Section 1.3: Supervised Machine Learning

Monday, September 5, 2022 17:14

[00:00] Plan:

- Formal definition of supervised machine learning.
- Examples: regression, classification, ranking.

[**00:19] Examples of Machine Learning:**

Recall the previous example about price prediction pertaining to a car. We supervised the model by teaching the it about the different features to consider. We showed them the data, and set guidelines that dictated, "This is what the car's price should be." This is the essence of the idea behind machine learning.

In all these cases, we teach the machines by showing them examples and from these examples the machine is able to learn the patterns; as a result, the machine can generalize new examples. So, if we have a car for which we are not sure about pricing, the machine can use the previous examples as a guideline to generate a price.

Another example surrounded spam emails and this was meant to juxtapose rule-based systems against machine learning processes. While rule-based systems (such as programs) are effective in performing their tasks, they are not necessarily sustainable against changing externalities. With machine learning, you can incorporate the rules outlined in rule-based programming systems in addition to examples (or data) in order for it to generate a prediction based on probability. These rules would also include binary features like true (1) or false (0).

> With the spam email, you might accrue a set of arrays something like the following:

We would need to tell the models exactly what we want them to predict, in this case, we would want the models to predict if an email was either **spam** (**1**) or **not spam**(**0**).

[03:00] Explanation:

Machine learning is actually a branch of computer science and applied mathematics. It uses mathematics and statistics among other concepts to be able to extract the pattern.

Suppose, in the above figure, the column boxed in green was to convey the "**deposits**" feature; the machine will likely predict more often than not that if the word "**deposits**" shows up in an email, that the target will be categorized as "**spam**" as opposed to "**not spam**". A model tries to extract patterns using concepts based in mathematics and statistics.

The **Feature Matrix** is highlighted in the above figure, and is designated with variable, **X**. The **Feature Matrix** is a two-dimensional array.

$$\textbf{Feature Matrix} \mid\mid \textbf{Features} \equiv \mathbf{X}$$

- The two parallel lines are the logical operator, or.

A matrix is a two-dimensional array where rows are our observations or objects for which we want to predict something, and our columns are features. So in other words, referencing the above figure, it depicts each row as one email containing a set of features.

In the above example, this is a 6x6 square matrix where the ordered pair is arranged as "row x columns."

For each row, you have a corresponding matrix (read as: a vector), **y**, with our **Target Array** containing our **Target Variables**; for each row of the vector, we know if the email is spam or not spam. The array, y is a one-dimensional array.

$$\textbf{Target Array} \mid\mid \textbf{Targe Variable} \equiv \mathbf{y}$$

[06:05] Notation:

$$g(X) \approx y$$

So, this equation is sort of like the machine learning equivalent to the fundamental theorem of calculus; we have our **Feature Matrix**, **X**; when we train a model we want to converge as close as possible on the **Target Variable**, **y**. This is convergence is possible by training the function, **g**, which is our **Model**.

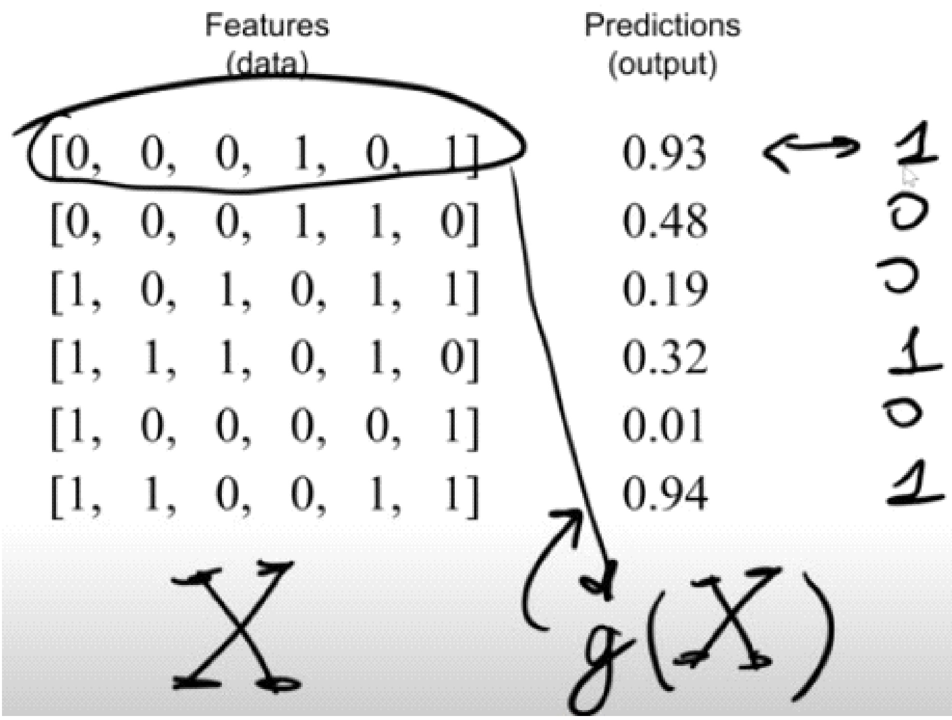
$$\textbf{Model} \equiv \ g$$

The **Model** takes in the **Feature Matrix** and it produces something that is approximately close to the **Target Variable**.

