Machine learning
Concept
Machine learning

Concept

Data → Program → Output

Traditional Programming
Machine Learning

Concept

Data → Program

Output → Machine Learning
Machine learning

Concept

Machine Learning

Output Data

Program

Traditional Programming

Data

Program

Output
Machine learning
Concept

Human Learning process:

- **Generalisation**: Predicting Assuming past predict the future
- **Memorisation**: Capability to memorise (Storage) Observation (Time)

Source: A. Wang and Zirui Wang, "Pedestrian Trajectory Prediction with Graph Neural Networks", 2019.
Machine learning
Concept

Variations:

Supervised

Machine learns a function that maps from input to output by relying on tabled example of pairs of (input, output)

Unsupervised

Machine learns how to group (natural cluster) a set of unlabelled data or feature vector.
Machine learning
Concept

\[ \{(x_n, c_n)\}_n \]

Data pairs

Features \rightarrow Classes

\[ c_m = f(x_m, \theta) \]

Discriminative Classification
Machine Learning

Definition

**Definition:** Machine Learning

Machine learning in Data Science can be defined as a method of data analytics that help automate the building process of analytical model. It is also based on the idea that computers can learn from data in order to identify patterns and make decision with less human intervention.
Machine Learning

Topics

• Classifiers
• Neural Networks and Deep learning
Classifiers
Machine Learning
Classifiers

Definition: Classifiers
Classifiers are algorithms that can categories or order information automatically into classes.

- Linear Discrimination Analysis
- Nearest Neighbour
- Template Matching
- Naive Bayes Classification
Machine Learning

Classifiers

• Linear Discrimination Analysis
• Nearest Neighbour
• Template Matching
• Naive Bayes Classification
Machine Learning
Classifiers

**Definition:** Linear Discriminant Analysis (LDA)

LDA is the result of generalising Fisher’s linear discriminant, which is a method for finding a linear combination of features for separating two or more classes of objects/events. It is also considered as a technique for dimensionality reduction such as PCA.

In mobility and transportation

Weigh Transportation modal Characteristics

Choice of mode
Machine Learning
Classifiers
Machine Learning

Classifiers

Find a new dimensions through projection

$\mu_{car} - \mu_{train}$

maximum separation between the means

minimum variance

$\sigma_{car}$

$\sigma_{train}$

$max \frac{(\mu_{car} - \mu_{train})^2}{\sigma_{car}^2 - \sigma_{train}^2}$
Machine Learning

Classifiers

• Linear Discrimination Analysis
• Nearest Neighbour
• Template Matching
• Naive Bayes Classification
Nearest Neighbour (k=1) or K-nearest neighbours

NN or K-NN is a non-parametric classification method which classify object by popularity vote of its neighbours.
Machine Learning

Classifiers

\[ C = C_m \]

\[ m = \text{arg}_n \min |x - x_n|^2 \]
Machine Learning

Classifiers

\[ C = C_m \]

\[ m = \arg\min_n |x - x_n|^2 \]
Machine Learning
Classifiers

Defining K is needed
Time complexity
Storage
Meaningful distance function

Flexible to feature
Handles multi class
Easy to implement
Machine Learning
Classifiers

• Linear Discrimination Analysis
• Nearest Neighbour
• Template Matching
• Naive Bayes Classification
Template matching is technique that relies on a set of reference patterns to find the best match for an unknown pattern.
Machine Learning
Classifiers

• Example:

\[ \bar{x}(c) = \frac{1}{\sum (c_n = c)} \sum x_n(c_n = c) \]

• Classification

\[ c = \arg_{c \in \text{class}} \min |x - \bar{x}(c)| \]
Machine Learning
Classifiers

Example:
Machine Learning

Classifiers

Example:

Template Matching Results
Machine Learning
Classifiers

• Linear Discrimination Analysis
• Nearest Neighbour
• Template Matching
• **Naive Bayes Classification**
Machine Learning
Classifiers

Definition: Naive Bayes Classification

Naive Bayes is a discriminative classifier derived from a simplified generative Bayesian model from the data.

Bayesian Theorem

\[
P(A \mid B) = \frac{P(A) \times P(B \mid A)}{P(B)}
\]
Machine Learning
Classifiers

Example: Eating at the cafeteria in Delta Building

Students and Professors

Day

Special Menu (SM)

Special Offer (SO)

Weekdays

Yes

Yes

Weekends

No

No

Holidays

Eat at the cafeteria

Don’t eat at the cafeteria

Predict whether a person on campus will eat at the cafeteria on a specific combination of factors such as day, special offer or special menu?
## Machine Learning

