Introduction to Machine Learning

Lecture 2

Data Representation

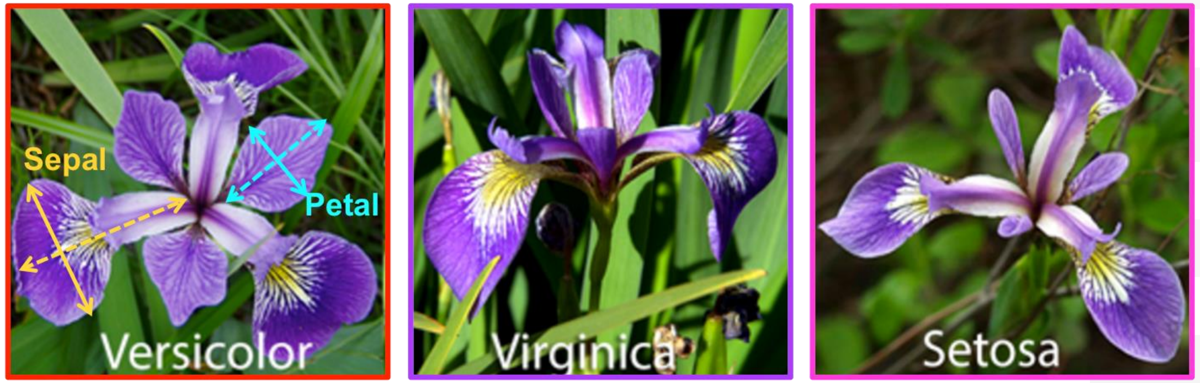
A. S. M. Sanwar Hosen

Email: sanwar@wsu.ac.kr

Date: 5 Oct., 2023

Data Representation in Machine Learning

- **Data Representation:** The main objective of ML is to build models by interpreting data. To do so, it is highly important to feed the data in a way that is readable by the model. For example, to feed data into a scikit-learn model (ML data representation library or standard), the data must be represented as a table or matrix of the required dimension.



	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	6.4	3.2	4.5	1.5	Versicolor
4	6.9	3.1	4.9	1.5	Versicolor
5	5.5	2.3	4.0	1.3	Versicolor
6	5.8	2.7	5.1	1.9	Virginica
7	7.1	3.0	5.9	2.1	Virginica
8	6.3	2.9	5.6	1.8	Virginica

Figure: An example of data representation of Iris species: this data sets consists of 3 different types of irises (Setosa, Versicolor, and Verginica) with their features sepal.length, sepal.width, petal.length, and petal.width.

Data Representation in Machine Learning

- ❑ **Data Representation:** The format of representing data fed into ML model is:

✓ **Tables or Matrix of Data:** A table/matrix of data contains the rows and columns. Each row represents an observation (an instance) and each column represents a characteristic of each observation. Most tables fed into ML problems are two-dimensional. The purpose of the dataset is to differentiate the types/classes or to predict a new output.

Example:

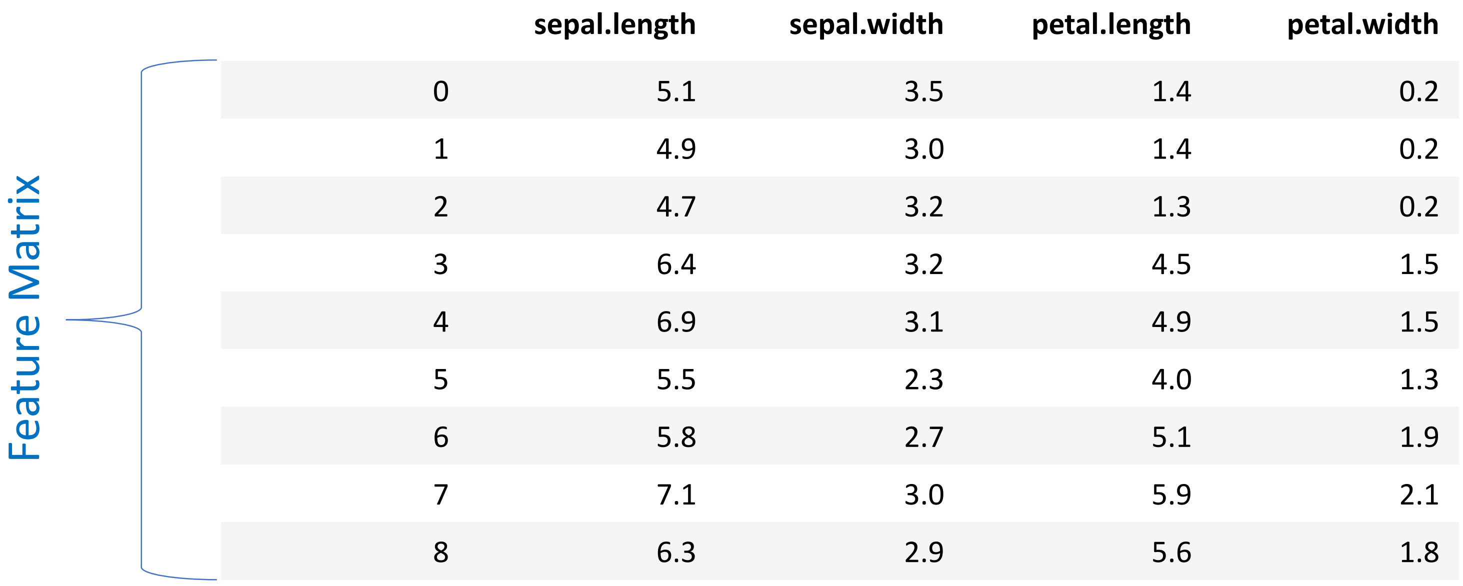
	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	6.4	3.2	4.5	1.5	Versicolor
4	6.9	3.1	4.9	1.5	Versicolor
5	5.5	2.3	4.0	1.3	Versicolor
6	5.8	2.7	5.1	1.9	Virginica
7	7.1	3.0	5.9	2.1	Virginica
8	6.3	2.9	5.6	1.8	Virginica

The example dataset of Irises plants is to differentiate from among three types of iris plants based on their characteristics. Hence, in the above table each row embodies a plant, and each column denotes the value of that feature for every plants.

