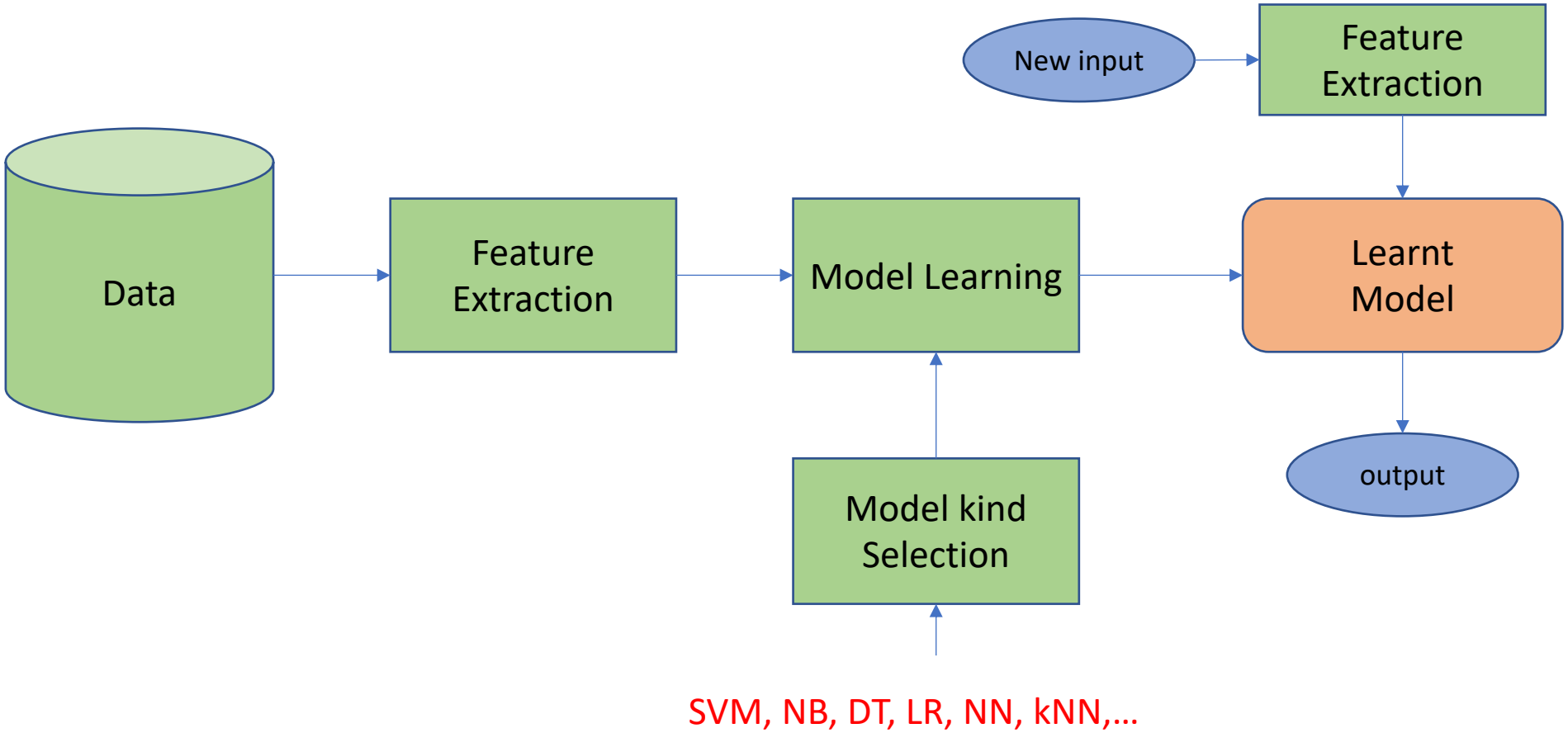
ĐẠI HỌC TÔN ĐỨC THẮNG
TON DUC THANG UNIVERSITY

