CPSC 483
Introduction to Machine Learning

Dr. Anand Panangadan
apanangadan@fullerton.edu

What we will cover today

- Introductions
- What is Machine Learning?
- Activity: some issues to think about
- Types of ML approaches
- A ML project example
- Course Outline (syllabus overview)

A bit about myself

Associate Professor
Computer Science Department
Office: CS 548
Office hours: Thu 3-4pm, 7-9pm

Research interests

- Applications of machine learning
  - Transportation
- Social network analysis
  - Text mining
- Sensor networks
  - Classification of time-series

A bit about yourself?

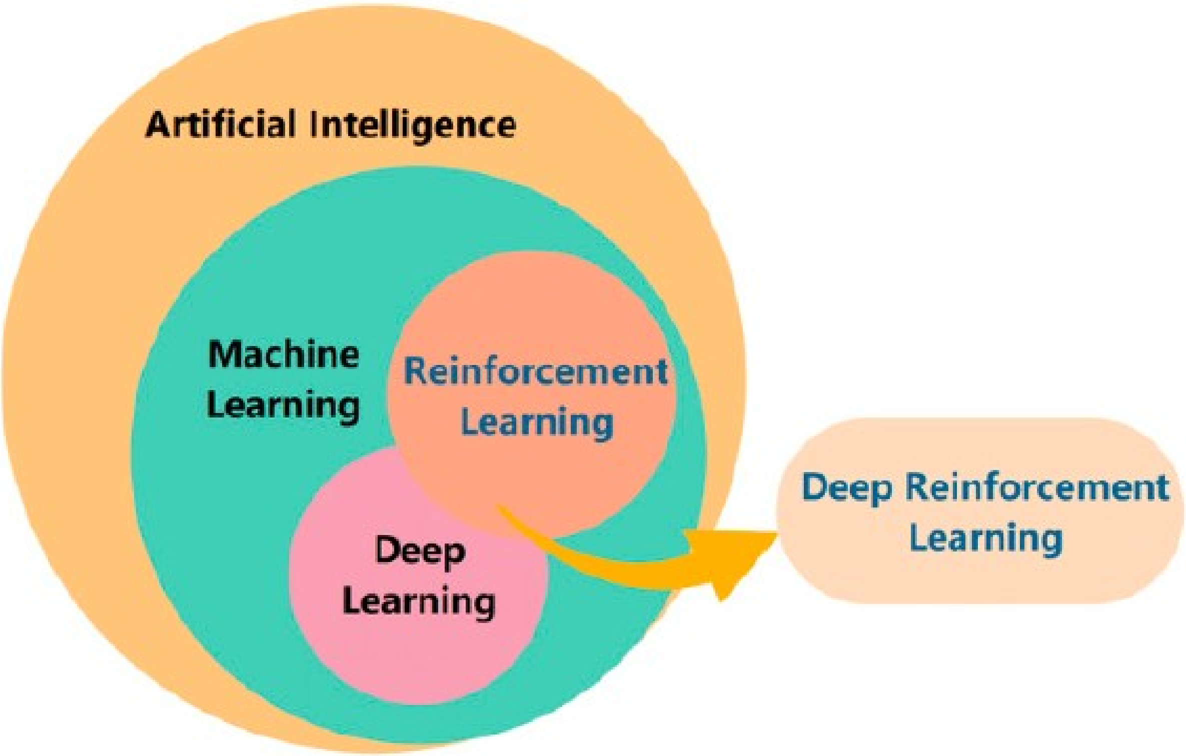
- In next 5 minutes:
  - Form groups of 2-3 students
  - Introduce yourselves to each other
  - Learn your partner's name
  - What they have heard of ML
  - Anything else they would like to share
- When everyone is done
  - Introduce your new friends to the class

What is Learning?

- **Learning** is a process of **acquiring *knowledge***
- **Knowledge**
  - Facts (data), patterns, concepts, rules, models, skills
- **Example forms of knowledge**
  - Facts (A thing that is known to be true, e.g., a red car, π=3.141592….)
  - Patterns (repeated forms that represent the nature of facts)
  - Concepts (classes, class hierarchies created by generalization from patterns)
    - 0, 2, 4, 6, … => 2n (even number concept)
    - Dog, Cat, Bird => Animal (animal class hierarchy)
  - Rules
    - P ☾ Q, e.g., if it rains, ground is wet. If oil price is too high, people travel less.
  - Models
    - Abstract representation, e.g., mathematical functions such as v = d/t, f(n) = f(n-1) + f(n-2), data model, design model, simulated systems.
  - Skills
    - Being able to do a complex activities, e.g., playing soccer, writing programs, etc.

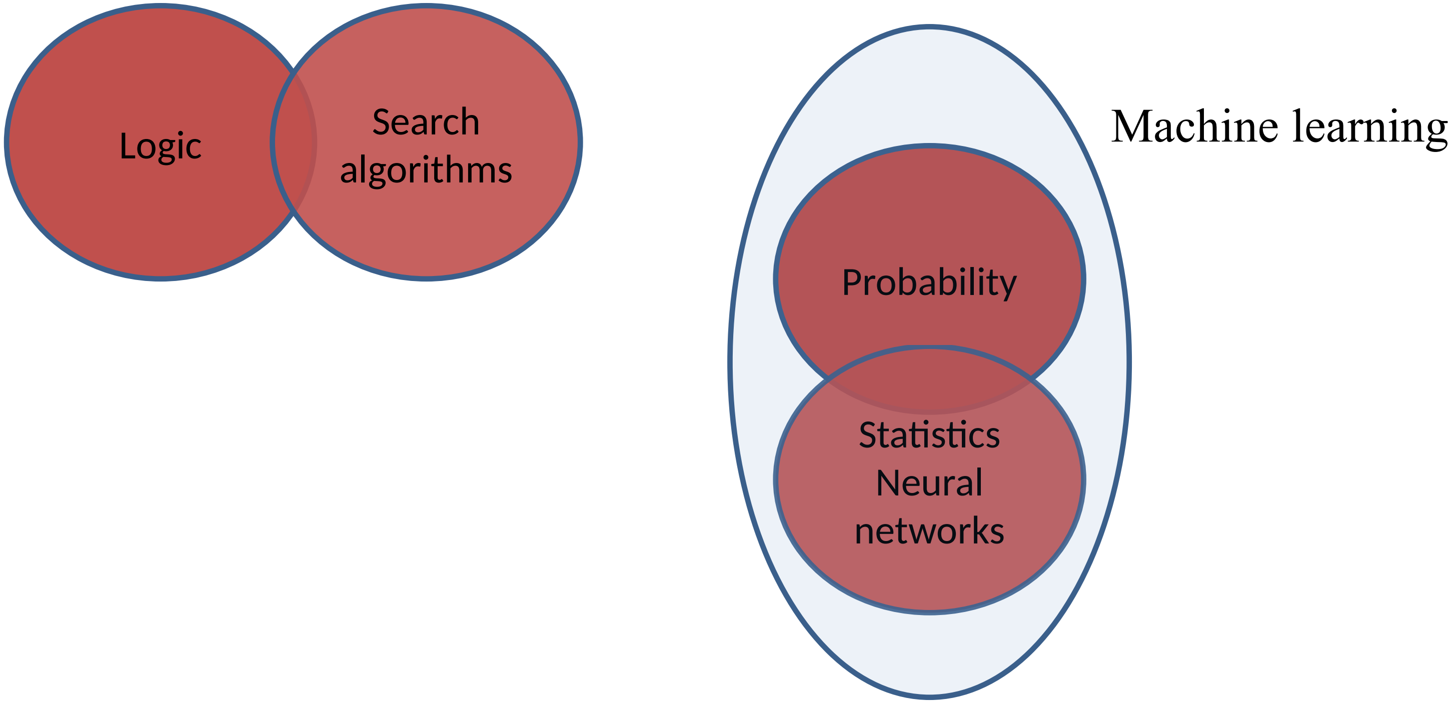
Use of Knowledge

- **Knowledge enables Intelligence**
  - Recognition (Matching sensory information or patterns with the existing knowledge in memory)
  - Understanding (Perception of intended meaning of the sensory information by integrating it with the existing knowledge to make sense out it)
  - Communication (Exchanging information)
  - Reasoning (Reaching a conclusion through generalization, logical, and statistical approaches)
  - Planning (Process of thinking about the activities to achieve a goal)
  - Decision making (Selection of a course of action among several alternative options)
  - Problem solving (Finding solutions)
  - Imagination, abstract thinking, creativity (Thinking or creating something that may not exist based on the existing knowledge)
  - Learning (Acquiring new knowledge and skills)



Artificial Intelligence
Machine Learning
Reinforcement Learning
Deep Learning
Deep Reinforcement Learning

ML is one approach to AI



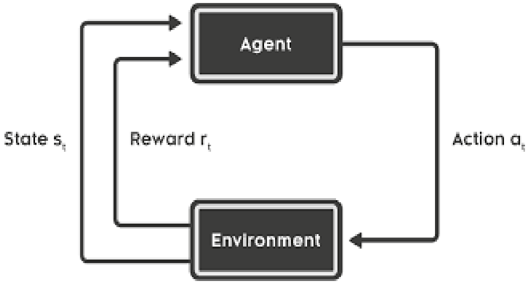
Machine Learning (ML)

- **Definitions**
  - "A field of study that gives computers the ability to learn without being explicitly programmed." (Samuel, 1959)
  - "Any change in a system that allows it to perform better the second time on repetition of the same task or on another task drawn from the same population" (Simon, 1983)
  - "A computer program that learns from experience **E** with respect to some class of tasks **T** and performance measure **P**. If its performance at tasks in T, is measured by P, it **improves** with experience E." (Tom Mitchell, 1997)
  - **ML is a study of methods to develop a system that can learn**.

Key Components for Machine Learning

- **Goal** (Learning goals determine learning tasks)
  - To acquire knowledge by finding patterns, relationships, or policies
- **Input data set or experience** and **learning algorithm**
  - **Learning** or **training** from the **input data** to find the **parameters**
  - **Types of input data**
    - Example data with label/class/target often called "**training data**"
    - Example data **without** label, just "**data**"
    - Example experience data in an environment

- **Output:** Knowledge in different forms
  - Patterns (e.g., repeated data, forms, or shapes)
  - (Concepts) classes (e.g., Yes/No, high/middle/low)
  - Clusters (e.g., a group of data with similar properties)
  - Rules (e.g., If credit score > 800, the loan application is appr
  - Functions (e.g., $f(x) = x^2 + 2x + 3$)
  - Policies of actions (e.g., making decision or a series of actions to perform from data and experience)



An Example Data Set

Class RISK	CREDIT HISTORY	DEBT	COLLATERAL	INCOME
high	bad	high	none	$0 to $15k
high	unknown	high	none	$15 to $35k
moderate	unknown	low	none	$15 to $35k
high	unknown	low	none	$0 to $15k
low	unknown	low	none	over $35k
low	unknown	low	adequate	over $35k
high	bad	low	none	$0 to $15k
moderate	bad	low	adequate	over $35k
low	good	low	none	over $35k
low	good	high	adequate	over $35k
high	good	high	none	$0 to $15k
moderate	good	high	none	$15 to $35k
low	good	high	none	over $35k
high	bad	high	none	$15 to $35k

Data Set vs Training Data Set

Label

ID	# of legs	# of ears	Fur	Sound
O1	4	2	Y	Meow
O2	4	2	Y	Bark
O3	4	2	Y	Meow
O4	3	2	Y	Bark
O5	4	2	Y	Bark

ID	# of legs	# of ears	Fur	Sound	Class
O1	4	2	Y	Meow	Cat
O2	4	2	Y	Bark	Dog
O3	4	2	Y	Meow	Cat
O4	3	2	Y	Bark	Dog
O5	4	2	Y	Bark	Dog

Data set without label. Most common form of data

Training Data

Questions to think about

1. Most people believe that all humans have some universal rights and that animals have some rights too. Will there be a time that “thinking” machines will have rights too?
2. Who should be responsible if a system based on a machine learning model causes harm? (the designer, the user, …)
3. Bias and fairness are present in ML systems trained on biased real-world data. Is it necessary to understand the internals of an ML system to mitigate bias (or is it sufficient to consider ML systems as “black boxes”)?
4. Read about the energy and water costs of training large language models. Briefly present some of the steps that are being taken to address this issue.
5. Is it necessary to pattern ML after what is known about how humans learn, or strict “engineering” approaches sufficient?

Class exercise

- In next 5 minutes:
  - Groups of 3 students
    - Roles: manager, recorder+presenter, timekeeper
  - Discuss one of the questions
- When everyone is done
  - recorder+presenter tells the answer to the class

Classwork

- On Google Drive
  - **Link on Canvas**
- Go to today's date and section
  - Editable and viewable by all
- Create a new Google Document and name it with your groupmates' names
  - E.g.: Garcia-Joshi-Nguyen
  - Write the names of your group members
  - Add your answers
- Will not be graded
- Can easily share with rest of class

K-nearest neighbors algorithm

CLASSIFICATION

- Write a function that will predict if a name is masculine or feminine

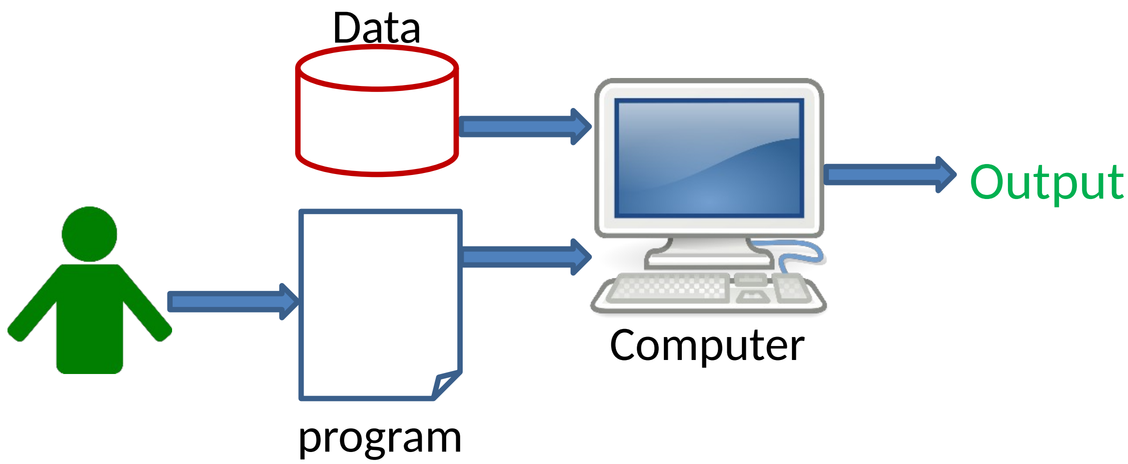
Input	Output
Laura	F
Carlos	
Jose	
Maria	
Belen	

Write a function to guess if a name is typically masculine/feminine

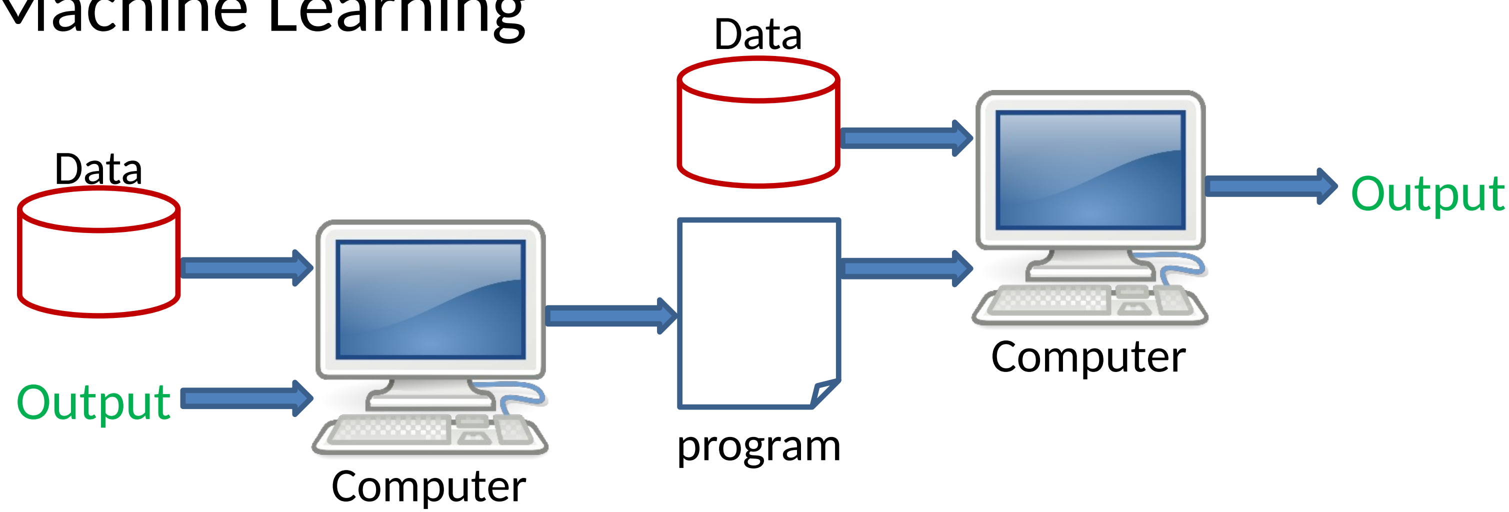
```
def m_or_f(name):



    ??
```

```
def m_or_f(name):
  if name[-1]:
    return "F"
  else:
    return "M"
}
```