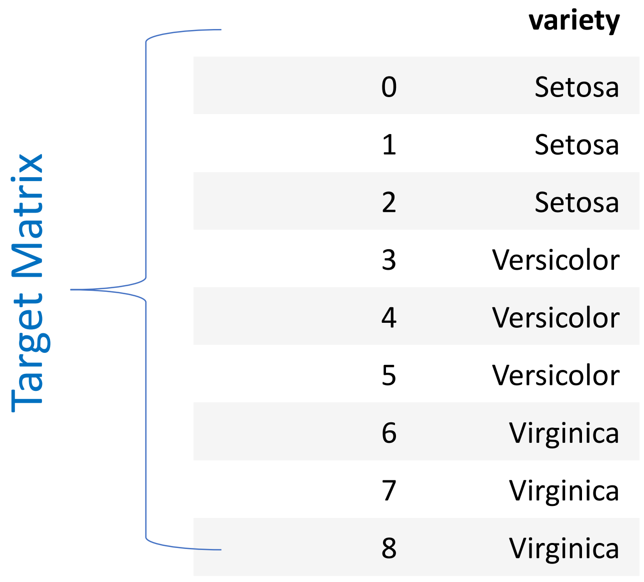
Data Representation in Machine Learning

- **Table of Data:** A table of data contains Features and Target matrices. These are as follows:
- ✓ **Feature Matrix:** The feature matrix comprises data from each instances for all features, except the target.
- ✓ **Target Matrix:** Different than the feature matrix, the target matrix is usually one-dimensional since it only carries one feature for all instances meaning that its length is of value of number instances (n). Nevertheless, there are some occasions where multiple targets are required, and the dimension of the matrix becomes $(n \times n)$.

Example:


Feature Matrix

	sepal.length	sepal.width	petal.length	petal.width
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	6.4	3.2	4.5	1.5
4	6.9	3.1	4.9	1.5
5	5.5	2.3	4.0	1.3
6	5.8	2.7	5.1	1.9
7	7.1	3.0	5.9	2.1
8	6.3	2.9	5.6	1.8

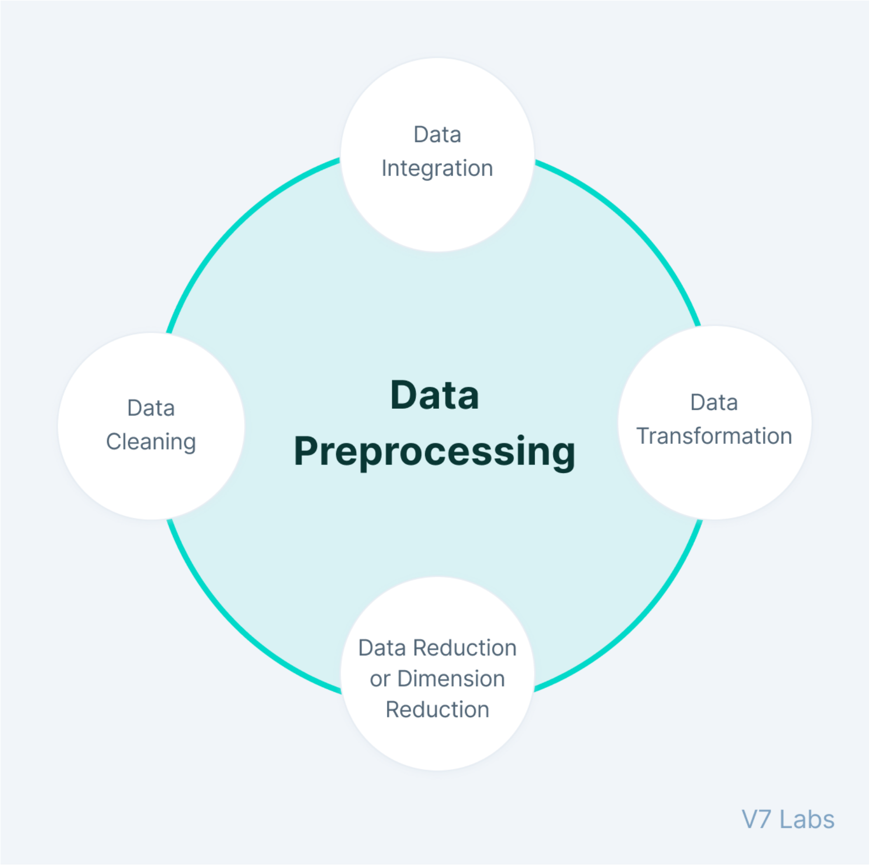

Target Matrix

	variety
0	Setosa
1	Setosa
2	Setosa
3	Versicolor
4	Versicolor
5	Versicolor
6	Virginica
7	Virginica
8	Virginica

