ECON484 Machine Learning

1. Concepts of Machine Learning (1/2)

Lecturer:
Bora GÜNGÖREN
bora.gungoren@atilim.edu.tr

Why Machine Learning

- Computer software include algorithms and data structures, both of which are either **designed from scratch for a particular very well defined problem**, or **selected among alternatives based on fitness to a more or less well defined problem**.

Why Machine Learning

- What happens **when we cannot well define our problem**?
  - This happens more often than expected.
  - When this happens **within a tiny scope**, **data obtained from observations** on human (or systems’) behavior can be used to **estimate and/or imitate what would be done by human (or systemic) actors**.
  - This is called machine learning, the computer algorithm **learns and mimics existing behavior**.

Why Machine Learning

- Example: Price-setting for toilet paper rolls.
  - Our scope is simple :
    - Observe the **prices set by our competitors**. Try to be cheaper than some, try to be more expensive than some, based on our **tacit understanding of quality**.
    - Try to be **cash positive**. We buy ingredients that will be used for over several weeks, but sell our product over time. We should **estimate ingredients’ cost, several weeks ahead** so that when the time comes to replenish inventory, we will have enough cash.

Why Machine Learning

- Example: Price-setting for toilet paper rolls.
    - Our initial model is very simple: **Moving averages**.
        - Take **moving averages** of competitors’ pricing, and our pricing.
            - This can be used to **predict systemic price increases** in the market. **Why?**
            - Observe the ratio of our price to this average. This is **expected to be stable**. Why? How would you **define stability**?

Why Machine Learning

- Example: Price-setting for toilet paper rolls.
  - Our initial model is very simple: **Moving averages**.
    - Calculate (moving) **mean time between price increases** and (moving) **mean price increase percentage** of ingredients.
      - Are these means **stable**? How do we **define stability**?
    - If we are **planning to order after the next expected price increase**, we should also increase prices by the mean price increase to stay cash positive.

Why Machine Learning

- How does learning come into play?
  - Time series for time between price increases (days): 90, 90, 75, 90, 75, 75, 60, 60, 45, ...
    - Time series for mean time: 90, 90, 85, 86, 84, 83, 79, 76, ...
    - Time series for moving average (w=3): 90, 90, 85, 85, 80, 80, 70, 65, 55…

Why Machine Learning

- How does learning come into play?
  - Time series for price ratio (without intervention): 1.5, 1.5, 1.5, …, 1.45, 1.45, ….., 1.40, ….., 1.40, 1.30, 1.30, …
    - Our relative price is going down, which means, **our competitors are increasing prices**.
    - There should be **a reason behind their collective price increase**. Whatever the reason, it will effect us as well.

Why Machine Learning

- How does learning come into play?
  - Time series after intervention: 1.5, 1.5, …, 1.45, **1.5**, …., 1.44, 1.5, …., 1.40, 1.5, …
    - We catch up with the price increase.
    - Our system **learns to increase (or decrease) prices**, without the need to **learn the underlying reason**.

Why Machine Learning

- How does learning come into play?
  - We **catch up** with the changing behavior quickly, therefore we learn.
    - Of course we **lag behind**, but we do catch up.
    - If there is a systemic change, and the system becomes stable after the change, we will **eventually catch up** (to the new steady state).

Why Machine Learning

- How does this help our organization?
  - Eventually this helps us better understand how we conduct business, because it makes us ask **better questions**.
    - Can we make an agreement on prices with our suppliers, so that when our estimation lags behind (i.e. we make a poor decision), we will not be effected as much.
    - Does our understanding of price-quality relation **actually represent the reality**?

Why Machine Learning

- Machine learning mimics human (or systemic) behavior.
  - If what we mimic is inferior, then machine learning will provide inferior results.
- Any discussion for a better model has two aspects.
  - The ability to mimic behavior with better qualities (such as how much we lag).
    - Compare mean vs moving mean vs exponential smoothing.
  - The ability to explain underlying reasons.