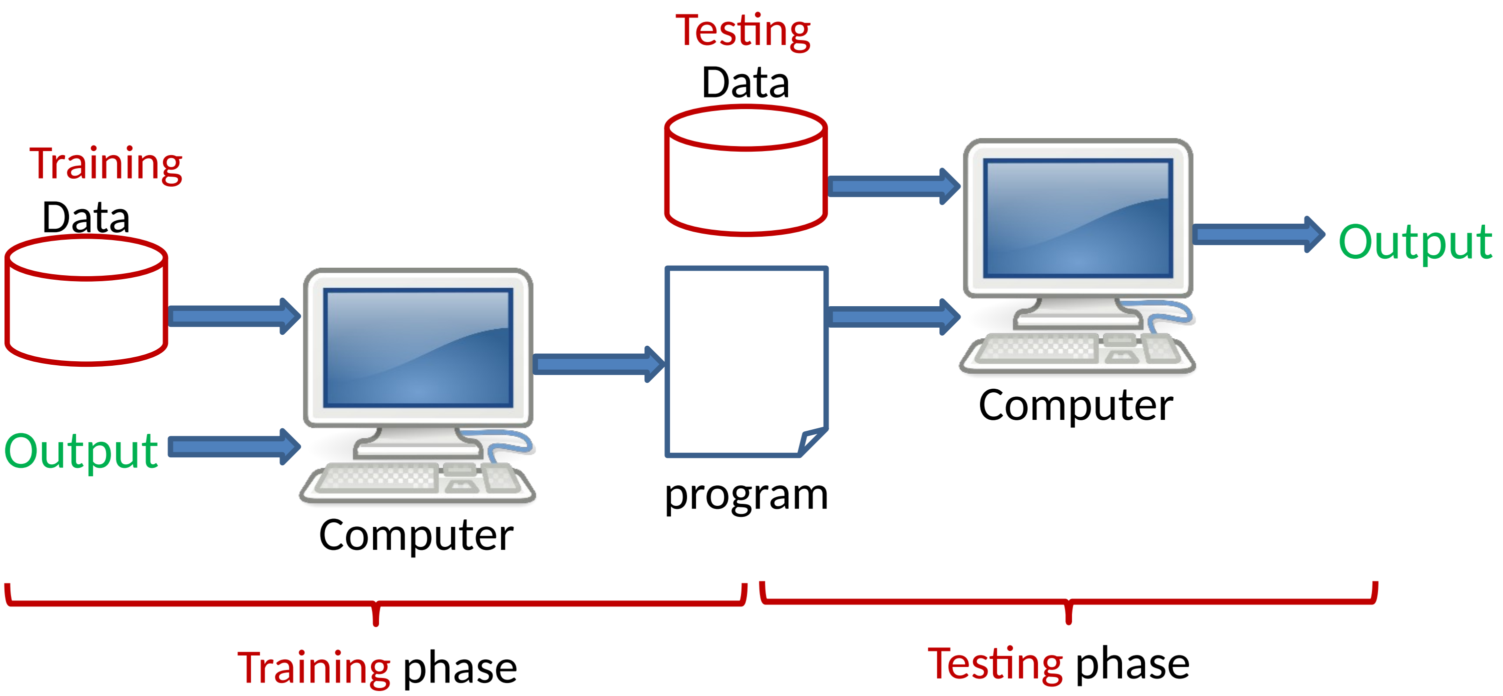
Traditional Programming



Machine Learning



Machine Learning



The machine learning approach

```
def m_or_f(name):
```

mydata =

Input	Output
Laura	F
Carlos	M
Jose	M
Maria	F
Belen	F

compare name with mydata.Input and find "closest" match – the *nearest neighbor –* and return corresponding output

Structure of training and test data

Input	Output
Laura	F
Carlos	M
Jose	M
Maria	F
Belen	F

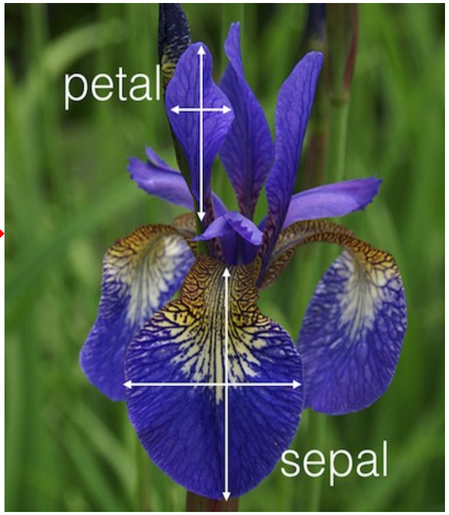
Iris setosa

Iris versicolor

Iris virginica

Input	nchars	lastchar	nvowels	Output
Laura	5	‘a’	3	F
Carlos	6	‘s’	2	M
Jose	4	‘e’	2	M
Maria	5	‘a’	3	F
Belen	5	‘n’	2	F

$\underbrace{\text{nchars} \quad \text{lastchar} \quad \text{nvowels}}_{\text{Features/Feature vector}}$ $\underbrace{\text{Output}}_{\text{Label/ Class}}$



Sepal. Length	Sepal. Width	Petal. Length	Petal. Width	Output
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
7.0	3.2	4.7	1.4	versicolor
6.3	3.3	6.0	2.5	virginica
…	…	…	…	…

Supervised Learning

Features

Labels/Classes

Petal Length	Petal Width	Sepal Length	Sepal Width	Species
5.1	3.5	1.4	0.2	Setosa
4.9	3.0	1.4	0.2	Setosa
6.2	2.9	4.3	1.3	Versicolor
6.3	3.3	6.0	2.5	Virginica
5.1	2.5	3.0	1.1	Versicolor
7.1	3.0	5.9	2.1	Virginica

an *instance*

Training data

5.1	3.5	1.4	0.2	?
7.1	3.0	5.9	2.1	?

Predictions

Supervised Learning

Features

Labels/Classes

Petal Length	Petal Width	Sepal Length	Sepal Width	Species
5.1	3.5	1.4	0.2	Setosa
4.9	3.0	1.4	0.2	Setosa
6.2	2.9	4.3	1.3	Versicolor
6.3	3.3	6.0	2.5	Virginica
5.1	2.5	3.0	1.1	Versicolor
7.1	3.0	5.9	2.1	Virginica

Training data

Split for evaluation

Test data

Supervised Learning

Features

Labels/Classes

Petal Length	Petal Width	Sepal Length	Sepal Width	Species
5.1	3.5	1.4	0.2	Setosa
4.9	3.0	1.4	0.2	Setosa
6.2	2.9	[illegible]	1.3	[illegible]
6.3	3.3	6.0	2.5	Virginica
5.1	2.5	3.0	1.1	Versicolor
7.1	3.0	[illegible]	2.1	[illegible]

train features

train labels

test features

test labels

Training data

Split for evaluation

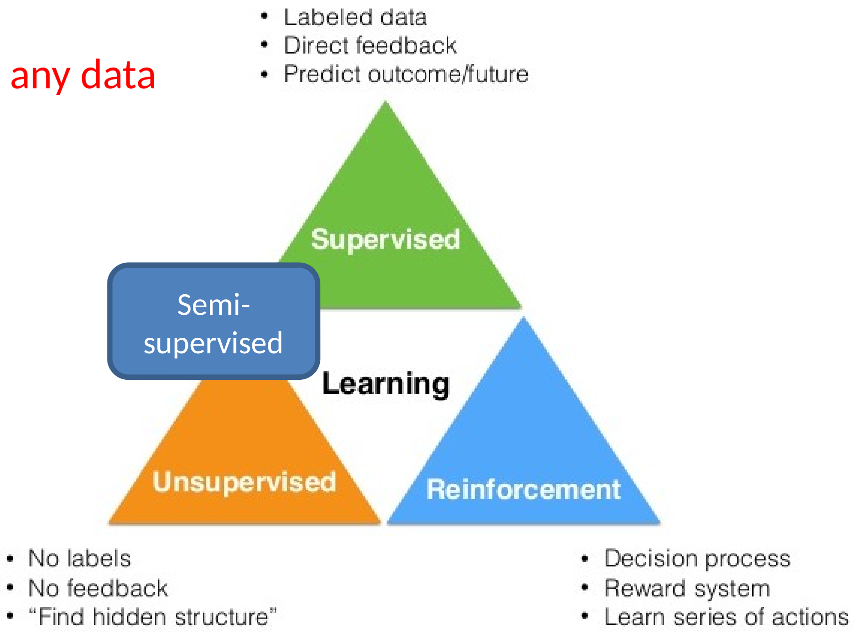
Test data

Learning Methods

- **Memorization**
  - Weak learning. Computer implementation is trivial, but **still important**.
- **Finding patterns**
  - Identifying repeated forms, finding associations and relationships
- **Categorization** and **classification**
  - Grouping data or patterns by similarity or relationships
- **Generalization**
  - Establishing abstract concepts from examples
- Analogy
  - Transferring information from one form to another for the purpose of explanation or clarification
- Synthesizing
  - Combining results or knowledge (obtained from different learning methods) into a coherent whole for advanced or higher-level knowledge

Taxonomy of Machine Learning

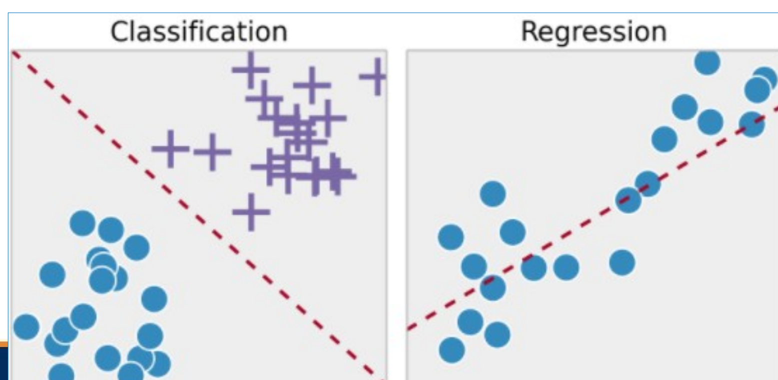
- Rote learning (Memorization and simple matching)
- Supervised learning
  - Learning by examples in a training data set through generalization or induction (called training)
  - **Two types of problems**: regression and classification
- Unsupervised learning
  - **Clustering**, **finding associations** or **features** in any data
- Semi-supervised learning
  - Mixing supervised and unsupervised learning
- Reinforcement learning
  - Learning rules or actions by trial-and-error
- **Other types of machine learning**
  - Deep learning
  - Online or adaptive learning
  - Transfer learning
  - Ensemble learning



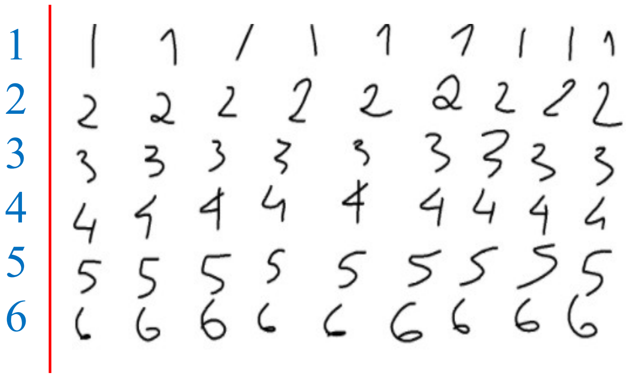
SUPERVISED LEARNING

Supervised Learning

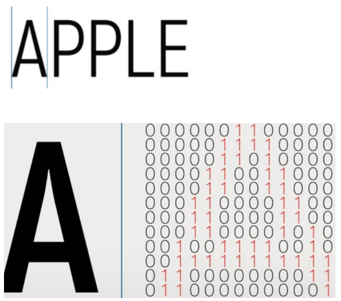
- Given a <u>training</u> data set
  - Learn a function or classifier $\mathbf{f: X \rightarrow Y}$
    - The process of learning **f** is called training.
  - Predict outcomes for a given X (**new data**):
    - If the learned function f is used to predict a continuous value, , it's called regression.
    - If the learned classifier f is used to determine a discrete value, , it's called classification.
- **Classification vs. regression**
  - Classification is to find the *decision boundary* that **separates** the different groups of data as clearly as possible.
  - Regression is to find the *function* that **fits** the data the best.



Classification: Digit or Character Recognition



Numbers in training data

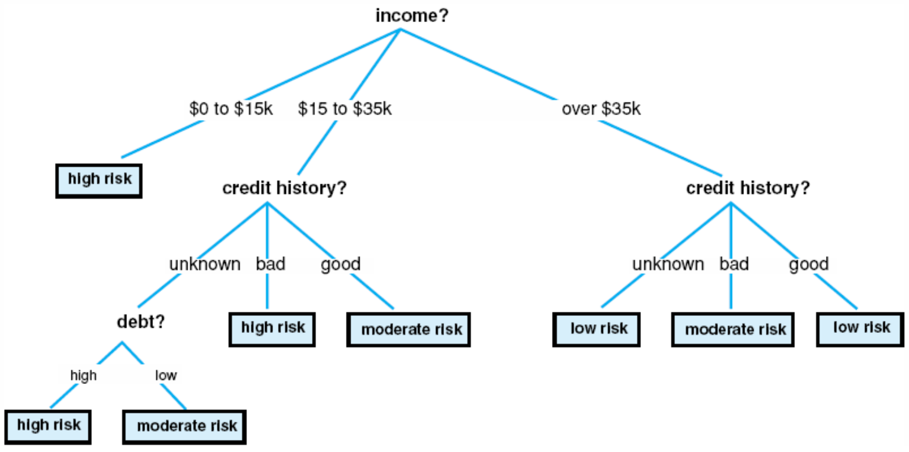


Character representation

- **ML problem to solve**
  - Recognize the numbers and characters in a training data set.
- **Many applications**
  - License plate recognition in police cars, information extraction from multimedia (images, videos), etc.

Classification: Classify Loan Applications

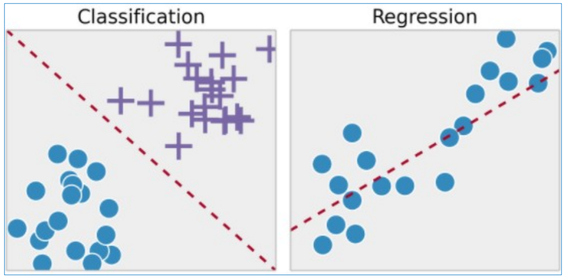
RISK	CREDIT HISTORY	DEBT	COLLATERAL	INCOME
high	bad	high	none	$0 to $15k
high	unknown	high	none	$15 to $35k
moderate	unknown	low	none	$15 to $35k
high	unknown	low	none	$0 to $15k
low	unknown	low	none	over $35k
low	unknown	low	adequate	over $35k
high	bad	low	none	$0 to $15k
moderate	bad	low	adequate	over $35k
low	good	low	none	over $35k
low	good	high	adequate	over $35k
high	good	high	none	$0 to $15k
moderate	good	high	none	$15 to $35k
low	good	high	none	over $35k
high	bad	high	none	$15 to $35k



High risk rule
IF (income = $0-$15k) **or** (income = $15k-$35k **and** credit history = unknown and debt = high) **or** (income = $15k-$35k **and** credit history = bad)
THEN **High risk**

Low risk rule
IF (income > $35k) **and** (credit history = unknown **or** good)
THEN **Low risk**

Supervised Learning Methods



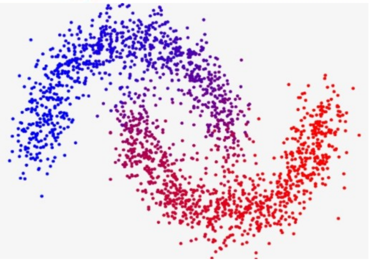
- **Regression**
  - The least-square method
  - Gradient descent
  - K-Nearest Neighbors (KNN), Artificial Neural Network (ANN), Naïve Bayes, Decision tree, Random forests, Support Vector Machines (SVM), Logistic regression, etc.