Data Representation in Machine Learning

❑ **Data Preprocessing:** It is the process of transforming raw data into an understandable format. The steps of data preprocessing in ML are:

- ✓ Data Cleaning
- ✓ Data Integration
- ✓ Data Transformation
- ✓ Data Reduction



Data Representation in Machine Learning

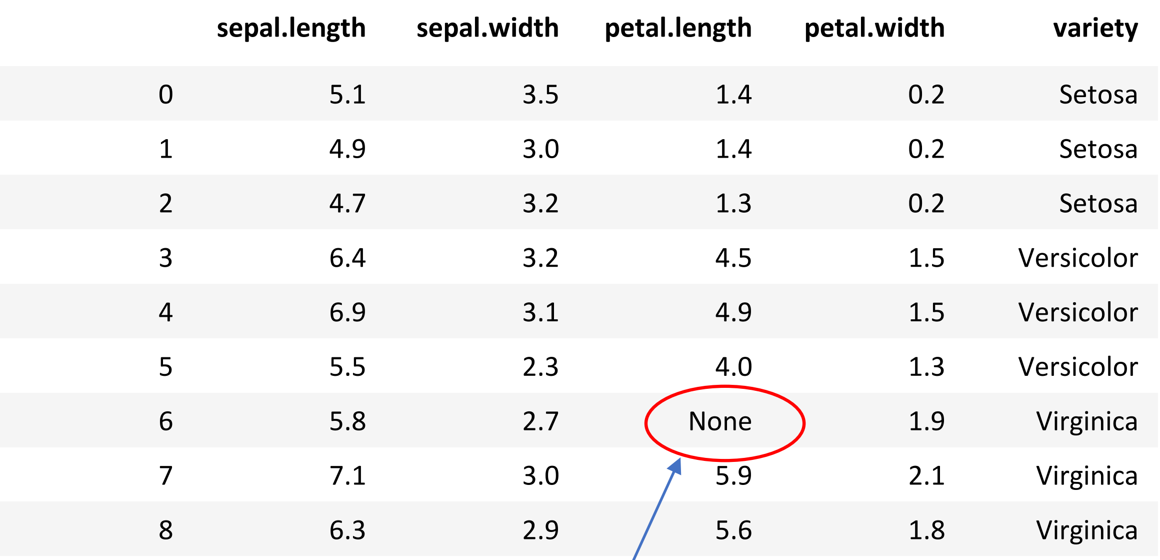
- **Data Cleaning:** This process is to clean data by filling missing values, smoothing the noisy data, resolving the inconsistency, and removing outliers.

- ✓ **Missing Values:** A value of a feature is missing/empty (exists in the form of NaN, None, Inf, -Inf) in a dataset. There are few ways to solve this issue:

  1) **Fill in the Ignore those Tuples:** This method should be considered when the dataset is huge and numerous, missing values are present within a tuple.

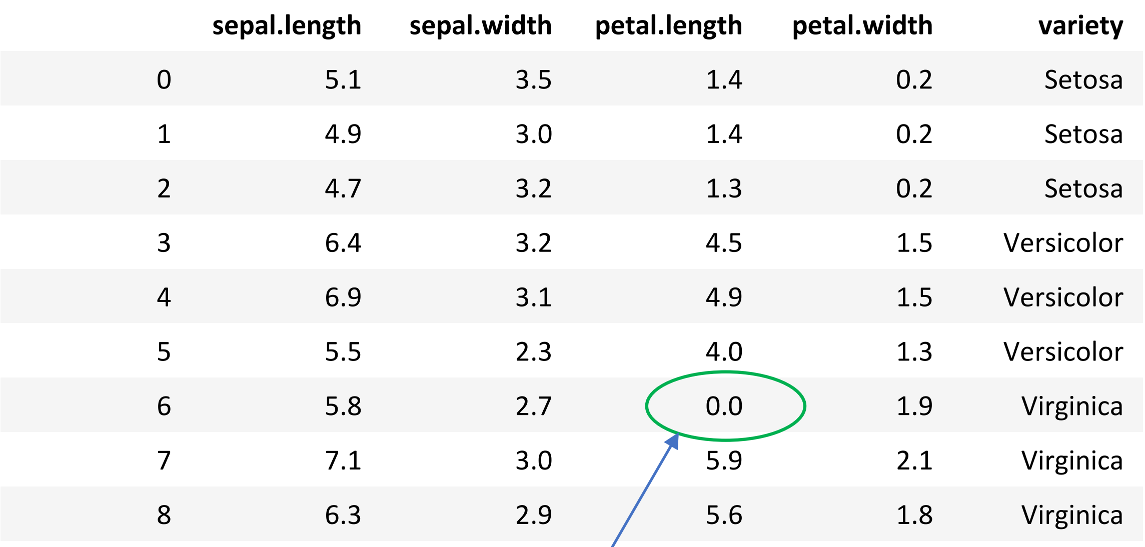
  2) **Missing Values:** Fill/replace in the missing values with known value or by 0.