Why Machine Learning

- Eventually our price model becomes:
  - $P(t) = P_{\text{comp. markup}}(t-1) \times F_{\text{est. supp. price inc.}}(t+L)$
- Any model is better than having no model.
  - The quality of a model is evaluated with its contribution to the organization.
- Simulate for past years.
  - Calculate the Oracle’s version of pricing and profits that case. See the performance gap with this and reality.

Why Machine Learning

- How do we evaluate performance?
  - Simulate for past years.
    - Calculate the **The Oracle’s version** of pricing and profits.
    - See the **performance gap** with this and reality.
    - Simulate with our model. See **how much of the performance gap** can be closed with this model.
    - Simulate with alternate model. See how much of the performance gap can be closed with that model.
  - Choose best model based on **real world performance indicator**.

Why Machine Learning

- How do we evaluate performance?
  - What if we **do not have** a real world performance indicator?
    - From a practitioner’s point of view, this is very problematic. This means, decision makers cannot evaluate the performance based on an objective criteria. So they will use some subjective criteria. This **undervalues your effort**.
    - From an academic point of view, this is not a very big problem. We always have **statistical techniques and indicators**.
    - We learn the statistical indicators and **propose them as objective (but less useful) performance criteria**.

Why Machine Learning

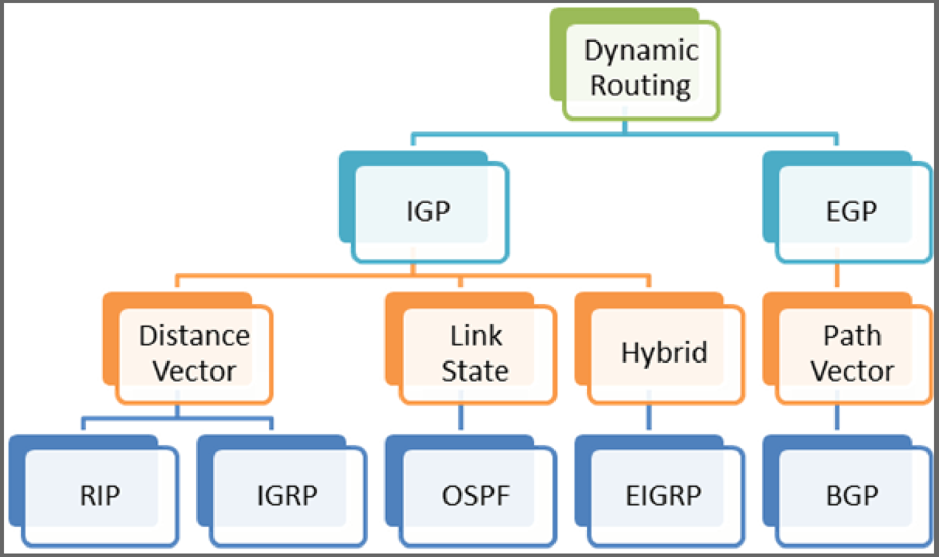
- Another example: Packet routing.
  - This is from computer networks, but can easily be adapted to physical delivery methods.
    - Any (data) packet has a source and destination address.
    - The packet “hops” from router to router many times.
    - Each router decides on where to send the packet next, based on descriptive statistics on the traffic between other (available) routers.
    - “Static routes” are predetermined routes, whereas “dynamic routes” are based on these decisions.

Why Machine Learning

- Another example: Packet routing.
  - Any source to destination journey is usually three-fold:
    - A static part that ends at an exit gateway of your service provider.
    - A dynamic part that starts at your exit gateway, and ends at an entry gateway.
    - Another static part that starts at an entry gateway and ends at the destination.
  - The dynamic part is of our interest.
    - Routers need to learn about the “best” routes.
    - Best with respect to which criteria?

