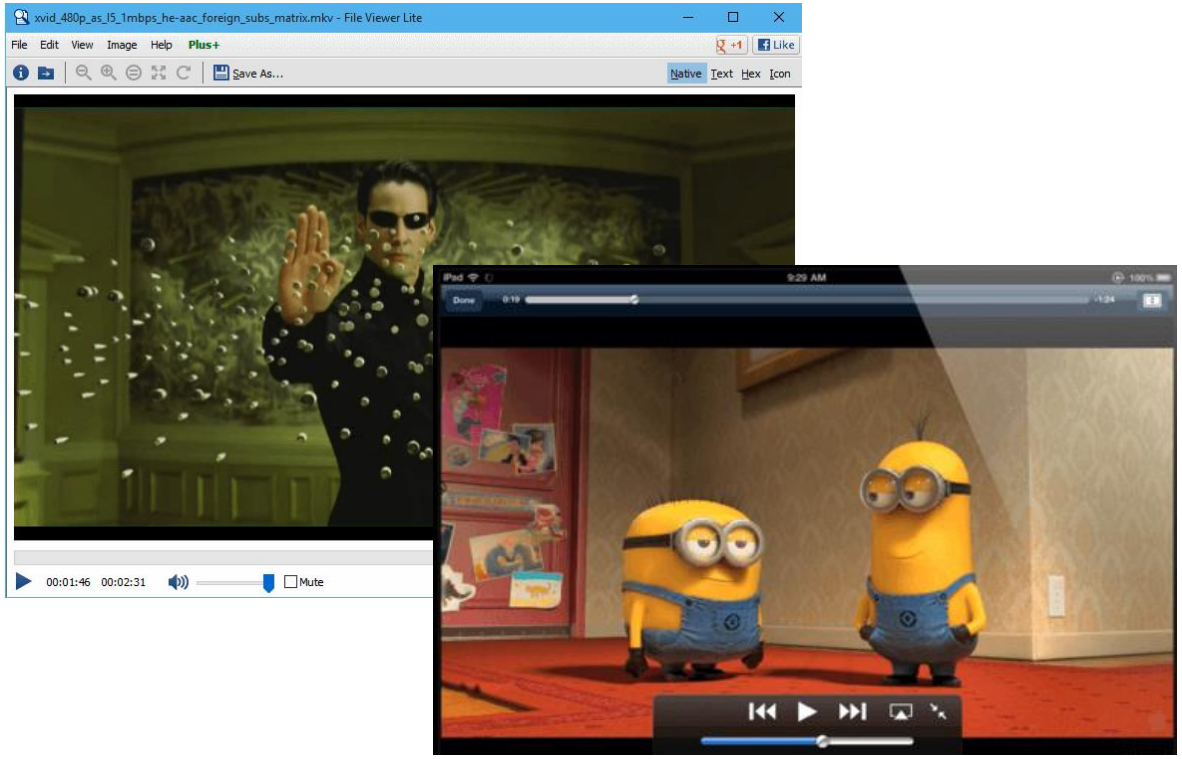
Un-Structured Data

- Data that is not mapped to any identifiable predefined data model

Examples:



Text and image files (e.g., .txt, .doc, .pdf, .jpg, .png)



Audio and Video files (e.g., .mp3, .wav, .mp4, .avi)

Semi-structured Data

- Mixture of structured and unstructured data
- Data that does not reside in a relational database but with some process they can be stored in a relational database
- Semi-structured data is typically used to ease the storage overhead of data

Example:

Open standard JSON files


```
{"widget": {
    "debug": "on",
    "window": {
        "title": "Sample Konfabulator Widget",
        "name": "main_window",
        "width": 500,
        "height": 500
    },
    "image": {
        "src": "Images/Sun.png",
        "name": "sun1",
        "hOffset": 250,
        "vOffset": 250,
        "alignment": "center"
    },
    "text": {
        "data": "Click Here",
        "size": 36,
        "style": "bold",
        "name": "text1",
        "hOffset": 250,
        "vOffset": 100,
        "alignment": "center",
        "onMouseUp": "sun1.opacity = (sun1.opacity / 100) * 90;"
    }
}}
```

Semi-structured Data Example: System log file in JSON format

{"datum":{"com.bbn.tc.schema.avro.cdm20.Host":{"uuid":"/□¢©¹¶W□»e\u000B\fØü□Ù",
"hostName":"","ta1Version":"debug","hostIdentifiers":null,"osDetails":null,"hostType":"HOST_DESKTOP",
"interfaces":{"array":[{"name":"eth0","macAddress":"52:54:00:0c:f2:ce","ipAddresses":{"array":["192.168.122.123
"]}}]}}},"CDMVersion":"20","type":"RECORD_HOST","hostId":"/□¢©¹¶W□»e\u000B\fØü□Ù","sessionNumber":1342
21299,"source":"SOURCE_LINUX_THEIA"}

Data type named Host that records the properties of host system

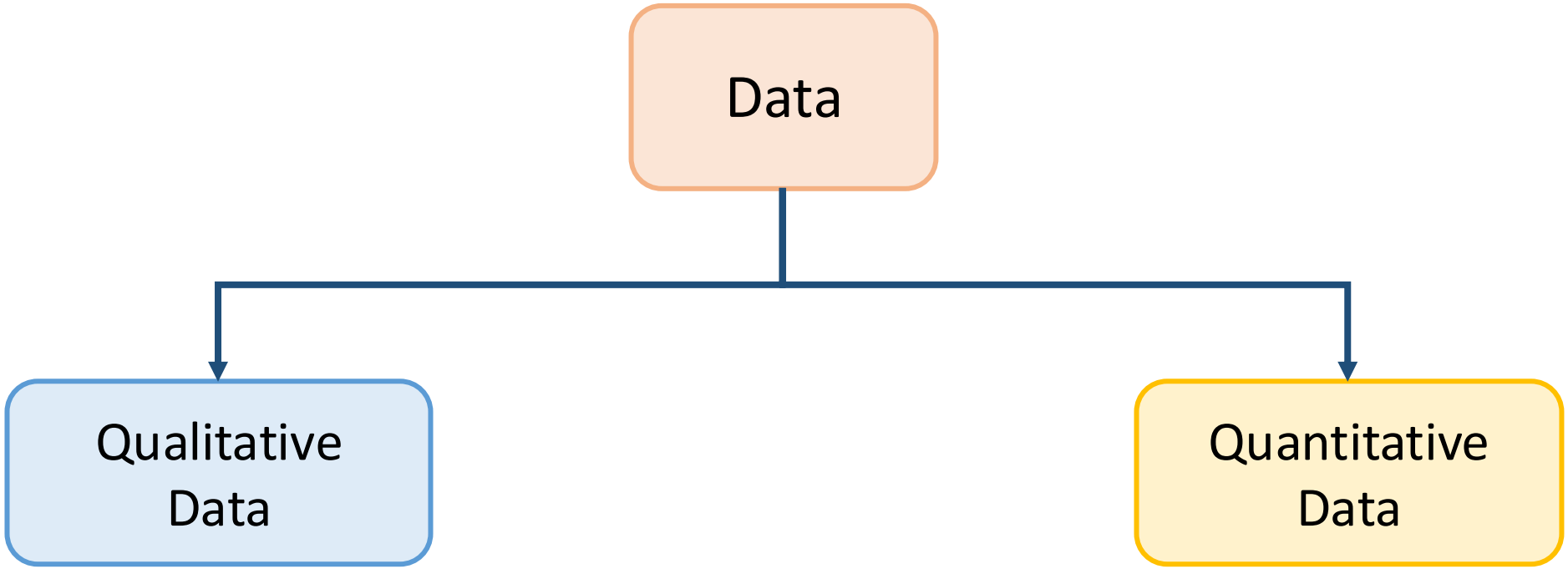
{"datum":{"com.bbn.tc.schema.avro.cdm20.Event":{"uuid":"/ÿX¬IÉ[□□TÍ□ë\u0002 Ä","sequence":{"long":74},
"type":"EVENT_OPEN","threadId":{"int":4674},"subject":{"com.bbn.tc.schema.avro.cdm20.UUID":"□î\r^µ÷\\Ð□<□§
îwà"},"predicateObject":{"com.bbn.tc.schema.avro.cdm20.UUID":"\u0001\u0000Ð\u000FQ÷\u0002\u0000\u000
0\u0000\u0000\u0000l\u0000Ð^"},"predicateObjectPath":null,"predicateObject2":{"com.bbn.tc.schema.avro.cdm
20.UUID":"\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u
0000\u0000"},"predicateObject2Path":null,"timestampNanos":1591292235615848458,"names":{"array":["open"]}
,"parameters":null,"location":null,"size":null,"programPoint":null,"properties":{"map":{"flags":"O_CLOEXEC"}}}},"C
DMVersion":"20","type":"RECORD_EVENT","hostId":"/□¢©¹¶W□»e\u000B\fØü□Ù","sessionNumber":134221299,"
source":"SOURCE_LINUX_THEIA"}

Data type named Event that records the properties of system calls

{"datum":{"com.bbn.tc.schema.avro.cdm20.Subject":{"uuid":"ÐðÜvú□Q\u0000□Ab°\u001E;Qý",
"type":"SUBJECT_PROCESS","cid":1042,"parentSubject":{"com.bbn.tc.schema.avro.cdm20.UUID":"ÝVµ□□tXÃ³è,b7□
¸í"},"localPrincipal":{"com.bbn.tc.schema.avro.cdm20.UUID":"¶ÅD□8 _P¦\ný□v!□®"},"startTimestampNanos":{"lon
g":1591292195676198843},"unitId":null,"iteration":null,"count":null,"cmdLine":{"string":"/usr/sbin/sshd -
D"},"privilegeLevel":null,"importedLibraries":null,"exportedLibraries":null,"properties":{"map":{"fsuid":"0","path":"
/usr/sbin/sshd","uid":"0","egid":"0","gid":"0","euid":"0","fsgid":"0","tgid":"1042","sgid":"0","suid":"0","ppid":"1"}}
}},"CDMVersion":"20","type":"RECORD_SUBJECT","hostId":"/□¢©¹¶W□»e\u000B\fØü□Ù","sessionNumber":13422
1299,"source":"SOURCE_LINUX_THEIA"}

Data type named Subject that records the properties of processes

Categorization of Data - II



Qualitative Data

- Descriptive, non-numeric information that represents **categories or qualities** (e.g., color, brand, “spam/not spam,” sentiment label).

Example:

CustomerID	FavoriteColor	MembershipTier	PreferredPayment	FeedbackSentiment
C001	Blue	Gold	Card	Positive
C002	Red	Silver	Cash	Neutral
C003	Green	Bronze	Mobile	Negative
C004	Blue	Silver	Card	Positive
C005	Yellow	Gold	Mobile	Neutral

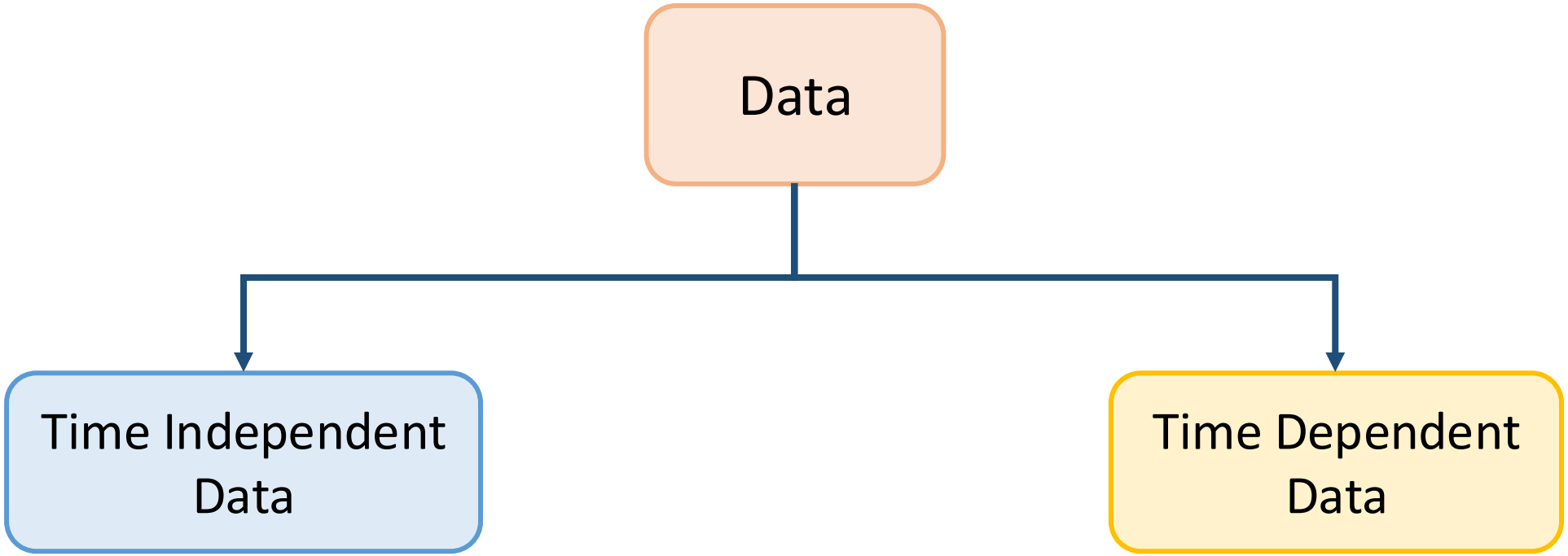
Quantitative Data

- Numeric information that represents **measurable amounts** (e.g., height, temperature, income, number of clicks).