- **Classification**
  - Logistic regression
  - K-Nearest Neighbors (KNN)
  - Artificial Neural Network (ANN), Naïve Bayes, Decision tree, Random forests, Support Vector Machines (SVM), etc.

- **Both** regression and classification
  - **Most supervised learning methods** can be used for both **regression** and **classification** via small changes.

UNSUPERVISED LEARNING

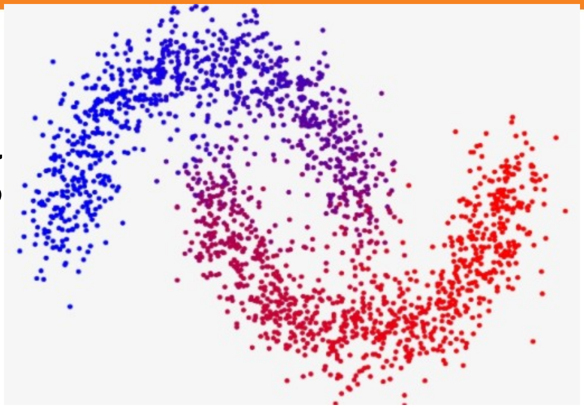
Unsupervised Lear

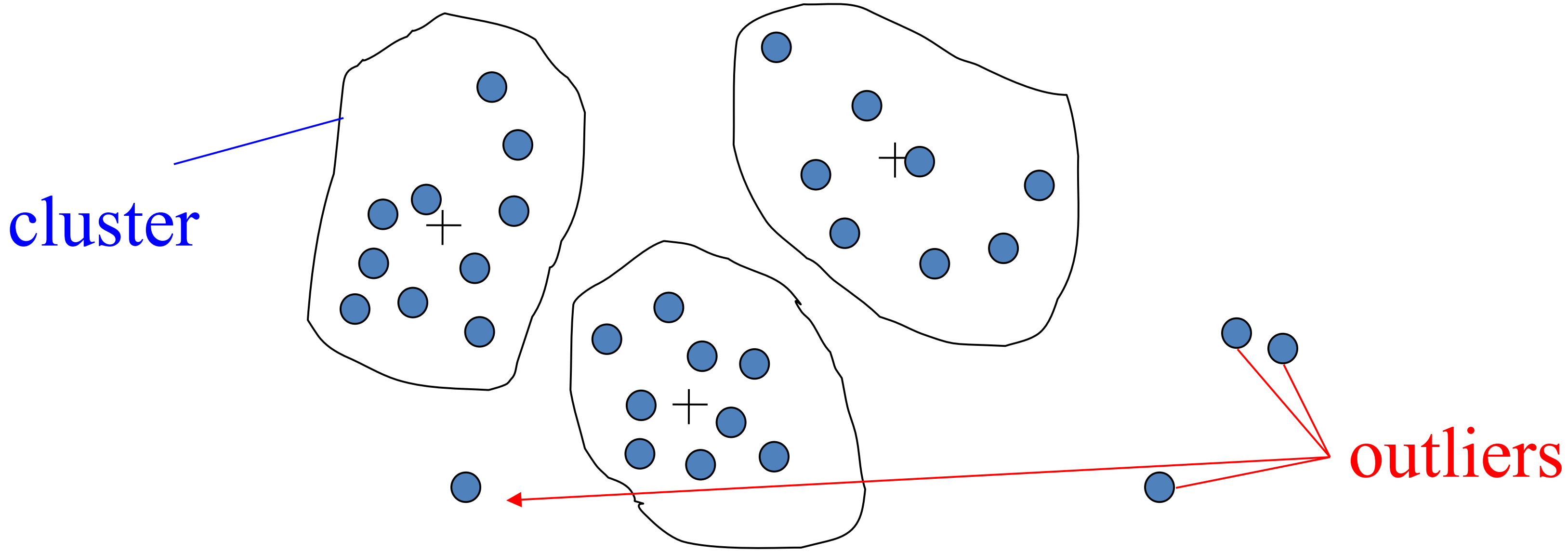


TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

- Given any data set $\{(x_1, x_2, \ldots, x_i) \mid i=1\ldots m\}$
  - No training data and training process required
  - Can use any form of data
    - However, if training data are available, better to use supervised learning.
  - Useful when no training data available
- Can find (outcomes of learning):
  - Clusters C with similar properties: Clustering
  - Associations C with high occurrences together: Association rule mining
  - Dependencies C with high correlations or variations: Feature selection
- Applications
  - Identifying fake news
  - Object detection, recommendation systems
  - Identifying fraudulent or criminal activities
  - Biological data analysis, etc.
  - Can also be used to create training data set

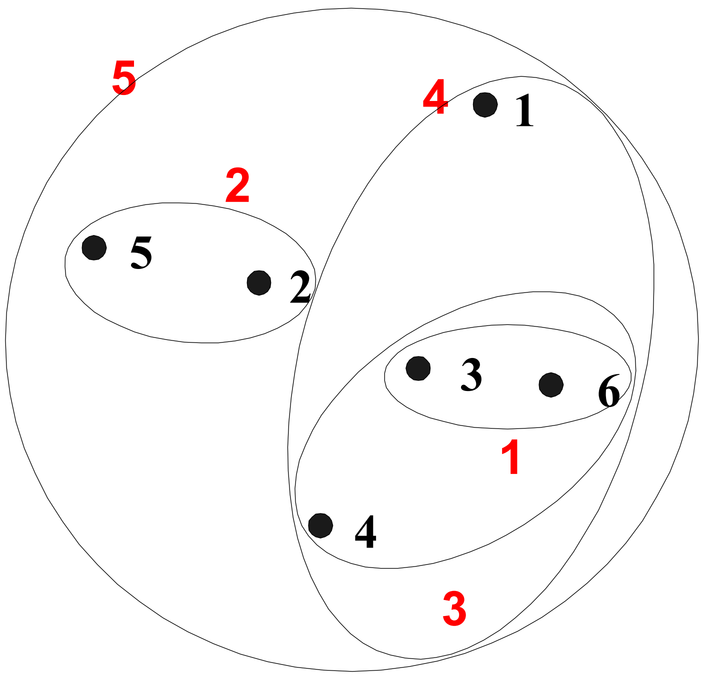
Partitioning Clustering



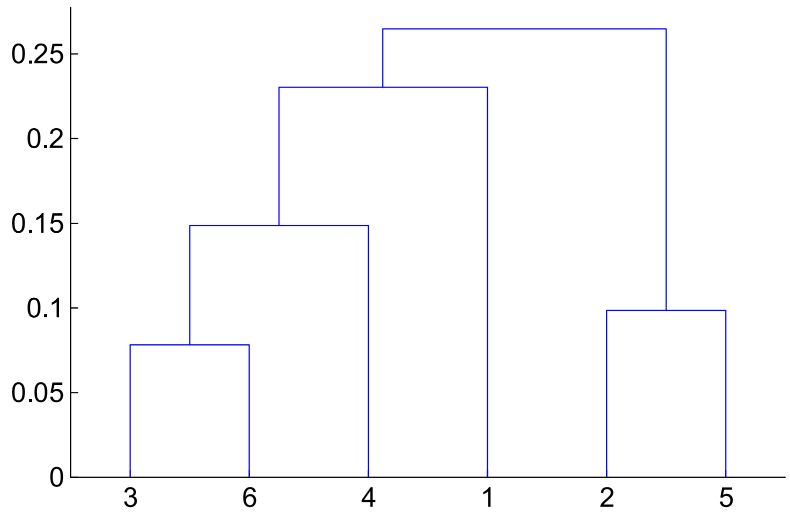


- In **some applications** we are interested in **discovering outliers**, not clusters (outlier analysis)

Hierarchical Clustering



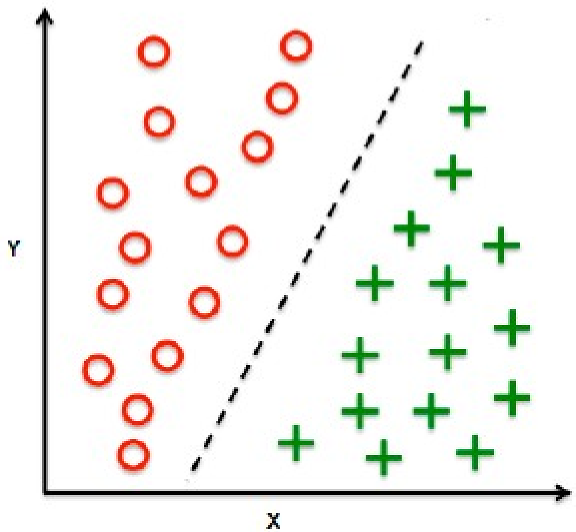
Nested Clusters



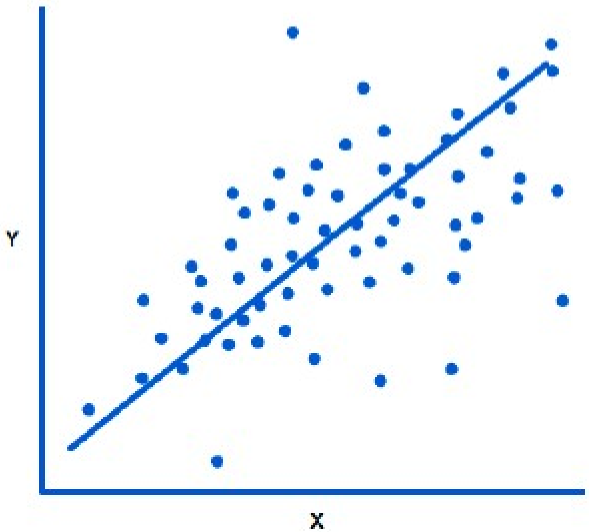
Dendrogram

Popularly used in biological data analysis

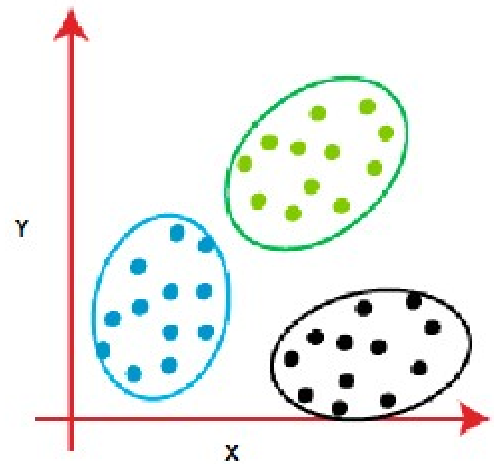
Classification, Regression, Clustering



Classification



Regression



Clustering

Association Rule Mining

- Given a set of transactions, **find association rules** that will predict the occurrence of an item based on the occurrence of other items in a transaction.
  - **Association rule**: X ☾ Y where X and Y are itemsets, e.g., {Bread, Milk} ☾ {Diaper}.

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Rule evaluation metrics (strength of cause-effect)

- Support (s): Fraction of transactions that contain $X \cup Y$, e.g., s = σ(Bread, Milk, Diaper)/|T| = 2/5 = 0.4
- Confidence (c): Measures how often items in Y appear in transactions that contain $X \cup Y$, e.g., c = σ(Bread, Milk, Diaper)/σ(Bread, Milk) = 2/3 = 0.67

Useful in **discovering interesting relations** between **variables** in **large databases**, for market basket analysis

Feature Selection

Temperature, T	Pressure, P	Composition			Thermal Conductivity, k × 10^3	Sound Velocity, v
		CH_4	C_2H_6	CO_2		
° C	psi	%	%	%	W/(m.K)	m/s
-20	600	94.00	6.00	0.00	27.36	350.86
-20	600	91.83	5.40	2.77	27.02	343.90
-20	600	89.66	4.80	5.54	26.68	336.48
-20	600	87.49	4.20	8.31	26.33	330.29
-20	600	85.32	3.60	11.08	25.99	324.43
-20	600	83.15	3.00	13.85	25.64	318.88
-20	600	80.98	2.40	16.62	25.30	313.60
-20	600	78.81	1.80	19.39	24.95	307.78
-20	600	76.64	1.20	22.16	24.61	303.00
-20	600	74.47	0.60	24.93	24.26	298.44
-20	600	72.3	0.00	27.7	23.92	294.08

Do I need all these features?

How about a data set with >100 attributes?

- Feature selection can be considered as *feature learning* or *representation learning*.

- **Why feature selection?**
  - Checking multicollinearity that impacts the model accuracy
  - Identification of important features (based on variations)
  - Finding hidden variables

Methods for Unsupervised Learning

- Clustering
  - Partitional clustering
  - Hierarchical clustering
  - Density-based clustering, model-based clustering

- Association rule mining
  - Apriori algorithm, FP growth algorithm

- Feature selection
  - Correlation/covariance matrix (to identify multicollinearity)
  - Information gain (the degree of uncertainty)
  - Fisher's linear discriminant analysis (LDA) (variance of the score)
  - Principal Component Analysis (PCA)
- Language modeling
  - Learning text data

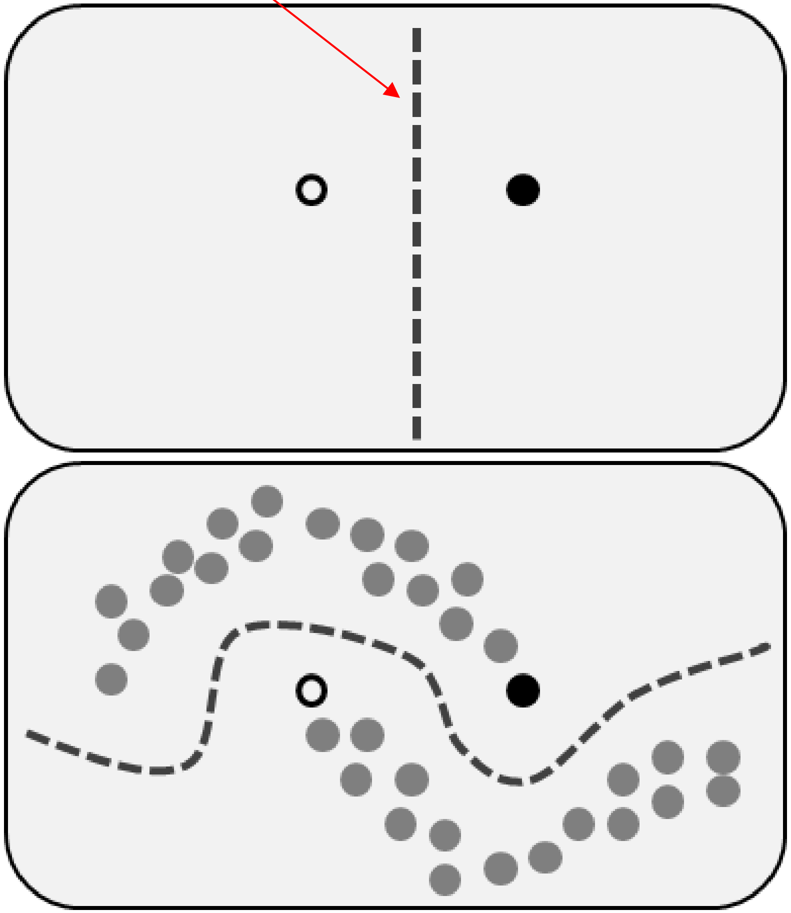
SEMI-SUPERVISED LEARNING

Semi-Supervised Learning (SSL)

- **Lack of training data in real-world**
  - Labeled data (training data) is hard to get.
    - Human annotation is time consuming, and labeling require experts. Expensive!
  - Unlabeled data (any data) is abundant.
    - Automatically measured data or databases. Cheap!
- **How to create a training data set with unlabeled data**
  - SSL makes use of a small amount of labeled data to create a training data set with a large amount of unlabeled data.
  - The idea is like a combination of supervised and unsupervised learning.

