Supervised Learning

Supervised learning's tasks are well-defined and can be applied to a multitude of scenarios—like identifying spam or predicting precipitation. Supervised learning uses historical data better than reinforcement learning.

Foundational supervised learning concepts

Supervised machine learning is based on the following core concepts:

- Data
- Model
- Training
- Evaluating
- Inference

Data

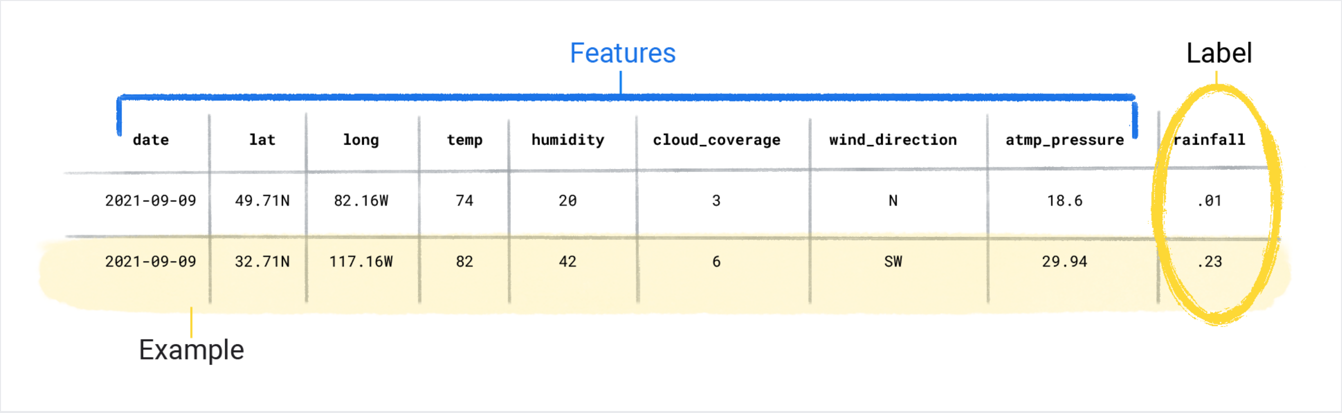
Data is the driving force of ML. Data comes in the form of words and numbers stored in tables, or as the values of pixels and waveforms captured in images and audio files. We store related data in datasets. For example, we might have a dataset of the following:

- Images of cats
- Housing prices
- Weather information

Datasets are made up of individual examples (/machine-learning/glossary#example) that contain features (/machine-learning/glossary#feature) and a label (/machine-learning/glossary#label). You could think of an example as analogous to a single row in a spreadsheet. Features are the values that a supervised model uses to predict the label. The label is the "answer," or the value we want the model to predict. In a weather model that predicts rainfall, the features could be *latitude*, *longitude*, *temperature*, *humidity*, *cloud coverage*, *wind direction*, and *atmospheric pressure*. The label would be *rainfall amount*.

Examples that contain both features and a label are called labeled examples (/machine-learning/glossary#labeled-example).

Two labeled examples

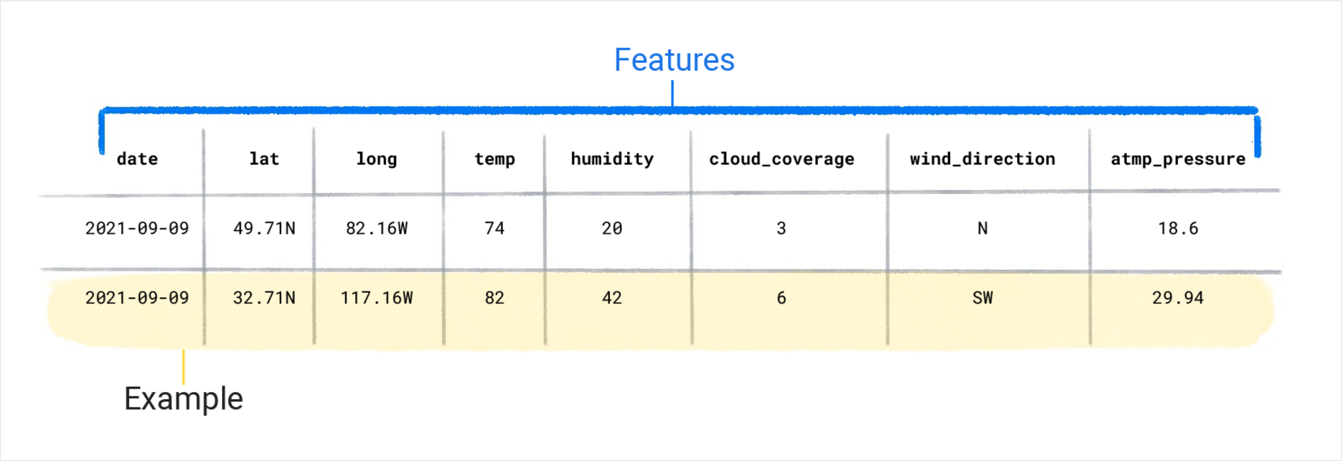


Features
Label
Example

date	lat	long	temp	humidity	cloud_coverage	wind_direction	atmp_pressure	rainfall
2021-09-09	49.71N	82.16W	74	20	3	N	18.6	.01
2021-09-09	32.71N	117.16W	82	42	6	SW	29.94	.23

In contrast, unlabeled examples contain features, but no label. After you create a model, the model predicts the label from the features.