Example:



	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	6.4	3.2	4.5	1.5	Versicolor
4	6.9	3.1	4.9	1.5	Versicolor
5	5.5	2.3	4.0	1.3	Versicolor
6	5.8	2.7	None	1.9	Virginica
7	7.1	3.0	5.9	2.1	Virginica
8	6.3	2.9	5.6	1.8	Virginica

Missing data



	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	6.4	3.2	4.5	1.5	Versicolor
4	6.9	3.1	4.9	1.5	Versicolor
5	5.5	2.3	4.0	1.3	Versicolor
6	5.8	2.7	0.0	1.9	Virginica
7	7.1	3.0	5.9	2.1	Virginica
8	6.3	2.9	5.6	1.8	Virginica

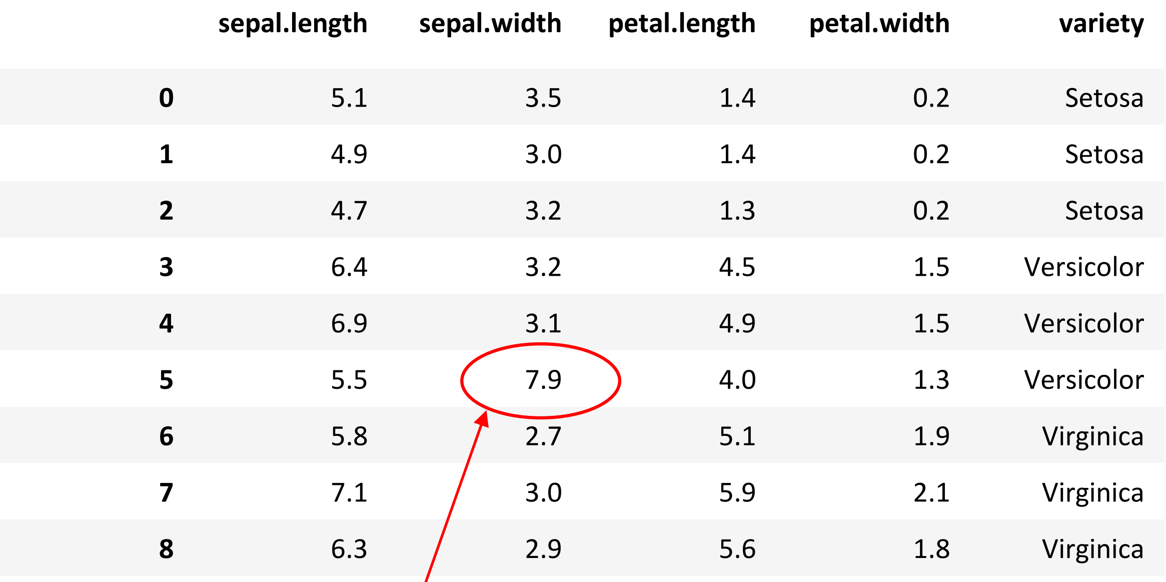
Replaced by 0

Data Representation in Machine Learning

❑ Data Cleaning

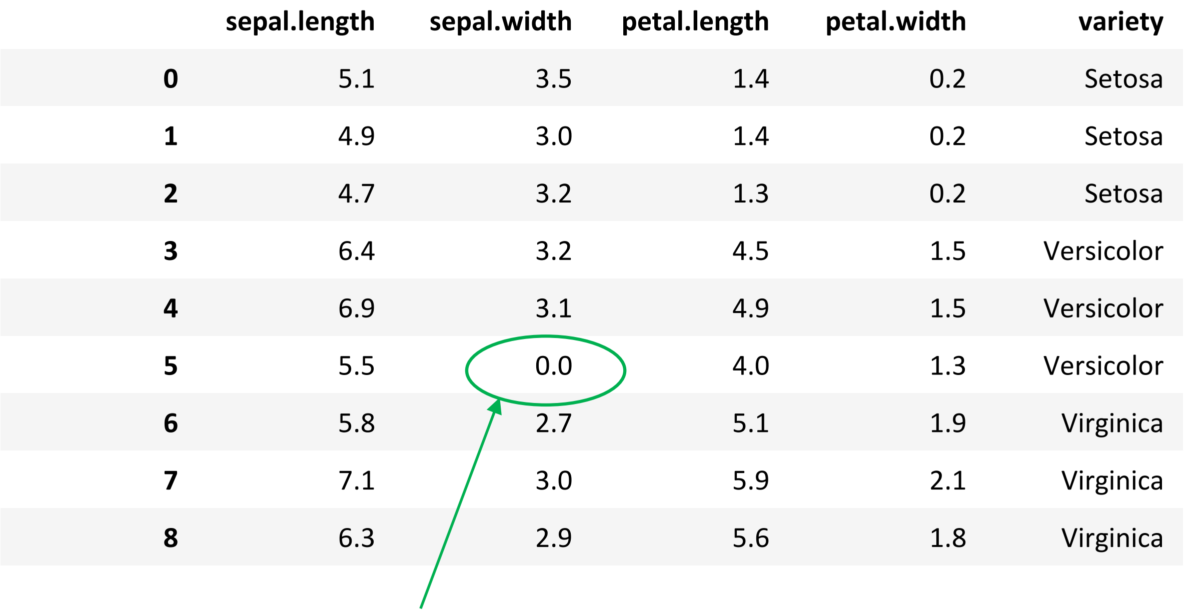
✓ **Noisy Data:** Noisy data clearing involves removing a random error or variance in a measured variable.

