Machine Learning Process

Wu Zhuoyuan, Wu Tingan, Qiu Zixin

WHU Fintech Stu

November 2, 2022

- How machine learning from data works
  - The challenge of machine learning
  - Supervised learning
  - Unsupervised learning
  - Reinforcement learning
- The machine learning workflow
  - Workflow
  - Regression
  - Classification
  - Exploring, extracting, and engineering features
  - Selecting an ML algorithm
  - Design and tune the model

What is ML?

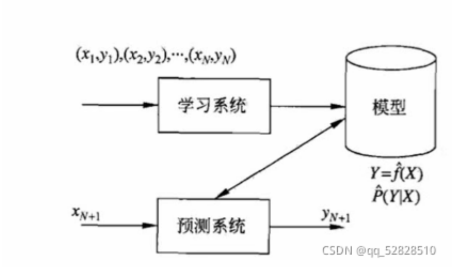
- A subfield of computer science that gives computers the ability to learn **without being explicitly programmed**. (Arthur Samuelson,1959)
- A computer program **learns from experience** with respect to a task and a performance measure of whether the performance of the task improves with experience. (Tom Mitchell,1997)

The key challenge of ML

- The **hypothesis space** limits the functions that the data can possibly represent.
- The **key challenge** is how to choose a model with a hypothesis space.
  - **large enough** to contain a solution to the learning problem
  - **small enough** to ensure reliable learning and generalization.
- The **no-free-lunch theorem (NFL)** states that a learner's hypothesis space has to be tailored to a specific task.

Supervised learning: teaching by example

- What is supervised learning?
  - to capture a functional **input-output relationship** from individual samples.
  - to apply its learning by **making valid statements** about **new data**.

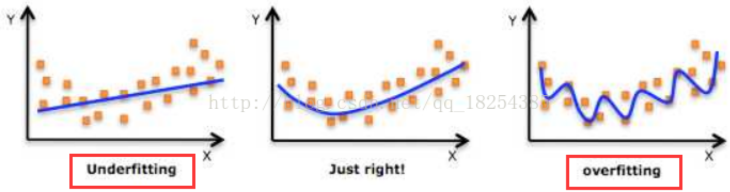


Supervised learning: teaching by example

- The **output variable**
  - We will use y_i for outcome observations $i = 1, \ldots, N$, or y for a (column) vector of outcomes.
- The **input variable**
  - We use x_i for a vector of features with observations $i = 1, \ldots, N$, or $\mathbf{X}$ in matrix notation, where each column contains a feature and each row an observation.
- The **solution** is a function $f(\hat{X})$ that represents what the model learned about the input-output relationship.

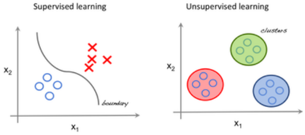
Supervised learning: teaching by example

- The **bias-variance** trade-off
  - Overly simple models will miss complex signals and deliver biased results.
  - More complex models are more likely to learn random noise particular to the training sample.



Unsupervised learning: uncovering useful patterns

- Unsupervised algorithms aim to **identify structure in the input** that permits a new representation of the information contained in the data.
- **Only observe the features** and have no specific measurements of the outcome.
- Frequently, **the measure of success** is the contribution of the result to a downstream task.



src

Unsupervised learning: Use cases

- **Grouping** securities with similar risk and return characteristics (*Chapter 13*)
- Finding a small number of risk factors driving the performance of a much larger number of securities using **principal component analysis** or **autoencoders** (*Chapter 13,19*)
- Identifying latent topics in a body of documents (for example, earnings call transcripts) that comprise the most important aspects of those documents (*Chapter 13,14*)

Cluster algorithms

- **Cluster algorithms** summarize a dataset by assigning a large number of data points to a smaller number of clusters.
- Some prominent examples:
  - K-means clustering
  - Gaussian mixture models
  - Density-based clusters
  - Hierarchical clusters

Dimensionality reduction

- Dimensionality reduction **produces new data** that captures **the most important information** contained in the source data.
- Some prominent examples:
  - Principal component analysis (PCA)
  - Manifold learning
  - Autoencoders

What is reinforcement learning?

- It centers on **an agent** that needs to choose the action that yields the **highest reward** over time, based on a set of observations that describes the current state of **the environment**.
- The **trade-off** between exploitation and exploration.

Reinforcement learning: learning by trial and error

- **Interactive**: actions taken now may influence both the environment and future rewards.
- **Dynamic**: the stream of positive and negative rewards impacts the algorithm's learning.

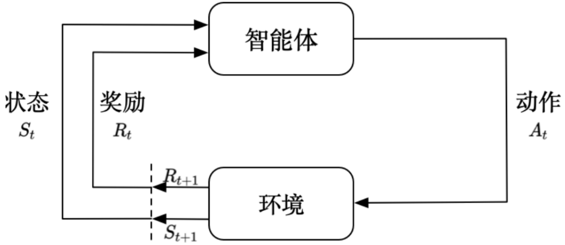


图 1.1 强化学习示意图

Reinforcement learning: learning by trial and error

- **RL vs supervised learning**
  - The **outcomes only become available over time**.
  - Aims to **find the optimal strategy** instead of the input-output relationship.
- **RL vs unsupervised learning**
  - Although with a delay, the **feedback** on the actions will be available.