An Idea of Semi-Supervised Learning

A **classifier** for only two labeled examples

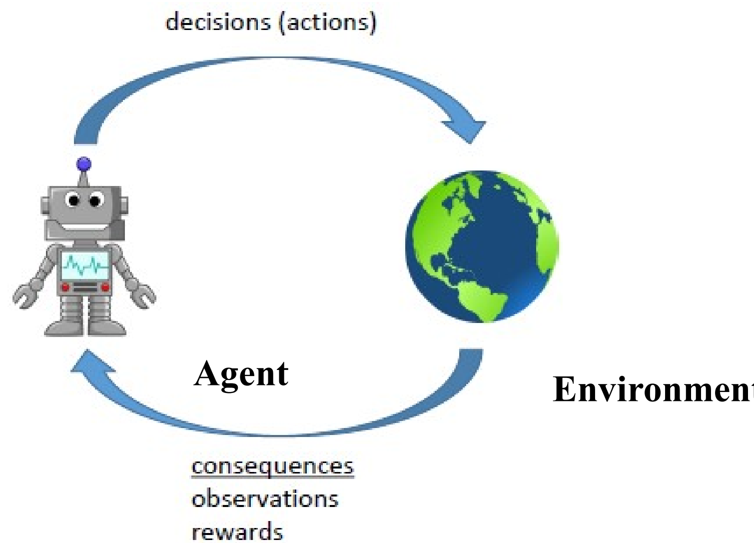


Better classifier for both two labeled examples and the unlabeled examples added later

- **Self-training** (one of many algorithms):
  1) A supervised learning algorithm is trained based on the labeled data only to have a classifier.
  2) The classifier is applied to the unlabeled data to generate labeled examples (like clustering).
  3) Feed those new labeled examples as training data to the supervised learning algorithm to learn a better model.
     - Generally, only the labels the classifier is most confident of are added at each step.

REINFORCEMENT LEARNING

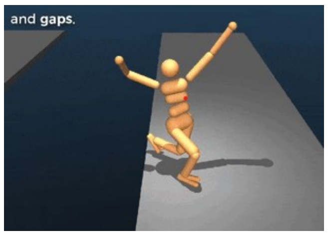
Reinforcement Learning (RL)

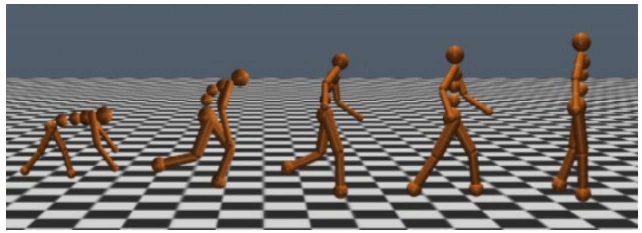


- **RL** has been studied in many disciplines such as game theory, control theory, decision theory, operation research, simulation-based optimization, etc.
- **RL** is a **goal-directed learning** through a **series of actions** that result in consequences as **rewards**.
  - The agent adapts by exploring a variety of actions while progressively favoring those actions that produce the most short-term reward (exploiting), ultimately aiming for the maximum long-term reward.
  - Training data can be used to build an initial model but NOT required.

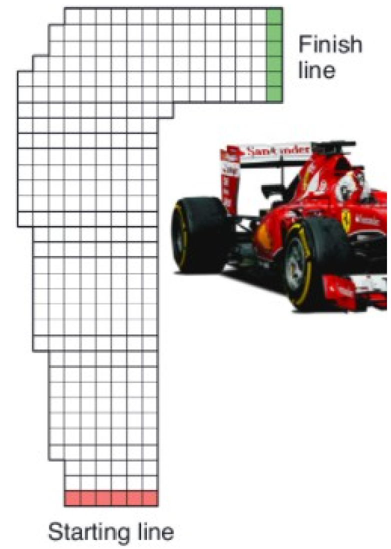
Reinforcement Learning Applications

Robot walking





Autonomous vehicles



- **Playing Go games**
  - AlphaGo beats the human Go champion, Se Dol Lee.
  - AlphaZero can even learn the game rules (self-taught) and beat both AlphaGo and the human champions

- **Other applications**
  - Control problems and robotics
  - Optimal decision making in business
  - Simulation-based optimization in scientific studies

Methods for Reinforcement Learning

- Model-based reinforcement learning
  - Dynamic programming to solve the Bellman optimality equations through
    - Policy iteration
    - Value iteration

- Model-free reinforcement learning
  - Monte-Carlo method
  - Temporal-difference
  - State-Action-Reward-State-Action (SARSA)
  - On-policy and off-policy methods

  - Q-Learning
    - Deep Q-learning Network (DQN)

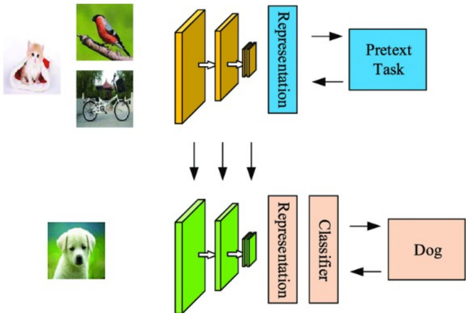
Is RL Supervised or Unsupervised?

- **RL is a supervised learning method.**
  - Supervised: A policy can be improved and optimized by reward experience.
- **RL is an unsupervised learning method.**
  - Unsupervised: Explicit goals are not given but forced to learn those optimal goals by trial and error.

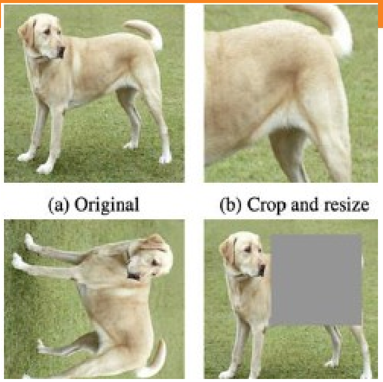
- **Therefore, RL** is **neither supervised nor unsupervised** but both.
  - The agent keeps on adapting by **exploring** a variety of actions while progressively **exploiting** (favoring) those actions producing the most reward.
  - The best action is taken by exploiting a reward through the improved policy.

Self-Supervised Learning

Self-Supervised Learning (SSL)



- **Idea**:
  - Use the labels that are naturally part of the data or use unsupervised learning ideas to generate supervisory signals rather than relying on labeled data.
- **SSL steps**
  - Generate supervisory signals from unlabeled data using methods such as unsupervised learning, representation learning, or autoencoding (encoding-decoding). This step is also called pretext task.
  - Use the labeled data for supervised learning
- **Examples**
  - Running text as training data
    - From a sentence, "The cat sat on the mat.", randomly mask a word "The cat [ ] on the mat.", then train the model to predict the masked word.
  - Computer vision
    - Image annotation and classification, object detection, semantic segmentation, etc.
  - Representation learning
    - Autoencoders (encoding-decoding)



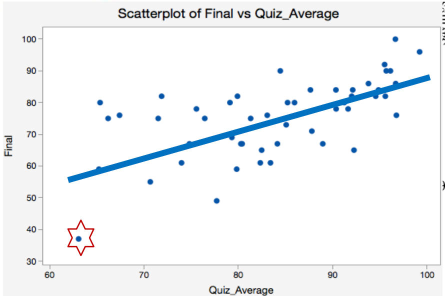
Self-Supervised Learning (SSL)

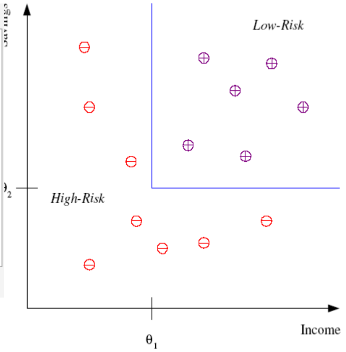
- **Capabilities of SSL**
  - Predict any part of the input from any other part
  - Predict the future from the past or the past from the present
  - Predict the obscure from the visible vice versa
  - Pretend there is a part of the input you don't know and predict that.
- **Connections with other ML approaches**
  - Supervised learning
  - Unsupervised learning
  - Semi-supervised learning
  - Reinforcement learning
- **Benefits**
  - Scalability
  - Understanding and mimicking how the human mind works.
    - Machine's capability of automatically generating labels without any humans.
- **Limitations**
  - Requires a lot of computational power
  - Low accuracy

Other Categories of Machine Learning

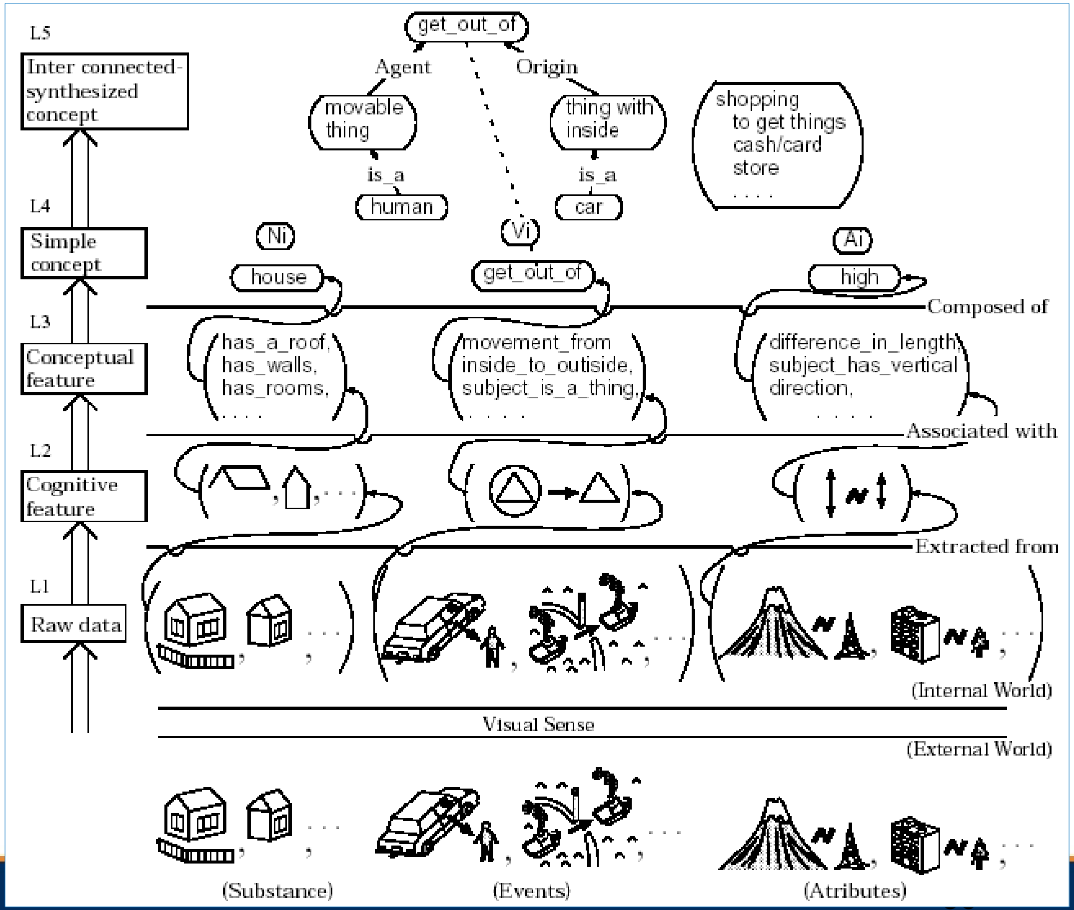
Shallow Learning

QuizAvg	Final
64	38
66	59
67	80
68	77
71	58
...	...





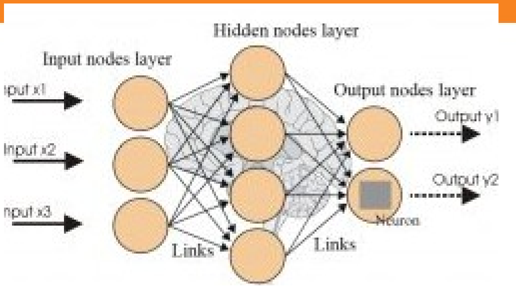
- **The ML methods** discussed so far do not learn any "deep" concepts.
  - Unable to find complex patterns that consist of many smaller patterns or patterns that may be hidden
  - How can a learning algorithm learn these **complex** or **hidden patterns**?



Deep Learning

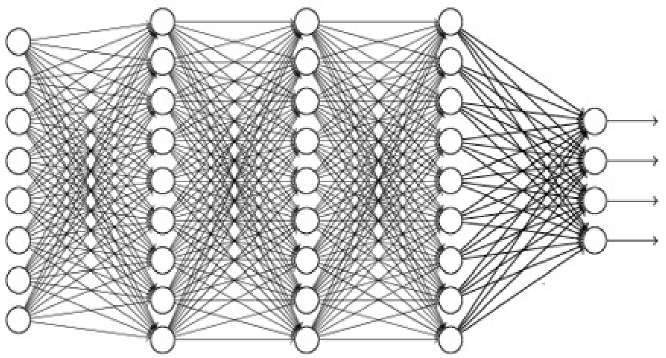
x_1	x_2	Output t
1.0	1.0	1
9.4	6.4	-1
2.5	2.1	1
8.0	7.7	-1
0.5	2.2	1
7.9	8.4	-1
7.0	7.0	-1
2.8	0.8	1
1.2	3.0	1
7.8	6.1	-1

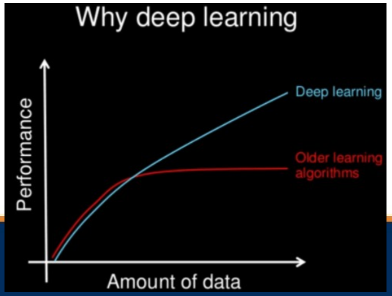
After 500 iterations with Perceptron with one neuron
$f(x) = -1.3x_1 - 1.1x_2 + 10.9$



How long will it take with multi-layers and neurons?

- **Deep learning**
  - Attempting to mimic the human brain with many layers of processing to extract features from data, optimize, and refine the model for accuracy.
    - Self-taught learning, unsupervised feature learning, hierarchical feature learning
    - Very large neural networks such as CNN
- **Computing requirements**
  - High-performance computer
  - A large amount of data for learning performance
- **Many applications**
  - Image recognition
  - Automated driving
  - Text generation
  - Medical research



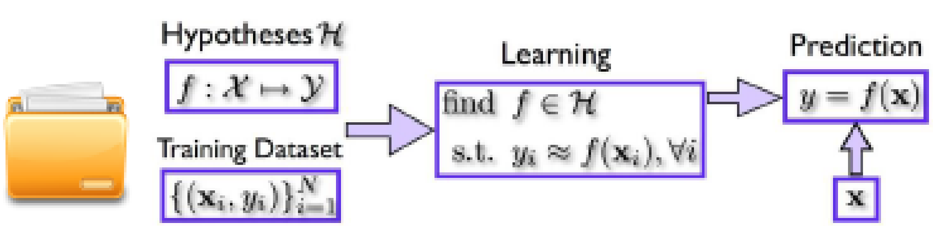


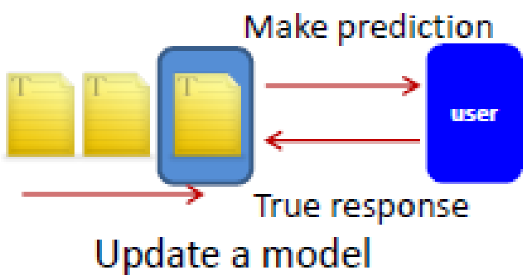
Online/Adaptive Learning

- **Updating prediction model** in **real-time** without a comprehensive training process for **new data** (critical for <u>big data analytics</u>)

- Batch/Offline learning
  - Observe a **batch** of training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$
  - Learn a model from them
  - Predict new samples accurately

- Online learning
  - Observe a **sequence** of data $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_t, y_t)$
  - Learn a model **incrementally** as instances come
  - Make the sequence of online predictions accurately





Transfer Learning

- **Basic idea**
  - A model trained on one task is adapted (transferred) to work on other related tasks
  - Transferring knowledge learned from a source to other learning tasks.
- Examples
  - Pretrained CNN models on large image data sets (e.g., VGG, ResNet)
  - Pretrained language models such as BERT, GPT