Example:



	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	6.4	3.2	4.5	1.5	Versicolor
4	6.9	3.1	4.9	1.5	Versicolor
5	5.5	7.9	4.0	1.3	Versicolor
6	5.8	2.7	5.1	1.9	Virginica
7	7.1	3.0	5.9	2.1	Virginica
8	6.3	2.9	5.6	1.8	Virginica

Error data



	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	6.4	3.2	4.5	1.5	Versicolor
4	6.9	3.1	4.9	1.5	Versicolor
5	5.5	0.0	4.0	1.3	Versicolor
6	5.8	2.7	5.1	1.9	Virginica
7	7.1	3.0	5.9	2.1	Virginica
8	6.3	2.9	5.6	1.8	Virginica

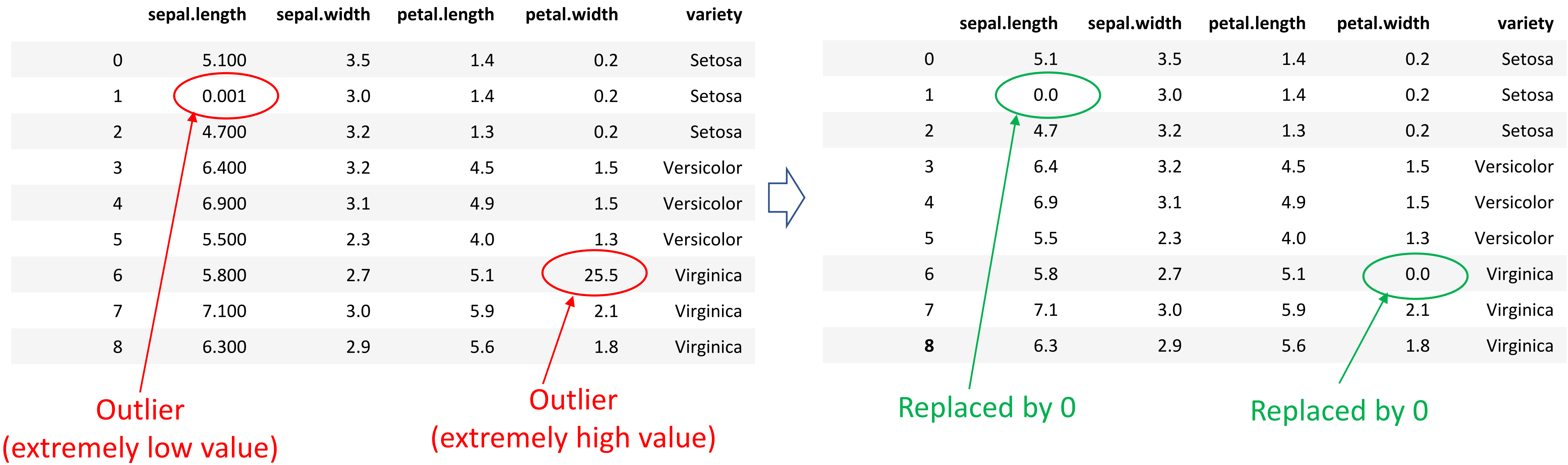
Replaced by 0

Data Representation in Machine Learning

❑ Data Cleaning

✓ **Removing Outliers:** It removes an extremely high or extremely low value in the dataset.

Example:



	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.100	3.5	1.4	0.2	Setosa
1	0.001	3.0	1.4	0.2	Setosa
2	4.700	3.2	1.3	0.2	Setosa
3	6.400	3.2	4.5	1.5	Versicolor
4	6.900	3.1	4.9	1.5	Versicolor
5	5.500	2.3	4.0	1.3	Versicolor
6	5.800	2.7	5.1	25.5	Virginica
7	7.100	3.0	5.9	2.1	Virginica
8	6.300	2.9	5.6	1.8	Virginica

Outlier
(extremely low value)

Outlier
(extremely high value)

	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	0.0	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	6.4	3.2	4.5	1.5	Versicolor
4	6.9	3.1	4.9	1.5	Versicolor
5	5.5	2.3	4.0	1.3	Versicolor
6	5.8	2.7	5.1	0.0	Virginica
7	7.1	3.0	5.9	2.1	Virginica
8	6.3	2.9	5.6	1.8	Virginica

Replaced by 0

Replaced by 0

Data Representation in Machine Learning

❑ **Data Integration:** It is a data preprocessing step used to merge the data present in multiple sources into a single data store or variables.

Example:

	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa

Data1: Setosa

	sepal.length	sepal.width	petal.length	petal.width	variety
51	6.4	3.2	4.5	1.5	Versicolor
52	6.9	3.1	4.9	1.5	Versicolor
53	5.5	2.3	4.0	1.3	Versicolor

Data2: Versicolor

	sepal.length	sepal.width	petal.length	petal.width	variety
101	5.8	2.7	5.1	1.9	Virginica
102	7.1	3.0	5.9	2.1	Virginica
103	6.3	2.9	5.6	1.8	Virginica

Data3: Virginica

	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	6.4	3.2	4.5	1.5	Versicolor
4	6.9	3.1	4.9	1.5	Versicolor
5	5.5	2.3	4.0	1.3	Versicolor
6	5.8	2.7	5.1	1.9	Virginica
7	7.1	3.0	5.9	2.1	Virginica
8	6.3	2.9	5.6	1.8	Virginica

Data integration

Dataset = Data1+Data2+Data3

Data Representation in Machine Learning

❑ **Data Transformation:** It is the process of converting data from one format to another, typically from the format of a source system into the required format of a destination system (i.e., ML model). The ways of data transformation are as follows:

- ✓ Data Generalization
- ✓ Encoding
- ✓ Data Normalization/Scaling

Data Representation in Machine Learning

❑ Data Transformation (cont..)

