AI and ML: Introduction and Beyond

Varad Meru

Azure Functions, App Services,
Microsoft Corp.


linkedin.com/in/vmeru


github.com/vrdmr


@vrdmr

Who am I

- Current: Senior SDE @ Microsoft Corp.
  - Working on Python Azure Functions.
  - Previously worked on Compute Platform within Azure Core.
- Previous Experiences:
  - Worked for Orzota (Chennai) and Persistent Systems (Pune).
  - Internship experiences at Facebook and Nomura.
- Education:
  - MS in CS (ML and Data Systems) from UC Irvine (2015)
  - BE in CSE from Shivaji University (DYPCET) (Class of 2011)
- Interests: Distributed Systems & Cloud Infra, ML, Economics.

Outline

- Introduction
- History of AI and Landscape
- Introduction to Machine Learning
- Getting started

Introduction

Q. What is artificial intelligence?

A. It is the science and engineering of making intelligent machines, especially intelligent computer programs.

It is related to the similar task of using computers to understand human intelligence, but AI does not have to confine itself to methods that are biologically observable.

AI Examples I

Here are some examples which give us the perception of AI

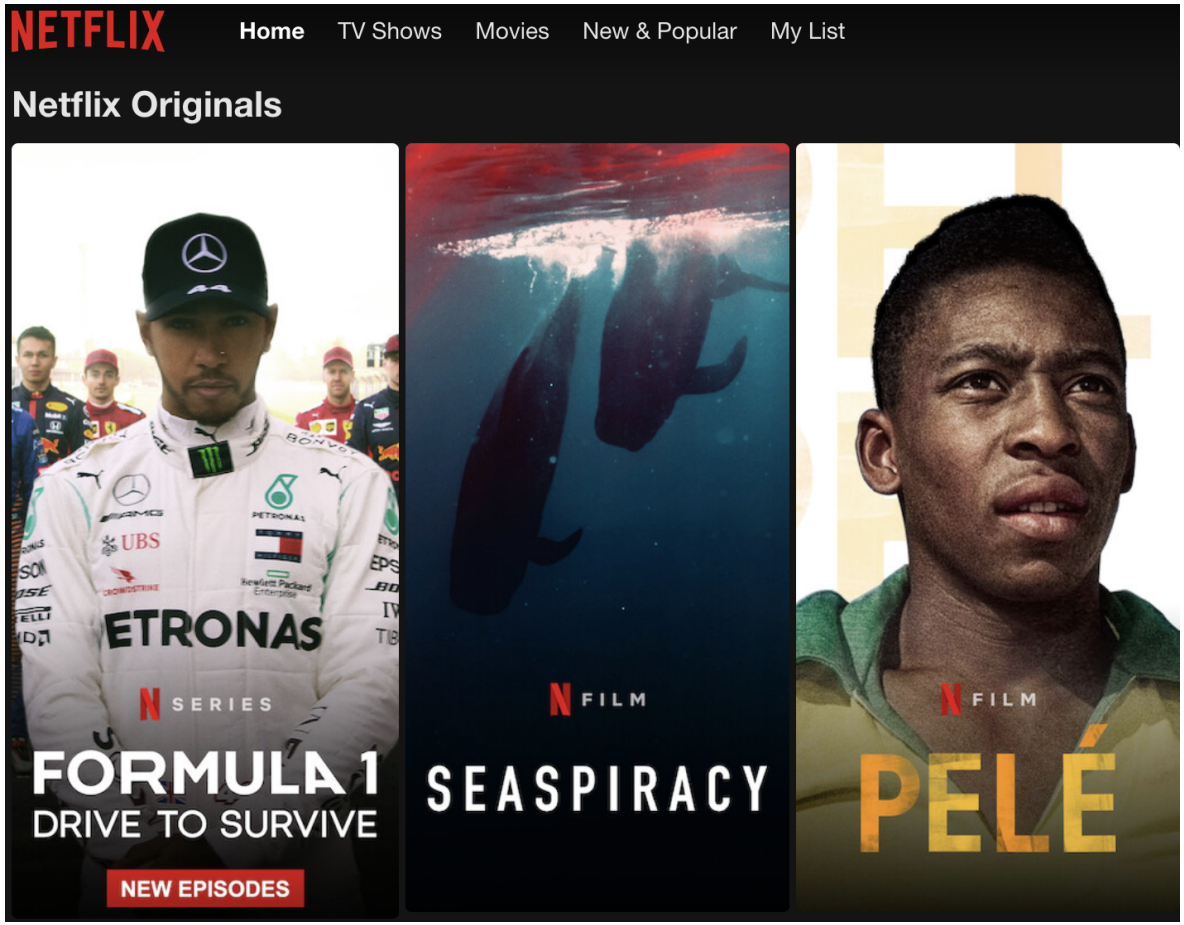


Figure 1: Netflix Recommendations

AI Examples II

Here are some examples which give us the perception of AI

Figure 2: AlphaGo vs Lee Sedol

AI Examples III

Here are some examples which give us the perception of AI

Figure 3: Tesla Autopilot

AI Examples IV

Here are some examples which give us the perception of AI

Figure 4: Robot from Boston Dynamics

History of AI I

- 1940's — Interest in neurons, neural networks and their relationship to mathematics and learning
- 1950 — Turing's paper
- 1956 — Dartmouth conference (Defining AI)
- 1950's and 1960's — enthusiasm and optimism; big promises
- Late 1960's and 1970's — Realization that further progress was really hard; disillusionment
- 1980's — Expert Systems, neural networks, etc.; AI now a little different; quiet successes
- 1990's — Intelligent agents, probabilistic reasoning, machine learning, DeepBlue.
- 2000's — robot pets, self-driving cars.
- 2010's — Deep Learning, AlexNet, AlphaGo, GPT-3.

Calls for clarity

31 Mar 2021 | 17:00 GMT

Stop Calling Everything AI, Machine-Learning Pioneer Says

Michael I. Jordan explains why today's artificial-intelligence systems aren't actually intelligent