### Classifiers

#### Example:

<table>
<thead>
<tr>
<th>Day</th>
<th>Special Offer (SO)</th>
<th>Special Menu (SM)</th>
<th>Eat</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weekday</td>
<td>yes</td>
<td>yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Weekday</td>
<td>yes</td>
<td>yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Weekday</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Holiday</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Weekday</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Holiday</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Weekend</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
</tbody>
</table>

Sample of the 50 Rows of data
Example:

Step 1: Getting the frequencies of each attribute

<table>
<thead>
<tr>
<th>Day</th>
<th>Eat</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Yes</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Weekday</td>
<td>15</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Weekend</td>
<td>11</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Holyday</td>
<td>3</td>
<td>13</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SO</th>
<th>Eat</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Yes</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>28</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>11</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SM</th>
<th>Eat</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Yes</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>29</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>
Example:

\[
P(A \mid B) = \frac{P(A) \times P(B \mid A)}{P(B)}
\]

Bayes
Machine Learning
Classifiers

Example:

<table>
<thead>
<tr>
<th>Day</th>
<th>Eat</th>
<th>likelihood</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weekday</td>
<td>Yes</td>
<td>15/29</td>
</tr>
<tr>
<td>Weekend</td>
<td>Yes</td>
<td>11/29</td>
</tr>
<tr>
<td>Holyday</td>
<td>Yes</td>
<td>3/29</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>2/21</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>6/21</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>13/21</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>16/50</td>
</tr>
</tbody>
</table>

| P(B) = P(weekday) = 17/50 = 0.34 |
| P(A) = P(No Eat) = 21/50 = 0.42 |
| P(B/A) = P(Weekday/No Eat) = 2/21 = 0.10 |
Machine Learning
Classifiers

Example:

<table>
<thead>
<tr>
<th></th>
<th>Calculation</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>P(B)=P(weekday)</td>
<td>17/50 = 0.34</td>
<td></td>
</tr>
<tr>
<td>P(A)=P(No Eat)</td>
<td>21/50 = 0.42</td>
<td></td>
</tr>
<tr>
<td>P(B/A)=P(Weekday/No Eat)</td>
<td>2/21 = 0.10</td>
<td></td>
</tr>
</tbody>
</table>

\[
P(A | B) = \frac{P(A) \times P(B | A)}{P(B)} = \frac{P(NoEat) \times P(Weekday | NoEat)}{P(Weekday)}
\]
## Machine Learning

### Classifiers

**Step 3: Computing the conditional probabilities**

**Example:**

<table>
<thead>
<tr>
<th>Event</th>
<th>Probability</th>
<th>Calculation</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>P(B)=P(weekday) =</td>
<td>17/50 =</td>
<td></td>
<td>0.34</td>
</tr>
<tr>
<td>P(A)=P(Eat) =</td>
<td>29/50 =</td>
<td></td>
<td>0.58</td>
</tr>
<tr>
<td>P(B/A)=P(Weekday/Eat) =</td>
<td>15/29 =</td>
<td></td>
<td>0.52</td>
</tr>
<tr>
<td>P(A/B)=P(No Eat/Weekday) =</td>
<td>(0.34x0.58)/0.51</td>
<td></td>
<td>0.88</td>
</tr>
</tbody>
</table>
### Machine Learning

#### Classifiers

**Example:**

**Step 2: Computing the likelihood**

<table>
<thead>
<tr>
<th></th>
<th>Eat</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SO</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yes</td>
<td>28</td>
<td>5</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td>No</td>
<td>6</td>
<td>11</td>
<td>17</td>
<td></td>
</tr>
</tbody>
</table>

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SO</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yes</td>
<td>28/34</td>
<td>5/16</td>
<td>33/50</td>
<td></td>
</tr>
<tr>
<td>No</td>
<td>6/34</td>
<td>11/16</td>
<td>17/50</td>
<td></td>
</tr>
</tbody>
</table>

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Eat</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yes</td>
<td>28/34</td>
<td>5/16</td>
<td>33/50</td>
<td></td>
</tr>
<tr>
<td>No</td>
<td>6/34</td>
<td>11/16</td>
<td>17/50</td>
<td>34/50</td>
</tr>
</tbody>
</table>
Machine Learning
Classifiers

Example:

Step 2: Computing the likelihood

Frequency

<table>
<thead>
<tr>
<th>SM</th>
<th>Eat</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Yes</td>
<td>29</td>
<td>2</td>
</tr>
<tr>
<td>Yes</td>
<td>29</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>11</td>
<td>8</td>
</tr>
<tr>
<td>No</td>
<td>11</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>40</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>40</td>
<td>10</td>
</tr>
</tbody>
</table>

Likelihood

<table>
<thead>
<tr>
<th>SM</th>
<th>Eat</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Yes</td>
<td>29</td>
<td>2</td>
</tr>
<tr>
<td>Yes</td>
<td>29/40</td>
<td>2/10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>11</td>
<td>8</td>
</tr>
<tr>
<td>No</td>
<td>11/40</td>
<td>8/10</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>40/50</td>
<td>10/50</td>
</tr>
</tbody>
</table>
Machine Learning
Classifiers