Two unlabeled examples



Features
Example

date	lat	long	temp	humidity	cloud_coverage	wind_direction	atmp_pressure
2021-09-09	49.71N	82.16W	74	20	3	N	18.6
2021-09-09	32.71N	117.16W	82	42	6	SW	29.94

Dataset characteristics

A dataset is characterized by its size and diversity. Size indicates the number of examples. Diversity indicates the range those examples cover. Good datasets are both large and highly diverse.

Some datasets are both large and diverse. However, some datasets are large but have low diversity, and some are small but highly diverse. In other words, a large dataset doesn't guarantee sufficient diversity, and a dataset that is highly diverse doesn't guarantee sufficient examples.

For instance, a dataset might contain 100 years worth of data, but only for the month of July. Using this dataset to predict rainfall in January would produce poor predictions. Conversely, a dataset might cover only a few years but contain every month. This dataset might produce poor predictions because it doesn't contain enough years to account for variability.

Check Your Understanding

What attributes of a dataset would be ideal to use for ML?

Large size / High diversity □

Large size / Low diversity □

Small size / Low diversity □

Small size / High diversity □

A dataset can also be characterized by the number of its features. For example, some weather datasets might contain hundreds of features, ranging from satellite imagery to cloud coverage values. Other datasets might contain only three or four features, like humidity, atmospheric pressure, and temperature. Datasets with more features can help a model discover additional patterns and make better predictions. However, datasets with more features don't *always* produce models that make better predictions because some features might have no causal relationship to the label.

Model

In supervised learning, a model is the complex collection of numbers that define the mathematical relationship from specific input feature patterns to specific output label values. The model discovers these patterns through training.

Training

Before a supervised model can make predictions, it must be trained. To train a model, we give the model a dataset with labeled examples. The model's goal is to work out the best solution for predicting the labels from the features. The model finds the best solution by comparing its predicted value to the label's actual value. Based on the difference between the predicted and actual values—defined as the loss (/machine-learning/glossary#loss)—the model gradually updates its solution. In other words, the model learns the mathematical relationship between the features and the label so that it can make the best predictions on unseen data.

For example, if the model predicted `1.15 inches` of rain, but the actual value was `.75 inches`, the model modifies its solution so its prediction is closer to `.75 inches`. After the model has looked at each example in the dataset—in some cases, multiple times—it arrives at a solution that makes the best predictions, on average, for each of the examples.

The following demonstrates training a model:

1. The model takes in a single labeled example and provides a prediction.

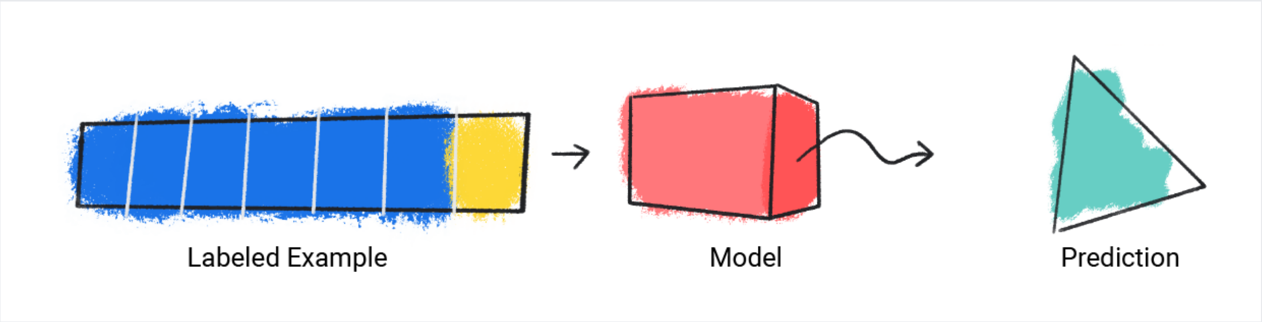


Figure 1. An ML model making a prediction from a labeled example.