By **Kathy Pretz**

Artificial Intelligence—The Revolution Hasn't Happened Yet

by Michael I. Jordan

Published on Jul 02, 2019

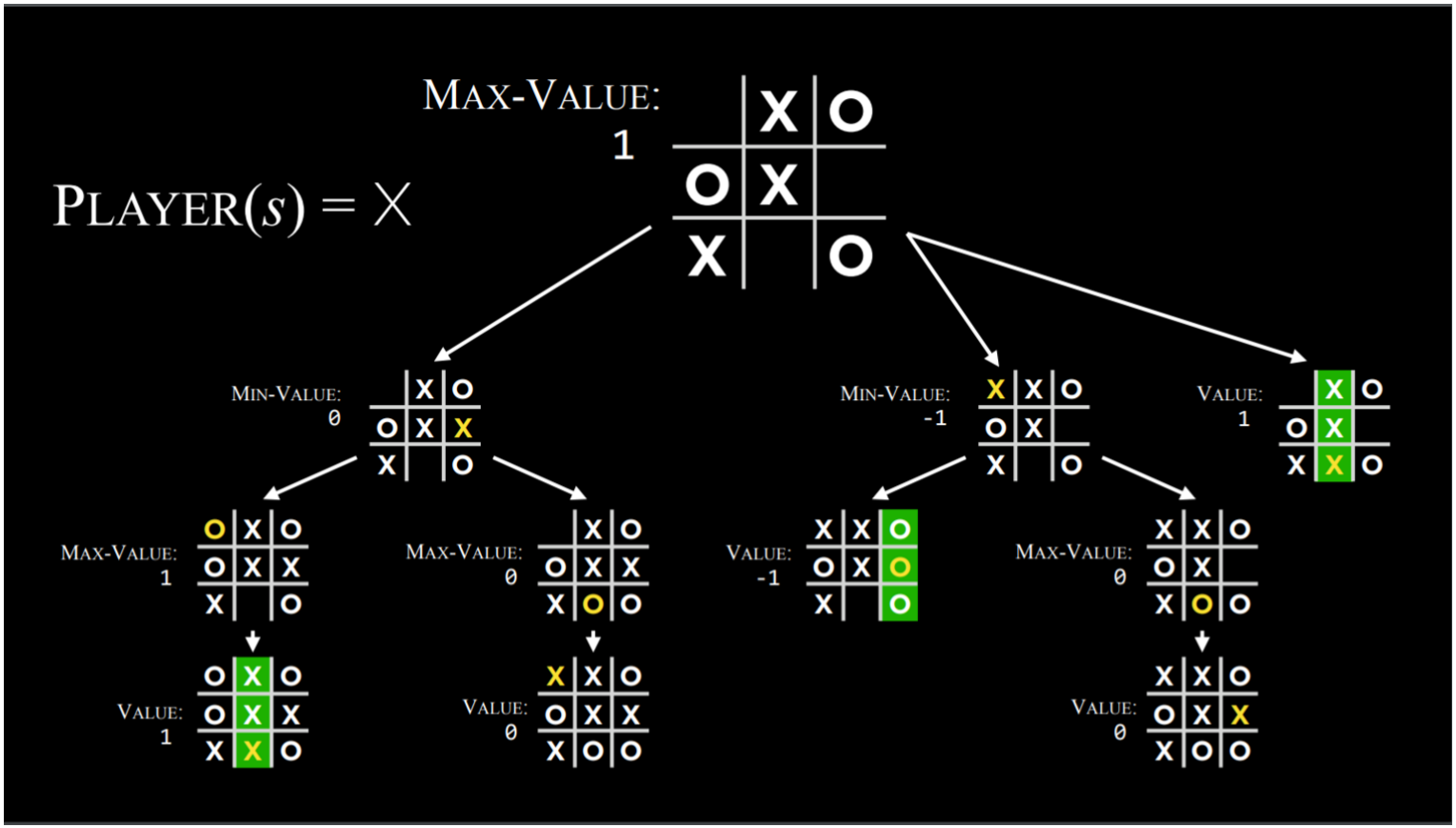
AI Landscape I

- Knowledge representation (including formal logic)
- Search, especially heuristic search (puzzles, games)
- Planning
- Reasoning under uncertainty, including probabilistic reasoning
- Learning
- Agent architectures
- Robotics and perception
- Natural language processing

AI Landscape II

Example: Search

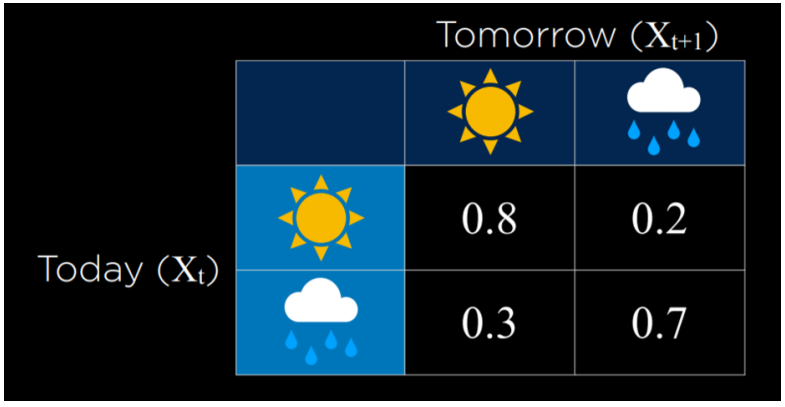
Solving search Problems using `minmax` adversarial search



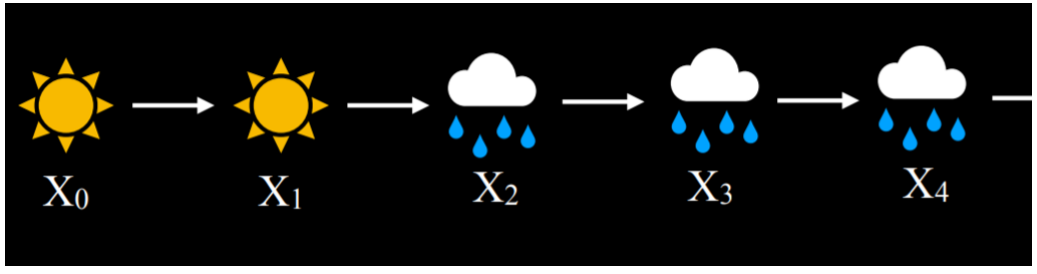
AI Landscape III

Example: Predicting under Uncertainty - Markov Chains

To start constructing a Markov chain, we need a **transition model**



We can now answer questions such as *"what is the probability of having four rainy days in a row?"*



AI Landscape IV

- Knowledge representation (including formal logic)
- Search, especially heuristic search (puzzles, games)
- Planning
- Reasoning under uncertainty, including probabilistic reasoning
- **Learning**: *The main driver of recent successes in AI*
- Agent architectures
- Robotics and perception
- Natural language processing