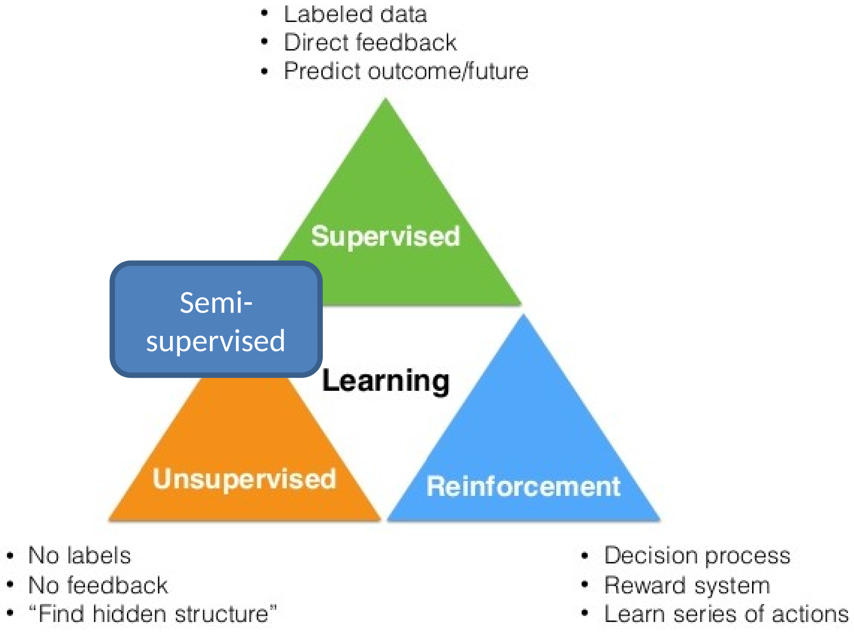
- **Applications**
  - Computer vision
    - Pretrained CNN models can be used as a starting point for specific image-related tasks such as object detection in images.
  - Natural Language Processing (NLP) tasks
    - Pretrained language models can be used for information extraction from text documents, language generation, and language translation

Ensemble Learning

- Basic idea
  - Use of multiple learning algorithms to obtain better predictive performance => deep learning
- Examples
  - Random forests (bootstrapping and learning from multiple decision trees)
  - Bayes optimal classifiers
- **Applications**
  - Computer security, e.g., malware detection, intrusion detection, etc.
  - Computer vision, emotion recognition, etc.
  - Fraud detection
- **Challenges**
  - Incompatibility of data and models between different algorithms due to the limitation of knowledge representation
  - Computing resources

Summary of Machine Learning

- Rote learning (Learning by memorization)
- Supervised learning
  - Learning by training data through training, also called inductive learning (generalization)
  - **Two types of problems**: classification and regression
- Unsupervised learning
  - **Clustering** or finding associations in any data
- Semi-supervised learning
  - Mixing supervised and unsupervised learning
- Reinforcement learning
  - Learning rules or tasks by trial-and-error
- **Other categories**
  - Deep learning
  - Online or adaptive learning
  - Transfer learning
  - Ensemble learning



Choosing the Right ML Methods

- **Availability of training data**
  - Supervised, unsupervised, or semi-supervised
- **Value prediction, classification, clustering, finding associations**
  - Regression, classification, clustering, association rule mining
- **Finding the best policy for actions for optimal decision making**
  - Reinforcement learning
- **Available computing power**
  - Deep learning
- **Big data**
  - Online (real-time) or batch learning
- **Choosing ML algorithm(s)**
  - Accuracy
  - Training and prediction speed
  - Resilience to noise in data
  - Complexity of model (knowledge representation)
  - Interpretability of results

Python Packages for Machine Learning

- **Numpy**: Fast numerical data processing and computation (>10x faster than Python)
  - https://numpy.org
- **Pandas**: Data loading, structuring, cleaning, searching, processing
  - https://pandas.pydata.org, https://www.w3schools.com/python/pandas/default.asp
- **Statistics**: Statistics libraries
  - https://docs.python.org/3/library/statistics.html
  - https://scipy-lectures.org/packages/statistics/index.html
- **Scipy**: Scientific computation
  - https://docs.scipy.org/doc/scipy/getting_started.html
- **Matplotlib**: Plotting libraries
  - https://matplotlib.org/stable/tutorials/introductory/pyplot.html
- **Scikit-learn**: ML packages using numpy, pandas, scipy, matplotlib without ANN
  - https://scikit-learn.org
- **PyTorch**: ANN libraries including deep learning and natural language processing
  - https://pytorch.org/tutorials/
- **Keras**: Deep learning framework with APIs built on top of TensorFlow
  - https://keras.io/
- **Anaconda**: Python distribution including most packages
  - https://www.anaconda.com/

Other Machine Learning Packages

- MATLAB
- R, other statistics tools such as SPSS, SAS
- Microsoft ML.NET
- Java libraries
  - Weka, Mahout, MOA

- Cloud computing service providers
  - AWS
  - Azure
  - Google

Relevant Disciplines to ML

- **Computer Science**
  - Data structures, algorithms, databases, data engineering, high-performance computing, computational complexity theory
  - Artificial Intelligence (AI)
- **Mathematics**
  - Statistics and probability (descriptive statistics, sampling, hypothesis testing, regression analysis, probability distributions, conditional probability, Bayes theorem, etc.)
  - Linear algebra (vectors and matrix, eigenvalues and eigenvectors, matrix factorization, orthogonality, principal component analysis, etc.)
  - Calculus (differential and integral calculus, partial derivatives, finding maxima and minima of a function, sigmoid and logit functions, etc.)
  - Numerical analysis (computational approach to optimization problems)
- **Information Theory** (information, entropy)
- **Decision Science, Control Theory** (reinforcement learning, game theory)
- **Psychology, Neurobiology** (cognitive process of learning)
- **Philosophy** (logics and reasoning)
- …

Current Issues in Machine Learning

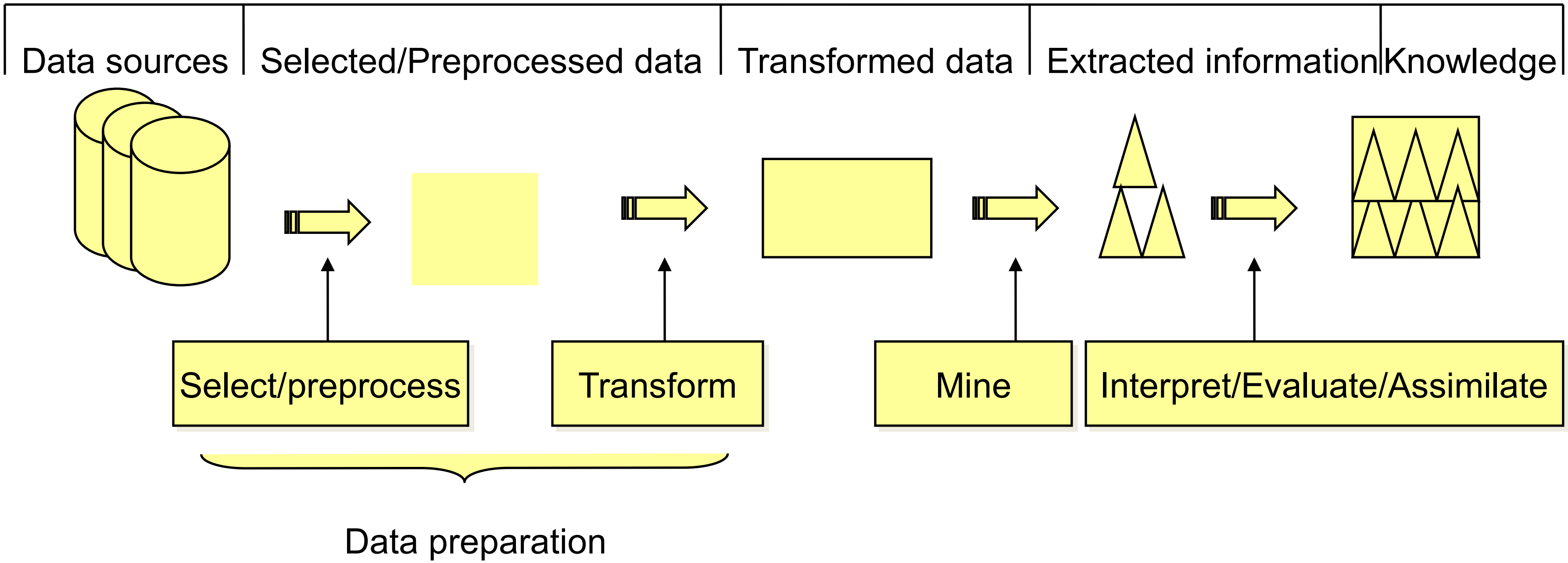
- How can we **model** applications as machine learning problems?
- What **algorithms** can be used?
- How does **the size of training data** influence accuracy?
- How does **noise data** influence accuracy?
- How can we gauge the **accuracy of a model** (hypothesis) on unseen data?
- How does **complexity of model representation** impact it?
  - Occam's razor: The simplest answer is usually correct.
- How can **prior knowledge** of leaner help?
- What are the **theoretical limits** of learnability?
- How can systems **alter their own representations**?
- What clues can we get from **biological learning systems**?

DATA SCIENCE AND BIG DATA ANALYTICS

Data Science

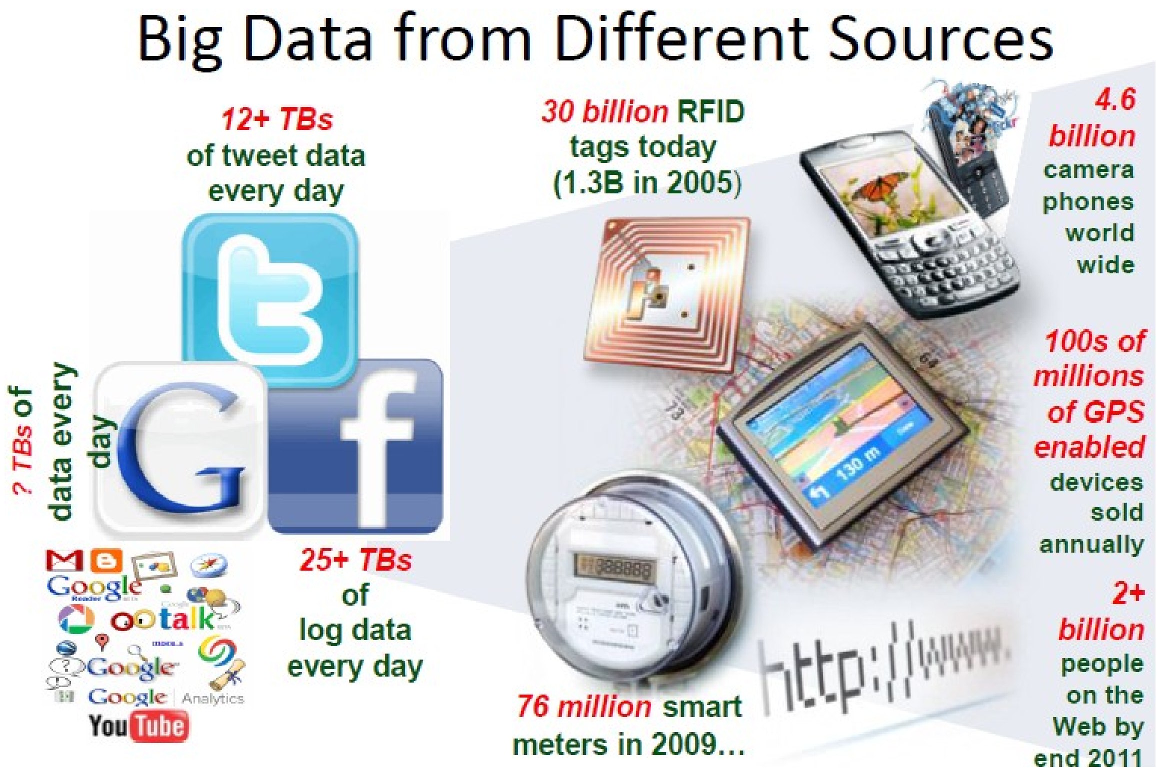
- **What is data science?**
  - An interdisciplinary field that uses statistics, probability theories, machine learning, databases, and computing to acquire knowledge and insights from data (Wikipedia)
    - Used to be called statistical data analysis or data mining
- **Common tasks required in data science**
  - Data collection, extraction, integration, preprocessing (data engineering)
  - Visualization
  - Predictive modeling and mining
  - Interpretation and reasoning
  - Communication with stakeholders
- **Required skills for data scientists**
  - Data engineering (database and computational) – majority effort
  - Analytics (analysis, mathematical and logical reasoning)
  - Communication

Data Mining and Big Data Analytics



A **non-trivial process** of identifying valid, novel, potentially useful, and understandable patterns in **typically large data**, and ultimately understanding the data => beyond the traditional statistical analysis

Challenges of Big Data Analytics



- Voluminous data
- A complex, noisy, heterogeneous, longitudinal data changing over time

- **Major issues**
  - Capturing, storing, searching, sharing, (real-time) processing, visualizing, and analyzing

Beyond the scope of traditional statistical analysis!

Technologies for Big Data Analytics

- Databases for data storage and indexing
  - Relational, NoSQL, Graph
- Statistical and machine learning methods for analysis
  - Statistical methods
  - Machine learning methods
    - Supervised learning and unsupervised learning
    - Deep learning and ensemble learning
    - Online or adaptive learning
- High-performance computing for processing, visualizing, and analyzing
  - Hardware
    - High performance computing using parallel processors and GPUs
  - Software algorithms and frameworks
    - Map-and-Reduce , Hadoop, Apache Spark, etc. for computation
    - JavaScript-based visualization frameworks

ML-related roles in the industry

ROLE	RESPONSIBILITIES
Data Engineer	Makes the appropiate data available for data scientists; focuses on data integration, modelling, optimization, quality and self-service.
Data Scientist	Identifies use cases, determines appropriate datasets and algorithms, experiments and builds AI models.
AI Architect	Is the glue between data scientists, data engineers, developers, operations (DevOps, DatOps, MLOps) and business unit leaders to govern and scale the AI initiatives.
ML Engineer	Deploys AI models through effective scaling and ensuring production readiness, ensures continuous feedback loop.

See Canvas page

SYLLABUS

Acknowledgement

- Slides from Dr. Christopher Ryu
- Many slides borrowed from machine learning classes @Stanford, Cornell, Pen State, UIUC, U of Toronto, UC-Berkeley, UCLA, NYU

References

- Tom Mitchell, Machine learning, McGraw-Hill, 2017.
- Simon Rogers and Mark Girolami, A First Course in Machine Learning, 2nd edition, CRC press, 2017.
- Kevin Murphy, Machine learning: A Probabilistic Perspective, MIT press, 2012.
- C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
- R.S. Sutton and A.G. Barto, Reinforcement Learning: An Introduction, 2nd edition, The MIT press, 2017.
- Simon J.D. Prince, Understanding Deep Learning, MIT press, 2024.
- S. Haykin, Neural Networks – A Comprehensive Foundation, Pearson-Prentice Hall, 1999.