Why Machine Learning

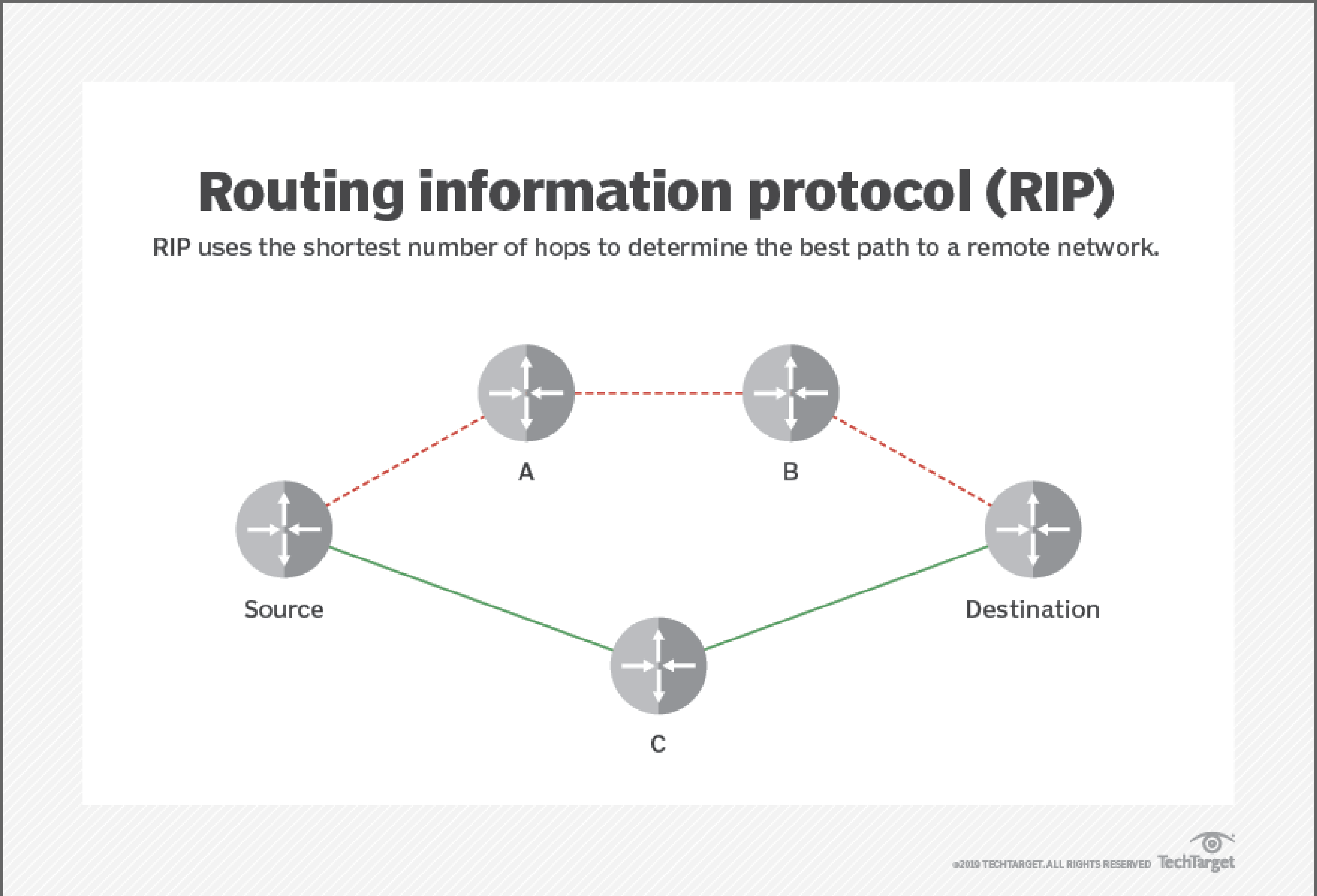
- Realms of "Dynamic Routing"
  - Interior routing protocols are designed for use within a single autonomous system (ie. Router).
  - Exterior routing protocols are designed for use between routers.