What is ML? I

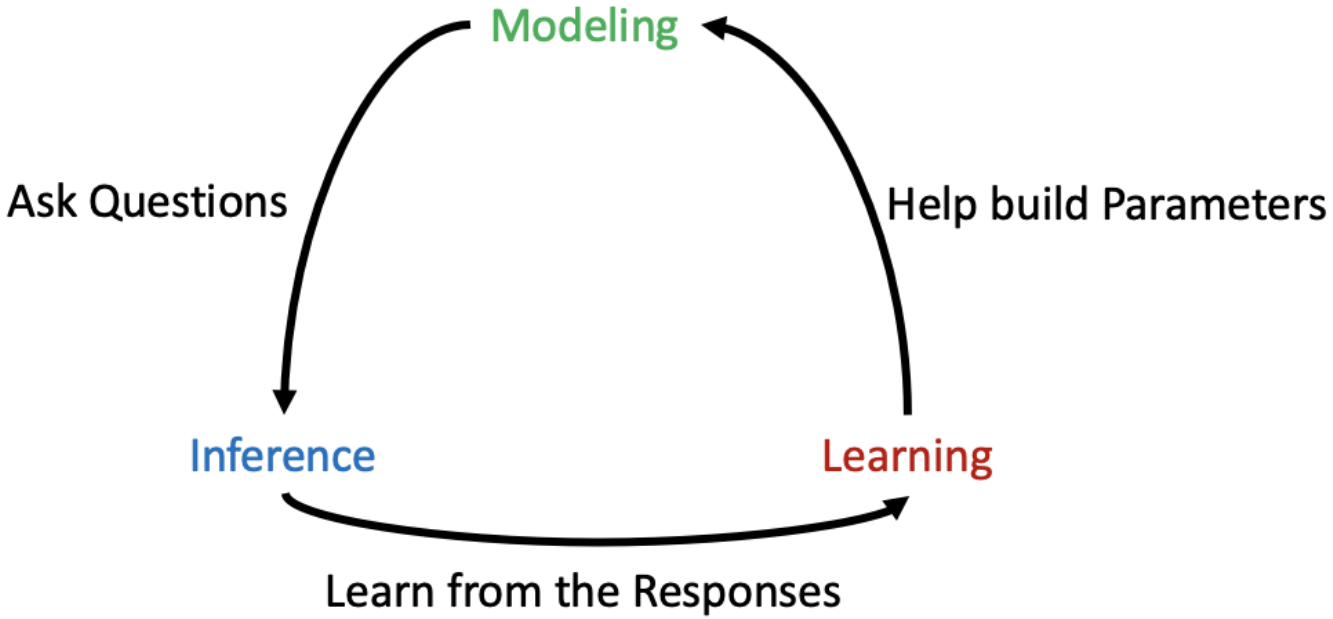
Machine learning algorithms are data analysis methods which search data sets for patterns and characteristic structures.

What is ML? II

Paradigms:

- *Modeling*: Modeling is the process of approximating real world problems using formal mathematical objects called **models**,
- *Inference*: Given a model, the task of **inference** is to answer questions about model.
- *Learning*: Machine learning is this process of turning an abstract model family that we can easily write down into a concrete model of the world that we can query

What is ML? III



ML Algorithms I

Types of Prediction problems:

- Supervised Learning
  - "Labeled" training data
  - Every example has a desired target value (a "best answer")
  - Reward prediction being close to target
    Examples
    - Classification: a discrete-valued prediction (often: action / decision)
    - Regression: a continuous-valued prediction

ML Algorithms II

- Supervised Learning: Some Algorithms
  - Classification: a discrete-valued prediction
    - K-Nearest Neighbors (KNN)
    - Support Vector Machines (SVM)
    - Decision Trees
    - Naïve Bayes Classifiers
    - Logistic Regression
  - Regression: a continuous-valued prediction
    - Linear Regression
    - Poisson Regression
- Supervised Learning: Applications
  - Spam Detection
  - Speech Recognition
  - Handwriting Recognition
  - Information Retrieval

