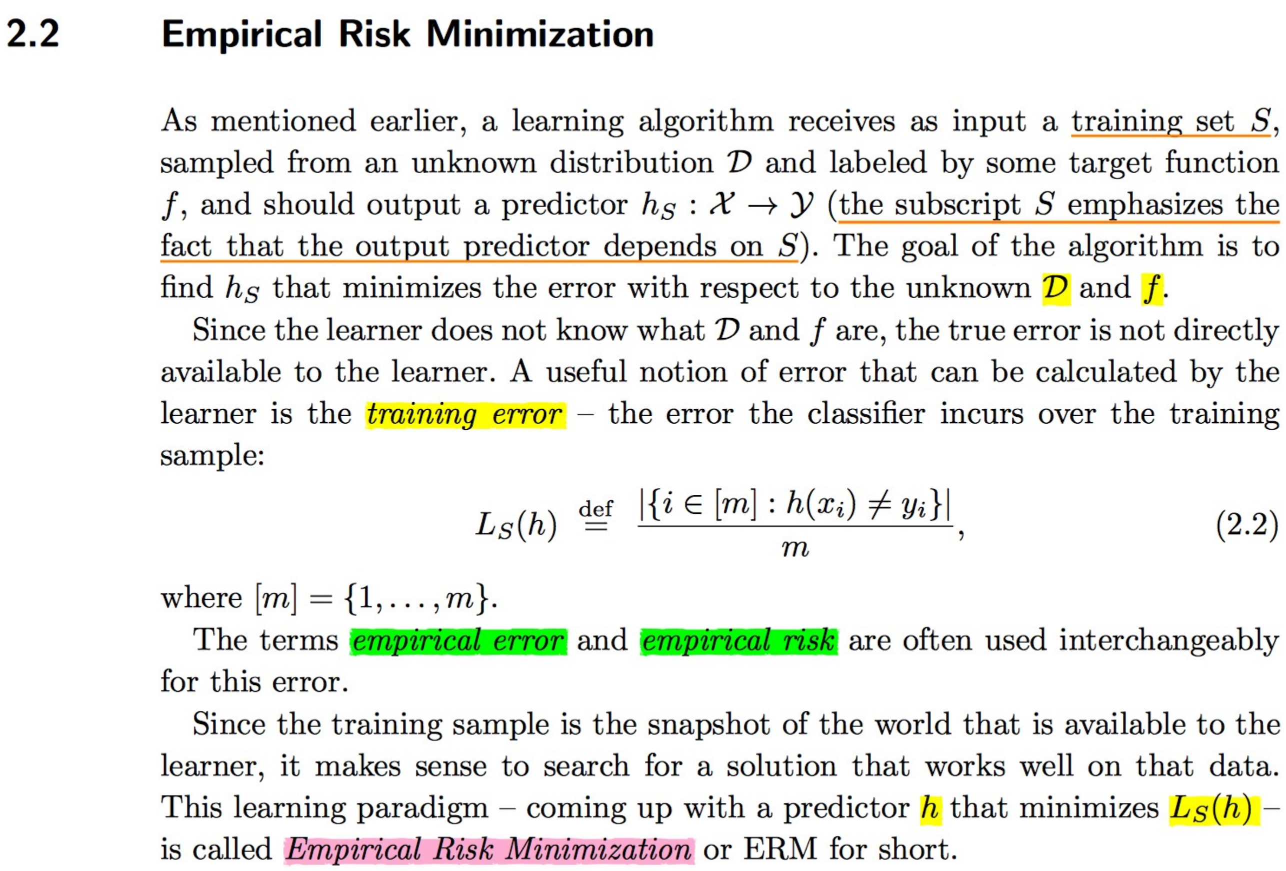


2.2 Empirical Risk Minimization

As mentioned earlier, a learning algorithm receives as input a training set S, sampled from an unknown distribution $\mathcal{D}$ and labeled by some target function f, and should output a predictor $h_S : \mathcal{X} \to \mathcal{Y}$ (the subscript S emphasizes the fact that the output predictor depends on S). The goal of the algorithm is to find h_S that minimizes the error with respect to the unknown $\mathcal{D}$ and f.

Since the learner does not know what $\mathcal{D}$ and f are, the true error is not directly available to the learner. A useful notion of error that can be calculated by the learner is the *training error* – the error the classifier incurs over the training sample:

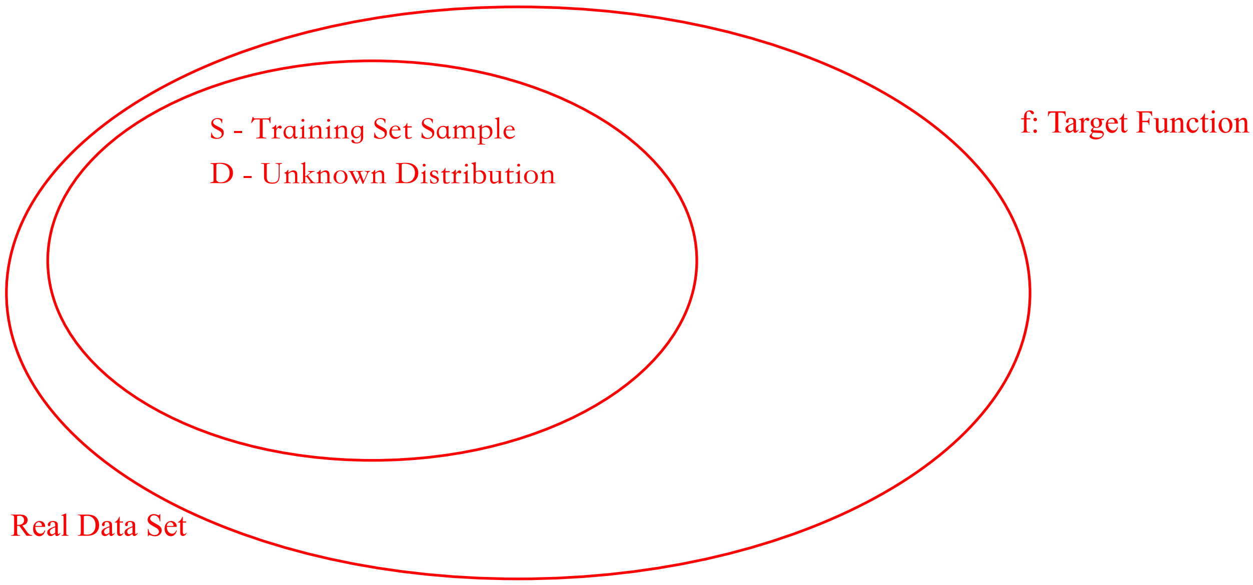
$$L_S(h) \stackrel{\text{def}}{=} \frac{|\{i \in [m] : h(x_i) \neq y_i\}|}{m}, \tag{2.2}$$

where $[m] = \{1, \ldots, m\}$.

The terms *empirical error* and *empirical risk* are often used interchangeably for this error.

Since the training sample is the snapshot of the world that is available to the learner, it makes sense to search for a solution that works well on that data. This learning paradigm – coming up with a predictor h that minimizes $L_S(h)$ – is called *Empirical Risk Minimization* or ERM for short.

Understanding Machine Learning: From Theory to Algorithms
- Shai Ben-David and Shai Shalev-Shwartz



What is Empirical Risk Minimization

Even though it has an ornate name, the underlying concept is actually quite simple and intuitive. The concept of Empirical Risk Minimization becomes relevant in the world of supervised learning. The actual goal of supervised learning is to find a model that solves a problem as opposed to finding a model that best fits the given dataset. Since we don't have every single data point that represents each class completely, we just use the next best thing available, which is a dataset that's representative of the classes. We can think of the process of supervised learning as choosing a function that achieves a given goal. We have to choose this function from a set of potential functions. Now how can we measure the effectiveness of this chosen function given that we don't know what the actual distribution looks like? Bear in mind that all the potential functions can achieve the given goal. How do we find the function that's the best representative of the true solution?

Understanding the concept of risk

To understand it, we need to talk a bit about the idea of a loss function. Given a set of inputs and outputs, this loss function measures the difference between the predicted output and the true output. But this is applicable only to the given set of inputs and outputs. We want to know what the loss is over all the possibilities. This is where "true risk" comes into picture.

True risk computes the average loss over all the possibilities. But the problem in the real world is that we don't know what "all the possibilities" would look like. In mathematical terms, we say that we don't know the true distribution over all the inputs and outputs. If we did, then we wouldn't need machine learning in the first place.

Give me an example

For example, let's say you want to build a model that can differentiate between a male and a female based on certain features. If we select 100 random people where men are really short and women are really tall, then the model might incorrectly assume that height is the differentiating feature. To build a truly accurate model, we need to gather all the men and women in the world to extract the differentiating features. Unfortunately, that's not possible! So we select a small number of people and hope that this sample is representative of the whole population.

What exactly is empirical risk minimization?

We assume that our samples come from this distribution and use our dataset as an approximation. If you compute the loss using the data points in our dataset, it's called empirical risk. It's "empirical" and not "true" because we are using a dataset that's a subset of the whole population.

When we build our learning model, we need to pick the function that minimizes the empirical risk i.e. the delta between the predicted output and the actual output for the data points in our dataset. This process of finding this function is called empirical risk minimization. Ideally, we would like to minimize the true risk. But we don't have the information that allows us to achieve that, so our hope is that this empiricial risk will almost be the same as the true empirical risk. Hence by minimizing it, we aim to minimize the true risk.

What does it depend on?

The size of the dataset has a big impact on empirical risk minimization. If we get more data, the empirical risk will approach the true risk. The complexity of the underlying distribution affects how well we can approximate it. If it's too complex, we would need more data to get a good approximation. We should also be careful about the family of functions we consider. If the size is too large, then the approximation error will be very high in certain situations. The behavior of the loss function itself can impact it. If we are not careful in choosing this function, then we might end up with very high loss values. L2 regularization is a very good example of empirical risk minimization.