Example:

SampleID	AgeYears	MonthlySpendUSD	StepsPerDay	ExamScore
101	19	42.5	8300	78.0
102	23	105.0	12000	91.5
103	21	60.0	5400	68.0
104	28	210.5	9800	88.0
105	25	150.0	7600	84.5

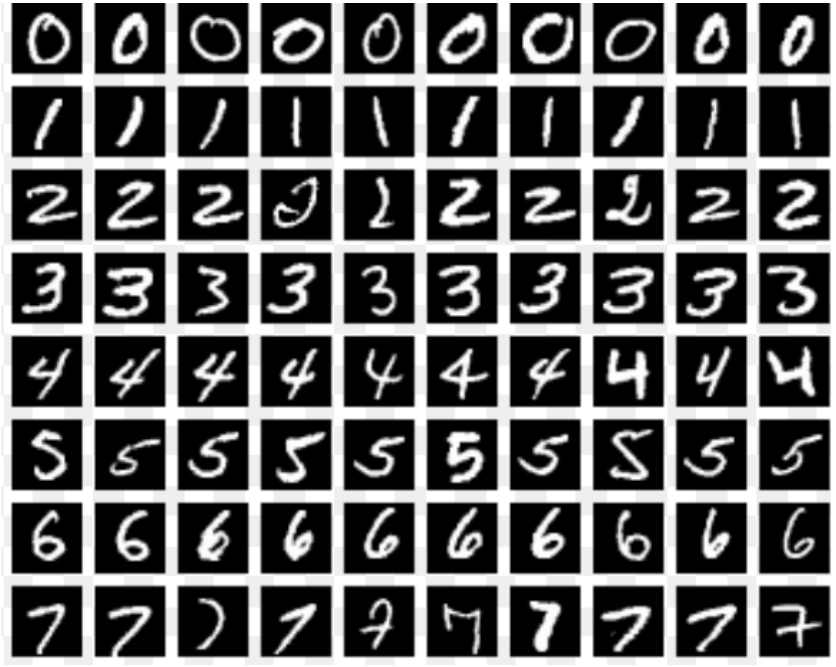
Categorization of Data - III



Time Independent Data

- Data where **time is not a meaningful factor** and each record can be treated as standalone (e.g., a dataset of house features and prices, a set of images for classification, survey responses without timestamps).

Examples:



The MNIST Dataset:
Dataset of Handwritten Digits

Variables

Student	Hours Studied	Exam Score
1	2	76
2	4	93
3	3	90
4	4	91
5	4	87
6	5	97
7	6	94
8	5	92
9	4	81
10	4	80

Cases

Student Survey containing # of hours studied vs. Exam Score

Time Dependent Data

- Data where **time order matters** and values are linked across time (e.g., sensor readings over time, stock prices, heart rate signals, website traffic per hour).
- **Examples:**

An object moving with a constant speed of 6 m/s

Time (s)	Position (m)
0	0
1	6
2	12
3	18
4	24

An object moving with a changing speed

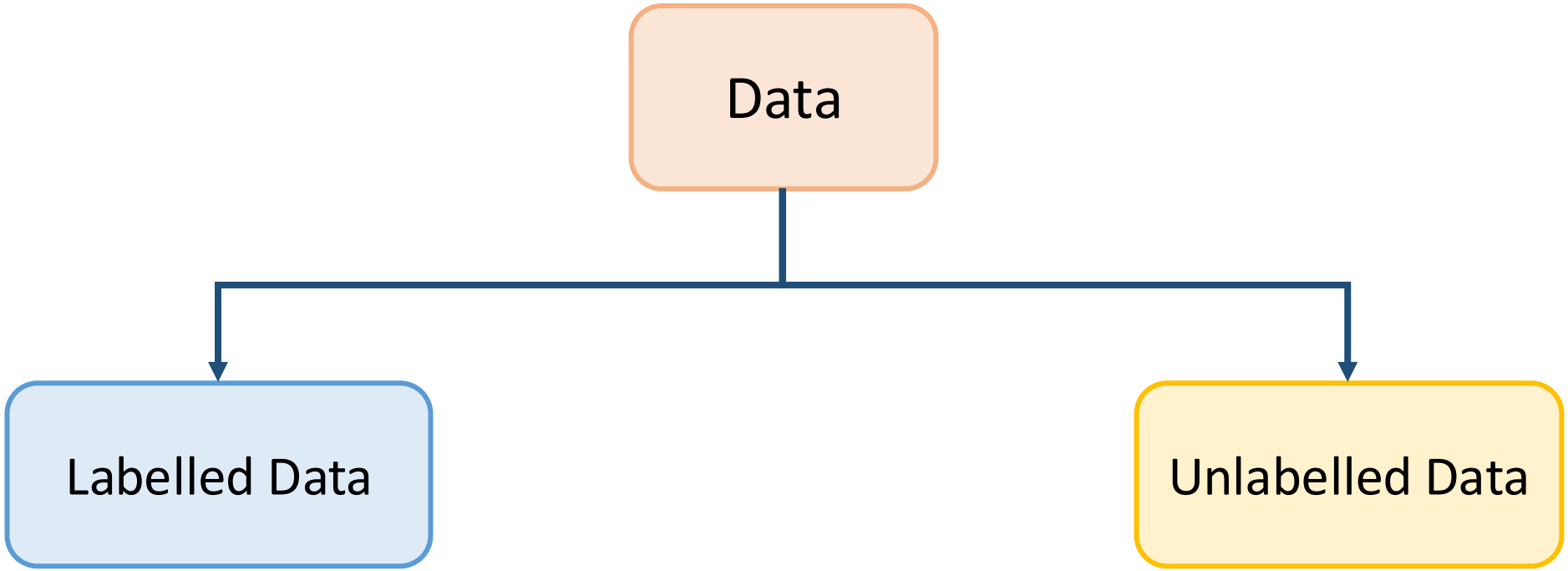
Time (s)	Position (m)
0	0
1	1
2	4
3	9
4	16

Time Vs. Speed of an Object

Date	Open	High	Low	Close	Volume
27-12-11	67.7	67.8	66.8	66.9	738600
28-12-11	67.9	68	66.5	66.6	626000
29-12-11	67.9	67.9	65.5	65.8	1080800
01-01-12	67.9	67.9	65.9	66.8	638600
02-01-12	44.4	45.3	43.3	43.8	339800
04-01-12	43.2	44.9	41.5	44	1025600
08-01-12	35.5	35.8	35	35.3	140200
10-01-12	32.6	35.5	32.5	34.1	405000
11-01-12	33.2	33.4	32.9	32.9	66200
15-01-12	63	63.6	61.5	61.6	409600
16-01-12	60	60.8	57	59.7	474400
18-01-12	58	60.9	56.9	57.7	492800
19-01-12	59.5	61.8	58.3	60.9	461400

Stock Prices

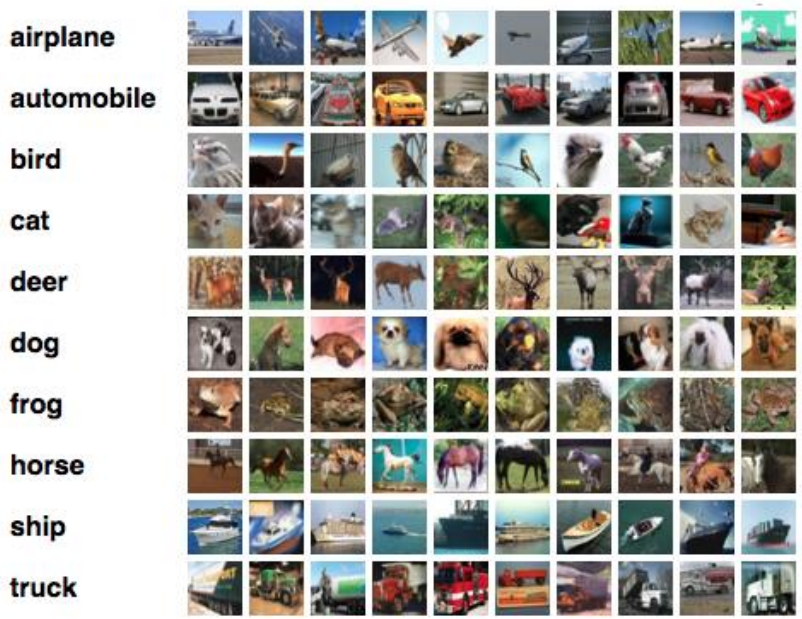
Categorization of Data - IV



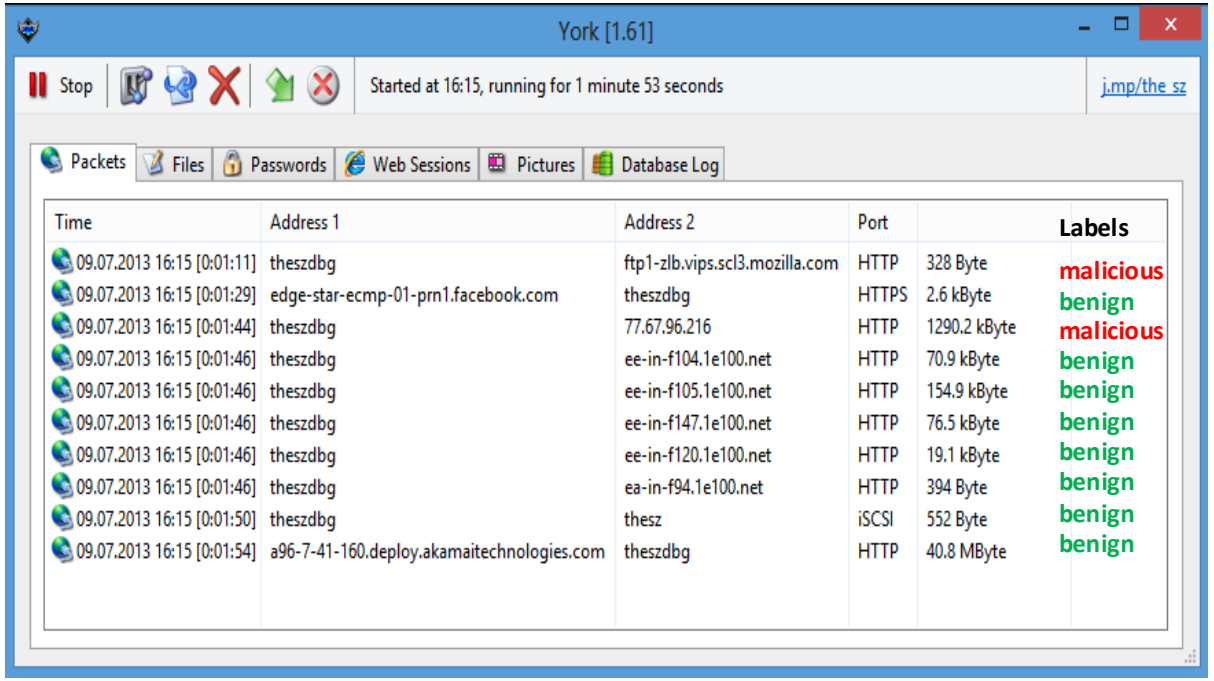
Labelled Data

- Data that has been (manually) tagged with headers and meaning is called **labelled data**

Examples:



The CIFAR-10 Dataset:
labeled images belong to 10 different classes



Labelled (**benign**/**malicious**) network traffic log

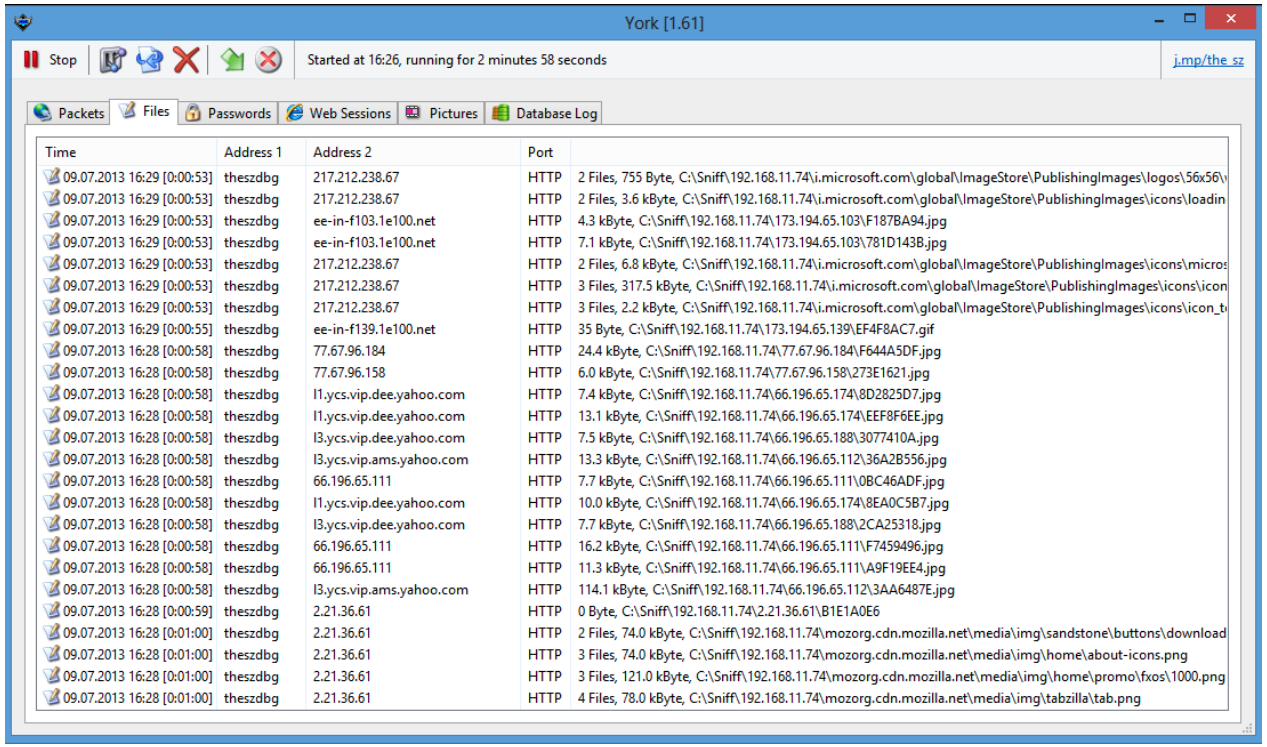
Unlabelled Data

- Data that has not been tagged is called **unlabelled data**

Examples:

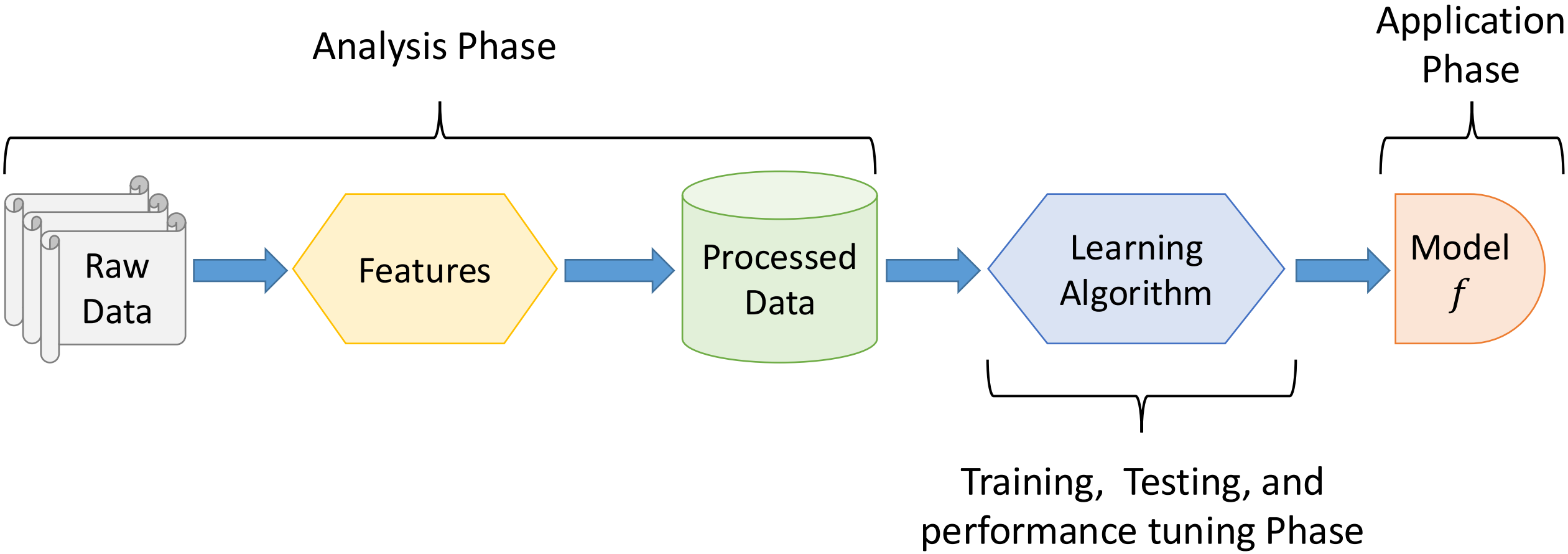
	A	B
1	date	value
2	9/6/2017	531974.19
3	9/7/2017	484704.26
4	9/8/2017	693635.27
5	9/9/2017	420176.65
6	9/10/2017	257548.74
7	9/11/2017	212416.06
8	9/12/2017	410240.57
9	9/13/2017	559267.26
10	9/14/2017	556496.67
11	9/15/2017	813277.37
12	9/16/2017	600138.13
13	9/17/2017	371246.62
14	9/18/2017	319319.61
15	9/19/2017	561685.94
16	9/20/2017	650536.61
17	9/21/2017	599229.88

Timeseries data

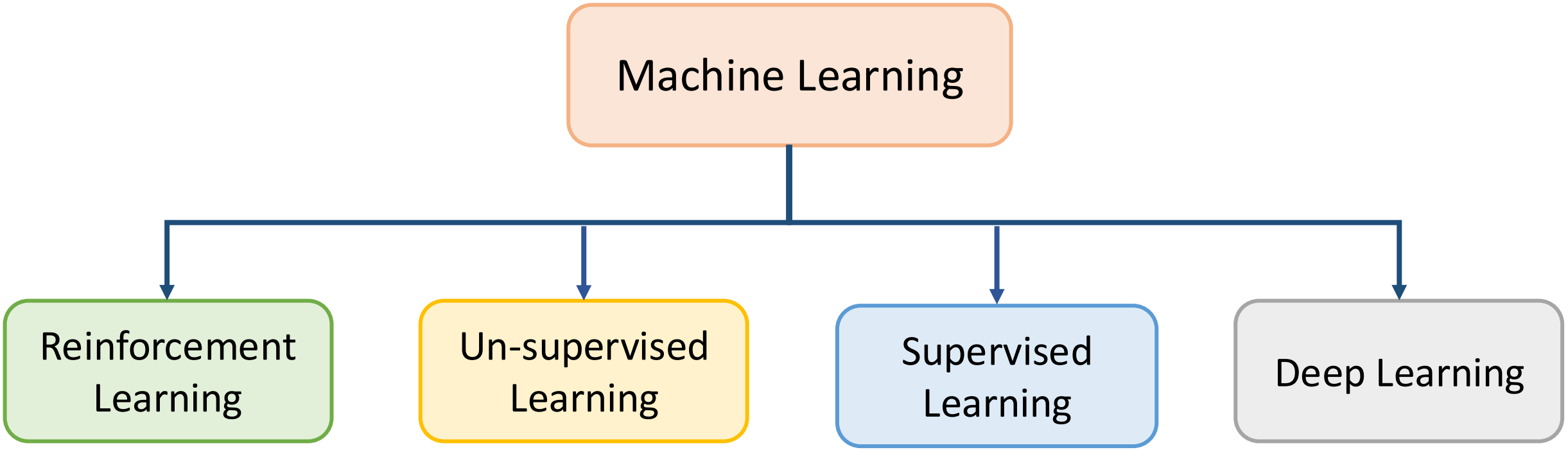


Unlabelled network file transfer log

A Schematic View of Machine Learning Pipeline



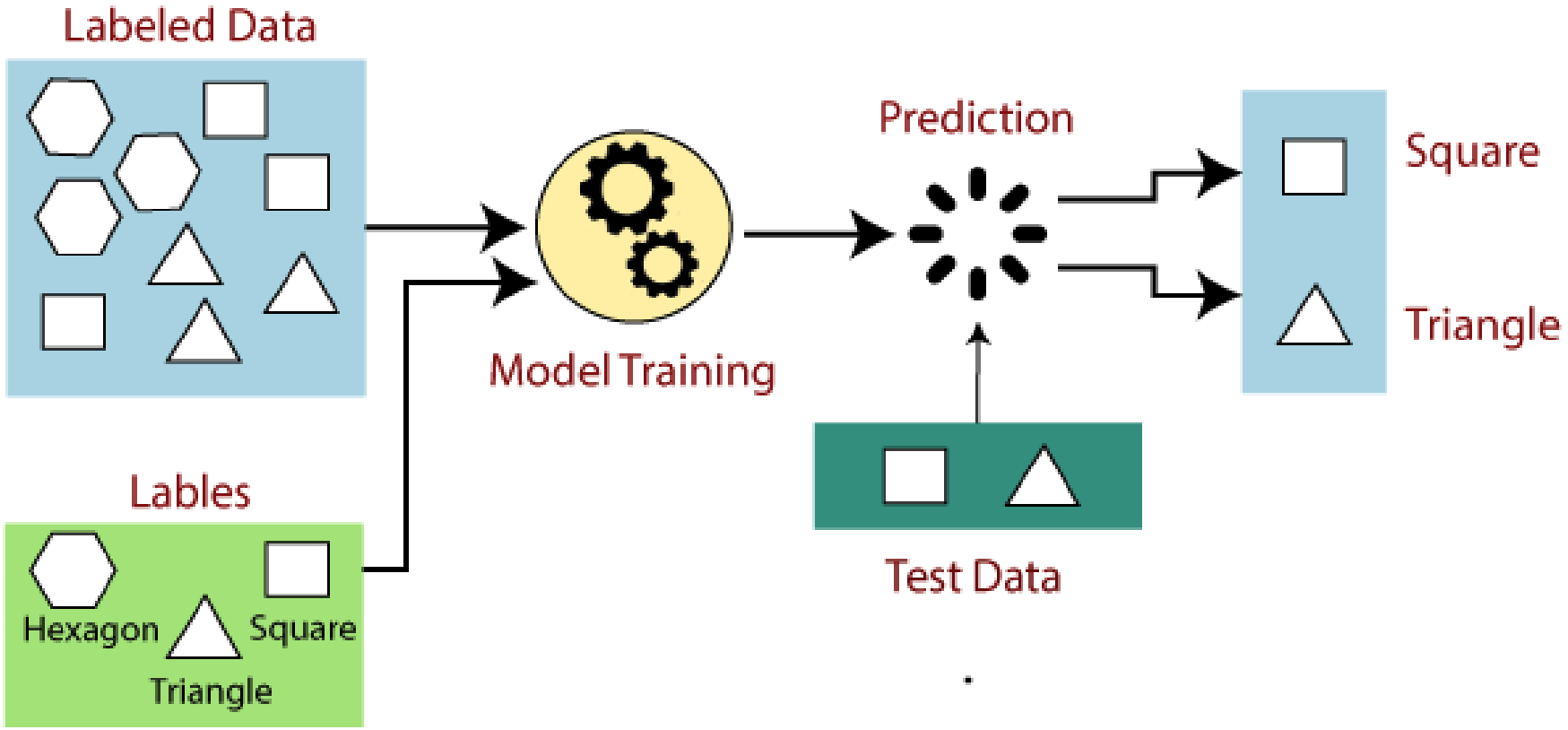
Categorization of Machine Learning Algorithms



5 min Break

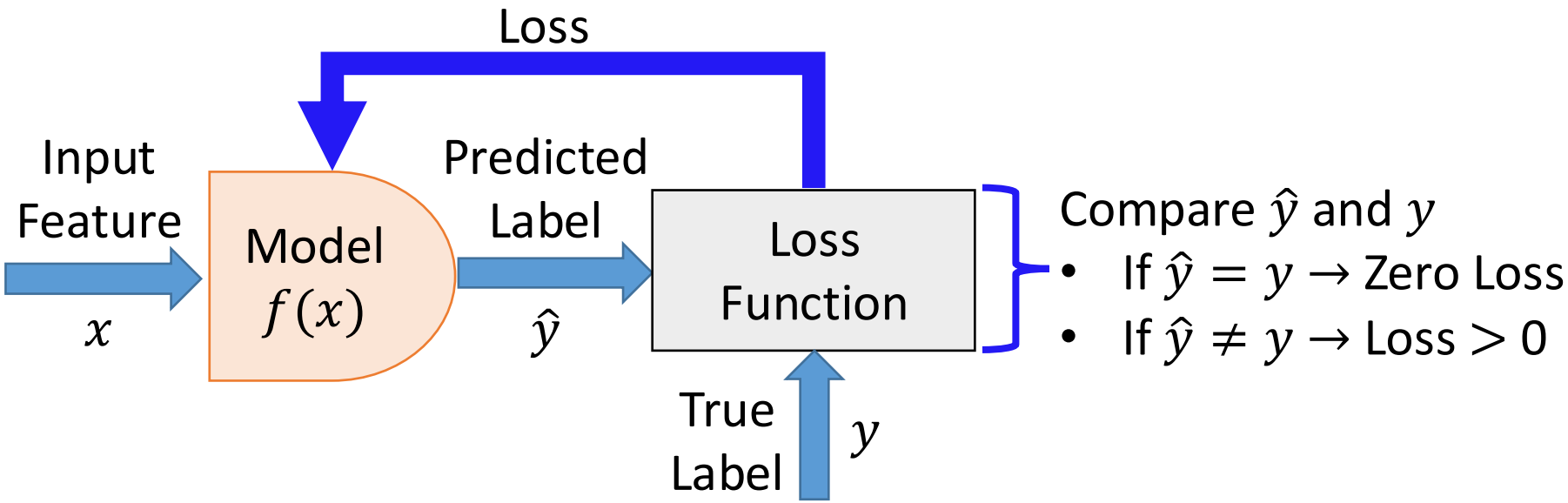
Supervised Learning

- Supervised learning methods **learn** from **labelled data** and use the **insight** gained to make decisions on the operational/testing data

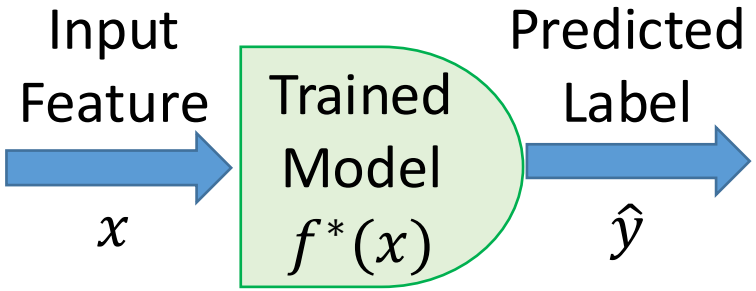


Supervised Learning

- **Training:**



- **Inference/Deployment:**



Supervised Learning Model

- Consider a dataset $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$ of feature vectors $x_i \in \mathcal{X} \subseteq \mathbb{R}^m$ and labels $y_i \in \mathcal{Y}$ where each data point $\{x_i, y_i\}$ is independently generated by an unknown distribution $\mathcal{P}$