✓ **Data Generalization:** The term ‘generalization’ refers to a model’s ability to adapt and react appropriately to previously unseen, fresh data chosen from the same data sample as the model’s initial input. In other words, generalization assesses a model’s ability to process new data and generate accurate predictions after being trained on a training set.

Data Representation in Machine Learning

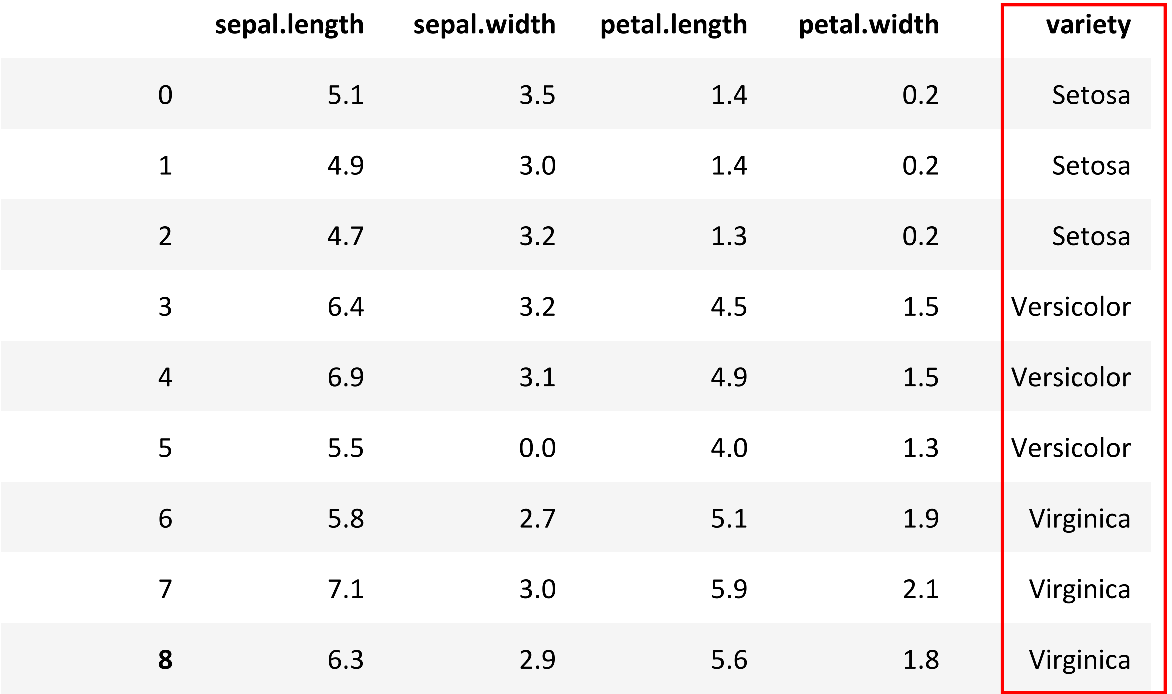
❑ Data Transformation (cont..)

✓ **Encoding:** It refers to converting the labels into a numeric form so that a machine can understand it better. There are several types of encoding as follows:

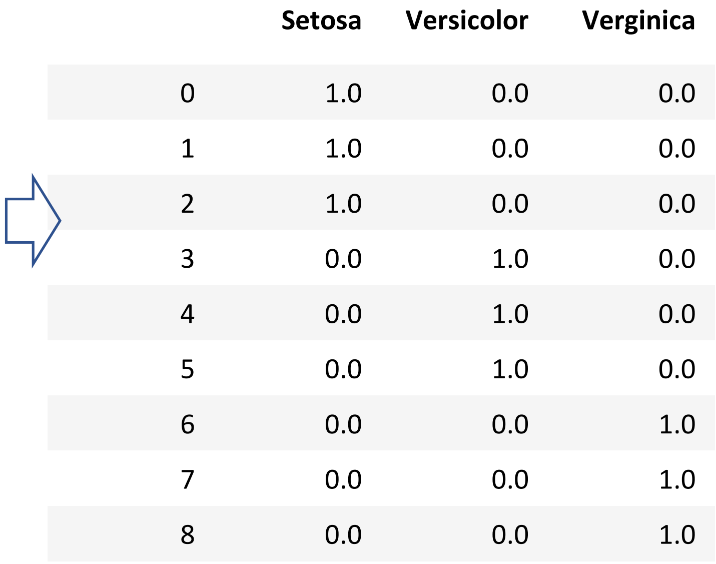
1) **OneHotEncoding:** It is the process by which categorical variables are converted into a binary vector form.

2) **Label Encoding:** It is the process by which categorical variables are converted into a number vector form.