Automatic Identification of Use of Public Transportation from Mobile Sensor Data

Mohammadreza Hajy Heydary
Pritesh Pimpale
Anand Panangadan

Introduction

- GPS has been vastly used by people mainly to find their current location and to navigate to a destination.
  - E.g., Google Map, Waze
- People use their smartphones more and more and they all have GPS
- GPS data can be used for more than just locating a place
- The data can be used to determine the type of activity one is doing
  - Walking
  - Running
  - Driving
- However
  - There is not yet a way to determine automatically if a person is using public transportation

Applications

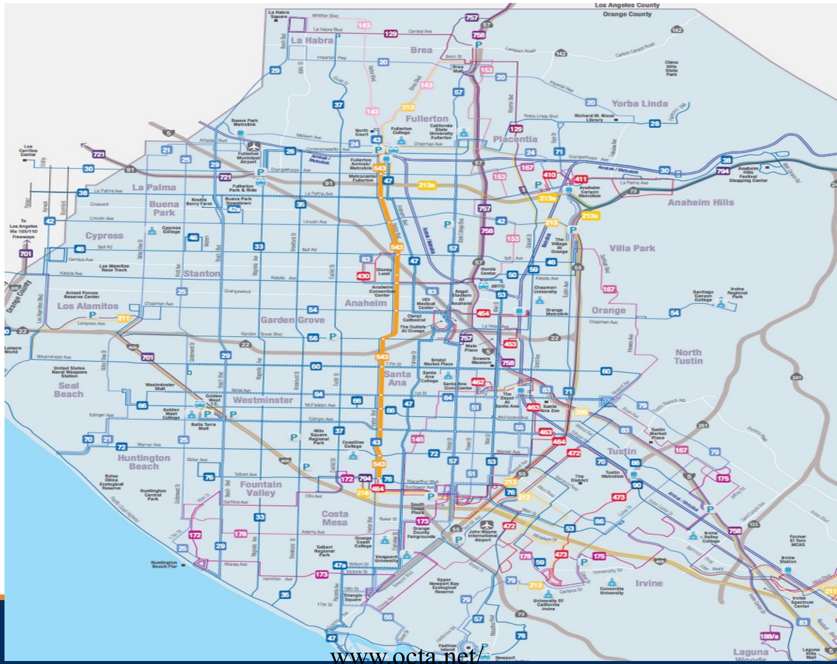
- Location-based recommendations
  - Yelp, Google Maps, …
- Do not consider mode-of-transit; based only on current location
- Transit-based recommendations
  - Recommend locations based on the ease of reaching it using public transportation
- Encourage public transportation
  - More sustainable than personal transportation
  - Rewards (coupons, discounts) when using the bus

Project Goal

- Develop a method to automatically detect when one is using public transportation
- This information can then be used to promote public transportation (route-specific product/shopping recommendations/coupons)

Challenges:

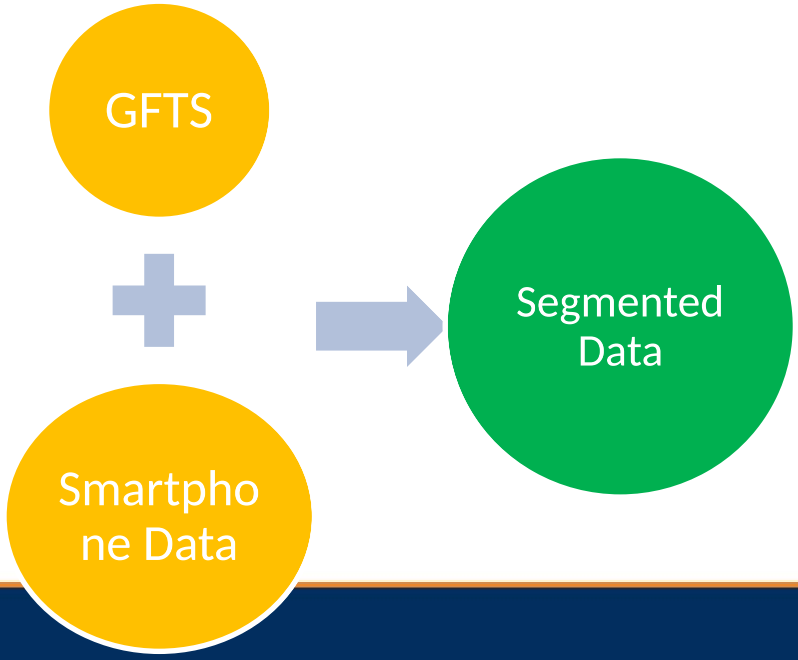
- Distinguishing between being in a car vs in a bus
- There are many bus routes in a small area



www.octa.net/

Requirements

- Input
  - An automatically recorded GPS track
  - A set of files in GTFS format
- Output
  - Segments of the user's path, each representing time on a bus or another motor vehicle
    - A file for the points belonging to a bus route



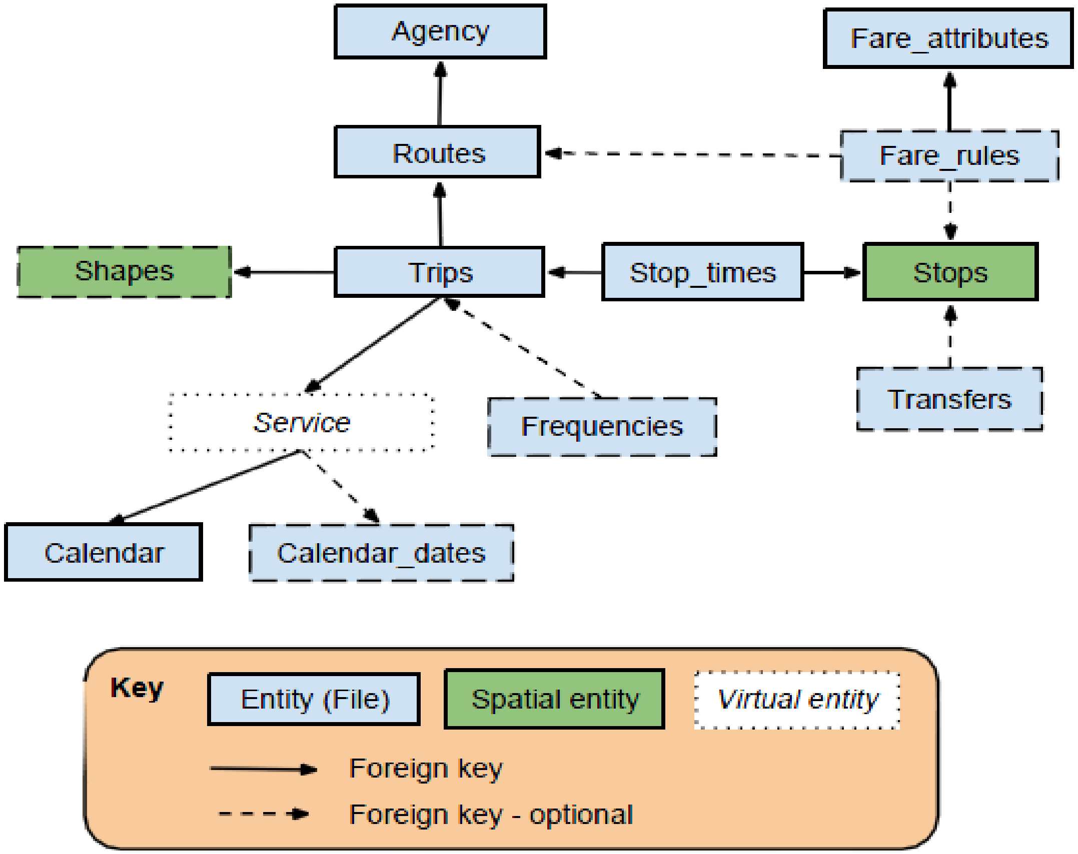
What is GTFS?

- *General Transit Feed Specification*
- A common format for public transportation schedules and associated geographic information
- Used by many public transit agencies
  - OCTA
  - LACMTA
  - CTA
- US DOT advocates broad GTFS contribution a registry of GTFS
- Open-source
- World Bank has also heavily advocated and assisted international adoption of GTFS

addtransit.com

More about GTFS

- GTFS contains multiple CSV files
- Limited to scheduled data



www.transitwiki.org

Analysis and Design

- Modeling GTFS files
- Pre-processing smartphone data
- Matching the closest route
- Matching the closest time
  – Check the weekly schedule
  – Check the time table for a given date

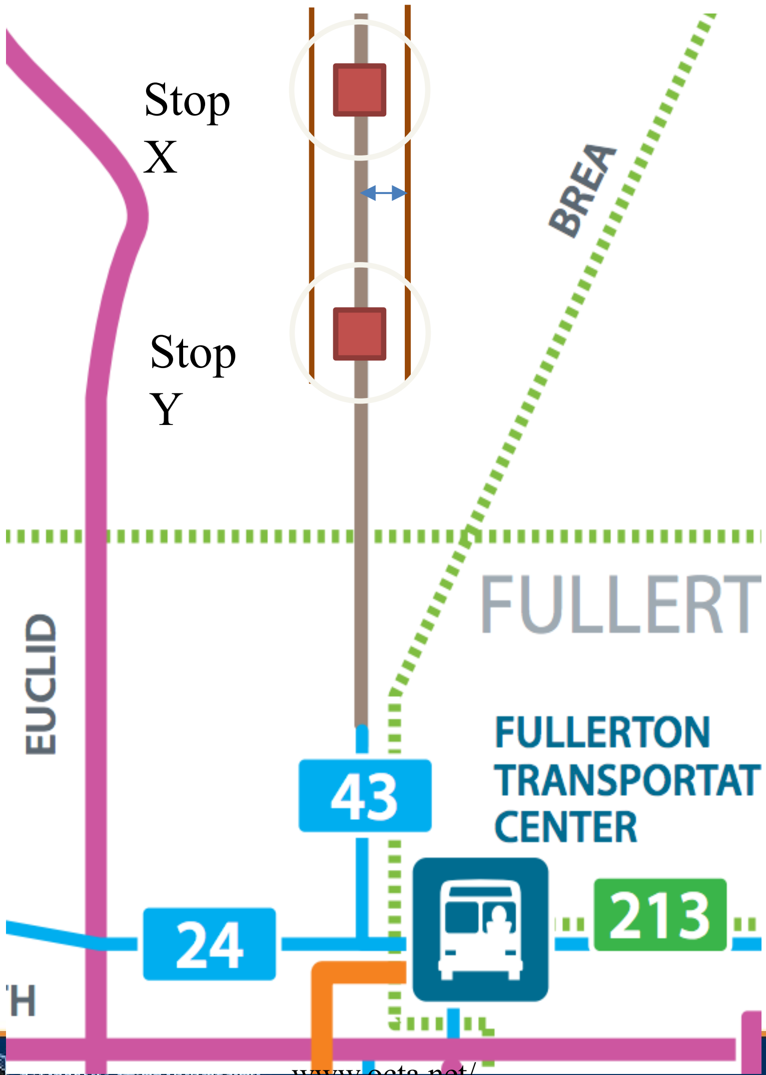
Is there a bus route? → check the weekly schedule → check the time table

Analysis and Design

- Multiple constants must be optimized
  - Pre-processing constants
    - Maximum time difference between points
      - Substitute average instead of points that are *t* seconds apart
  - Main program constants
    - Maximum distance from a route
    - Maximum distance from a bus stop
    - Time difference between GPS-time and bus schedule

www.zentut.com

Optimized Constants Values



www.octa.net/

A. Distance from a route

- 30m, 60m, 90m

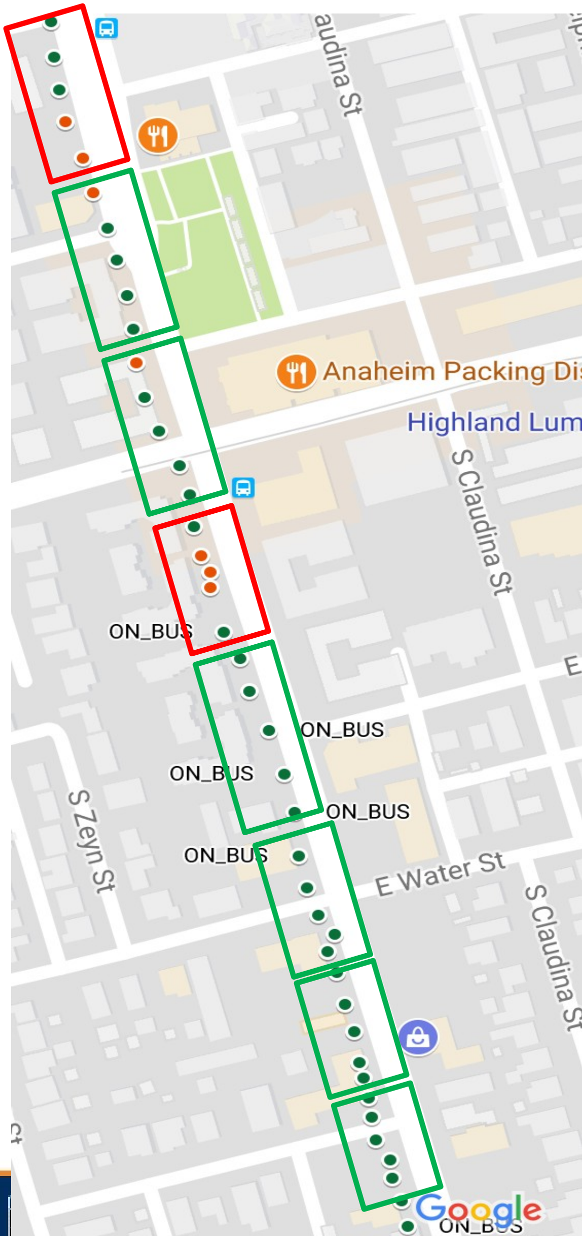
➢ 60m

B. Distance from a bus stop

- 300m, 600m, 900m

➢ 900m

C. Time



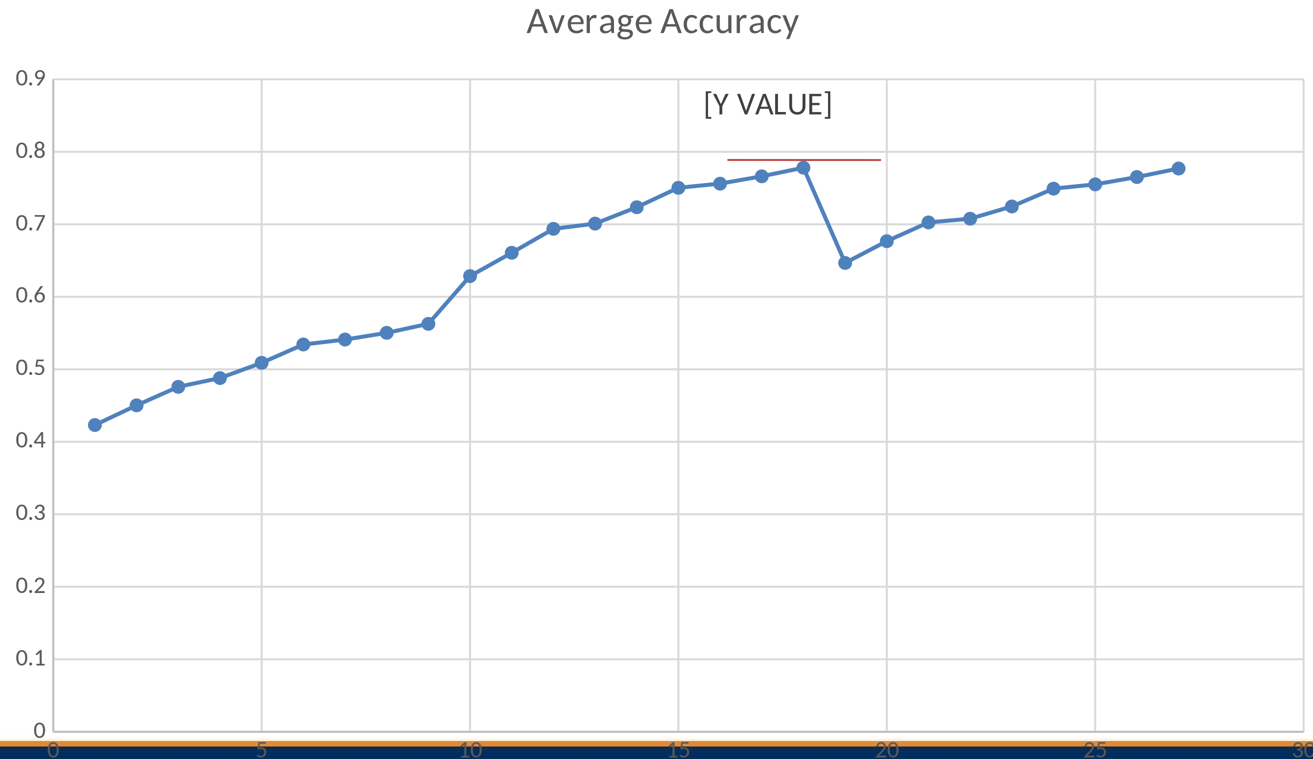
Accuracy

- Classification results are aggregated
  - Aggregation window: 5
- Ratio is defined as

- If the ratio 0.75
  - ➢ Label as on bus
- Otherwise,
  - ➢ Label as not on bus

Results

- 14 different trips are included (March 23, 2017)
  - More than 20,000 GPS points
- 27 different combinations of constants are tested



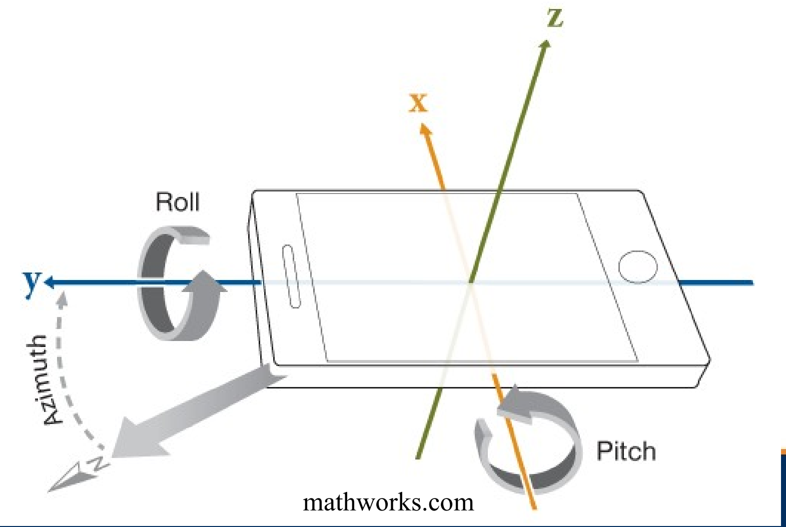
Observations and Problems

- GPS sensor in not power efficient
- GPS data is sometimes inaccurate
  – Dense Urban areas with tall buildings
    - Downtown
- GTFS files give a pre-scheduled timetable
  – Unexpected changes
  – Nonstandard traffic

plslogistics.com

What is next?