ML Algorithms III

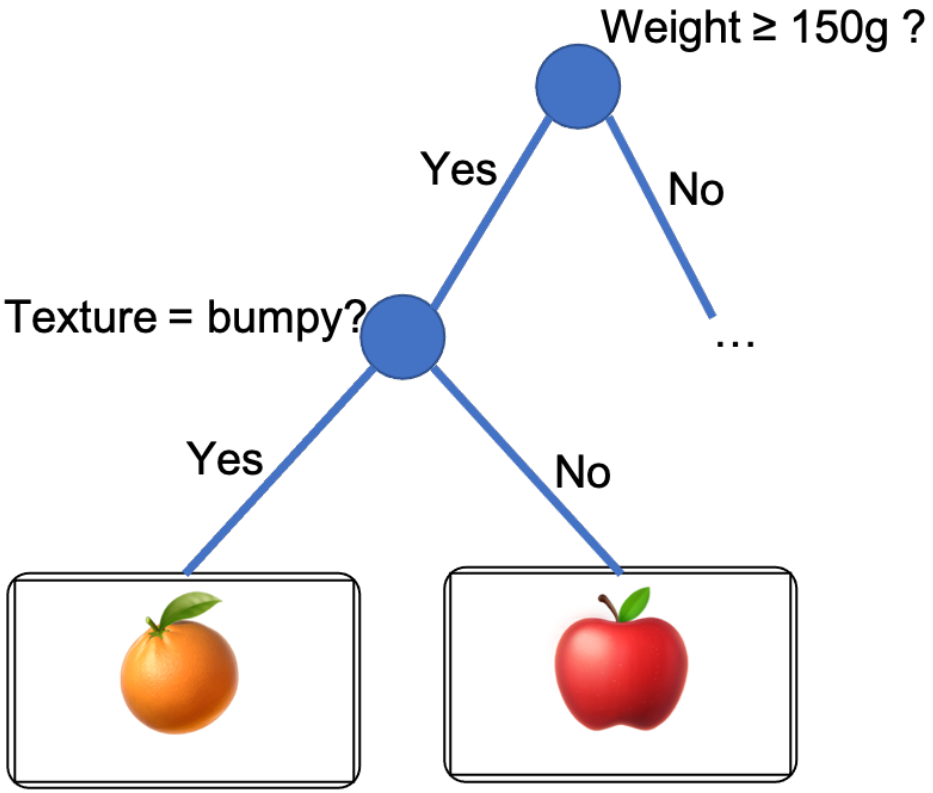
Example - Decision Tree Classifier

Weight	Texture	Label
150g	Bumpy	Orange
170g	Bumpy	Orange
130g	Smooth	Apple
140g	Smooth	Apple
...	...	...

Table 1: Fruit Classification Dataset

ML Algorithms IV

Example - Decision Tree Classifier



ML Algorithms V

Example - Decision Tree Classifier

```
from sklearn import tree
label_map = {1:  "Apple", 0:  "Orange"}
features = [[140, 1], [130, 1], [150, 0], [170, 0]]
labels = [0, 0, 1, 1]