Example:

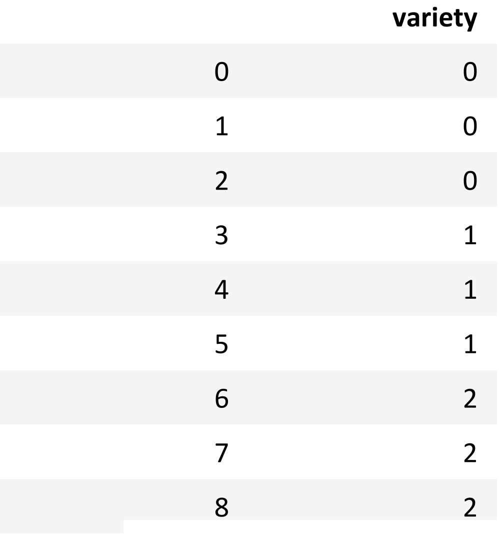


	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	6.4	3.2	4.5	1.5	Versicolor
4	6.9	3.1	4.9	1.5	Versicolor
5	5.5	0.0	4.0	1.3	Versicolor
6	5.8	2.7	5.1	1.9	Virginica
7	7.1	3.0	5.9	2.1	Virginica
8	6.3	2.9	5.6	1.8	Virginica



	Setosa	Versicolor	Verginica
0	1.0	0.0	0.0
1	1.0	0.0	0.0
2	1.0	0.0	0.0
3	0.0	1.0	0.0
4	0.0	1.0	0.0
5	0.0	1.0	0.0
6	0.0	0.0	1.0
7	0.0	0.0	1.0
8	0.0	0.0	1.0

One Hot Encoded



	variety
0	0
1	0
2	0
3	1
4	1
5	1
6	2
7	2
8	2

Label Encoded

Data Representation in Machine Learning

❑ Data Transformation (cont..)

✓ **Data Normalization/Scaling:** In this process, the numerical attributes are scaled up or down to fit within a specified range commonly between 0 to 1. Normalization can be processed in multiple ways as follows:

1) Decimal Scaling Normalization
2) Min-Max Normalization
3) Standard Normalization

Data Representation in Machine Learning

❑ Data Normalization

✓ **Decimal Scaling Normalization:** In this technique, the values of the features are moved to decimal point of values of the feature. A value x of feature X can be normalized as:

$$Normalized\ value\ of\ feature\ = \frac{x^i}{10^j}$$

Where j is the number of digits in the largest number.

Example 1:

CGPA	Formula = $\frac{x^i}{10^j}$	CGPA normalized after Decimal scaling
2	2/10	0.20
3	3/10	0.30

Why divided by 10? Here, maximum absolute value is 3 containing 1 digit, so put 1, and put one '0' after 1.

Example 2:

Salary bonus	Formula = $\frac{x^i}{10^j}$	Salary bonus normalized after Decimal scaling
400	400/1000	0.4
310	310/1000	0.31

Why divided by 1000? Here, maximum absolute value is 400 containing 3 digits, so put 1, and put three '0s' after 1.

Data Representation in Machine Learning

❑ Data Normalization (cont..)

✓ **Min-Max Normalization:** It is one of the most common ways to normalize data. For every feature, the minimum value gets transformed into a 0, the maximum value gets transformed into a 1, and every other values gets transformed into a decimal between 0 and 1. The equation of Min-Max scaler is defined below:

$$x_{scaled} = \frac{x - X_{min}}{X_{max} - X_{min}}$$

Where, x_{scaled} is the normalized values of x element in X, x is an instance value in X, X_{min} is the minimum value in X, and X_{max} is the maximum value in X.

Data Representation in Machine Learning

❑ Data Normalization (cont..)

✓ **Min-Max Normalization**

Example:

	sepal.width
0	3.5
1	3.0
2	3.2
3	3.2
4	3.1
5	2.3
6	2.7
7	3.0
8	2.9

Min-Max normalization steps:

Step 1: Find the minimum value among the values of a feature (column wise)

$$x_{min}(sepal.width) = x_{min}(3.5, 3.0, 3.2, 3.2, 3.1, 2.3, 2.7, 3.0, 2.9) = 2.3$$

Step 2: Find the maximum value among the values of the feature (column wise)

$$x_{max}(sepal.width) = x_{max}(3.5, 3.0, 3.2, 3.2, 3.1, 2.3, 2.7, 3.0, 2.9) = 3.5$$

Step 3: Find the x_{scaled} using the equation $x_{scaled} = \frac{x-X_{min}}{X_{max}-X_{min}}$

$$x0_{scaled} = \frac{x0-X_{min}}{X_{max}-X_{min}} = \frac{3.5-2.3}{3.5-2.3} = 1.000000$$

$$x1_{scaled} = \frac{x1-X_{min}}{X_{max}-X_{min}} = \frac{3.0-2.3}{3.5-2.3} = 0.583333$$

$$x2_{scaled} = \frac{x2-X_{min}}{X_{max}-X_{min}} = \frac{3.2-2.3}{3.5-2.3} = 0.750000$$

$$\mathrm{x3_{scaled}} = \frac{\mathrm{x3-X_{min}}}{\mathrm{X_{max}-X_{min}}} = \frac{3.2-2.3}{3.5-2.3} = 0.750000$$

$$\mathrm{x4_{scaled}} = \frac{\mathrm{x4-X_{min}}}{\mathrm{X_{max}-X_{min}}} = \frac{3.1-2.3}{3.5-2.3} = 0.666667$$

$$\mathrm{x5_{scaled}} = \frac{\mathrm{x5-X_{min}}}{\mathrm{X_{max}-X_{min}}} = \frac{2.3-2.3}{3.5-2.3} = 0.000000$$

$$\mathrm{x6_{scaled}} = \frac{\mathrm{x6-X_{min}}}{\mathrm{X_{max}-X_{min}}} = \frac{2.7-2.3}{3.5-2.3} = 0.333333$$

$$x7_{scaled} = \frac{x7-X_{min}}{X_{max}-X_{min}} = \frac{3.0-2.3}{3.5-2.3} = 0.583333$$

$$x8_{scaled} = \frac{x8-X_{min}}{X_{max}-X_{min}} = \frac{2.9-2.3}{3.5-2.3} = 0.500000$$

	speal.width
0	1.000000
1	0.583333
2	0.750000
3	0.750000
4	0.666667
5	0.000000
6	0.333333
7	0.583333
8	0.500000

Normalized values

Data Representation in Machine Learning

❑ Data Normalization (cont..)

