Data Mining

The art of extracting knowledge from large bodies of structured data.

What does the process look like?
Data Mining Tasks

Descriptive Tasks
Here, the objective is to derive patterns (correlations, trends, clusters, trajectories, and anomalies) that are able to summarize the underlying relationships in data. Descriptive data mining tasks are often exploratory in nature and frequently require postprocessing techniques to validate and explain the results.

Predictive Tasks
The objective of these tasks is to predict the value of a particular attribute based on the values of other attributes. The attribute to be predicted is commonly known as the target or dependent variable, while the attributes used for making the prediction are known as the explanatory or independent variables.
Data Mining Tasks

• **Anomaly Detection**: the task of detecting unusual deviations.
• **Association Analysis**: the task of discovering patterns that describe relationships.
• **Clustering**: the task of discovering groups and structures.
• **Classification**: the task of assigning (discrete) target variables to one of several predefined categories.
• **Regression**: the task of finding a function that models (continuous) target variables.
• **Collaborative Filtering**: the task of filtering patterns for an unknown user based on patterns for known users.
Defining a Data Mining Task

• Generate a problem statement.
• Utilize background knowledge.
• Posit the right question.
• Understand the data.
• Implement one or more modeling approaches.
• Identify performance measurement criteria.
• Interpret the model(s).
• Visualize and present the results.
The Goal

Using the knowledge discovery process to turn data into knowledge.
Knowledge Discovery Process

Let's look at what we mean by *data*.
What’s in Data?
What’s in Data?

• **Instance:**
  – Thing to be classified, associated, or clustered.
  – Individual, independent example of target concept.
  – Characterized by a predetermined set of features or attributes.

• **Input to learning scheme:** set of instances
  – Usually represented as a single relation.
  – Traditional form for data mining.
  – Advanced methods now exist for relational data.
Preliminaries

What's in an Instance?

• Each instance is described by a set of “features.”
• A feature is a property or characteristic of an instance.
• A feature can take several values (feature values).
  – Can be categorical (nominal or ordinal)
  – Can be numeric (interval or ratio)
• Features can discrete or continuous.
Discrete Features

• Qualitative features.
  – Enough information to distinguish one object from another.

• Has only a reasonable set of values.
  – Thumb-rule: count with your fingers.

• Often represented as integer variables.
  – For example: 0 for red; 1 for blue; etc.

• Note: binary attributes are a special case of discrete features.
Continuous Features

• Most numeric properties hold.
• Can be integer or real number.
• Examples: temperature, height, weight, age, counts.
• Practically, real values can only be measured and represented using a finite number of digits.
Supervised Learning

- Data in the form of tuples or input-output pairs \((x_i, y_i)\) that comes from a deterministic mapping of \(X \rightarrow Y\).
  \[
  \forall \tilde{x}_i \in X
  \]
  \[
  \tilde{x} \equiv \{x_1, x_2, \ldots, x_n\}
  \]
  \[
  \forall y_i \in Y \text{ (set of classes/concepts } c(x))
  \]
  - An example of “supervised” learning.

- Develop an approximation to the mapping that is “consistent” with the data and “not too complex.”
  - Learn a function of the hypothesis.
Instances, Features, and Classes

- **Input Features**: $X = \{x_1, x_2, \ldots, x_m\}$
- **Class**: $Y = \{y\}$

Instances:

<table>
<thead>
<tr>
<th>(i)</th>
<th>(x_i)</th>
<th>(y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>\vdots</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(n)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Instances, Features, and Classes

<table>
<thead>
<tr>
<th>Make</th>
<th>Cylinders</th>
<th>Length</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Honda</td>
<td>Four</td>
<td>150</td>
<td>1956</td>
</tr>
<tr>
<td>Toyota</td>
<td>Four</td>
<td>167.9</td>
<td>2280</td>
</tr>
<tr>
<td>BMW</td>
<td>Six</td>
<td>176.8</td>
<td>2765</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Style</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hatchback</td>
</tr>
<tr>
<td>Wagon</td>
</tr>
<tr>
<td>Sedan</td>
</tr>
</tbody>
</table>

Given car make, cylinders, length, and weight, learn a function for the body style.
## Instances, Features, and Classes