Why Machine Learning

- Another example: Packet routing.
  - Routing Information Protocol (RIP, 1988, very old but simple)
    - Routing information protocol is an IGP that bases its internal logic on distance-vector; **this vector describes the information a router knows about a route**.
    - Each RIP router maintains a **routing table**, which is a list of all the destinations the router knows how to reach. Each router broadcasts its entire routing table to its closest neighbors every 30 seconds.
    - Routers decide where to route based on this table.

Why Machine Learning



Why Machine Learning

- Another example: Packet routing.
  - RIP updates its knowledge based on an algorithm, hence it is learning (or not?).
  - This algorithm has several weaknesses:
    - RIP “counts” hops, and assumes all hops are identical. This is a very bad assumption.
    - If a router crashes, it is discovered after some errors at neighboring routers and the change in the routing table is propagated accordingly. However, this takes a lot of time.
    - It is **both memory and computation intensive**. There are $2^{32} \sim 10^{9}$ IPv4 addresses, hence many routes. So it is **observed to be not scalable**.

Why Machine Learning

- Another example: Packet routing.
  - Our evaluation of RIP, shows that some models work at small scale, but cannot handle larger scale. This is due to the algorithmic complexity and memory requirements of the chosen algorithm.
  - RIP uses a table (a linear data structure, such as an array) and a search algorithm.
    - Discussion: Could you speed up the search with an advanced data structure? What would you use? A tree, a hash map, or a trie?
    - Discussion: If there is limited memory, would you have a method to select which route to dump?

Why Machine Learning

- Another example: Packet routing.
  - Open Shortest Path First (OSPF) is a protocol that creates a complete view of the network by gathering information from all the other routers.
    - Protocols that make such a comprehensive view of a network are referred to as link-state protocols.
    - As a link-state protocol, OSPF builds its routes using a mathematical algorithm known as the Dijkstra Shortest Path First (SPF) algorithm.
    - This algorithm analyses the link-state database and then builds the local routes with the router's information to add to the routing table.

Why Machine Learning

- Another example: Packet routing.
  - Open Shortest Path First (OSPF) uses advanced data structures:
    - Adjacency database: This database create a table known as the neighbor’s table; this table list all neighbor routers that a router has established a bidirectional connection, and it is unique to all routers.
    - Link-state database (Topology table): One purpose of forming an OSPF adjacency is to allow two neighbors to exchange their database. This table stores information about all other routers in that network, and it is identical for all routers within an area with identical LSDB.

Why Machine Learning

- Another example: Packet routing.
  - Open Shortest Path First (OSPF) uses advanced data structures:
    - Forwarding database: This database creates the routing table that contains all known networks connected to the router or learned from adjacent routers.
  - OSPF also used advanced algorithms.
    - Dijkstra Algorithm makes a Shortest Path First (SPF) tree by placing every router at the base of this tree and computing the shortest path to every router, and this path is then added to the routing table.

Why Machine Learning

- Another example: Packet routing.
  - Does OSPF facilitate machine learning?
    - Routers running OSPF use five packet types to convey their routing information.
      - Hello Packet,
      - Database Description packet,
      - Link-state request packet,
      - Link-state update packet,
      - Link-state acknowledgment packet.
    - OSPF routing policies are dictated by link metrics associated with each routing interface, **typically the interface speed which is a preset value**.

Why Machine Learning

- Another example: Packet routing.
  - Does OSPF facilitate machine learning?
    - The **database of available routes changes over time**, but the selection procedure **does not use a historical record of actual network performance**.
    - No use of
      - Actual transmission times.
      - Actual packet error rates.
      - Actual throughput values.

Why Machine Learning

- Another example: Packet routing.
  - Does OSPF facilitate machine learning?
    - Therefore there is **very limited, if at all, adaptation** to network behavior.
      - Adaptation quality (real world performance indicator):
        - OSPF adapts to a router disappearing.
        - OSPF does not adapt to a router slowing down.
        - Which happens at what frequency?
        - Can you predict disappearance from slowing down?
      - Most professionals, and almost all academics would **disqualify OSPF as machine learning**.