Data Preparation for Machine Learning

LÊ ANH CƯỜNG

Ton Duc Thang University

A General Machine Learning Diagram



Basic Steps in Building ML Systems

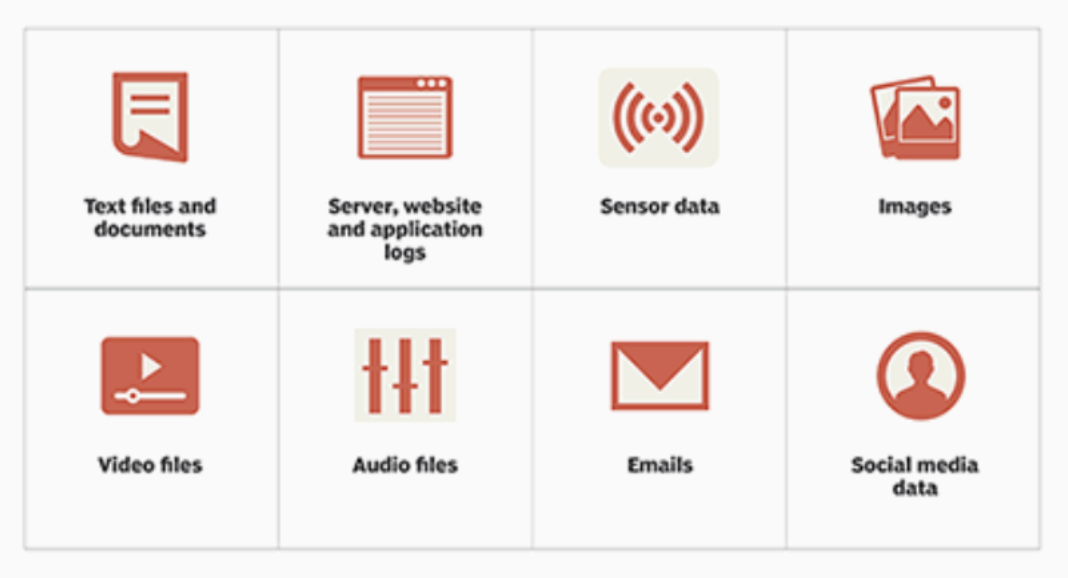
- **Step 1**: Define Problem.
- **Step 2**: Prepare Data.
- **Step 3**: Evaluate Models.
- **Step 4**: Finalize Model.

Structured data vs unstructured data

- Structured data is data that has been predefined and formatted to a set structure before being placed in data storage, which is often referred to as schema-on-write. The best example of structured data is the relational database.

	A	B	C	D	E	F	G
1	Purchase ID	Last name	First name	Birthday	Country	Date of purchase	Amount of purchase
2	1	Davidson	Michael	04/03/1986	United States	10/12/2016	37
3	2	Vito	Jim	09/01/1994	United Kingdom	02/02/2016	85
4	3	Johnson	Tom	23/08/1972	France	02/11/2016	83
5	4	Lewis	Peter	18/10/1979	Germany	22/11/2016	27
6	5	Koenig	Edward	13/05/1983	Argentina	26/03/2015	43
7	6	Preston	Jack	16/06/1991	United States	06/11/2016	77
8	7	Smith	David	11/03/1965	Canada	15/11/2016	23
9	8	Brown	Luis	03/09/1997	Australia	03/07/2015	74
10	9	Miller	Thomas	07/01/1980	Germany	07/11/2016	13
11	10	Williams	Bill	26/07/1960	United States	20/11/2015	80
12	11	Gemini	Alexia	12/09/1995	Canada	11/03/2017	35
13	12	Bond	James	25/02/1975	United Kingdom	12/08/2017	40
14	13	Burgle	Patricia	01/12/1990	United States	18/01/2015	55
15	14	Reding	Michelle	07/04/1985	Canada	23/02/2017	28
16	15	Harvey	Billy	14/07/1971	United Kingdom	12/01/2016	41
17							

- Unstructured data is information that is not arranged according to a preset data model or schema, and therefore cannot be stored in a traditional relational database or RDBMS. Text and multimedia are two common types of unstructured content.



ChatGPT

LE

structured data vs unstructured data

Structured data refers to data that is organized in a specific format, such as tables, spreadsheets, and databases, and can be easily searched, sorted, and analyzed. Examples include customer names and addresses, product inventory, and financial transactions.

Unstructured data, on the other hand, refers to data that doesn't have a clear or defined format, such as text documents, emails, audio, and video files. This type of data is often more difficult to process and analyze, as it may contain inconsistencies, subjectivity, and a lack of structure.