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Wind Speed</th>
<th>Decision</th>
</tr>
</thead>
<tbody>
<tr>
<td>80°</td>
<td>Low</td>
<td>Bike Day</td>
</tr>
<tr>
<td>40°</td>
<td>Low</td>
<td>Couch Day</td>
</tr>
<tr>
<td>60°</td>
<td>Medium</td>
<td>Couch Day</td>
</tr>
<tr>
<td>80°</td>
<td>High</td>
<td>Bike Day</td>
</tr>
</tbody>
</table>

Go out and bike or laze on the couch.
Features and Classes

1. If wind-speed = high, then Bike Day.
2. If wind-speed = medium, then Couch Day.
3. If wind-speed = low and temp \(\leq 40\), then Couch Day.
4. If wind-speed = low and temp > 40, then Bike Day.
Now Consider this Problem

• An advertising company wants to group customers based on similarities. There are no predefined labels for this group, and based on the groups on demographics and past buying behavior, they will have targeted marketing and advertising initiatives.

• What is this?
  – An example of unsupervised learning.
  – No predefinition of groups, a.k.a. classes.
  – Find similarities in data based on features.
  – This is the simplistic view of clustering.
Unsupervised Learning

• Given data Points $X$:
  – Develop a model or representation of the data such that “important” structure or “regularities” (and irregularities) are captured.
  – Organizing instances into groups that share similar features.
  – Model, for example, can be probability distribution estimation: $p(x)$ of the entire $X$. 

$X$:
Examples of Kinds of Data

- Financial transactions
- Genetic sequence data
- Documents
- WWW
- Molecular structures
- Medical data
- Geographical data
Knowledge Discovery Process

So that’s data. The process also often involves the concept of learning.
What is Machine Learning?
Definition of Machine Learning

“A computer program is said to learn from experience $E$ with respect to some class of tasks $T$ and performance measure $P$, if its performance at tasks $T$, as measured by $P$, improves with experience $E$.”

—Tom Mitchell/Machine Learning
Definition of Machine Learning

“A computer program is said to learn from experience $E$ with respect to some class of tasks $T$ and performance measure $P$, if its performance at tasks $T$, as measured by $P$, improves with experience $E$.”

—Tom Mitchell/Machine Learning

• Improve over class of tasks $T$
• With respect to performance measure $P$
• Based on experience $E$

We need to be able to formulate $T$, $P$, and $E$. 
Learning to Play Checkers

• **Task:** Playing checkers.
• **Performance:** Percent of games won in world tournament.
• What training experience?
• What exactly should be learned? (*Target function*)
• How to represent the target function?
• What specific algorithm to learn it?
Formalizing the Learning Task

- Training experience → training data.
- Task → target function required.
  - What is the outcome or what is to be predicted?
- Identify the objective or learning function required to fit the data.
  - For example, rules or decision trees.
- Performance measurement criteria → evaluate the learning function on the testing data.
- How accurate is the function?
Concept Learning

• Acquire general concepts from a set of training examples.
• A concept can describe some objects or events.
  – People, continually, attach “description” to objects or events.
    • I will enjoy sports today if the sky is sunny, air temperature is warm, humidity is normal, and wind is not strong.
• Typically, inferring a boolean-valued function.
  – True or false.
• Definition of concept learning: approximate a boolean valued function from training examples.
Instances, Features, and Classes

<table>
<thead>
<tr>
<th>Make</th>
<th>Cylinders</th>
<th>Length</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Honda</td>
<td>Four</td>
<td>150</td>
<td>1956</td>
</tr>
<tr>
<td>Toyota</td>
<td>Four</td>
<td>167.9</td>
<td>2280</td>
</tr>
<tr>
<td>BMW</td>
<td>Six</td>
<td>176.8</td>
<td>2765</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Style</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hatch back</td>
</tr>
<tr>
<td>Wagon</td>
</tr>
<tr>
<td>Sedan</td>
</tr>
</tbody>
</table>

Given car make, cylinders, length, and weight, learn a function for the body style.
The Weather Problem