- Goal: Given a class of models $\mathcal{F}$ and a dataset $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$, finding a model $f \in \mathcal{F}$ which is close to the true model h as possible. i.e., Find a model $f \in \mathcal{F}$ with property,

$$\mathbb{E}_{(x,y)\sim\mathcal{P}}[l(f(x), h(x))] \leq \mathbb{E}_{(x,y)\sim\mathcal{P}}[l(f'(x), h(x))] \text{ for all } f' \in \mathcal{F}$$

- $l(f(x), h(x))$: *loss function* that measures the error that $f(x)$ makes in predicting the true label $y = h(x)$
  - e.g., $l\big(f(x), h(x)\big) = \big||f(x) - h(x)|\big|_2^2 = \sum_{i\in\mathcal{D}}\big(f(x_i) - h(x_i)\big)^2$ ← *Sum of squared distance*

Supervised Learning Model

- Supervised learning use labeled training data to solve the following optimization problem

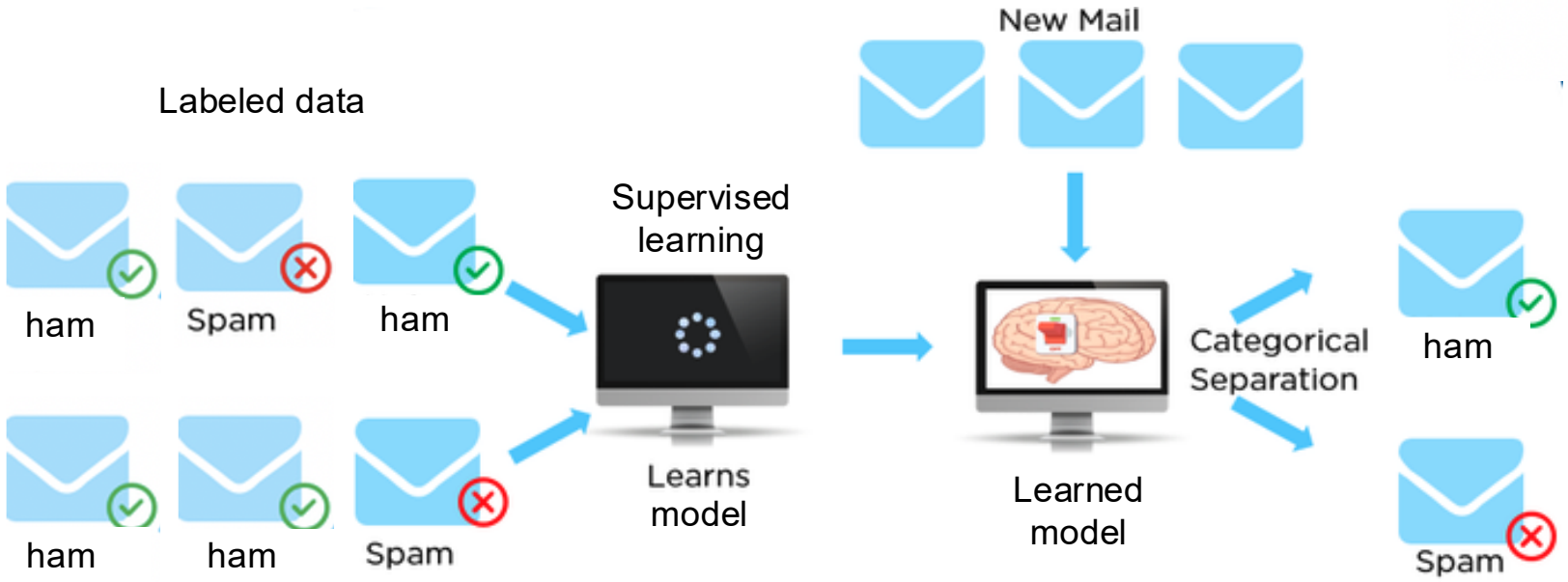
$$\min_{w} \sum_{i \in \mathcal{D}} l(f(x_i; w), y_i) + \gamma \rho(w)$$

  - w: Parametric representation of the models $f \in \mathcal{F}$ in real vector space
  - $\gamma > 0$: Regularization parameter
  - $\rho(w)$: **Regularization term** which penalizes the complexity of the candidate models $f \in \mathcal{F}$ can be used to **mitigate the over-fitting**
    - $\rho(w)$ often takes the form of an l_p norm of w (e.g., $\rho(w) = ||w||_2^2$ --- l_2 regularization)

- Supervised learning is typically divided into two categories
  - Regression where labels are real-valued. i.e., $\mathcal{Y} \in \mathbb{R}$
  - Classification where $\mathcal{Y}$ is a finite set of details

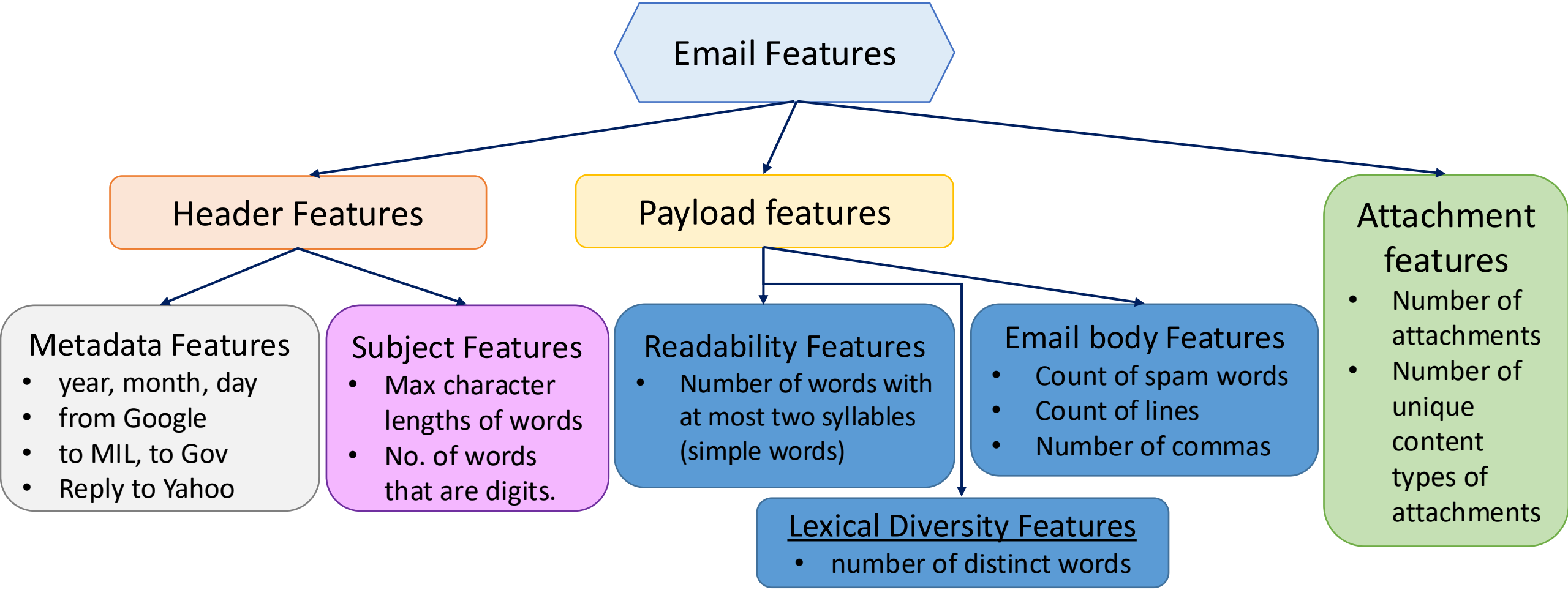
A Supervised Learning Example

- Spam detect:
  - Supervised learning helps distinguish spam emails in the inbox by separating them from legitimate emails also known as ham emails.
  - During this process, the training data enables learning, which helps such systems to send ham emails to the inbox and spam emails to the Spam folder



A Supervised Learning Example-Relevant Features

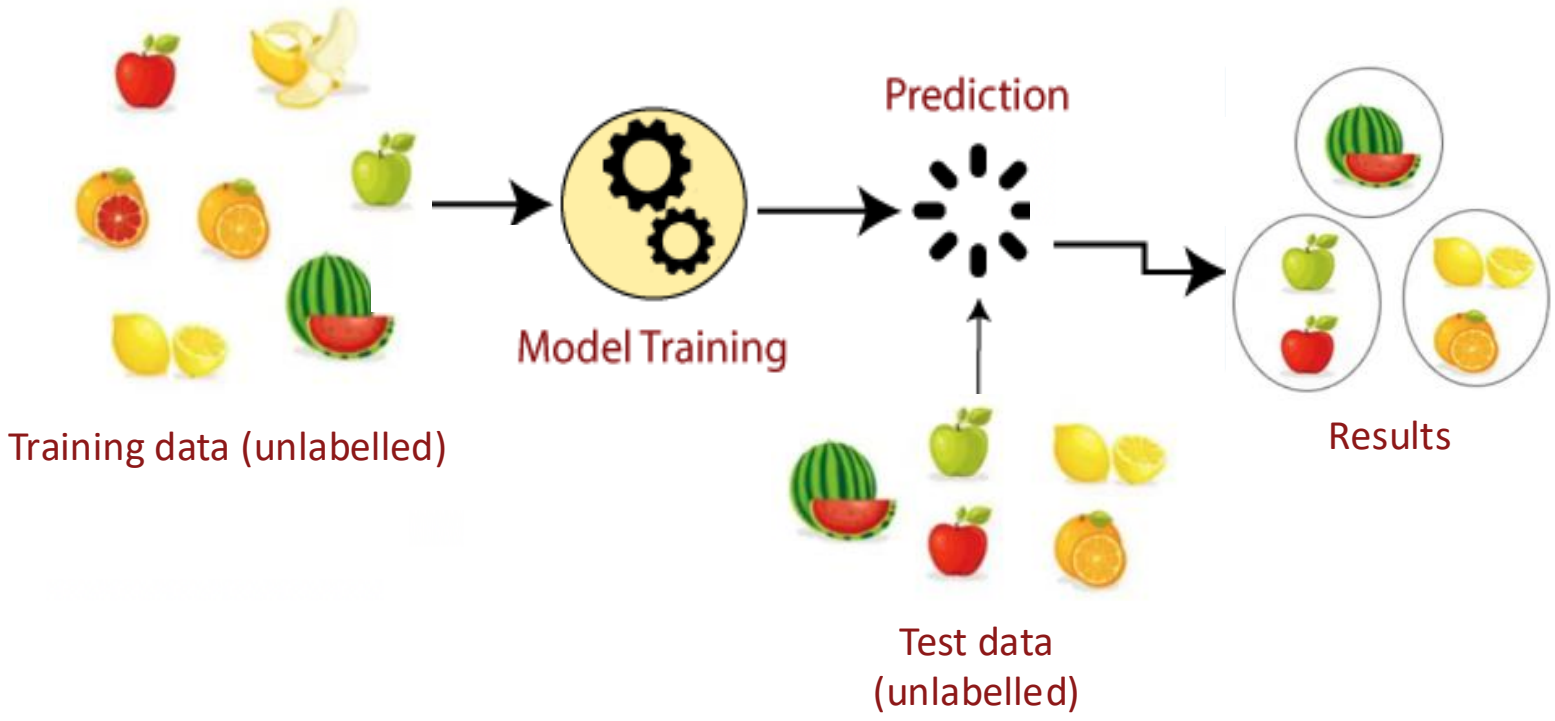
- Features of the emails extracted for the Spam detect



EMFET: E-mail Features Extraction Tool (https://arxiv.org/pdf/1711.08521.pdf)

Un-supervised Learning

- Un-supervised learning technique is used when the initial data is **unlabelled**
- Draws insights by processing data whose structure is not known before hand



Un-supervised Learning Model

- Dataset in Un-supervised learning is comprised of only the feature vectors, but has no labels. i.e., $\mathcal{D} = \{x_i\}$
- Goal: Identifying the aspects of the joint distribution of observed features (rather than predicting a target label as in supervised learning)

- Commonly studied problems under un-supervised learning
  - Clustering
  - Principle Component Analysis (PCA)
  - Matrix completion