Why Machine Learning

- Another example: Packet routing.
  - How does **a machine learning based router** work?
    - Is uses OSPF as a base. But the selection algorithm (Dijkstra) uses historical data to build the trees.
      - When you update the historical data about a router, you also update the tree computed using that router at base.
    - If a router becomes very slow, we disqualify it for a while.
      - We stop loading it with traffic, so that it can re-arrange its internal state and maybe even reboot.
    - Hence we learn about changes in router behavior and adapt.

Why Machine Learning

- Machine learning is not just another data processing or database problem.
  - It is part of the general notion of artificial intelligence.
  - To be intelligent, a system (algorithm) that is located within a changing environment (behavioral agents, producing data streams) should have the ability to learn, and adapt (change its own behavior) to the environment.
  - In machine learning we process historical data with incoming data to learn.

Why Machine Learning

- Another example: Customer classification.
  - We are a bank and we receive many requests for consumer spending credits.
    - These credits are usually small amounts, but there are many different people who obtain these credits.
    - We need to classify people into groups.

Why Machine Learning

- Another example: Customer classification.
  - We need to classify people into groups.
    - Those who will pay all installments at the designated time.
    - Those who will miss some payments but will eventually catch up.
    - Those who will go default on the credit, but can re-structure debt.
    - Those who will go default on the credit, and can-not re-structure.
  - Historical data has many people already assigned into these groups.

Why Machine Learning

- Another example: Customer classification.
  - Based on what historical data we have, and what personal data is provided at application stage, we should devise a model.
  - In other words, **a machine learning model needs to be designed based on the historical data set we are given, and the incoming data (both in content and pattern) we will evaluate**.

Why Machine Learning

- Another example: Customer classification.
  - A machine learning approach would require historical data.
    - Particular data on the person applying.
    - Relatable data on “similar” persons. (Define similarity)
    - Less relatable data on general population.

Why Machine Learning

- Most machine learning problems fall into some basic categories:
  - Estimation of (near) future behavior
  - Classification of a new item
  - Selecting parameter values to optimize a system.
- We first try to understand if the problem is of prediction or description nature.
  - If there is both, maybe we need two models instead of one.

Why Machine Learning

- Machine learning **uses theoretical base from statistics, in building mathematical models.**
- Machine learning **uses theoretical base from computer science to enable practical use of those models on modern, readily available computing resources**.

Why Machine Learning

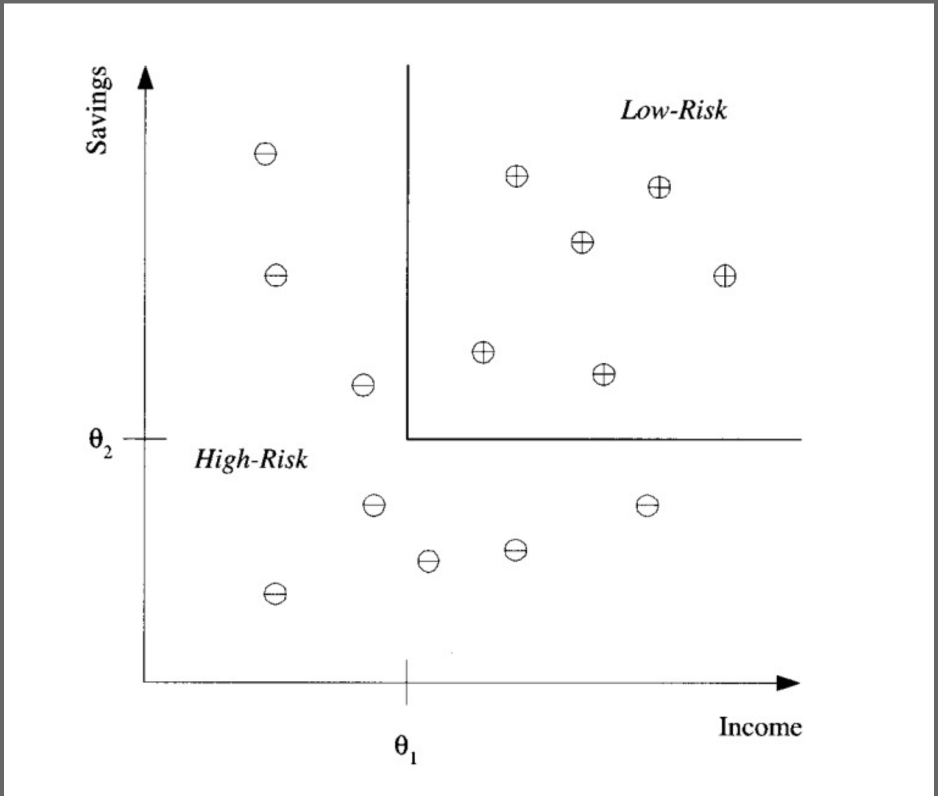
- Theoretical basis from statistics focuses on qualities such as predictive accuracy.
- Theoretical basis from computer science focuses on algorithmic complexity, parallel computing, and memory use.
- Computer science is important in both training the model and using it in practice.
  - Some models are always in training, some not.