Data Preparation for using in Machine Learning

1. **Data collection**: Gathering relevant data from various sources such as databases, APIs, and web scraping.
2. **Data cleaning**: Removing missing values, duplicates, and outliers, and handling errors and inconsistencies in the data.
3. **Data normalization**: Scaling the data so that it has the same range of values, allowing the machine learning algorithms to treat all the features equally.
4. **Data transformation**: Converting the data into a suitable format for the machine learning algorithm, such as encoding categorical variables, transforming skewed or imbalanced data, and creating new features.
5. **Data split**: Dividing the data into training, validation, and testing sets for model evaluation and selection.
6. **Data augmentation**: Creating additional synthetic data to overcome the limitations of limited data availability and to increase the robustness of the model.

Liblaries for ML

- Numpy
- Pandas

What is Numpy

- NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

- At the core of the NumPy package, is the *ndarray* object. This encapsulates *n*-dimensional arrays of homogeneous data types, with many operations being performed in compiled code for performance

Why Numpy?

```
for (i = 0; i < rows; i++) {
  for (j = 0; j < columns; j++) {
    c[i][j] = a[i][j]*b[i][j];
  }
}
```

NumPy gives us the best of both worlds: element-by-element operations are the “default mode” when an *ndarray* is involved, but the element-by-element operation is speedily executed by pre-compiled C code. In NumPy

```
c = a * b
```

Why Numpy?

NumPy (Numerical Python) is a library commonly used for scientific computing and data analysis, and it is widely used in the machine learning community for several reasons:

- Numerical Computation: NumPy provides fast and efficient operations for numerical computation, such as vectorized operations and linear algebra.
- Array Handling: NumPy provides a multi-dimensional array data structure, known as a numpy array, which is well suited for handling large arrays of numerical data.
- Broadcasting: NumPy supports broadcasting, which allows operations to be performed on arrays with different shapes. This is particularly useful in machine learning when performing element-wise operations on arrays of different shapes.
- Interoperability: NumPy is compatible with other popular libraries used in the machine learning community, such as SciPy, Matplotlib, and scikit-learn, tensorflow,... making it easier to integrate with existing machine learning workflows.
- Performance: NumPy provides fast execution speeds, due to its use of low-level, highly optimized C code for array operations.

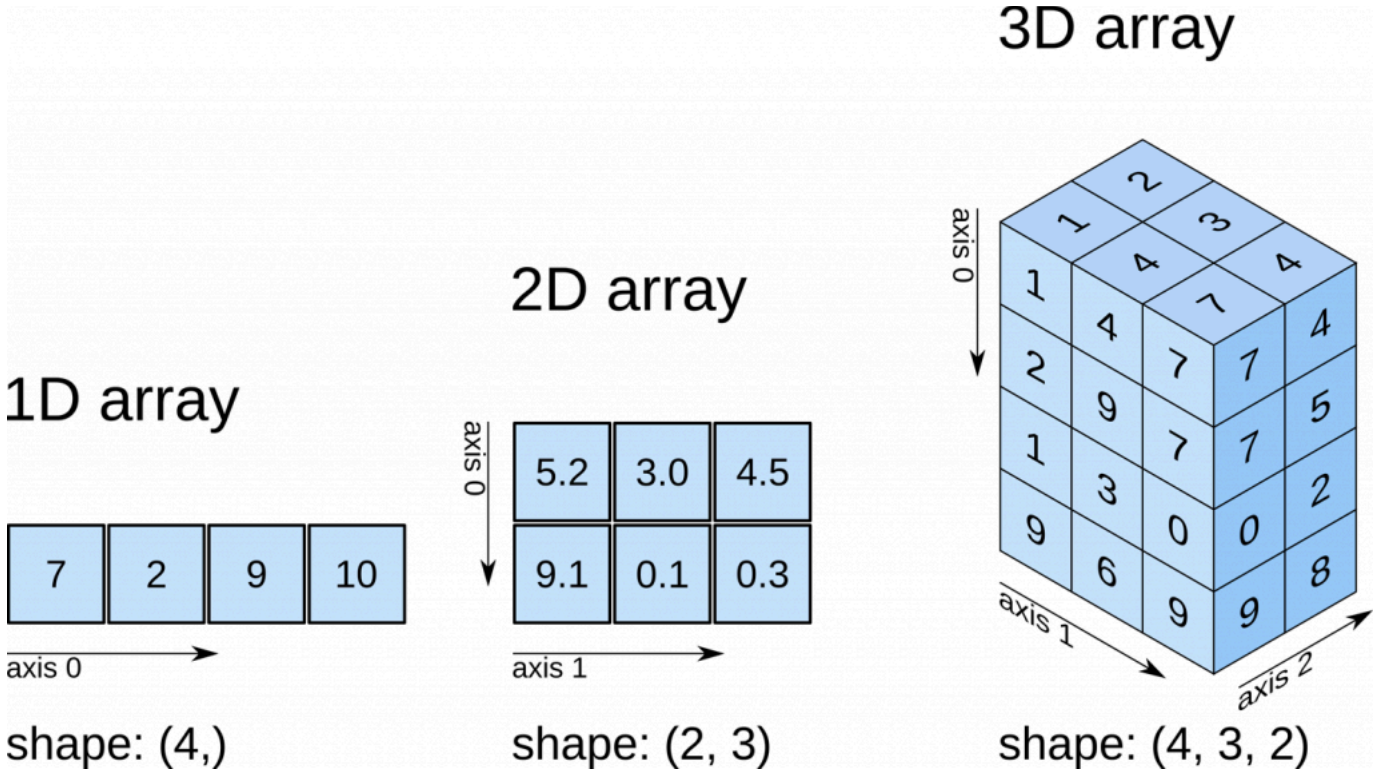
ndarray in Numpy

```
import numpy as np
arr = np.array([1, 2, 3, 5, 6])
print(arr)
```

```
[1 2 3 5 6]
```

```
import numpy as np
arr = np.array([[1, 2, 3],
                [4, 5, 6],
                [7, 8, 9]])
print(arr)
```

```
[[1 2 3]
 [4 5 6]
 [7 8 9]]
```