- We want better results!
- Smartphones have other sensors
  - Power efficient
  - Accurate
- Typical sensors on a smartphone:
  - Accelerometer
  - Gyroscope
  - Magnetometer
  - Wi-Fi



Machine Learning

- Hypothesis:
  - Private vehicles and buses will have subtle differences in the patterns of accelerations induced on the rider
- Challenges:
  - Similar tracks of cars and buses
  - Overlapping ranges of speed
  - Extracting location and smartphone orientation invariant features
- Contributions:
  - Defining a set of location and orientation invariant features
  - Use machine learning to classify streams of accelerometer and gyroscope data into instances of public transportation use.

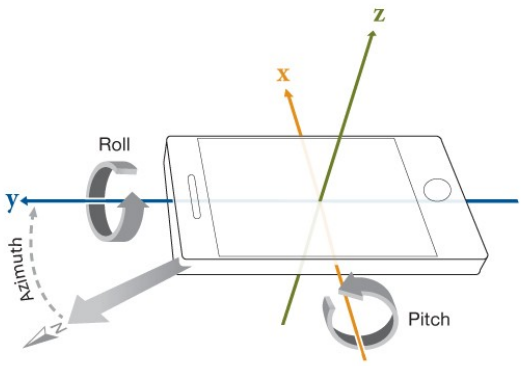


Figure from: mathworks.co

Data Collection

- Modern smartphones are equipped with a rich collection of sensors:
    1. Accelerometer: An accelerometer sensor reports the acceleration of the device along the 3 sensor axes
        - both the physical acceleration and the gravity

Accelerometer Data for Device : 23f92f65b4b15699			
Time	accX	accY	accZ
1501008420773	-1.647773504257200	5.721702098846430	7.963440418243400
1501008420785	-1.623823285102840	5.769602298736570	7.776628971099850
1501008420820	-1.657353639602660	5.781577587127680	7.623347282409660
1501008420833	-1.846560120582580	5.779182434082030	7.575447082519530
1501008420845	-1.961521029472350	5.640271186828610	7.565866947174070
1501008420855	-1.939965844154350	5.510940074920650	7.561077117919920
1501008420867	-1.963916063308710	5.534890174865720	7.683222770690910
1501008420878	-1.868115305900570	5.477409839630120	7.800579071044920

Data Collection

- Modern smartphones are equipped with a rich collection of sensors:

    2. Gyroscope: A gyroscope sensor reports the rate of rotation of the device around the 3 sensor axes.

Gyroscope Data for Device : 23f92f65b4b15699			
Time	X	Y	Z
1501008420795	-0.087618865072727	-0.243275016546249	-0.068806089460850
1501008420803	-0.058324091136456	-0.220904469490051	-0.070403985679150
1501008420828	-0.063650414347649	-0.219306573271751	-0.067208193242550
1501008420837	-0.080162011086941	-0.202794969081878	-0.060283970087767
1501008420850	-0.092412553727627	-0.194805487990379	-0.046968165785074
1501008420861	-0.113185212016105	-0.212914988398551	-0.039511311799288
1501008420872	-0.143545240163803	-0.239546597003936	-0.038446050137281
1501008420882	-0.166448429226875	-0.264047682285308	-0.027793403714895

Data Collection

- Modern smartphones are equipped with a rich collection of sensors:

  3. Gravity Sensor: This is a composite sensor measurement using data from both the accelerometer and gyroscope sensors.

Gravity Data for Device : 23f92f65b4b15699			
Time	X	Y	Z
1501008421734	-0.946596086025238	5.188220024108880	8.267811775207520
1501008421837	-0.846776545047760	5.119062900543210	8.321571350097650
1501008421944	-0.767900764942169	5.073137283325190	8.357272148132320
1501008422048	-0.783206760883331	5.073700904846190	8.355508804321280
1501008422154	-0.854259312152862	5.097428321838370	8.334077835083000
1501008422295	-0.922384202480316	5.126232624053950	8.309111595153800
1501008422459	-1.001546740531920	5.185049533843990	8.263325691223140
1501008422679	-1.064378976821890	5.196239948272700	8.248428344726560
1501008422859	-1.306375026702880	5.217032909393310	8.200386047363280

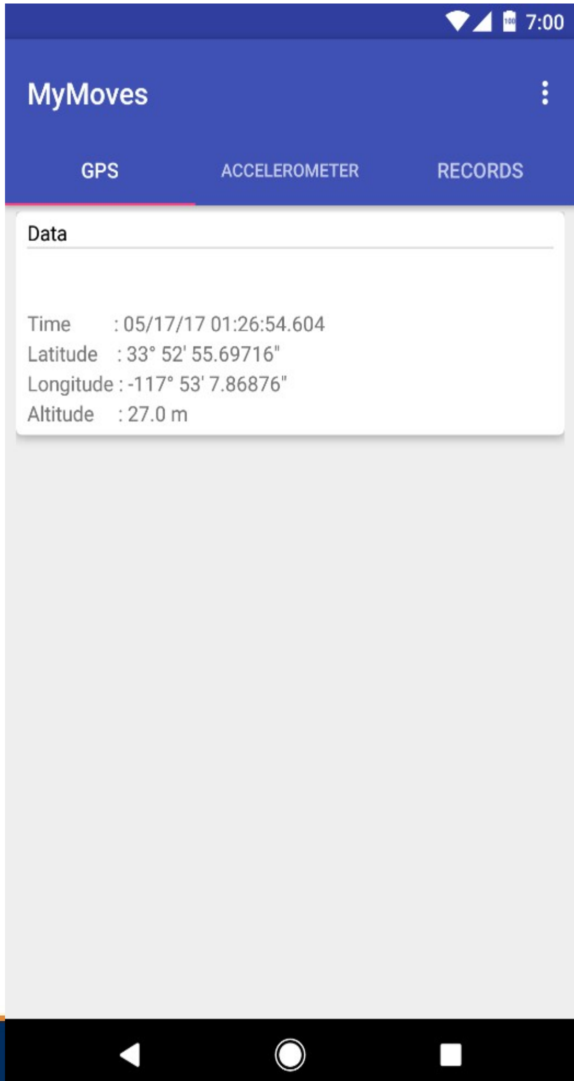
Data Collection

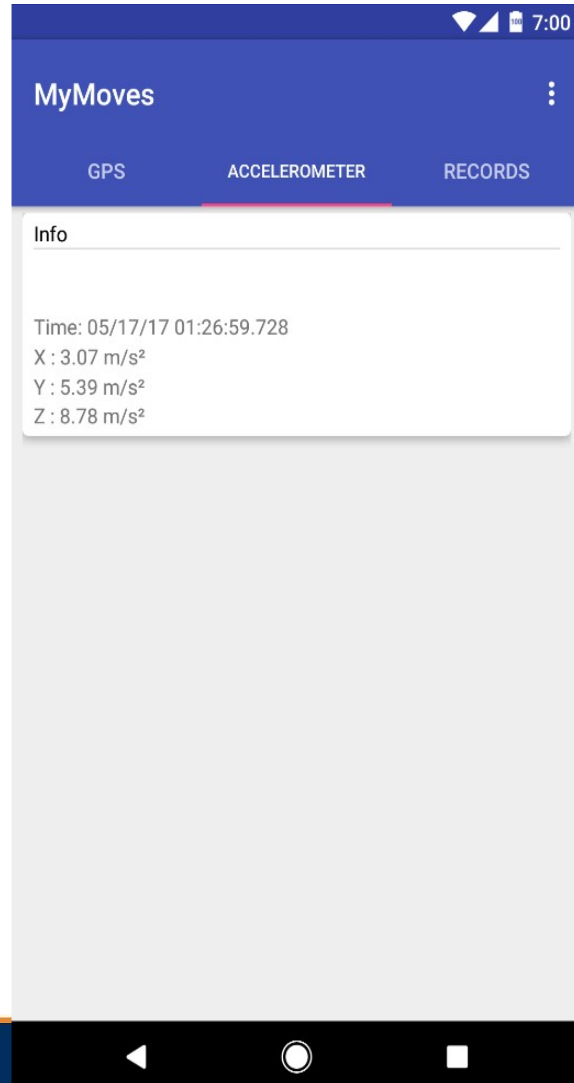
- Native API's on an Android device:
  - Location Service API:
    - Uses Assisted GPS and Wi-Fi localization
    - Provides location in Latitude, Longitude and Altitude values
  - Activity Recognition API
    - Uses the low power sensors of including the accelerometer, gyroscope, and magnetometer
    - In vehicle, on bicycle, on foot, running, still, tilting, walking, unknown

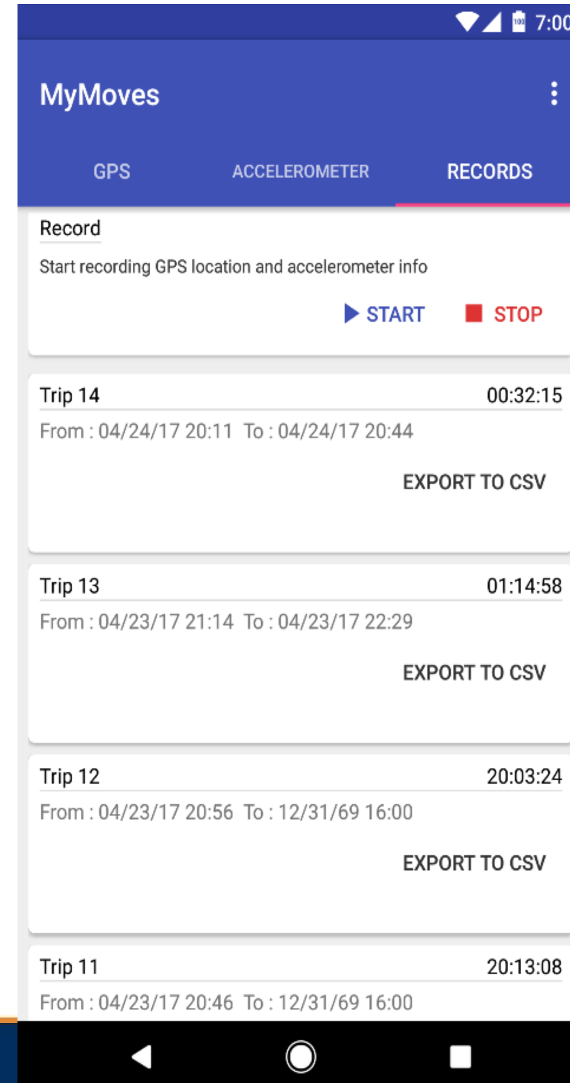
Activity Recognition Data for Device : 23f92f65b4b15699

Time	IN_VEHICLE	ON_BICYCLE	ON_FOOT	WALKING	RUNNING	STILL	TILTING	UNKNOWN	PROBABLE
1501008420979	75	5	10	5	5	5	0	5	IN_VEHICLE
1501008422575	75	5	10	5	5	5	0	5	IN_VEHICLE
1501008422806	75	5	10	5	5	5	0	5	IN_VEHICLE
1501008437558	5	5	10	5	5	5	0	75	UNKNOWN
1501008439365	5	5	10	5	5	5	0	75	UNKNOWN

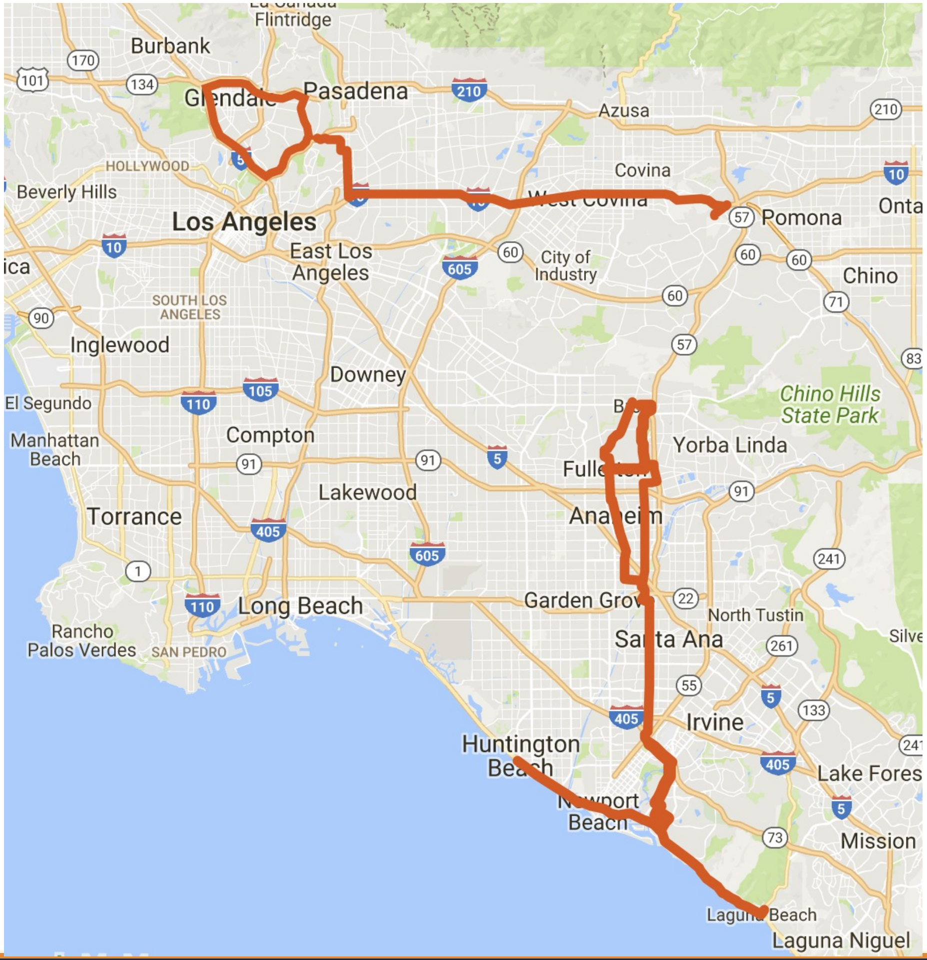
Data Collection



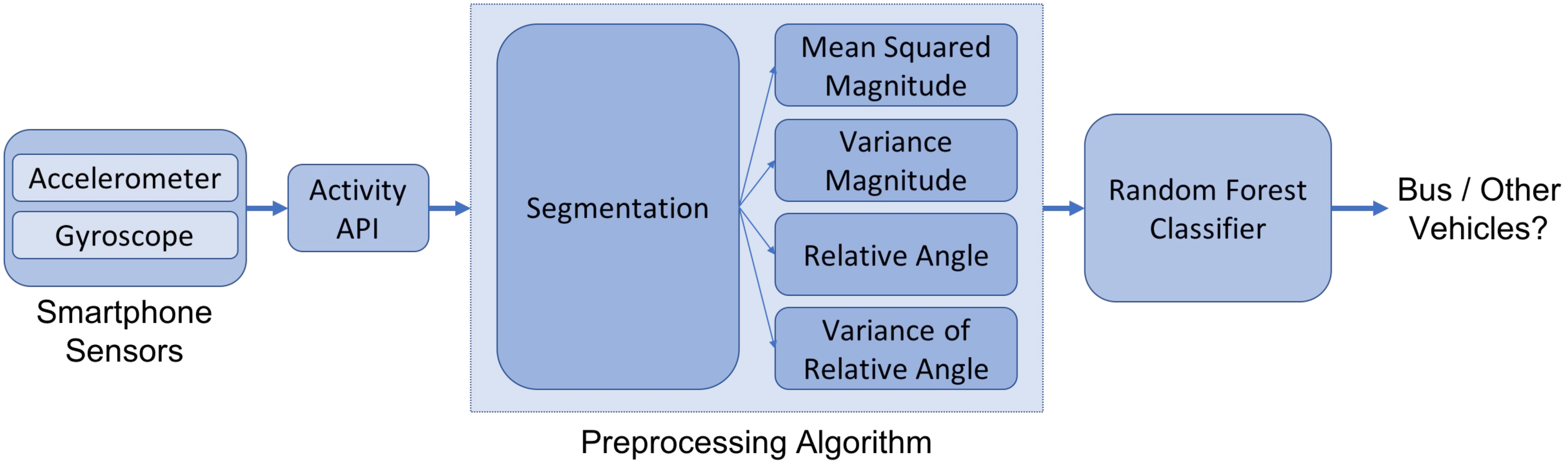




- Map of Southern California showing the locations where users traveled as sensor data was collected.