clf = tree.DecisionTreeClassifier()
clf = clf.fit(features, labels)
prediction = clf.predict([[150, 0]])[0]
print(label_map[prediction])
```

ML Algorithms VI

Types of Prediction problems:

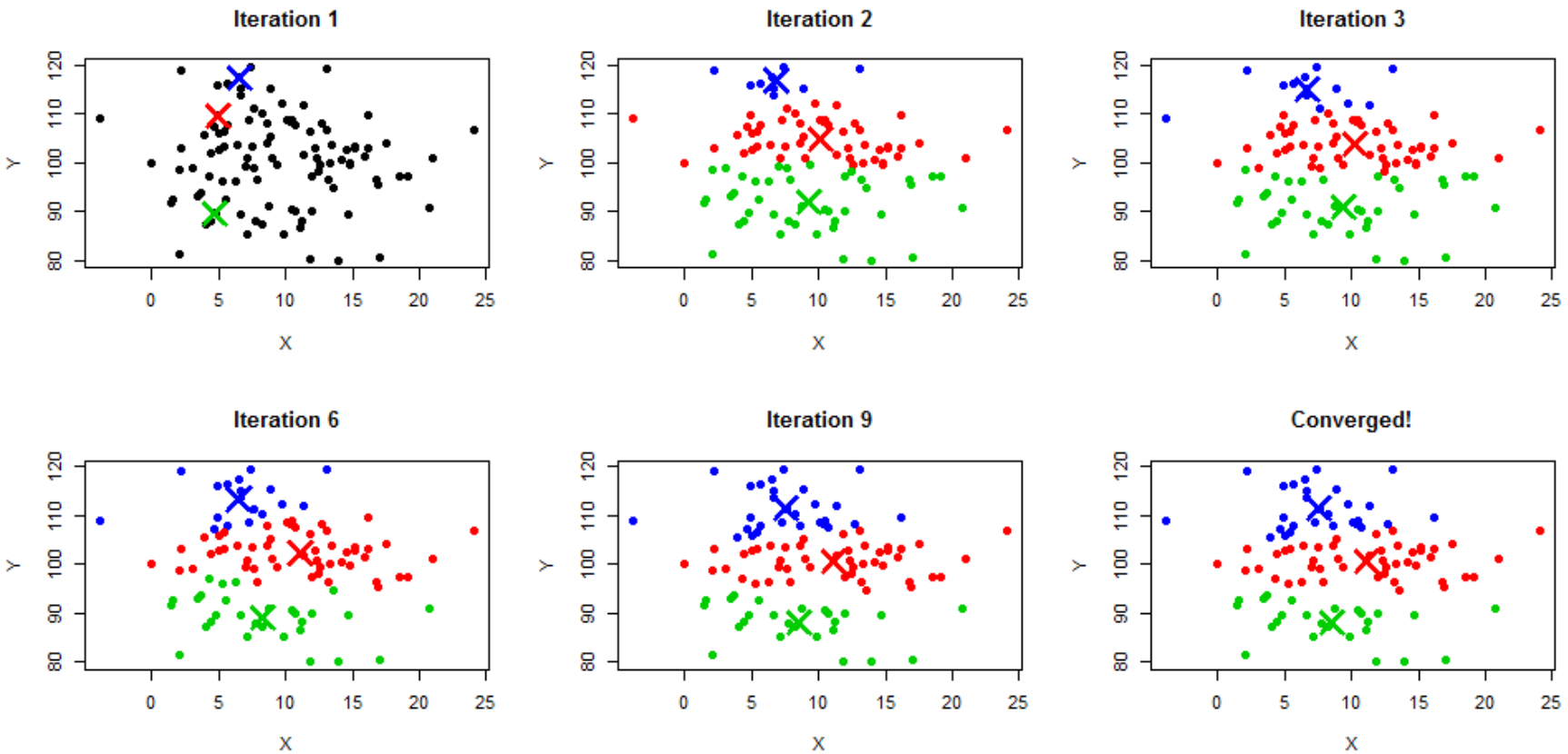
- Supervised Learning
- Unsupervised Learning
  - Data has lots of rich latent structures; want methods to discover this structure automatically.
  - No known target value
  - No targets = nothing to predict?
  - Reward "patterns" or "explaining features"
  - Often, data mining