Typical Machine Learning Tasks

- Learning Associations
  - Finding an association rule, in the form of a conditional probability P(Y|X) where Y is the state we want to condition on X, which is the existing state of the input.
    - Typical Example: P(bought milk | bought eggs) = 0.75. Then we can assume if someone has bought eggs, they can buy milk. But will they buy **from us**?
      - Re-organize shelves so that they are closer.
      - Provide an awards program for those who buy them bought.
      - Suggest buying milk to customers with eggs in their basket.

Typical Machine Learning Tasks

- Classification
  - Determine thresholds, centers of masses, distances, radii, etc. so that we can define categories to classify phenomenon (events, people, etc).



Typical Machine Learning Tasks

- Classification
  - Then evaluate an incoming item so that it is classified easily (i.e. without difficult computational resources) and quickly.
    - The determination of these parameters (threshold values, centers of masses, etc) is the training stage of our machine learning model.
    - Once the model is trained we use it with the simplicity of a few consecutive questions.
      - In this case the questions with the values are called a **discriminant**, and the result is called a **prediction**.
  - Once in a while the model is re-trained. **Why?**

Typical Machine Learning Tasks

- Pattern Recognition
  - Pattern recognition is a data analysis method that uses machine learning algorithms to automatically recognize **patterns** and **regularities** in data.
  - Pattern recognition systems should recognize familiar patterns quickly and accurately.

Typical Machine Learning Tasks

- Pattern Recognition
  - Pattern related data can be anything from text and images to sounds or other definable qualities.
    - Scanners and OCR software uses images
    - Face recognition uses images
    - Financial fraud detection uses financial transaction data
    - Medical diagnosis assistants use medical data from lab results as well as textual inputs from doctors
    - Speech to text software use voice patterns in the utterances of speakers.

Typical Machine Learning Tasks

- Knowledge Discovery / Knowledge Extraction
  - A nontrivial extraction of implicit, previously unknown, and potentially useful information from structured or unstructured data sets.
  - The extraction result goes beyond the creation of structured information or the transformation into a relational schema.

Typical Machine Learning Tasks

- Knowledge Discovery / Knowledge Extraction
  - Example. Named-entity recognition (NER) seeks to locate and classify named entities mentioned in unstructured text into predefined categories such as person names, organizations, locations, medical codes, time expressions, quantities, monetary values, percentages, etc.

Typical Machine Learning Tasks

- Knowledge Discovery / Knowledge Extraction
  - Example. Molecular mining. Since molecules may be represented by molecular graphs, these graphs can be used to search for potentially useful molecules for a particular effect. Variations of molecular mining is used in **drug discovery**.

Typical Machine Learning Tasks

- Outlier Detection / Anomaly Detection
  - Finding items that do not obey rules, and should be evaluated as exceptions.
  - Exceptions could mean many things.
    - Exceptionally good examples such as high performing employees.
    - Exceptionally bad examples such as accounts in extreme debt.