<table>
<thead>
<tr>
<th>Outlook</th>
<th>Temperature</th>
<th>Humidity</th>
<th>Windy</th>
<th>Play</th>
</tr>
</thead>
<tbody>
<tr>
<td>sunny</td>
<td>85°</td>
<td>85</td>
<td>false</td>
<td>no</td>
</tr>
<tr>
<td>sunny</td>
<td>80°</td>
<td>90</td>
<td>true</td>
<td>no</td>
</tr>
<tr>
<td>overcast</td>
<td>83°</td>
<td>86</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>rainy</td>
<td>70°</td>
<td>96</td>
<td>false</td>
<td>yes</td>
</tr>
</tbody>
</table>

Given past data, can you come up with the rules for determining the value of Play?
Formalizing Concept Learning

• Given:
  – Instances $X$: Possible days, each described by the attributes Outlook, Temperature, Humidity, Windy.
  – Target function $c$: $PlayGolf$: $X \rightarrow \{0,1\}$.
  – Hypothesis set $H$: Conjunction of attributes.
  – Training examples $D$: Positive and negative examples of the target function $\langle x_1, c(x_1), \ldots, x_n, c(x_n) \rangle$.

• Determine a hypothesis $h$ in $H$ such that $h(x) = c(x)$ for all $x$ in $D$. 
Consistency

“A hypothesis $h$ is consistent with a set of training examples $D$ of target concept $c$ if and only if $h(x) = c(x)$ for each training example $(x, c(x))$ in $D$.”

$$\text{Consistent}(h, D) \equiv (\forall \langle x, c(x) \rangle \in D) h(x) = c(x)$$
The Weather Problem

Conditions for playing sport:

<table>
<thead>
<tr>
<th>Outlook</th>
<th>Temperature</th>
<th>Humidity</th>
<th>Windy</th>
<th>Play</th>
</tr>
</thead>
<tbody>
<tr>
<td>sunny</td>
<td>hot</td>
<td>high</td>
<td>false</td>
<td>no</td>
</tr>
<tr>
<td>sunny</td>
<td>hot</td>
<td>high</td>
<td>true</td>
<td>no</td>
</tr>
<tr>
<td>overcast</td>
<td>hot</td>
<td>high</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>rainy</td>
<td>mild</td>
<td>normal</td>
<td>false</td>
<td>yes</td>
</tr>
</tbody>
</table>

If Outlook = sunny and Humidity = high, then Play = no.
If Outlook = rainy and Windy = true, then Play = no.
If Outlook = overcast, then Play = yes.
If Humidity = normal, then Play = yes.
If none of the above, then Play = yes.
The Weather Problem (Mixed Features)

Conditions for playing sport (with some numeric attributes):

<table>
<thead>
<tr>
<th>Outlook</th>
<th>Temperature</th>
<th>Humidity</th>
<th>Windy</th>
<th>Play</th>
</tr>
</thead>
<tbody>
<tr>
<td>sunny</td>
<td>85°</td>
<td>85</td>
<td>false</td>
<td>no</td>
</tr>
<tr>
<td>sunny</td>
<td>80°</td>
<td>90</td>
<td>true</td>
<td>no</td>
</tr>
<tr>
<td>overcast</td>
<td>83°</td>
<td>86</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>rainy</td>
<td>70°</td>
<td>96</td>
<td>false</td>
<td>yes</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>

If Outlook = sunny and Humidity > 83, then Play = no.
If Outlook = rainy and Windy = true, then Play = no.
If Outlook = overcast, then Play = yes.
If Humidity < 85, then Play = yes.
If none of the above, then Play = yes.
The Premise of Learning

• Given a training set, the (concept) learning algorithm has to estimate the function $f$ or hypothesize about it.

• We have just formed a premise behind inductive learning.
  – What is inductive learning?
Inductive Learning Hypothesis

“Any hypothesis found to approximate the target function well over a large set of training examples will also approximate the target function well over other unobserved examples.”

—Tom Mitchell/Machine Learning
Inductive Learning Hypothesis

• In other words:
  – Given a training set (known information), at best we can build (induce) a hypothesis around it.
  – Think about studying for an exam.
  – Finding a solution that is having a different “inductive” bias.
Inductive Learning Hypothesis

Use training instances to formulate or induce or discover a theory.