Example:

Let's take our factors as follows:
• Day = Weekend
• Special Offer= yes
• Special Menu= yes

B is equal to :
• B1 - Day = Weekend
• B2 - Special Offer= yes
• B3 - Special Menu= yes

A is equal to No Eat
### Machine Learning Classifiers

**Example:**

B is equal to:
- Day = holiday
- Special Offer = yes
- Special Menu = yes

A is equal to No Eat

\[
P(A/B) = P(\text{No Eat/SO=yes, SM=yes, Day=weekend})
\]

\[
= \frac{P(\text{SO=yes/No Eat}) \times P(\text{SM=yes/No Eat}) \times P(\text{Day=Weekend/No Eat}) \times P(\text{No Eat})}{P(\text{SO=yes}) \times P(\text{SM=yes}) \times P(\text{Day=Weekend})}
\]

\[
= \frac{5/16 \times 2/10 \times 6/21 \times 21/50}{33/50 \times 31/50 \times 17/50}
\]

\[
= 0.05
\]
Machine Learning
Classifiers

Example:

B is equal to:
- Day = weekday
- Special Offer= yes
- Special Menu= yes

A is equal to Eat

\[
P(A/B) = P(\text{Eat}/\text{SO=yes, SM=yes, Day=weekend})
\]

\[
= \frac{P(\text{SO=yes/Eat}) \times P(\text{SM=yes/Eat}) \times P(\text{Day=Weekend/Eat}) \times P(\text{Eat})}{P(\text{SO=yes}) \times P(\text{SM=yes}) \times P(\text{Day=Weekend})}
\]

\[
= \frac{28/34 \times 29/40 \times 11/29 \times 29/50}{33/50 \times 31/50 \times 17/50}
\]

\[
= 0.94
\]
The probability of eating is 0.94
The probability of not eating is 0.05

Now we need to normalise out results

Sum of the probabilities = 0.94 + 0.05 = 0.98

The likelihood of Eating = 0.94/0.98 = 95%
The likelihood of not Eating = 0.05/0.98 = 5%

We can conclude that there is high probability that the students and professor will come and eat at the cafeteria on a weekend when there is a special offer and special menu.
Posterior formula :

\[ P(A = a_k/B_1 \ldots B_n) = \frac{\prod_i P(B_i/A = a_k) \times P(A = a_k)}{\sum_j P(A = a_j) \prod_i P(B_i/A = a_j)} \]
Machine Learning

Classifiers

- Easy to implement
- Suitable for real time applications
- Highly scalable (predictors and datapoint)
- Works with continues and discrete data
- Needs less training data
Neural Networks
Machine Learning
Neural Networks
Machine Learning

Neural Networks

\[ X \rightarrow f(x) \rightarrow \text{Output} \]

Input

Activation Functions:
- Sigmoid
- Linear
- ReLU
Machine Learning
Neural Networks
Machine Learning
Neural Networks

Artificial Neural Networks
Machine Learning
Neural Networks

Deep Neural Networks
Machine Learning

Neural Networks

A mostly complete chart of

Neural Networks

©2019 Fjodor van Veen & Stefan Leijnen
asimovinstitute.org

Deep Convolutional Network (DCN)
Deconvolutional Network (DN)
Deep Convolutional Inverse Graphics Network (DCIGN)
Generative Adversarial Network (GAN)
Liquid State Machine (LSM)
Extreme Learning Machine (ELM)
Echo State Network (ESN)

Deep Residual Network (DRN)
Differentiable Neural Computer (DNC)
Neural Turing Machine (NTM)
Capsule Network (CN)
Kohonen Network (KN)
Attention Network (AN)

Markov Chain (MC)
Hopfield Network (HN)
Boltzmann Machine (BM)
Restricted BM (RBM)
Deep Belief Network (DBN)
Auto Encoder (AE)
Variational AE (VAE)
Denoising AE (DAE)
Sparse AE (SAE)

Input Cell
Backfed Input Cell
Noisy Input Cell
Hidden Cell
Probabilistic Hidden Cell
Spiking Hidden Cell
Capsule Cell
Output Cell
Match Input Output Cell
Recurrent Cell
Memory Cell
Gated Memory Cell
Kernel
Convolution or Pool

Source: The Neural Network Zoo
Machine Learning

Neural Networks

• For Example in image recognition we have CNN

Machine Learning

- Recurrent Neural Networks suitable for sound and signal
Machine Learning

- Implementing your own Architecture

![TensorFlow 2.0](image)

![Keras](image)

![PyTorch](image)