Un-supervised Learning Examples

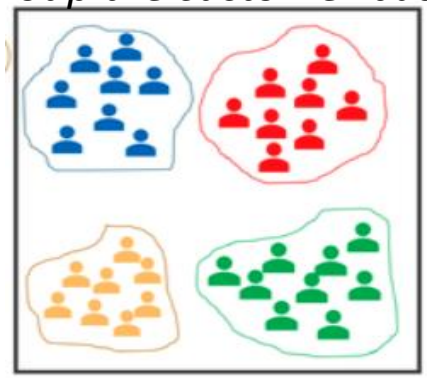
- User behavior analysis: Using unlabelled data about different human traits and human interactions to put each individual into different groups based on their behavior patterns

e.g., Identifying a potential customer base for selling a product

Clustering Algorithm to group the customer base

Customer data

e.g., gender, age

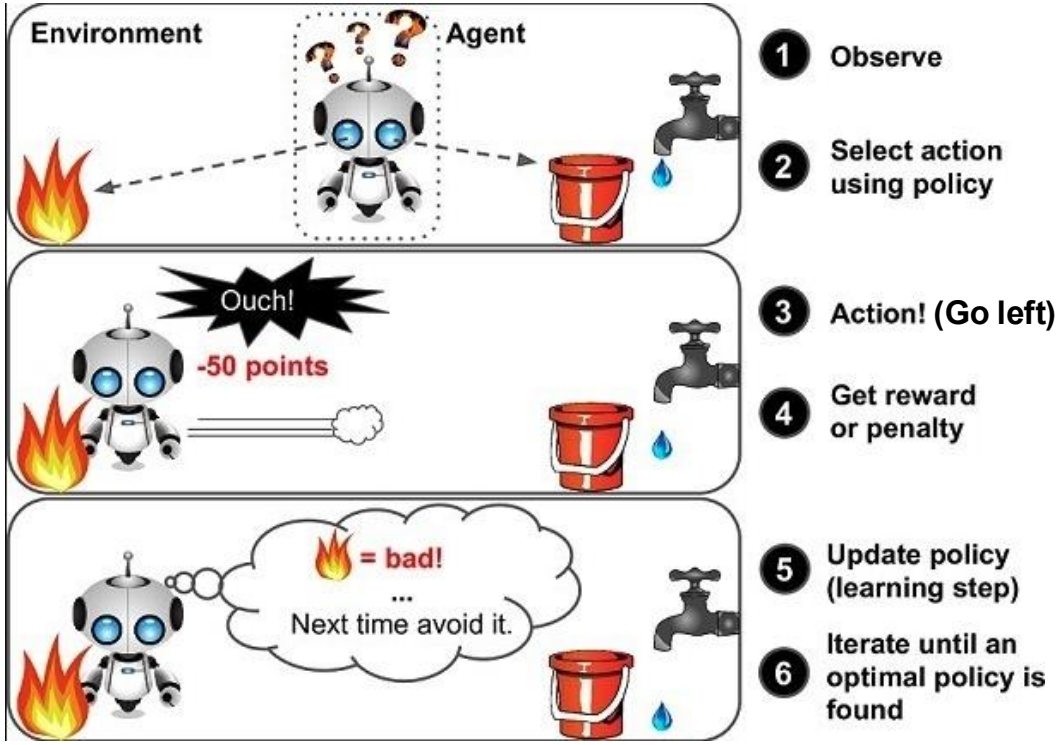


Selling the product to the identified customer group

- Market basket analysis: Identifying the likelihood that certain items will always appear together
  - Famous example: Men between 30- 40 years in age, shopping between 5pm and 7pm on Fridays, who purchased diapers were most likely to also have beer in their carts

Reinforcement Learning

- Consider an agent that interacts with a certain environment, changing its state, and receives rewards (or penalties) for its actions
- Goal: Finding patterns of actions for the agent, by trying them all and comparing the results, that yield the maximum cumulative expected reward points in achieving the goal

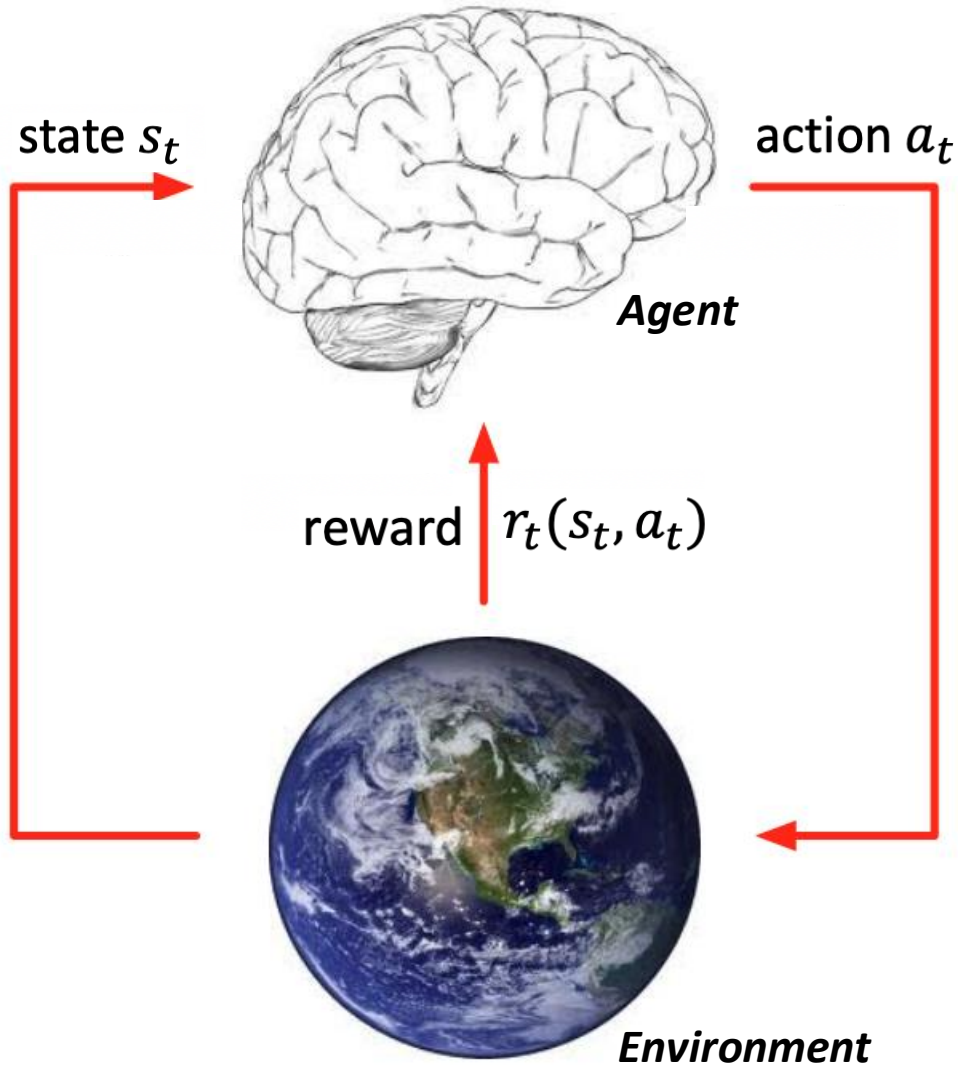


https://i.stack.imgur.com/gWmhs.jpg

Reinforcement Learning Model

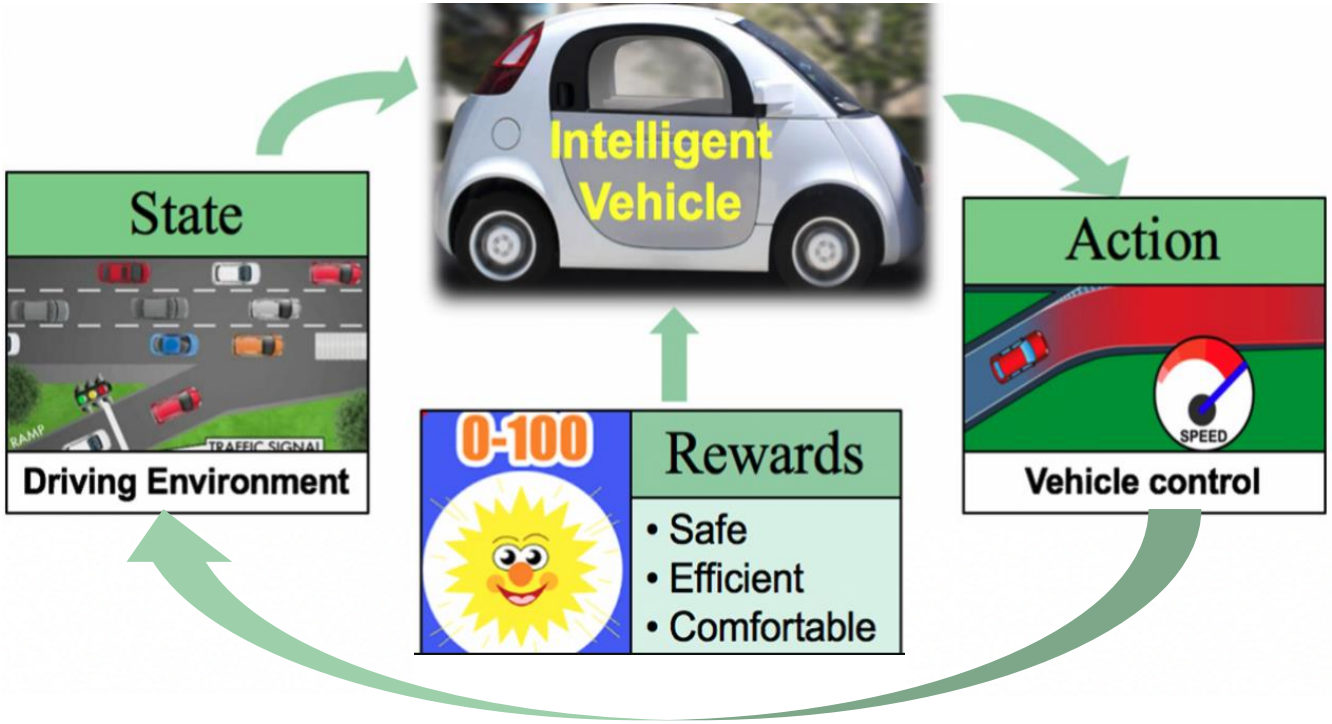
- At each time step $t = 1, 2, \ldots T$
  - Agent observes the current state s_t and take action a_t
  - Agent collects the reward $r_t(s_t, a_t)$
  - Next state s_{t+1} is randomly generated according to transition function P
  - Collection of the 3-tuple $\{(s_t, a_t, r_t)\}_{t=1}^{t=T}$ is called an episode

- **Note**: Agent's actions might not affect the immediate state of the environment but impact the subsequent ones (yields delayed rewards). Hence, the Agent doesn't learn whether a certain action is effective until much later in the episode



Reinforcement Learning Examples

Self driving cars

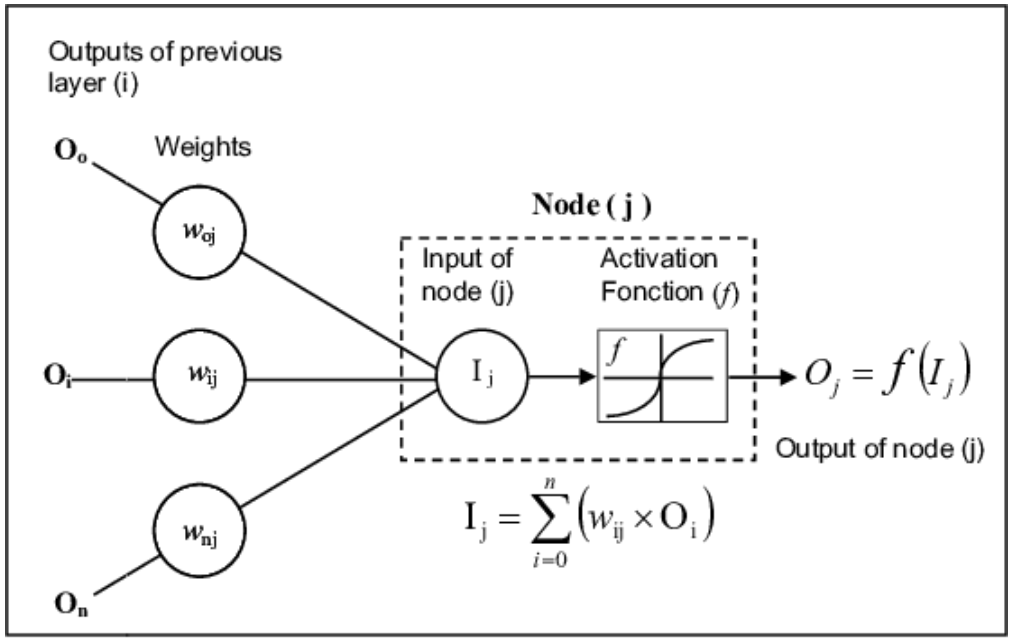


AlphaGo

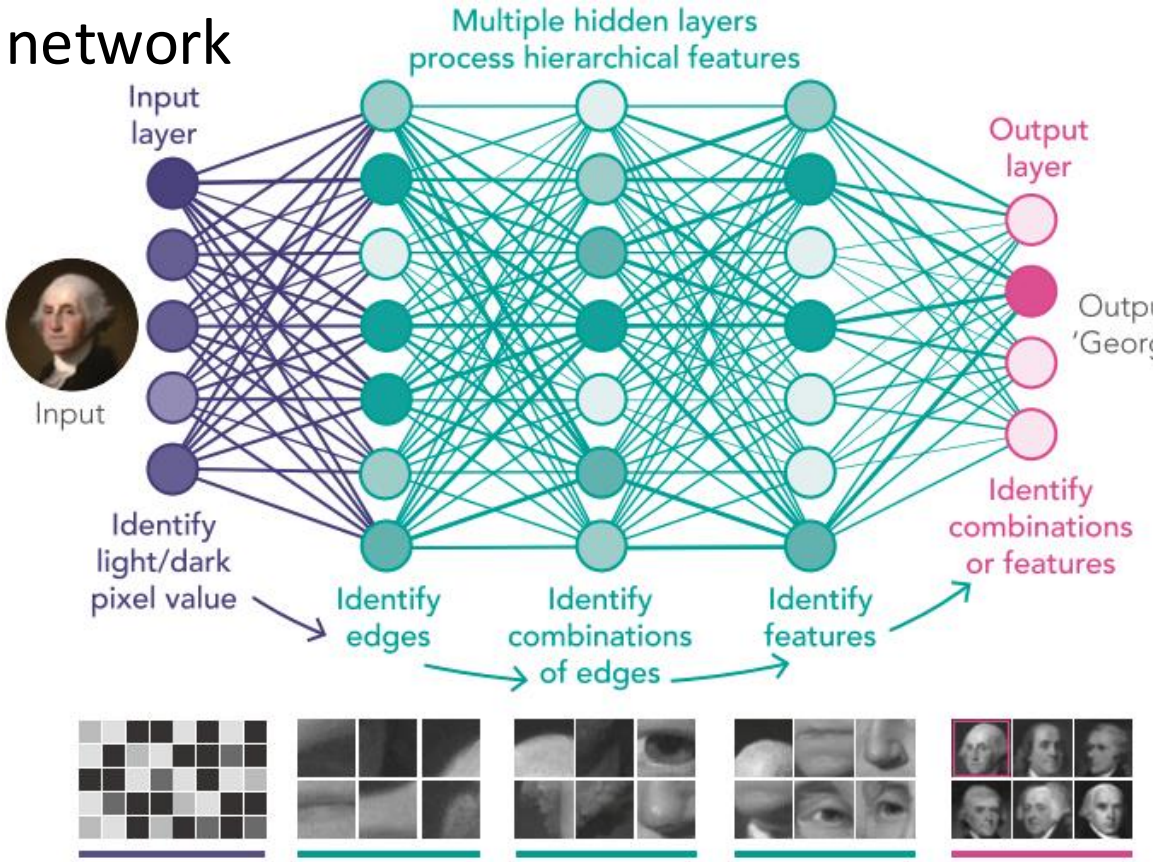
An AI-powered system that beats the reigning 3-times European Champion of the complex boardgame *Go*, by 5 points to 0.