Typical Machine Learning Tasks

- Regression
  - Most prediction problems are termed as regression, regardless of the mathematical technique used.
    - Broadly speaking, even a moving average is a regression model.
  - Regression is also used in optimization problems.
    - Example. Washing machine cycle time optimization.
  - Regression is also used in joining multiple models, trying to optimize their balance.

Typical Machine Learning Tasks

- Supervised Learning
  - Clustering and Regression based models are supervised learning models.
  - The aim is to discover / learn / optimize a **mapping (or function) from an input to an output with the help of a supervisor.**
    - The supervisor tells us the success level of our discovery process in terms of a true/false value or as some kind of distance to the reality.
    - In such mode, we are **training** our model **with supervision**.
    - Such training is usually done with a **training data set**, based on history and **well prepared**.

Typical Machine Learning Tasks

- Supervised Learning
  - The generic model of supervised learning is in the form of $y = g(x|\Theta)$ where y is the result, x is the inputs, and g is the model. Θ represents model parameters.
    - In classification models, y is a discrete result showing assignment into a particular classification.
    - In prediction models, y is the predicted value.
    - Machine learning optimizes the model parameters so that errors are minimized based on the training data set (where we know the results, y-values).

Typical Machine Learning Tasks

- Supervised Learning
  - Because parameter optimization is usually based on a type regression model at some point, most supervised learning tasks use regression.
  - Hence the common saying “… percent of machine learning is regression”.

Typical Machine Learning Tasks

- Unsupervised Learning
  - When there is no supervisor, there is no way to tell right from wrong, and often there is no definition of right or wrong.
  - We only want to understand the structure of or patterns within the input data set.
  - This is a case of unsupervised learning.
  - Clustering algorithms are typical examples of unsupervised learning.

Typical Machine Learning Tasks

- Reinforced Learning
  - When we are dealing with a system, where outputs of past execution is fed into the system as an input for now, then the system is reinforcing itself.
    - Previous price setting example is such an example of a reinforcing system.
      - We raise prices based on competitor’s prices.
      - Competitors use a similar (or same) model using our raised price as an input and raise prices.
      - Then we raise prices because competitors raised also.
      - This goes on until the moment our customers decide not to buy at all.

Typical Machine Learning Tasks

- Reinforced Learning
  - Reinforced learning is in many cases related to game theory, the difference being, we do not need to know the exact decision model of players.
  - The only shared theory is that there is a sequence of actions (turns in gameplay) and a sequence of good actions is usually more important than a single good action.

Typical Machine Learning Tasks

- Reinforced Learning
  - Reinforced learning is not easy because systems involving game play are only **partially observable**, and decisions are to be based on the data at hand.
  - When agents cooperate to a degree, reinforcement learning is easier to design.
    - The case of machine learning for network routing with interaction between routers is such an example of agent cooperation. Routers could **deliberately** ask for states or announce states to increase their knowledge about the overall state of the system.

Statistical Understanding of Machine Learning

- Change in Terminology
  - Forming general descriptions from particular experiences is explained as **statistical inference**.
  - Learning is all about adapting and responding to changing environment. This involves making the model run with particular inputs and getting a result (usually involving a regression model). Therefore learning is usually explained as **estimation**.
  - Classification is called **discriminant analysis**.

Statistical Understanding of Machine Learning

- Change in Terminology
  - Most neural network models have their theoretical basis in statistics.
    - They are based on versions of the back-propagation algorithm.
    - And we usually use neural networks as estimators or classifiers.
    - Note that:
      - An estimator is a predictor found from regression algorithm.
      - A classifier is a predictor found from a classification algorithm.

Statistical Understanding of Machine Learning

- Change in Terminology
  - Most neural network models are correctly classified as non-parametric regression models.
    - They could have advantages over parametric regression models.
    - But they also have several implementation issues.
    - Much research has been done and techniques have been developed to overcome these issues.
      - Although we are not focusing on neural networks exclusively in this course, some of these techniques will be within the subject of our course.

Questions?

CONTACT:
bora.gungoren@atilim.edu.tr