ML Algorithms VII

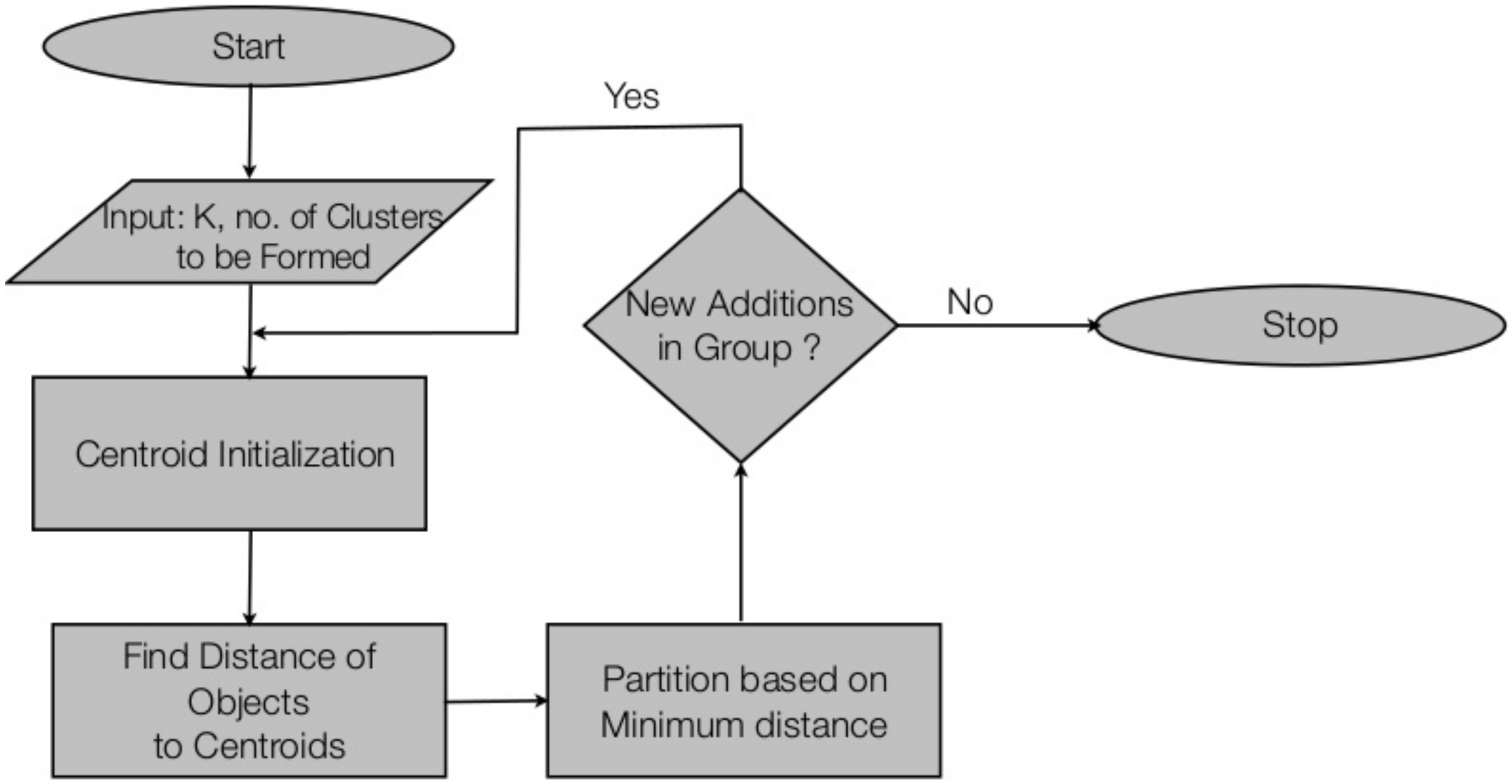
- Unsupervised Learning: Some Algorithms
  - Clustering
    - Hierarchical clustering
    - k-means
  - Anomaly Detection
    - Local Outlier Factor
    - Isolation Forest
  - Learning latent variable models
    - Expectation–maximization algorithms (EM)
    - PCA and SVD
  - Neural Nets: Hopfield, Boltzmann and RBMs, and Auto-Encoders
- Unsupervised Learning: Applications
  - Pattern Recognition
  - Grouping
  - Feature Engineering