2. The model compares its predicted value with the actual value and updates its solution.

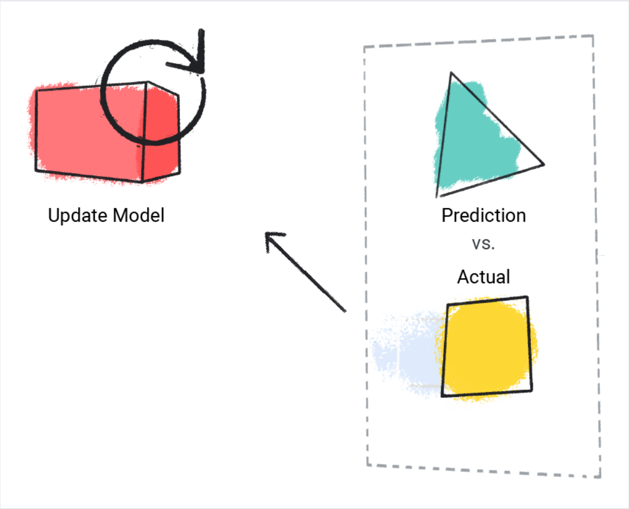


Figure 2. An ML model updating its predicted value.

3. The model repeats this process for each labeled example in the dataset.

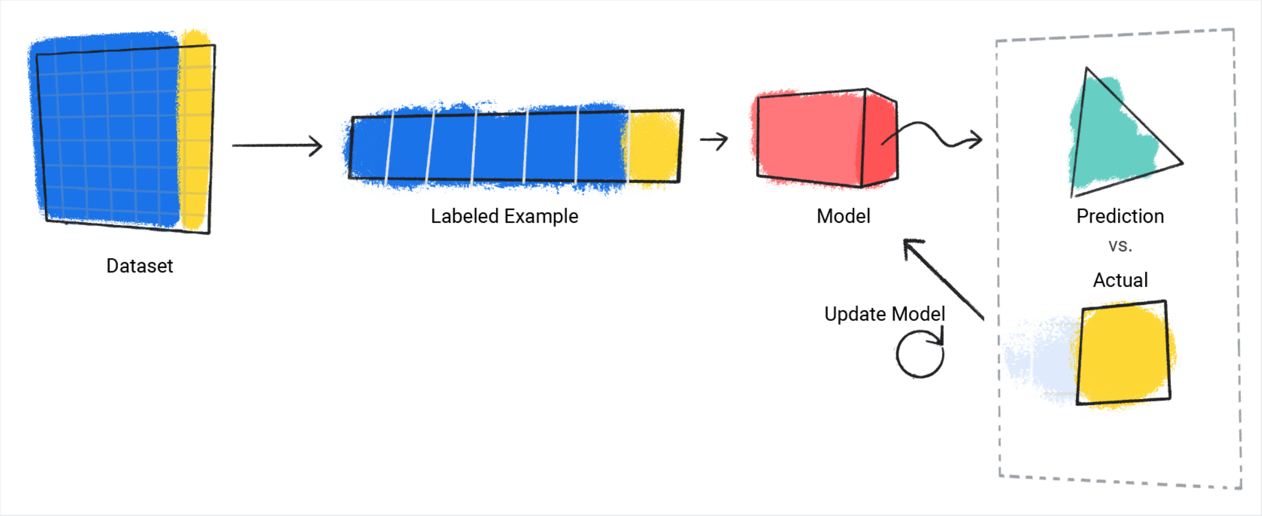


Figure 3. An ML model updating its predictions for each labeled example in the training dataset.

In this way, the model gradually learns the correct relationship between the features and the label. This gradual understanding is also why large and diverse datasets produce a better model. The model has seen more data with a wider range of values and has refined its understanding of the relationship between the features and the label.

During training, ML practitioners can make subtle adjustments to the configurations and features the model uses to make predictions. For example, certain features have more predictive power than others. Therefore, ML practitioners can select which features the model uses during training. For example, suppose a weather dataset contains`time_of_day` as a feature. In this case, an ML practitioner can add or remove `time_of_day` during training to see whether the model makes better predictions with or without it.

Evaluating

We evaluate a trained model to determine how well it learned. When we evaluate a model, we use a labeled dataset, but we only give the model the dataset's features. We then compare the model's predictions to the label's true values.