Deep Learning

- Basic unit is a node (also called “neuron”) which is composed of weights, biases, computing function, and an activation function
- Layers of nodes are used to form a deep neural network



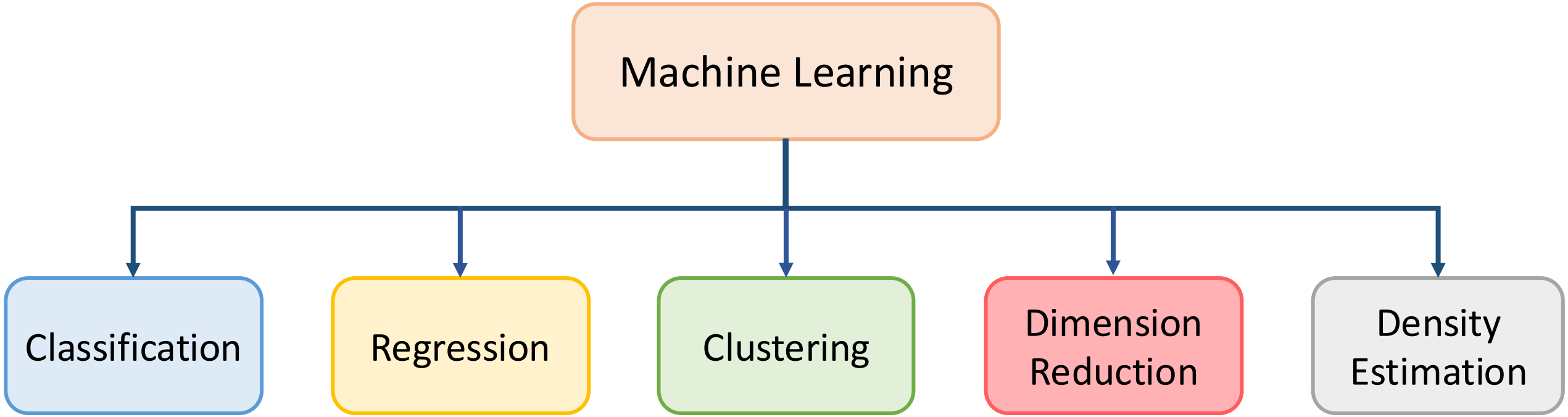
Schematic diagram of a node in deep neural network



Schematic diagram of a deep neural network

https://www.pnas.org/content/pnas/116/4/1074.full.pdf

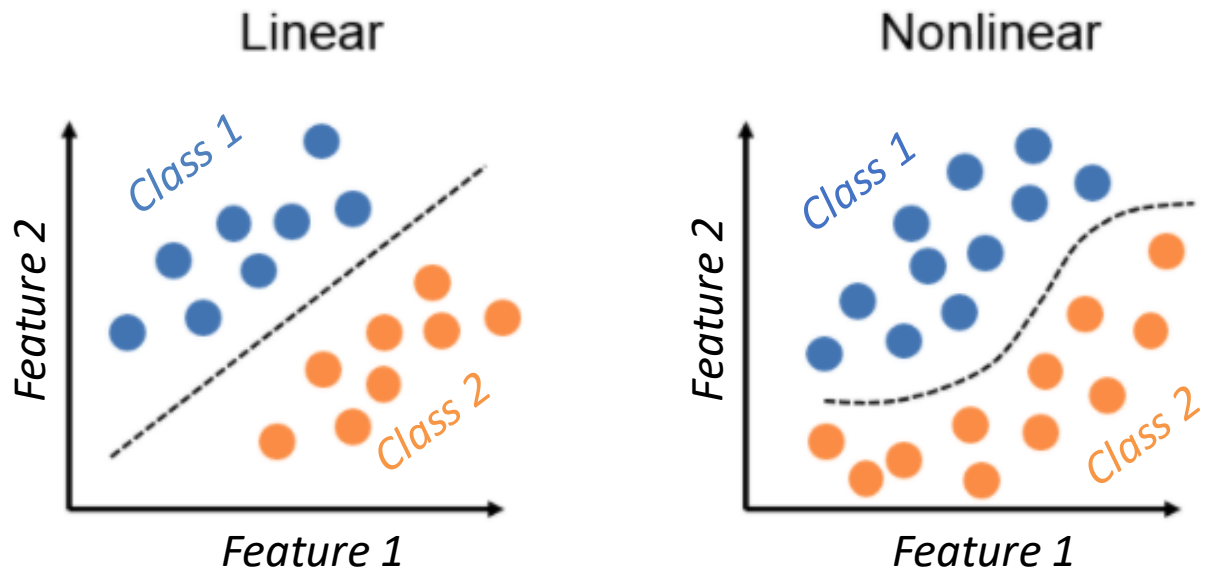
Another Categorization of Machine Learning Algorithms



Jump to ML tools

Classification

- An instant of supervised learning where the labels of the training data $\mathcal{Y}$ are from a finite set of labels
- Classification is the process of dividing data into multiple classes.
- Input data is divided into categories based on its characteristics or features.



Interpretation of linear and nonlinear binary (2-class) classification

Classification Model

- Most basic version of classification problem has two classes with labels $\mathcal{Y} = \{-1, +1\}$
- E.g., In security related problems -1 means “benign” (normal emails, legitimate network traffic) while +1 corresponds to “malicious” (spam, malware, intrusion attempt)
- Uses a classifier $f(x) = \text{sgn}\{g(x)\}$, where $g(x)$ is called **score function** (e.g., $g(x) = w^T x$)
- A real-valued function $g: x \to \mathbb{R}$ provides a convenient way of modeling the classifier f. i.e., when $g(x) < 0$, classifier returns -1 as the class and if $g(x) \geq 0$ it returns $+1$ as the class
- Ideal loss function: 0/1 loss function

$$l_{01}\big(yg(x)\big) = \begin{cases} 0, if\ yg(x) \geq 0 \\ 1, otherwise \end{cases}$$

Correct Classification

Incorrect Classification

- Challenge with l_{01} is that it is non-convex, making the following minimization problem associated with classification quite challenging

$\min_w \sum_{i \in \mathcal{D}} l_{01}\big(yg(x; w)\big) + \gamma\rho(w)$, where w is the real-valued parameter vector of f

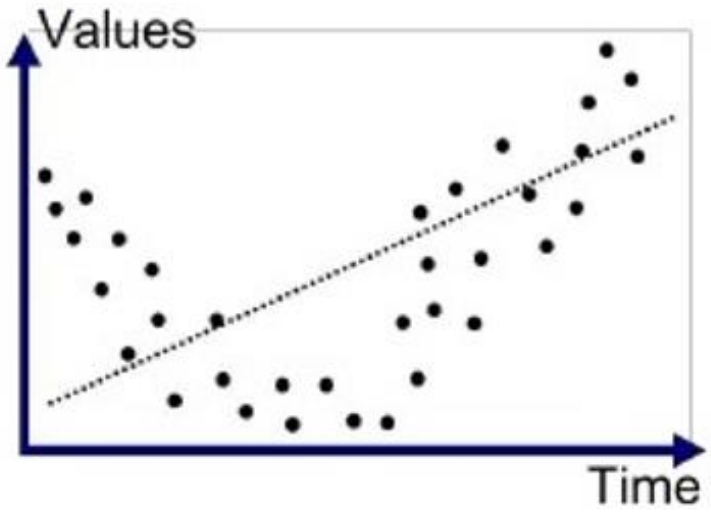
Classification Model

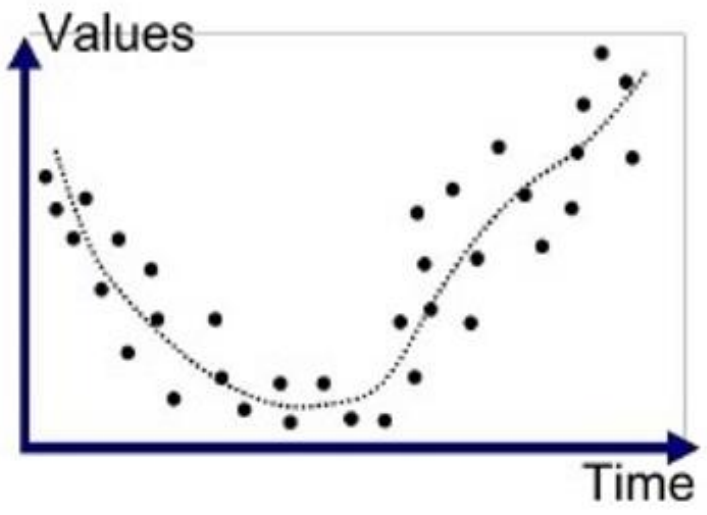
- For linear classifiers: $g(x) = w^T x$
- Linear classifier with hinge loss function (a convex relaxation of 0/1 loss function) solves the following optimization problem

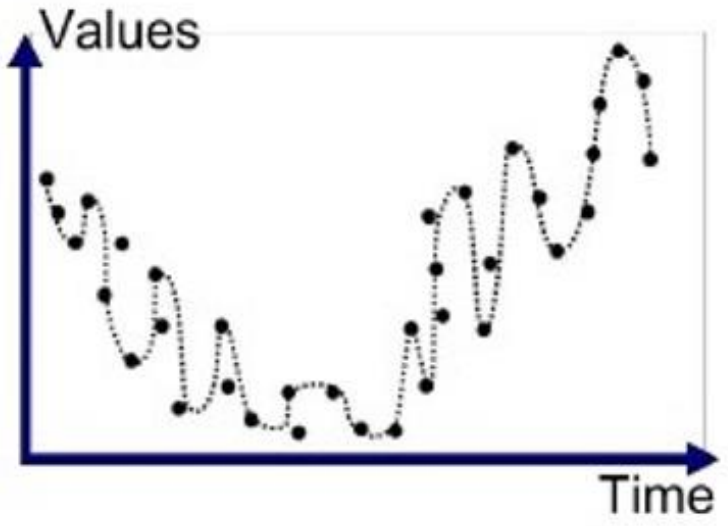
$$\min_{w} \sum_{i \in \mathcal{D}} \underbrace{\max\{0, 1 - y_i w^T x_i\}}_{\textbf{Hinge loss}} + \underbrace{\gamma ||w||_2^2}_{\boldsymbol{l_2 \text{ regularization with parameter } \gamma}}$$

- General form of a multi-class classifier: $f(x) = \arg\max_{y \in \mathcal{Y}} g(x, y)$
  - e.g., in image classification $g(x, y) = \Pr\{y|x\}$ --- Probability distribution over labels $\mathcal{Y}$ given feature vector x

Under-fitting and Over-fitting







Underfitted

Good Fit/Robust

Overfitted

High Bias: Average predictions from the learned model drastically deviate from the actual values

High Variance in the learned model

Under-fitting and Over-fitting of Data

- Both UF and OF are terms related to **NOT generalizing data outside of the training set**.
  Let
  - $D = \{(x_i, y_i)\}_{i=1}^{i=n}$: Dataset D with data values x and their corresponding real valued labels y
  - f: Learned model
  - Assume the label can be written as $y = h(x) + \epsilon$, where h is the true model and ϵ is zero mean noise with variance σ^2

- UF and OV describe the sources of the square errors as:

$$\mathbb{E}[\big(y - f(x)\big)^2] = (\text{Bias}[f(x)])^2 + \text{Var}[f(x)] + \sigma^2$$

  - $\text{Bias}[f(x)] = \mathbb{E}[f(x)] - h(x)$
  - $\text{Var}[f(x)] = \mathbb{E}[(f(x) - \mathbb{E}[f(x)])^2]$

Under-fitting and Over-fitting of Data

$$\mathbb{E}[(y - f(x))^2] = (\text{Bias}[f(x)])^2 + \text{Var}[f(x)] + \sigma^2$$