What is Pandas

- Pandas is widely used in the field of data analysis and data manipulation, and is an essential tool for many data scientists and machine learning practitioners.
- It provides an easy-to-use and efficient data structure (DataFrame) to store and manipulate data, as well as a variety of functions to clean, preprocess, and transform data into a suitable format for machine learning models.
- Additionally, Pandas integrates well with other popular data science libraries such as NumPy and Matplotlib, making it a versatile tool for end-to-end data analysis and machine learning workflows.

Highlights of Pandas

- A fast and efficient **DataFrame** object for data manipulation with integrated indexing;
- Tools for **reading and writing data** between in-memory data structures and different formats: CSV and text files, Microsoft Excel, SQL databases, and the fast HDF5 format;
- Intelligent **data alignment** and integrated handling of **missing data**: gain automatic label-based alignment in computations and easily manipulate messy data into an orderly form;
- Flexible **reshaping** and pivoting of data sets;
- Intelligent label-based **slicing**, **fancy indexing**, and **subsetting** of large data sets;
- Columns can be inserted and deleted from data structures for **size mutability**;
- Aggregating or transforming data with a powerful **group by** engine allowing split-apply-combine operations on data sets;
- High performance **merging and joining** of data sets;
- **Hierarchical axis indexing** provides an intuitive way of working with high-dimensional data in a lower-dimensional data structure;
- **Time series**-functionality: date range generation and frequency conversion, moving window statistics, date shifting and lagging. Even create domain-specific time offsets and join time series without losing data;
- Highly **optimized for performance**, with critical code paths written in Cython or C.
- Python with *pandas* is in use in a wide variety of **academic and commercial** domains, including Finance, Neuroscience, Economics, Statistics, Advertising, Web Analytics, and more.

Practice

- Numpy
- Pandas + KNN

Numpy

- Tạo mảng (Creating Array)
- Kích thước khuôn và ép khuôn (Shape and reshape)
- Chỉ số phần tử trong mảng (Array Indexing)
- Tách mảng con (Array Slicing)
- Lọc giá trị mảng bằng điều kiện (Array extraction by Condition)
- Tính toán trên ma trận (Operations on matrices)

Pandas

1. Read data (Đọc dữ liệu từ file, hiểu cấu trúc của dataframe)
2. Exact data (Trích xuất dữ liệu, trích chọn dữ liệu)
3. Draw graph (Vẽ đồ thị dữ liệu)
4. Transform data type (Chuyển đổi dữ liệu, xử lý dữ liệu)

Exercise

- https://www.machinelearningplus.com/python/101-numpy-exercises-python/
- https://www.machinelearningplus.com/python/101-pandas-exercises-python/