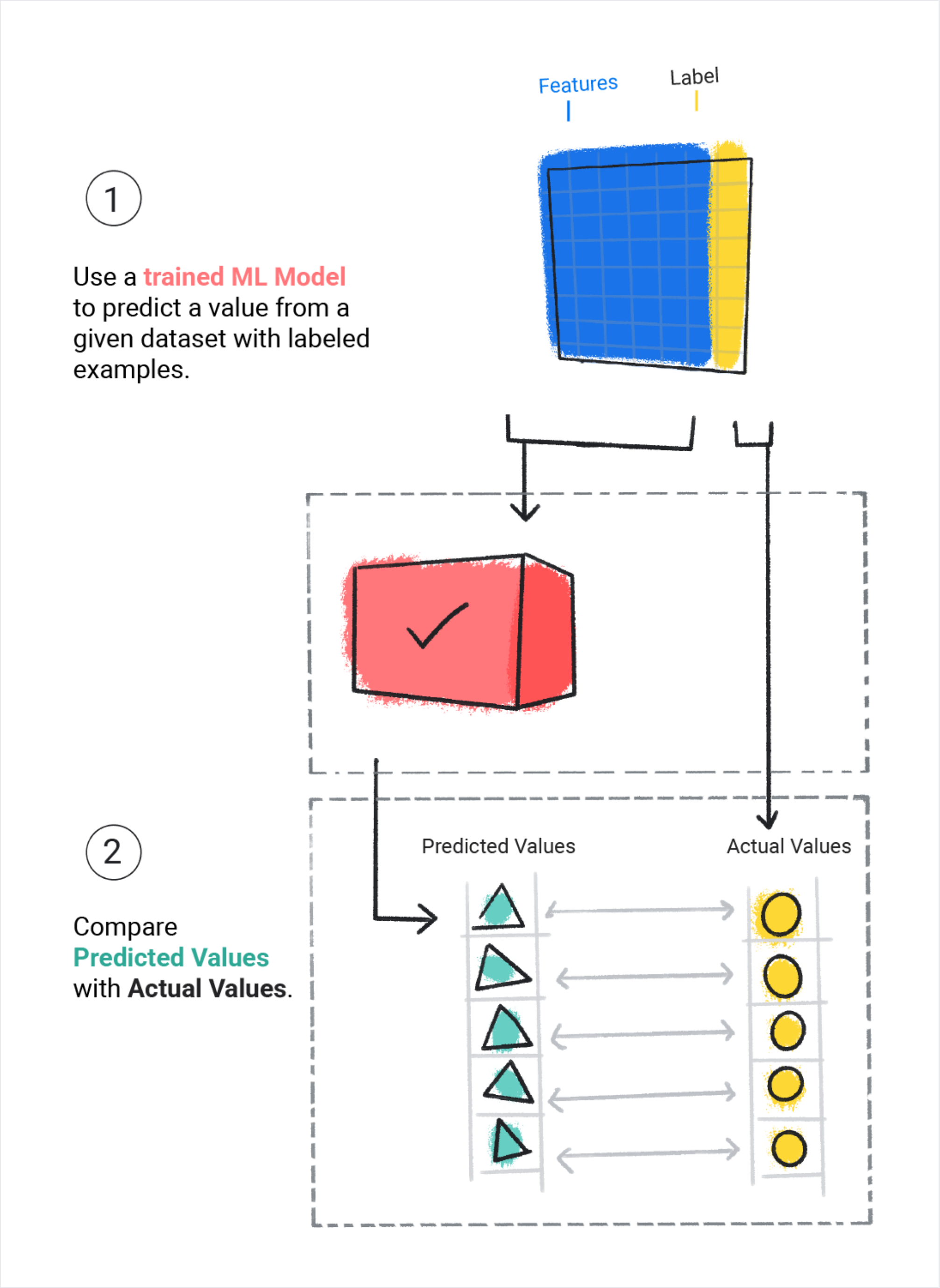

Features
Label
1
Use a trained ML Model to predict a value from a given dataset with labeled examples.
2
Compare Predicted Values with Actual Values.
Predicted Values
Actual Values

Figure 4. Evaluating an ML model by comparing its predictions to the actual values.

Depending on the model's predictions, we might do more training and evaluating before deploying the model in a real-world application.

Check Your Understanding

Why does a model need to be trained before it can make predictions?

A model doesn't need to be trained. Models are available on most computers. □

A model needs to be trained to learn the mathematical relationship between the features and the label in a dataset. □

A model needs to be trained so it won't require data to make a prediction. □

Inference

Once we're satisfied with the results from evaluating the model, we can use the model to make predictions, called inferences (/machine-learning/glossary#inference), on unlabeled examples. In the weather app example, we would give the model the current weather conditions—like temperature, atmospheric pressure, and relative humidity—and it would predict the amount of rainfall.

Key Terms:

- example (/machine-learning/glossary#example)
- feature (/machine-learning/glossary#feature)
- inference (/machine-learning/glossary#inference)
- labeled example (/machine-learning/glossary#labeled-example)
- label (/machine-learning/glossary#label)
- loss (/machine-learning/glossary#loss)
- prediction (/machine-learning/glossary#prediction)
- training (/machine-learning/glossary#training)

Previous

← What is Machine Learning? (/machine-learning/intro-to-ml/what-is-ml)

Next
Test Your Understanding (/machine-learning/intro-to-ml/understanding) →

Last updated 2023-08-03 UTC.