- **Bias:** Model not correct or doesn't have enough Degrees Of Freedom to fit the data (under-fitting). For example, trying to fit a linear model to a polynomial.

$$\text{Bias}[f(x)] = \mathbb{E}[f(x)] - h(x)$$

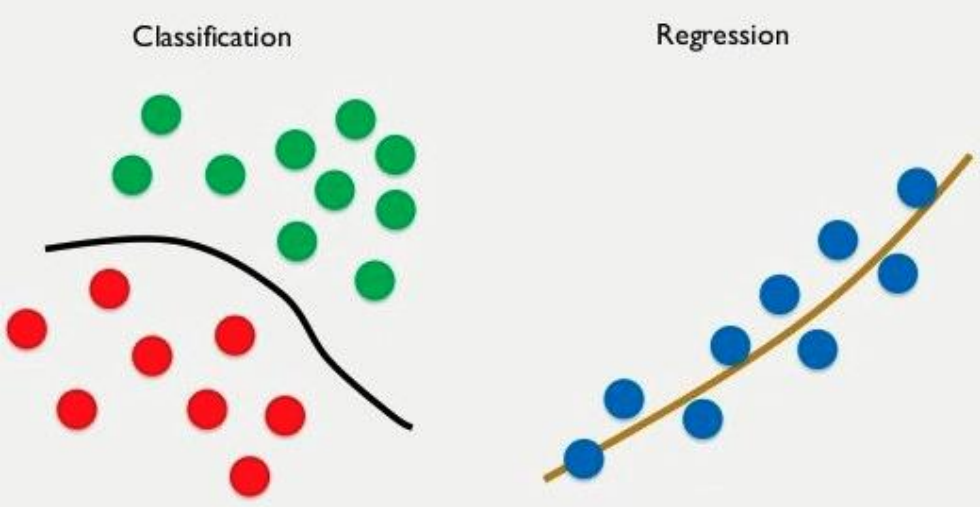
Trained Model *True Model*

- **Variance:** Caused by over-fitting. This leads to small error in training set but large errors in testing. This happens when there's too much freedom in the model.

$$\text{Var}[f(x)] = \mathbb{E}[(f(x) - \mathbb{E}[f(x)])^2]$$

Regression

- An instant of supervised learning where the labels of the training data are real-valued. i.e., $\mathcal{Y} \in \mathbb{R}$
- In regression, the relationship between two variables present in the data population is estimated by analyzing multiple independent and dependent variables



Regression Model

- Since the labels are real values, it is unlikely we will ever get them exactly correct
- An appropriate loss function should penalize the predictions that are far from the true labels
- Typically, l_p norm is used as the loss function. i.e.,

$$l(f(x), y) = \left|\left|f(x) - y\right|\right|_p^p$$

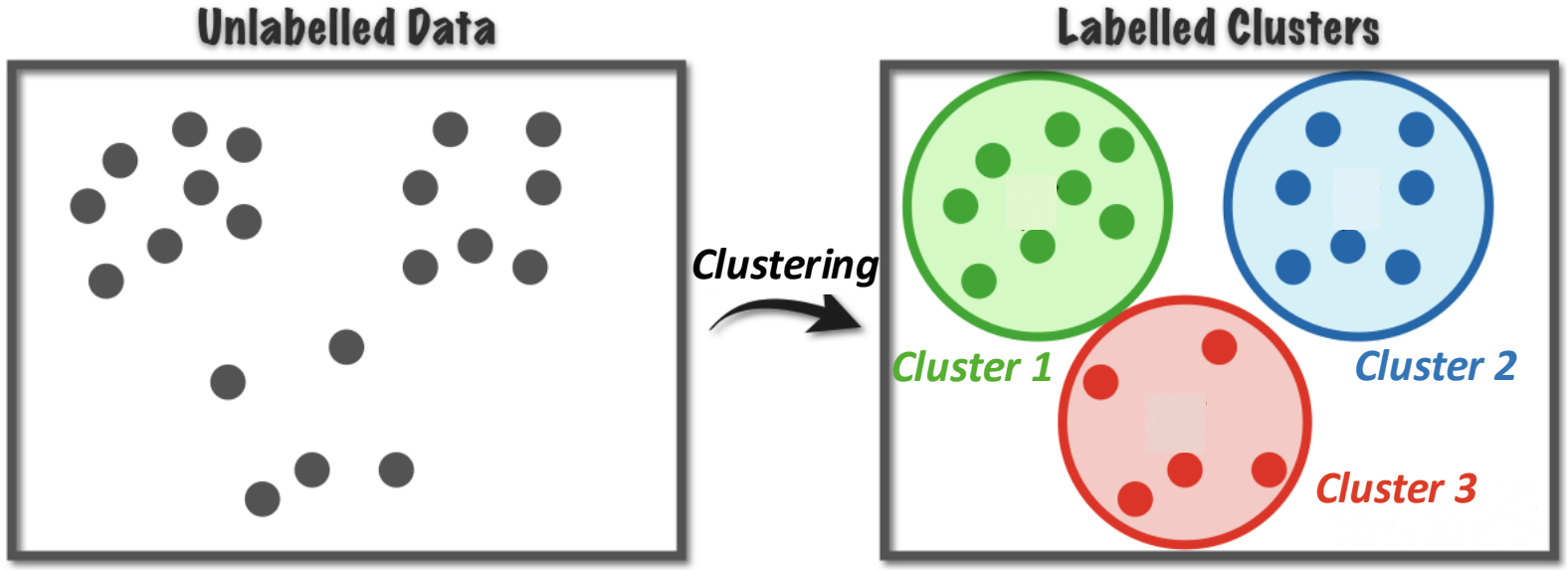
- **Linear Regression**: The model class $\mathcal{F}$ is set to all linear functions of dimension m.
  i.e., $f(x) = w^T b$, where $w \in \mathbb{R}^m$ is model parameters
- Goal of linear regression is to find set of model parameters w to minimize error on training data:

$$\min_{w \in \mathbb{R}^m} \sum_{i \in \mathcal{D}} l(w^T x_i, y_i) + \gamma \rho(w)$$

- **Lasso Regression**: Adding l_1 regularization ($\rho(w) = ||w||_1$) to the objective function
- **Ridge Regression**: Adding l_2 regularization ($\rho(w) = ||w||_2^2$) to the objective function

Clustering

- The most familiar example of unsupervised learning (use unlabeled data)
- Clustering is the process of grouping data and putting similar data into the same group
- Use a series of data parameters and go through several iterations before grouping data



Clustering Model

- Goal: Dividing the feature vectors x_i in a dataset $\mathcal{D} = \{x_i\}$ into a collection of subsets $\mathcal{S}$, such that feature vectors in each subset $S \in \mathcal{S}$ are "close" to the mean feature vector of S for some measure of closeness.
  - Set of subsets $\mathcal{S}$ forms a partition of the dataset $\mathcal{D}$
  - Common measure of closeness: l_2 norm --- sum of squared distance
- Clustering solves the following optimization problem

$$\min_{\mathcal{S},\mu} \sum_{S \in \mathcal{S}} \sum_{i \in S} l(x_i, \mu_S)$$

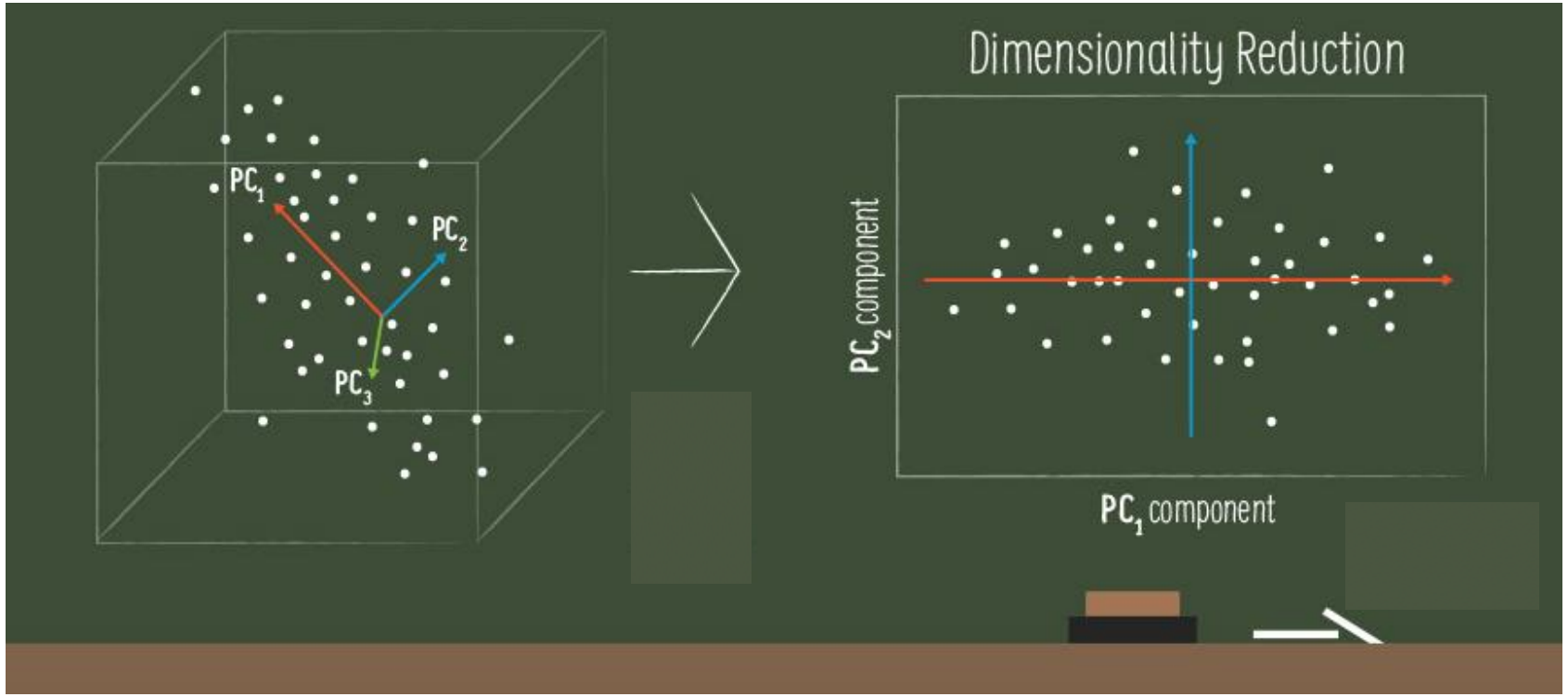
  - l: measure of closeness (distance function)
  - μ_S: An aggregation measure of the data in cluster $S \in \mathcal{S}$ (e.g., mean)
  - Regularization can be added to control the model complexity when number of clusters is unknown up front
- **k-means clustering**: Heuristic approximation of the above optimization problem when l_2 norm is used as the measure of closeness.
  - Iteratively updates cluster means and moves data points to a cluster with the closest mean

Classification vs. Clustering

Classification	Clustering
Dataset consists of features x and labels y	Dataset consists of only features x
Supervised (Class labels are known)	Unsupervised (Class labels have to be learned)
Learn a classifier model from set of labelled data and employ it to predict the labels of data points using their features	Identify similar groups (clusters) of data points (features)

Dimension Reduction

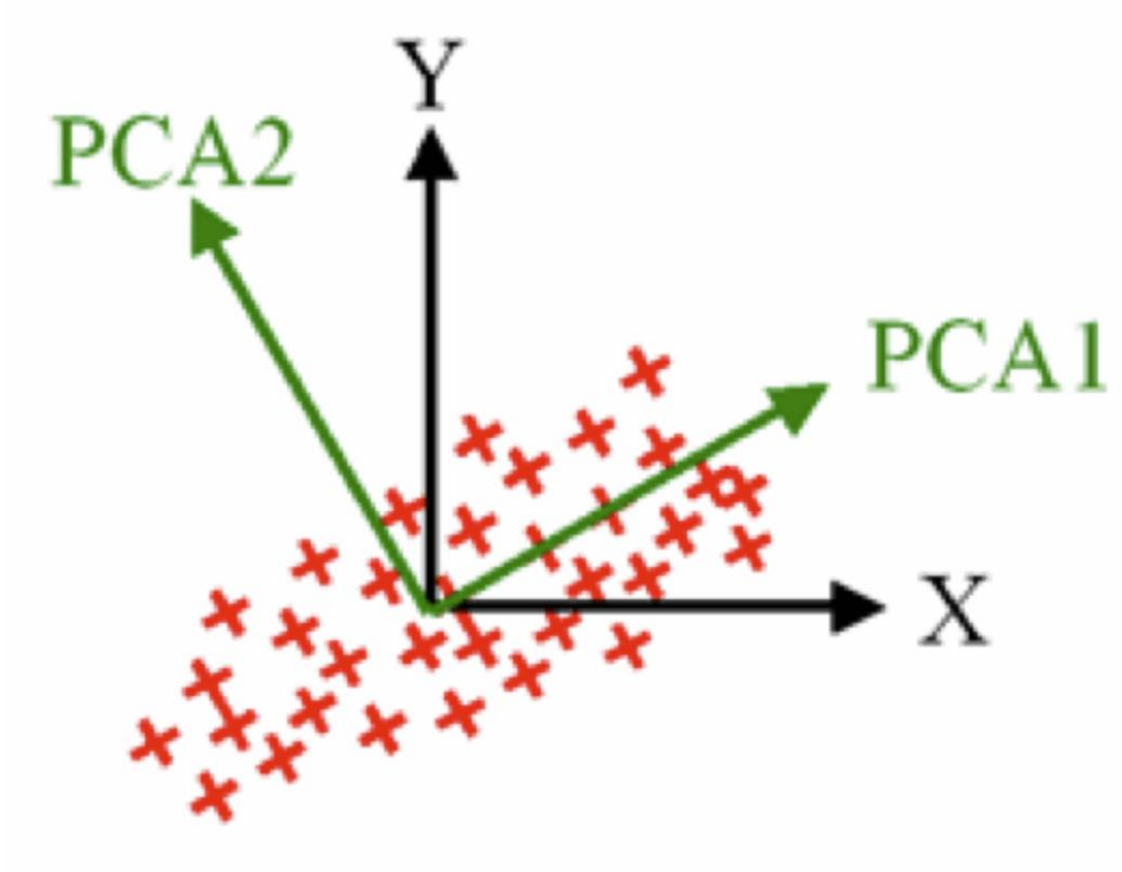
- Representation of high dimensional data with multiple variables using principle variables, without loosing any vital data.