✓ **Standard Normalization:** In this technique, the values are centered around the mean with a unit standard deviation. The formula of standard normalization is as follows:

$$x_{scaled} = \frac{x-\mu}{\sigma}$$

Where x_{scaled} is the standard normalized value of x element in X, x is an instance value in X, μ is the mean of X and σ is the standard deviation of X. Mean and standard deviation can be formulated as follows:

$$\text{mean } (\mu) = \frac{\sum_{i=1}^{N} x_i}{N},$$ N is the number of values in X

$$\text{variance } (\sigma^2) = \frac{\sum_{i=1}^{N} (x_i-\mu)^2}{N},$$ N is the number of values in X

$$\text{standard deviation } (\sigma) = \sqrt{\frac{\sum_{i=1}^{N} (x_i-\mu)^2}{N}},$$ N is the number of values in X

Data Representation in Machine Learning

❑ Data Normalization (cont..)

✓ **Standard Normalization**

Example:

	sepal.width
0	3.5
1	3.0
2	3.2
3	3.2
4	3.1
5	2.3
6	2.7
7	3.0
8	2.9

Standard normalization steps:

Step 1: Find the mean value of a feature

$$\mu(sepal.width) = (3.5+3.0+3.2+3.2+3.1+2.3+2.7+3.0+2.9)/9 = 2.988888$$

Step 2: Find the standard deviation of the values of the feature, $\sigma(sepal.width) =$

$$\sigma(3.5, 3.0, 3.2, 3.2, 3.1, 2.3, 2.7, 3.0, 2.9) = 0.321262$$

Step 3: Find the x_{scaled} using the equation $x_{scaled} = \frac{x-\mu}{\sigma}$

$$x0_{scaled} = \frac{x0-\mu}{\sigma} = \frac{3.5-2.988888}{0.321262} = 1.590943$$

$$x1_{scaled} = \frac{x1-\mu}{\sigma} = \frac{3.0-2.988888}{0.321262} = 0.034586$$

$$x2_{scaled} = \frac{x2-\mu}{\sigma} = \frac{3.2-2.988888}{0.321262} = 0.657129$$

$$\mathrm{x3_{scaled}} = \frac{x3-\mu}{\sigma} = \frac{3.2-2.988888}{0.321262} = 0.657129$$

$$\mathrm{x4_{scaled}} = \frac{x4-\mu}{\sigma} = \frac{3.1-2.988888}{0.321262} = 0.345857$$

$$\mathrm{x5_{scaled}} = \frac{x5-\mu}{\sigma} = \frac{2.3-2.988888}{0.321262} = -2.144315$$

$$\mathrm{x6_{scaled}} = \frac{x6-\mu}{\sigma} = \frac{2.7-2.988888}{0.321262} = -0.899229$$

$$x7_{scaled} = \frac{x7-\mu}{\sigma} = \frac{3.0-2.988888}{0.321262} = 0.034586$$

$$x8_{scaled} = \frac{x8-\mu}{\sigma} = \frac{2.9-2.988888}{0.321262} = -0.276686$$

	speal.width
0	1.590943
1	0.034586
2	0.657129
3	0.657129
4	0.345857
5	-2.144315
6	-0.899229
7	0.034586
8	-0.276686

Normalized values

Data Representation in Machine Learning

- ❑ **Data Reduction:** The size of a datasets can be too large to be handled by ML models. The data reduction process reduces the high dimensional data into low dimensional data. The following ways are applied in data reduction process:

✓ Data Cube Aggregation

✓ Dimensionality Reduction

✓ Data Compression

✓ Decentralization

✓ Numerosity Reduction

✓ Attribute Subset Selection

Data Representation in Machine Learning

- **Data Splitting:** It is a fundamental step in data preprocessing in ML. It involves dividing a dataset into multiple subsets for various purposes, typically for training and evaluating ML models. The primary reason for data splitting is to assess the performance of a model on unseen data and prevent overfitting, where a model performs well on the training data but poorly on new, unseen data. The most common data splits in ML are:

- ✓ **Training Set:** This is the largest portion of the dataset and is used to train the ML model. The model learns patterns and relationships in the data from this subset.
- ✓ **Validation Set:** Sometimes referred to as the development set or dev set, the validation set is used during model training to tune hyperparameters and monitor the model's performance. It helps in selecting the best model and prevents overfitting.
- ✓ **Test Set:** The test set is entirely separate from the training and validation sets. It is used to evaluate the final performance of the trained model. The model should not have seen this data during training or tuning, ensuring an unbiased assessment of its generalization ability.

Data Representation in Machine Learning

❑ **Lecture Overview**

✓ **Data Presentation**

- Feature Matrix

- Target Matrix

✓ **Data Preprocessing**

- Data Cleaning

(Missing Values, Noisy Data and Removing Outliers)

- Data Integration

- Data Transformation

(Data Generalization, Label Encoding (One Hot Encoding and Label Encoding) and Data Normalization (Decimal Scaling Normalization, Min-Max Normalization and Standard Normalization)

- Data Reduction

- Data Splitting