Go from specific to general.
Types of Learning

• Supervised learning
  – Given the value of an input vector $X$ and $c(x)$, predict $c$ on future unseen $x$’s.
  – ex., classification, regression

• Unsupervised learning
  – Given $X$, automatically discover the structure, representations, etc.
  – ex., clustering
Supervised Learning

• Classification
  – The predictions or outputs, \( c(x) \) are categorical while \( x \) can take any set of values (real or categorical). The goal is select correct class for a new instance.

• Regression
  – Given the value of an input \( X \), the output \( Y \) denoted by \( \hat{y} \) belongs to the set of real values \( \mathbb{R} \). Goal is to predict output accurately for new input.
Supervised Learning

• Time series prediction
  • Data is in the form of a moving time series. The goal is to perform classification/regression on future time series based on data known so far.
Classification

• Find ways to separate data items into pre-defined groups.
  – We know $X$ and $Y$ belong together, find other things in same group.

• Requires “training data”: data items where group is known.
Unsupervised Learning

• Anomaly detection
  • $X$ can be anything, goal is to detect deviations from normal behavior.

• Association rules
  • Find joint values of the variables $X = \{x_1, x_2, \ldots, x_n\}$, that tend to appear more frequently in the database.
Unsupervised Learning

• Clustering
  • $X$ is provided $c(x)$ or $Y$ is unknown. Grouping a collection of objects into “clusters” such that objects within a cluster are more closely related than those in different clusters.

• Density estimation
  • Describing your data.
Anomaly Detection

• Find unexpected values and outliers.
Association Rules

- Identify dependencies in the data.
  - $X$ makes $Y$ likely
- Indicate significance of each dependency.
Clustering

- Find groups of similar data items.
- Statistical techniques require some definition of “distance” (e.g., between travel profiles) while conceptual techniques use background concepts and logical descriptions.
Inductive Learning Bias

- Consider:
  - Concept learning algorithm $L$.
  - Instances $X$, target concept $c$.
  - Training examples $D_c = \{(x, c(x))\}$.
  - Let $L(x_i, D_c)$ denote the classification assigned to the instance $x_i$ by $L$ after training on data $D_c$.
  - The inductive bias of $L$ is any minimal set of assertions $B$ such that for any target concept $c$ and corresponding training examples $D_c$.

$$\forall x_i \in X)[(B \land D_c \land x_i) \leftrightarrow L(x_i, D_c)]$$
The Simpler the Better

• If there are two hypotheses describing a data, one complex and one simple, which one to take?

• William of Occam in the year 1320 said, “Prefer the simplest hypothesis that fits the data.”
  – “One should not increase beyond what is necessary, the number of entities required to explain anything.”
  – Solid theory in machine learning behind it.

• Remember our set of hypotheses $H$.
  – Given a set of data, there are multiple ways to model it, a set of $H$.
  – Choose the simpler $h$ of $H$ that fits the data.
Inductive Learning Issues

• Along the way, we also learned that the learning algorithm should be able to generalize well.

• The algorithm should be able to induce knowledge of a domain from given training examples, and not merely and completely “overfit” on the training examples.
Overfitting

- Overfitting means doing very well on training data but poorly on the test data, lest test data exactly mimics the training data
  - Counterintuitive?
  - Think about preparing for an exam. What is better: rote memorization or understanding a concept?
  - Learner could not induce a function that generalizes well.

- Generalization is often a tenet of machine learning.
Generalization Behavior

- **Fixed data size**
- **Mean Error**
  - Cross-validation error
  - Training error
  - High bias
  - High variance
- **Model Complexity**
Preliminaries

Summarizing the Basics of Learning

• Define the “domain” (training data and testing data); What are we trying to learn? What is our data source? What characteristics of data are relevant?
• Define the “learner.” Select what is appropriate (no free lunch).
• Do we have any prior knowledge about the domain?
• Define the objective function, $\Phi$. Adjust the learning to minimize the objective function.
• How do we define success? What is the output? How does the learner demonstrate that it has learned something? What decisions can I take based on what the learner tells me?
Summarizing the "Preliminaries"
Summarizing the “Preliminaries”

• Data are composed of *instances*.
• Each instance is described by a set of *features*.
• Sets of instances are used to perform a *task*.
• Data mining tasks can be *descriptive* or *predictive*.
• Often, these data mining tasks involve *learning*.
• Learning involves a tradeoff of *bias* and *variance*.