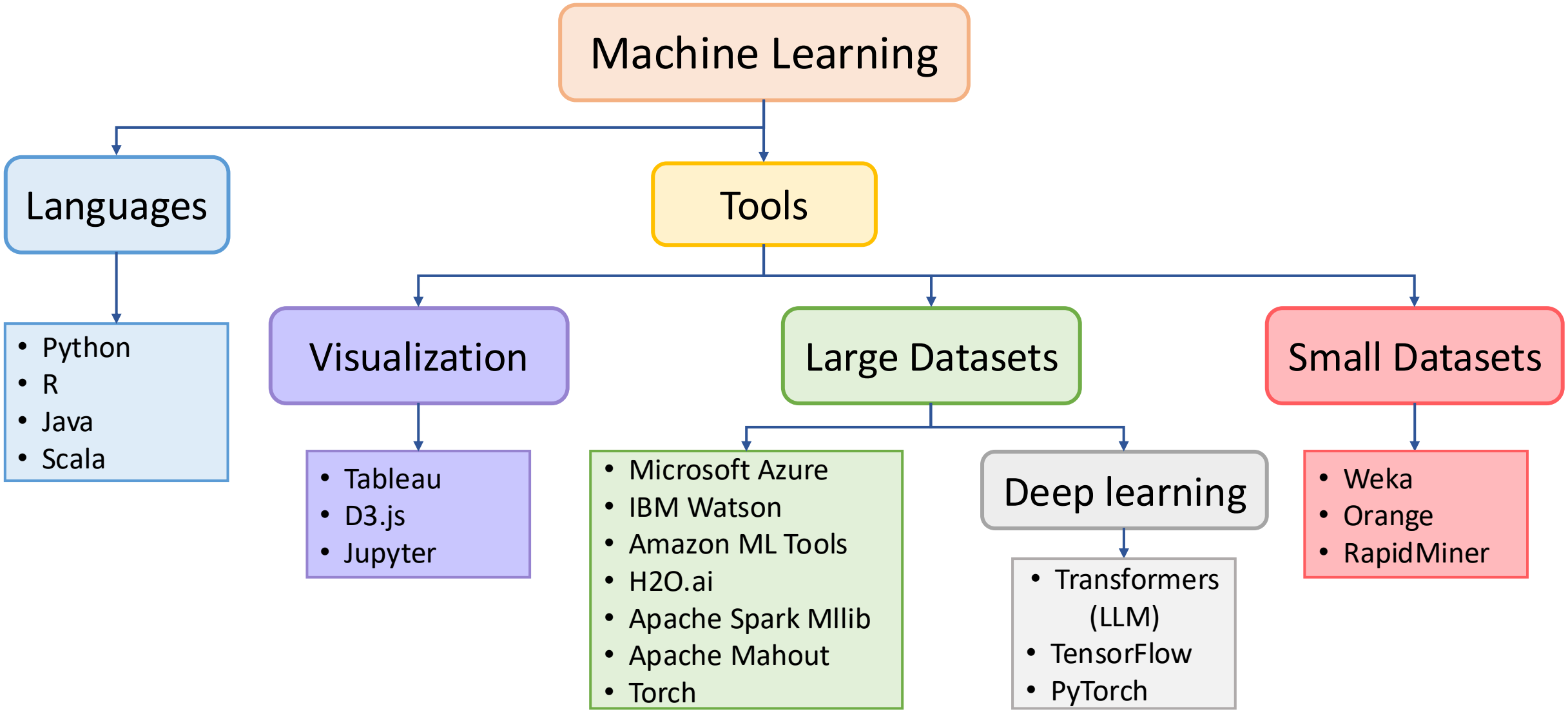
- Commonly used dimensionality reduction algorithm: Principle component analysis (PCA)

Principle Component Analysis (PCA)

- **Intuition**: Eigenvector corresponding to the largest eigenvalue of $\boldsymbol{X}^T\boldsymbol{X}$ (Covariance of data) represents the direction of the largest variance in the data



Machine Learning Tools



What about the Security of AI?

- While ML can be used to handle big data when the models are well trained, ML models are susceptible to adversarial attacks – adversarial inputs

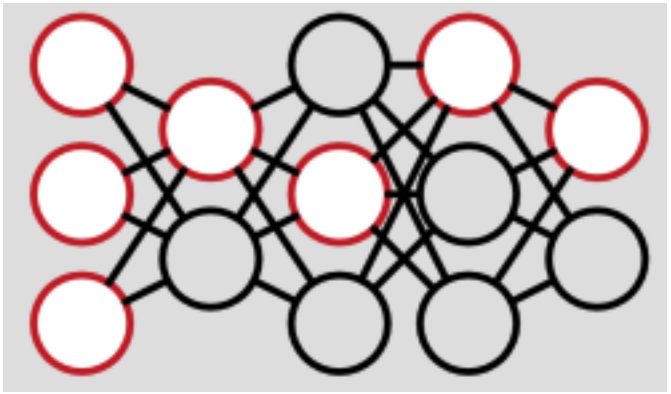
- Hence using ML in autonomous driving, robotic surgeries, drone-based missions, etc. can lead to catastrophic disasters in adversarial environments

- Developing ML algorithms that are robust to adversarial inputs is a growing research area (Adversarial Machine Learning)

Examples of Attacks on Machine Learning Algorithms

Normal image

Trained ML model

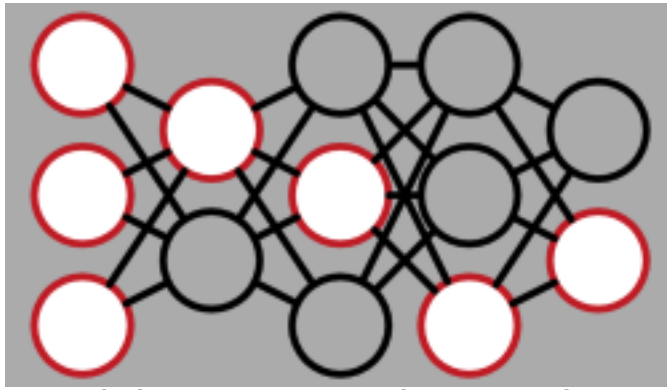


Predicted Label

Ant – 98%
Spider – 1.5%
Bee– 0.5%

Manipulated image by adversary (perturbed with noise)

Trained ML model



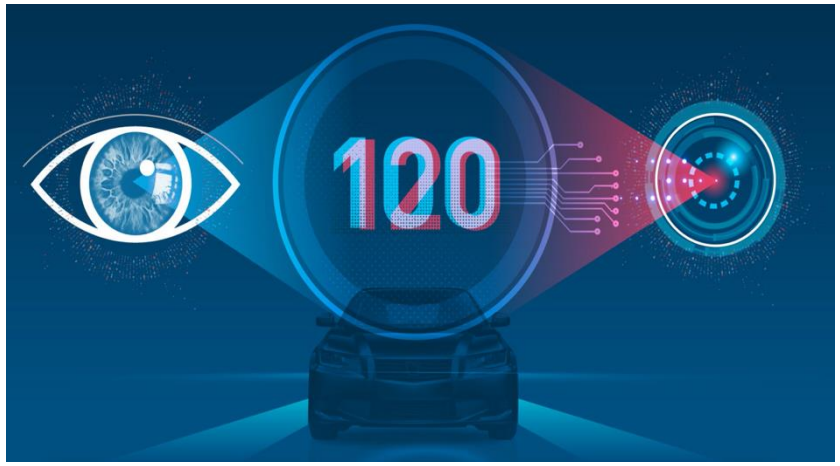
Predicted Label

Elephant – 93%
Bee – 6%
Spider --1%

Different model activations due to adversarial input

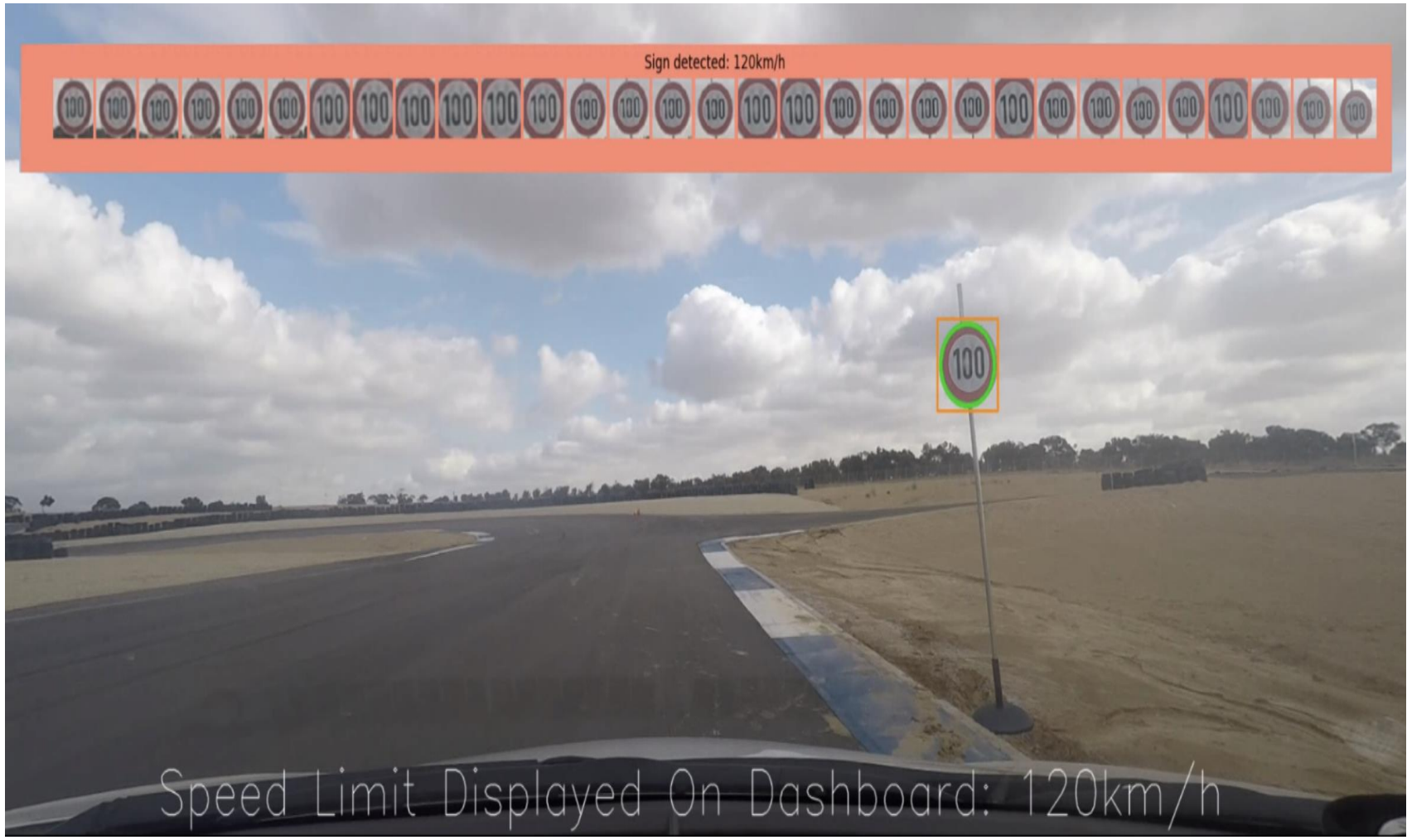
Examples of Attacks on Machine Learning Algorithms

Adversarial 100 km/h traffic sign forces ML algorithm to misclassify 100 km/h as 120 km/h



Adversarial 100 km/h traffic sign

Looks like dirt on the sign for human but malicious for a car rely on ML based image recognition system for navigation



Examples of Attacks on Machine Learning Algorithms

In the following examples adversarial modifications are shown in ~~**crossed-out**~~ texts and **Red** colored texts

Attack on Negative/Positive Review Classification

Predicted label before attack: 100% **Positive** **Predicted label after attack**: 93% **Negative**

This Starbucks location is located in the Bally's Grand Bazaar Shops. It's open 24/7 and it is huge. There is plenty of seating. Most of the seating is stadium type seating with benches. They also have an out door patio. The staff is very friendly and attentive to the guests. I do notice that they are under staffed sometimes when they are busy. They 'll get your drinks ~~out~~ pretty fast ~~though~~. Also, this ~~location~~ place is not owned by the ~~casino~~ property so they ~~don't~~ do n't charge outrageous prices ~~like the location~~ as a place on ~~the~~ an Linq promenade ~~does~~. Definitely one of my favorite Starbucks stores. Stop by if your on the Strip.

Today's Practice Lab

- *Review of Python syntax and libraries*

- *Python building blocks that we need for the coming weeks*