

training

Author and editors

Alexander Jung*, Department of Computer Science, Aalto University, Espoo, Finland

Editors: Konstantina Olioumtsevs, Ekkehard Schnoor, Aalto University, Espoo, Finland

*Corresponding author: alex.jung@aalto.fi

Synonyms

model fitting, fitting

Abstract

Training is the application of an optimization method to a model, so that its model parameters make a loss computed from data points small. Which method applies is settled by the machine learning (ML) setting rather than by the model alone. Offline empirical risk minimization (ERM) over a fixed training set is solved by gradient descent (GD), or by stochastic gradient descent (SGD) when that training set is large. In online learning, where data points arrive one at a time, online GD takes one step per arriving data point. Unsupervised settings use alternating optimization instead: the expectation-maximization (EM) algorithm trains a Gaussian mixture model (GMM) by alternating between responsibilities and component parameters, and the k -means algorithm alternates between cluster assignments and cluster means. All of these iterate a map from the current model parameters to the next, and training stops at or near a fixed point of that map.

Definition

Fitting a straight line to a year of electricity records, adapting an hourly forecaster to each new reading as it arrives, and splitting customer profiles into groups are all called training. Each one adjusts the model parameters of a model $\mathcal{H}$ until a loss computed from data points is small. Training is therefore the application of an optimization method, and which method applies is settled by the machine learning (ML) setting rather than by the model alone.

For a parametric model, each hypothesis $h^{(\mathbf{w})}$ is fixed by a choice of model parameters $\mathbf{w}$, so training searches over $\mathbf{w}$. Nearly every method used for that search is an iteration,

$$\mathbf{w}^{(t+1)} = T(\mathbf{w}^{(t)}),$$

whose operator T is what changes from one setting to the next. Training stops at, or near, a fixed point of T , where a further iteration leaves the model parameters unchanged. The learned hypothesis $\hat{h}$ is the one that these final model parameters define. Four settings and their operators are collected in Fig. 1.

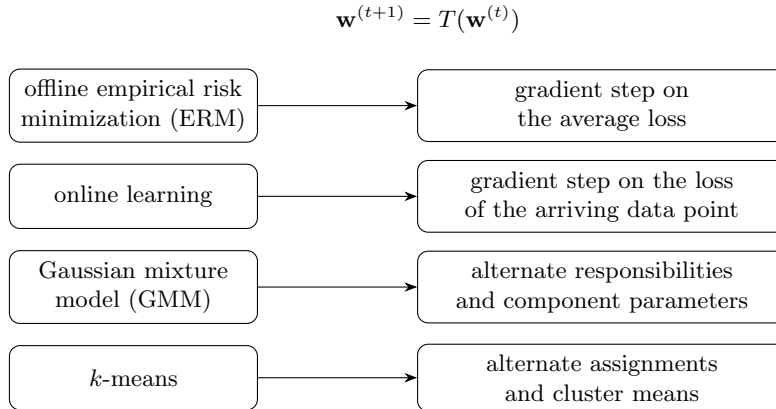


Figure 1: Training as the application of an optimization method. Each setting on the left poses a different search over the model parameters, and so is trained by a different operator T on the right. All four share the form $\mathbf{w}^{(t+1)} = T(\mathbf{w}^{(t)})$

Offline ERM minimizes the average loss $f(\mathbf{w})$ over a fixed training set, and its operator is a step of gradient descent (GD), $T(\mathbf{w}) = \mathbf{w} - \eta \nabla f(\mathbf{w})$ with step size η (Boyd and Vandenberghe, 2004, Sect. 9.3). When the training set is too large to pass through at every step, stochastic gradient descent (SGD) replaces $\nabla f(\mathbf{w})$ by the gradient of the average loss over a few data points (Bottou, 2010). In online learning there is no fixed training set: the data points arrive one at a time, and online GD takes one gradient step per arriving data point, on the loss of that data point alone (Zinkevich, 2003).

Unsupervised settings replace the single gradient step by alternating optimization, which splits the model parameters into groups and minimizes over one group while holding the others fixed. Training a GMM alternates between the responsibilities of the mixture components for each data point and the parameters of those components, which is the expectation–maximization (EM) algorithm (Dempster et al., 1977). The *k*-means algorithm alternates between assigning each data point to the nearest cluster and recomputing the cluster means (Lloyd, 1982). Both reach a fixed point of their operator, and neither is guaranteed to reach the smallest loss available in $\mathcal{H}$.

The setting therefore decides the method even when the model does not change. A shop forecasts daily bread demand from the weather with a linear model. It trains that linear model by GD on a year of past records. To let the forecaster follow a changing clientele instead, it trains the same linear model by online GD on each new day’s sales. One of the main challenges in ML is to control the discrepancy between the loss on the training set and the loss on data points outside it, and no choice of optimization method removes that challenge (Goodfellow et al., 2016).

Cross References

optimization problem, ERM, GD, SGD, online learning, EM, GMM, k -means, fixed point, model, model parameter, loss

References

1. Boyd and Vandenberghe (2004). Convex Optimization. Cambridge Univ. Press, Cambridge, U.K.
2. Goodfellow et al. (2016). Deep Learning. MIT Press, Cambridge, MA, USA.
3. Lloyd (1982). Least squares quantization in PCM. IEEE Trans. Inf. Theory, 28(2), pp. 129–137.
4. Zinkevich (2003). Online Convex Programming and Generalized Infinitesimal Gradient Ascent. In: Proc. 20th Int. Conf. Mach. Learn., pp. 928–936.
5. Bottou (2010). Large-scale machine learning with stochastic gradient descent. In: Proceedings of COMPSTAT’2010, pp. 177–186. Springer.
6. Dempster et al. (1977). Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Statist. Soc.: Ser. B (Methodological), 39(1), pp. 1–22.