ML Algorithms VIII

Example - KMeans Clustering



ML Algorithms IX



ML Algorithms X

Example - KMeans Clustering

```
from sklearn.datasets import make_blobs
from sklearn.cluster import KMeans
# create dataset
X, y = make_blobs(n_samples=150, n_features=2, centers=3,
cluster_std=0.5, shuffle=True,random_state=0)
km = KMeans(n_clusters=3, init='random', n_init=10, max_iter=300,
tol=1e-04, random_state=0)
y_km = km.fit_predict(X)
```

ML Algorithms XI

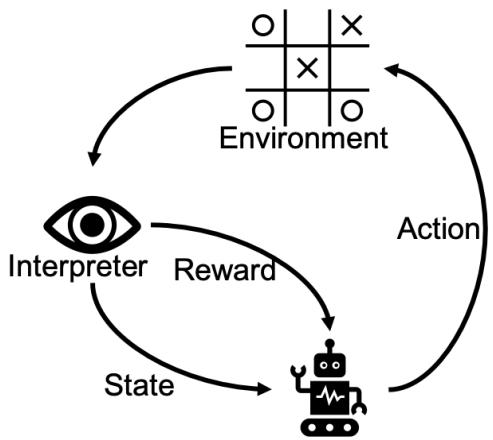
Types of Prediction problems:

- Supervised Learning
- Unsupervised Learning
- Semi-supervised learning
  - Similar to supervised
  - Some data have unknown target values
    Examples
    - Medical data - Lots of patient data, few known outcomes
    - Image tagging - Lots of images on Flikr/Facebook, but only some of them tagged

ML Algorithms XII

Types of Prediction problems:

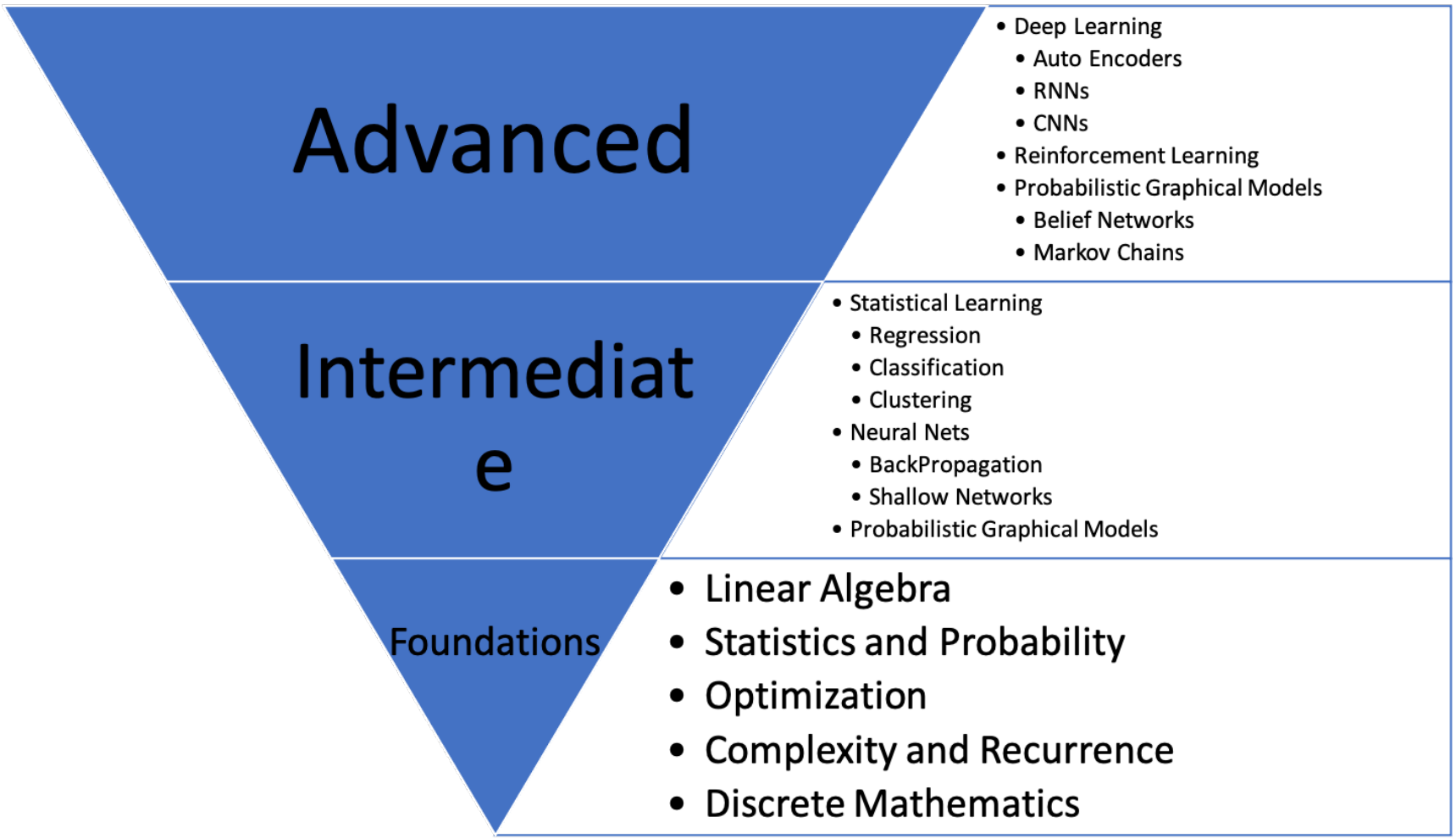
- Supervised Learning
- Unsupervised Learning
- Semi-supervised learning
- Reinforcement learning
  - "Indirect" feedback on quality - No answers, just "better" or "worse"
  - Feedback may be delayed



Tools and Libraries (Python)

- Scikit-learn: best library for classical ML algorithms: `scikit-learn.org`
- Tensorflow: Machine Learning and Deep Learning library: `tensorflow.org`
- Pandas: data extraction and preparation: `pandas.pydata.org`
- NumPy: arrays and linear algebra library: `numpy.org`
- SciPy: scientific computing library: `scipy.org`
- Matplotlib: plotting and data visualization: `matplotlib.org`
- PyTorch: alternative Deep Learning library: `pytorch.org`
- Keras: high-level wrapper around TensorFlow: `keras.io`

ML Pyramid


Advanced
• Deep Learning
• Auto Encoders
• RNNs
• CNNs
• Reinforcement Learning
• Probabilistic Graphical Models
• Belief Networks
• Markov Chains
Intermediat
e
• Statistical Learning
• Regression
• Classification
• Clustering
• Neural Nets
• BackPropagation
• Shallow Networks
• Probabilistic Graphical Models
Foundations
• Linear Algebra
• Statistics and Probability
• Optimization
• Complexity and Recurrence
• Discrete Mathematics

Getting Started I

Artifical Intelligence

- CS50's Introduction to Artificial Intelligence with Python (Harvard on edX)
- Artificial Intelligence: A Modern Approach by Peter Norvig

Machine Learning and Advanced ML

- Machine Learning Fundamentals with Python (Datacamp)
- Machine Learning with Python: from Linear Models to Deep Learning (MIT on edX)
- Learning From Data (CalTech on edX)
- Deep Learning Specialization (Deeplearning.ai)
- Probabilistic Graphical Models Specialization (Coursera)
- Introduction to TensorFlow for AI, Machine Learning, and Deep Learning (Coursera)

Getting Started II

Foundations

- *Introduction to Linear Algebra* (Book) and *Linear Algebra (18.06)* by Gilbert Strang (MIT) (Online Course)
- *Probability and Statistics* by Morris H. DeGroot, Mark J. Schervish (Book)
- *Convex Optimization* (Book) and *Convex Optimization* by Stephen Boyd (Stanford) (Online Course)
- *Introduction to Probability* (on edX) by Joseph Blitzstein (Harvard) (Online Course)
- *Probability and Statistics in Data Science using Python* (on edX) by Alon Orlitsky and Yoav Freund (UC San Diego) (Online Course)

Thank You

Questions?

linkedin.com/in/vmeru

github.com/vrdmr

@vrdmr