Workflow

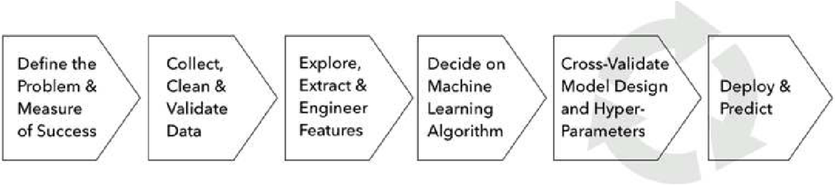


Figure 6.1: Key steps of the machine learning workflow

Framing the problem – from goals to metrics

- A continuous output variable poses a regression problem
- A categorical variable implies classification
- The special case of ordered categorical variables represents a ranking problem

Popular loss functions and error metrics

Name	Formula	scikit-learn function	Scoring parameter
Mean squared error	$\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$	mean_squared_error	neg_mean_squared_error
Mean squared log error	$\frac{1}{N}\sum_{i=1}^{N}(\ln(1+y_i) - \ln(1+\hat{y}_i))^2$	mean_squared_log_error	neg_mean_squared_log_error
Mean absolute error	$\frac{1}{N}\sum_{i=1}^{N}\lvert y_i - \hat{y}_i\rvert$	mean_absolute_error	neg_mean_absolute_error
Median absolute error	$\text{median}(\lvert y_1 - \hat{y}_1\rvert, \ldots, \lvert y_N - \hat{y}_N$	median_absolute_error	neg_median_absolute_error
Explained variance	$1 - \frac{(y - \hat{y})}{(y)}$	explained_variance_score	explained_variance
R^2 score	$1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y}_i)^2}$	r2_score	r2

Confusion matrix

		Actual (Truth)	
		Positive	**Negative**
Prediction	**Positive**	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

$$\text{Accuracy} = \frac{\text{\# Correct Predictions}}{\text{\# Cases}} = \frac{TP + TN}{TP + FP + TN + FN}$$

$$\text{True Positive Rate (Sensitivity, Recall)} = \frac{\text{\# Correct Positive Predictions}}{\text{\# Positive Cases}} = \frac{TP}{TP + FN}$$

$$\text{False Negative Rate (Miss Rate)} = 1 - \text{True Positive Rate}$$

$$\text{True Negative Rate (Specificity)} = \frac{\text{\# Correct Negative Predictions}}{\text{\# Negative Cases}} = \frac{TN}{TN + FP}$$

$$\text{False Positive Rate (Fall-Out)} = 1 - \text{True Negative Rate}$$

Figure 6.3: Confusion matrix and related error metrics

- Precision

$$Precision = \frac{TP}{TP + FP}$$

- Recall

$$Recall = \frac{TP}{TP + FN}$$

- f1 score

$$f1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

- **Receiver operating characteristics (ROC) curve** allows us to visualize, compare, and select classifiers based on their performance.
- **The area under the curve (AUC)** is defined as the area under the ROC plot that varies between 0.5 and the maximum of 1.

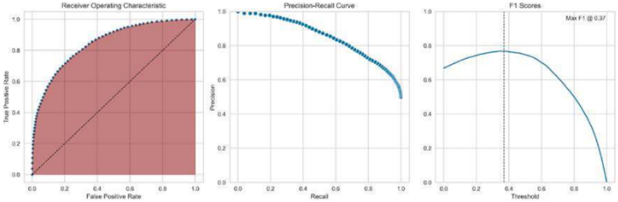


Figure 6.4: Receiver-Operating Characteristics, Precision-Recall Curve, and F1 Scores charts

Using information theory to evaluate features

- **Mutual Information**
  - The mutual information (MI) between a feature and the outcome is a measure of the mutual dependence between the two variables, which extends the notion of correlation to nonlinear relationships
  - The concept of MI is closely related to the fundamental notion of entropy of a random variable. Formally, the mutual information—I(X, Y)—of two random variables, X and Y, is defined as the following:

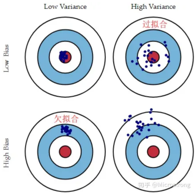
$$I(X,Y) = \int Y \int X p(x,y) \log\left(\frac{p(x,y)}{p(x)p(y)}\right)$$

Different Model Families

- **Linear Models**
  - need strong assumptions about the nature of the functional relationship between input and output variables.
- **Deep Neural Networks**
  - need fewer assumptions than linear models but will require more useful data.

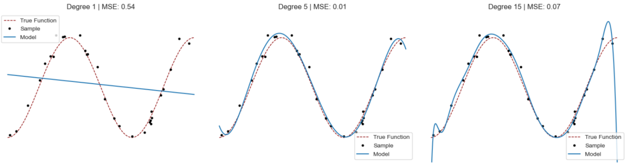
The bias-variance trade-off

- **Irreducible part**
  - the absence of relevant variables
  - natural variation
  - measurement errors
- **Reducible part**
  - bias
  - variance



Underfitting versus overfitting – a visual example

- *Illustration *
  - **A polynomial of degree 1**, but obviously wrong.
  - **A polynomial of degree 5**, approximates the true relationship reasonably well on the interval from about 0.5pi until 2.5pi.
  - **A polynomial of degree 15**, fits the small sample almost perfectly, but provides a poor estimate of the true relationship.

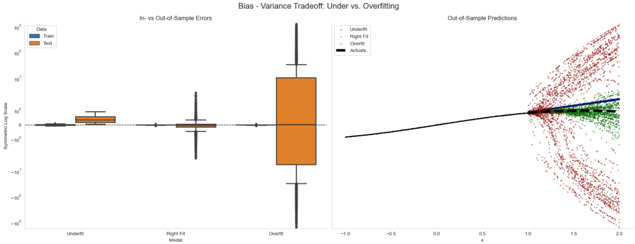


How to manage the bias-variance trade-off

- **Real Function**

$$f(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} + u$$

- **Illustration**



Learning curves