The goal of machine learning is to come up with the function **g**, that takes in the **Feature Matrix**, and produces something that is as close as possible to the **Target Variable**.

So, y could be the price of the car, **X** would be all of the data and features about the car such as mileage, make, etc. Then we want to train **g** in such a way that when it takes the **X**, it produces a **price** that is as close as possible to a car that meets the specific parameters. We want the prediction to be as close as possible.

So with respects to the **spam**/**not spam** example, the output was a probability that was as declared to be as close as possible to the [**true**/**false**] binary features established.



Based on the output generated, and based on the target variable, we have different types of supervised machine learning.

There are {3} categories of Supervised Machine Learning (SML):

[09:44] {1} Regression:

The first case, the car price prediction example, was an example of a **Regression** problem. In this case, **g** would return a number from 0 to a specified upper limit (could be infinity, could be a set value, etc.). In the car example, it was a currency value. The function, **g** outputs the **price**.

Regression problems output anything within the limit: $(-\infty, +\infty)$.

So, another example could be predicting the price of a house; this would also be an example of a regression problem. In the case of the house the **features would be characteristics such as number of bedrooms, number of bathrooms, square footage, proximity to areas of interest, etc.** that would be taken into a **model** to produce a **price for the house**. The output in these cases are always a number.

[11:15] {2}Classification:

The next type of supervised machine learning is **Classification**, where we do not output a number, but output a category. For example, if we look at a picture of a car; the picture would be the input, then the output of this would be a classification that the image is of a car.

The **spam**/**not spam** example is also a **Classification** problem. We fed some characteristics into the model, and the model used the information to classify the email as either **spam** or **not spam**. The function **g** produces the **target variable** that classifies the email accordingly based on the **features**.

There are (2) different subclasses of Classification:

- **Multiclass:** when we want to classify something into multiple categories.
  - In multiclass problems, there can be 10 categories, 1,000 categories, or as many categories as you need, ***so long as the number of categories is more than 2***.
- **Binary:** this is when we want the model to produce an output as close to either **0** or **1**.
  - The spam/**not spam** example had the function **g** output a probability between **0** and **1**. The target array consisted of **1's** and **0's** where the **model** was trying to produce an **output** as close as possible to either **1** or **0**.

[13:51] {3} Ranking:

This is usually used in cases where you want to rank system. If you want to check out a commerce website, you want to rank consumer items based on your user preferences, and the probability that you, as a consumer, will be satisfied with specific items, and sorts and ranks them accordingly. This practice is to show you what is more relevant.

[17:15] Summary:

Supervised Machine Learning is about teaching computers by showing different examples. The examples are placed in the **Feature Matrix** and run through a **model** in order to make a **prediction**.

The goal of machine learning is to come up with a function, **g** and in such a way, that when we apply **g** to the **Feature Matrix**, we get **results** in close proximity to our **target**.

Based on the **target** variable, we can have a Regression, Classification (which can be either multiclass or binary), and Ranking.

Binary classification is probably the most widely used type of supervised machine learning.

Notes by Community:

In Supervised Machine Learning (SML) there are always labels associated with certain features. The model is trained, and then it can make predictions on new features. In this way, the model is taught by certain features and targets.

> **Feature matrix (X):** made of observations (rows) and features (columns).
> **Target variable (y):** a vector with the target information we want to predict. For each row of X there's a value in y.

The model can be represented as a function **g** that takes the X matrix as a parameter and tries to predict values as close as possible to y targets. The obtention of the g function is what it is called **training**.

Types of SML problems:

- **Regression:** the output is a number (car's prize)
- **Classification:** the output is a category (spam example).
  - **Binary:** there are two categories.
  - **Multiclass problems:** there are more than two categories.
- **Ranking:** the output is the big scores associated with certain items. It is applied in recommender systems.

In summary, SML is about teaching the model by showing different examples, and the goal is to come up with a function that takes the feature matrix as a parameter and makes predictions as close as possible to the y targets.