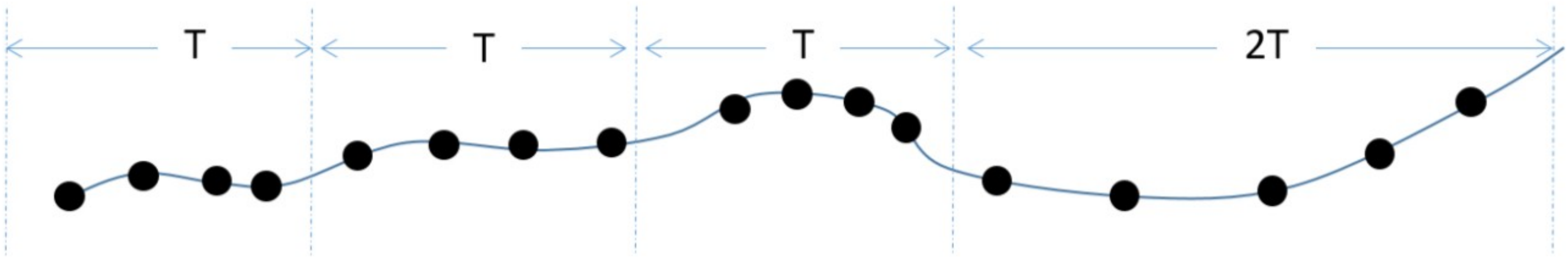


Approach


Accelerometer
Gyroscope
Smartphone Sensors
Activity API
Segmentation
Mean Squared Magnitude
Variance Magnitude
Relative Angle
Variance of Relative Angle
Preprocessing Algorithm
Random Forest Classifier
Bus / Other Vehicles?

Approach

- Segmentation
  - Intervals each at least T milliseconds long.
  - If the number of data points is less than 4
    - Time interval is dynamically increased to 2×T .



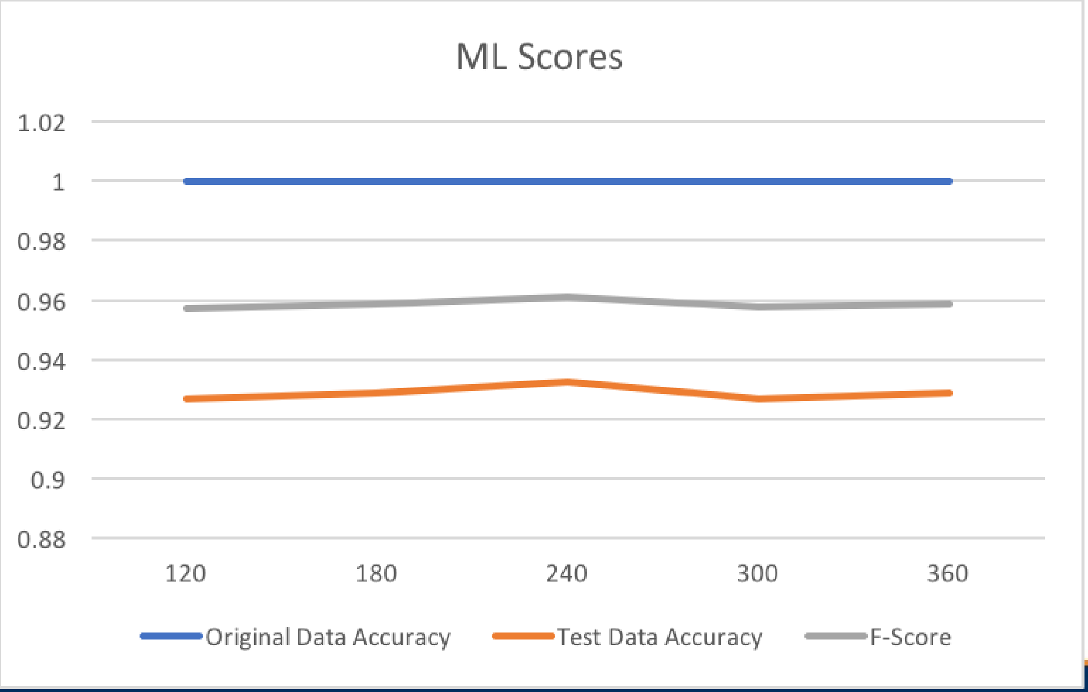
Approach

- Feature Extraction
  - mean squared magnitude of acceleration
  - mean squared magnitude of rate of rotation
  - variance of magnitude of acceleration
  - variance of magnitude of rate of rotation
  - relative angle
  - variance of relative angle

- Classification
  - Random Forests Classifier
  - Scikit-learn

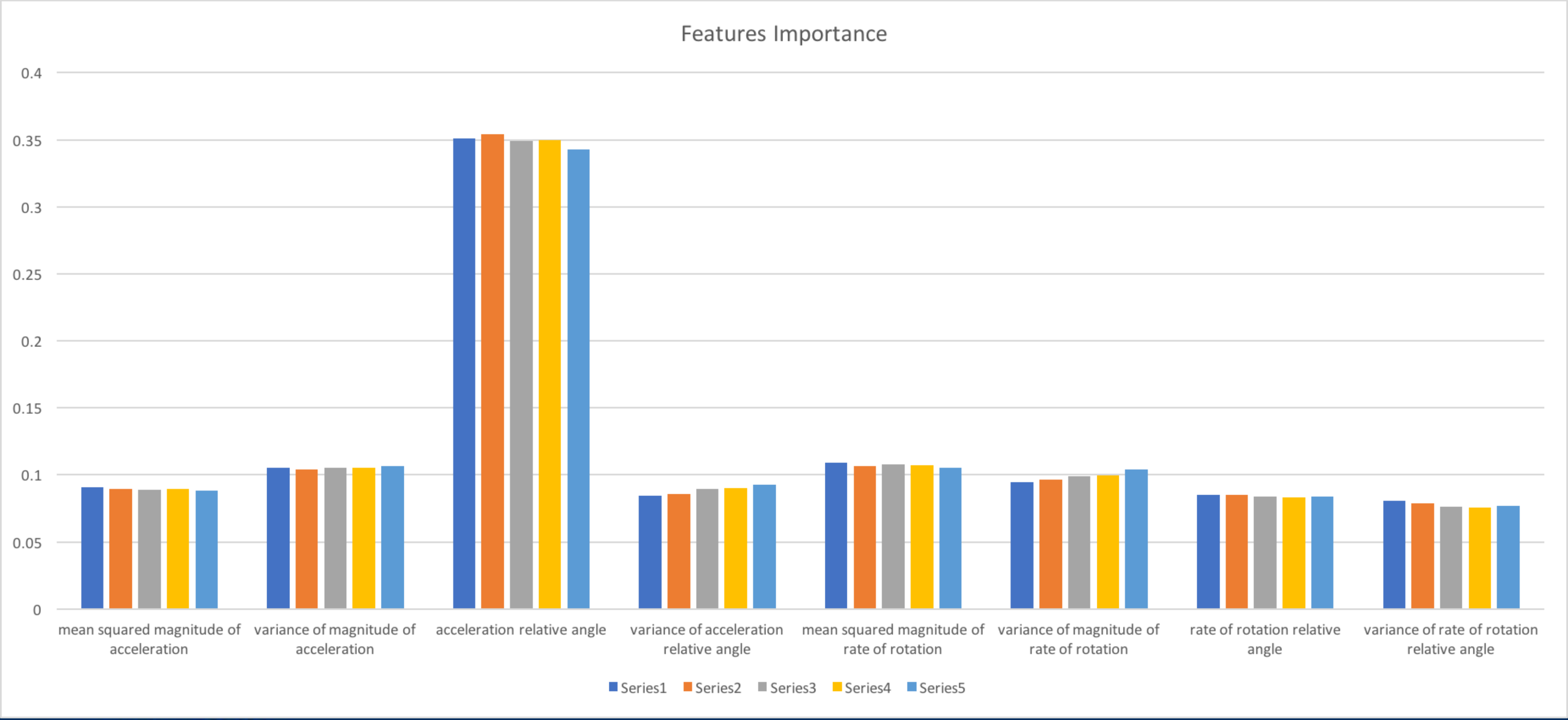
Results

- Data Set:
  - The final size of the data set is approximately 151,000 data points
  - Equivalent to more than 15 hours worth of activity
- Effect of segment length

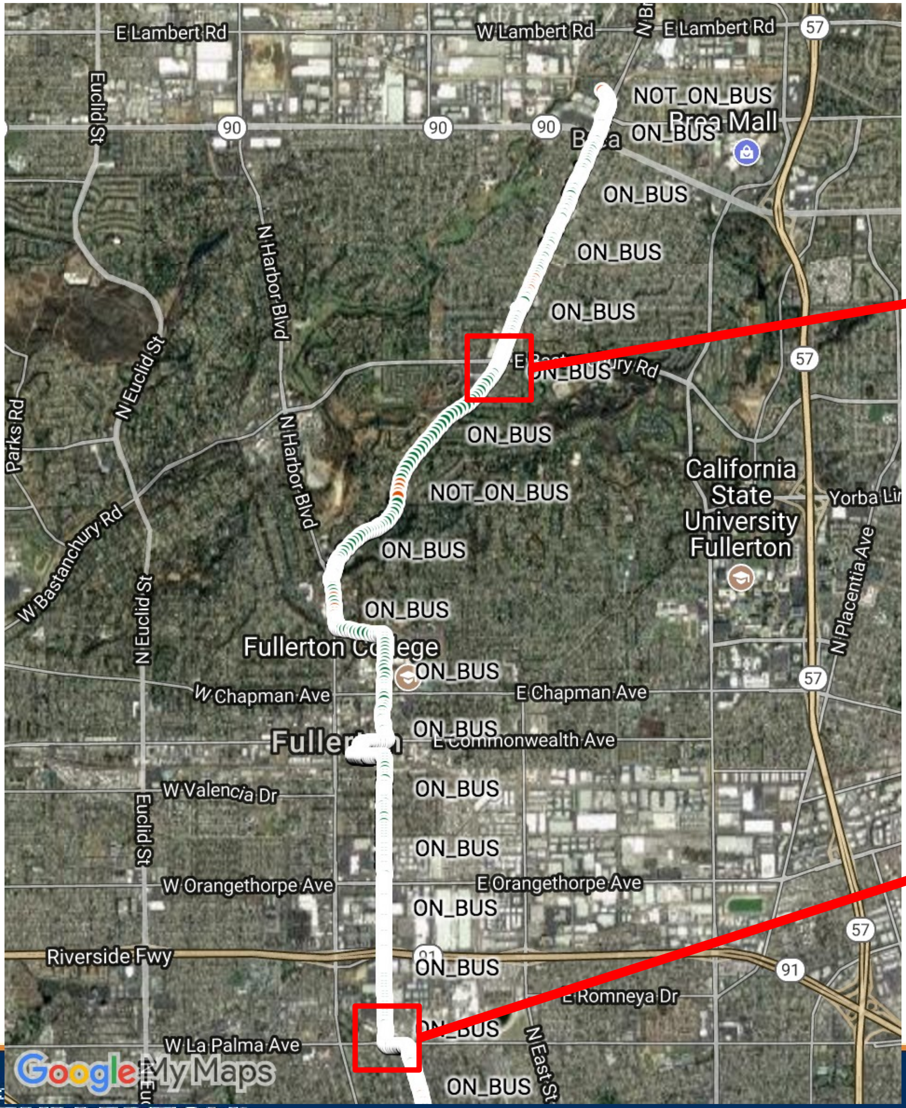


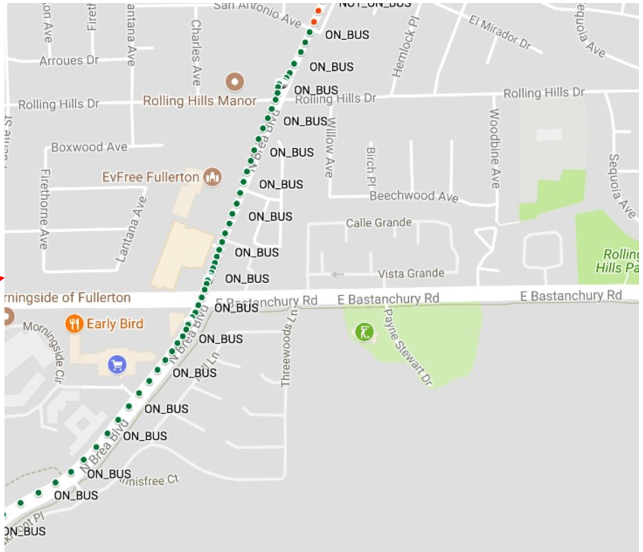
Results

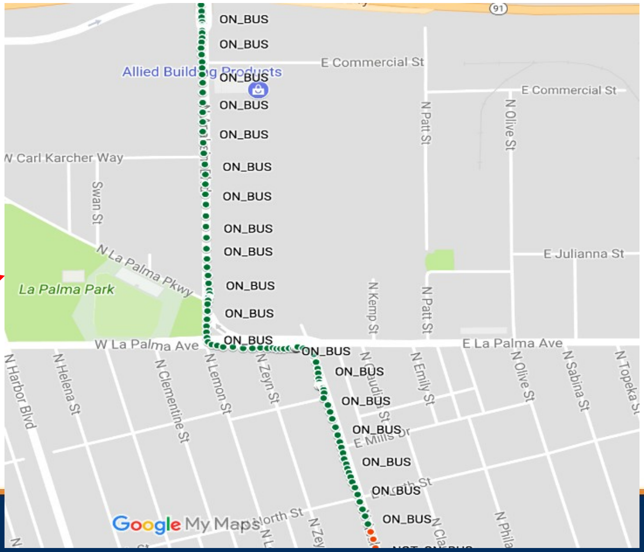
- Relative Importance of Features:
  - Depth of the decision node representing a feature in each decision tree



Sample Trip


E Lambert Rd
W Lambert Rd
E Lambert Rd
NOT_ON_BUS
Brea Mall
ON_BUS
ON_BUS
ON_BUS
ON_BUS
ON_BUS
NOT_ON_BUS
ON_BUS
ON_BUS
California State University Fullerton
Fullerton College
ON_BUS
W Chapman Ave
E Chapman Ave
ON_BUS
E Commonwealth Ave
W Valencia Dr
ON_BUS
ON_BUS
W Orangethorpe Ave
E Orangethorpe Ave
ON_BUS
Riverside Fwy
ON_BUS
E Romneya Dr
W La Palma Ave
ON_BUS
Google My Maps


ON_BUS
Rolling Hills Manor
Rolling Hills Dr
EvFree Fullerton
E Bastanchury Rd
Early Bird
Beechwood Ave
Calle Grande
Vista Grande
ON_BUS


ON_BUS
Allied Building Products
E Commercial St
W Carl Karcher Way
La Palma Park
W La Palma Ave
E La Palma Ave
E Julianna St
ON_BUS
Google My Maps

References

- Russel and Norvig, Artificial Intelligence: A Modern Approach, 4th edition, Prentice Hall, 2021.
- Luger, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, 6th edition, **Chapter 1**, Addison Wesley, 2009.