

Title: Lazy learning

URL: [https://en.wikipedia.org/wiki/Lazy\\_learning](https://en.wikipedia.org/wiki/Lazy_learning)

PageID: 10747879

Categories: Category:Machine learning

Source: Wikipedia (CC BY-SA 4.0).

-----

(Not to be confused with the lazy learning regime, see Neural tangent kernel ).

In machine learning , lazy learning is a learning method in which generalization of the training data is, in theory, delayed until a query is made to the system, as opposed to eager learning , where the system tries to generalize the training data before receiving queries. [ 1 ]

The primary motivation for employing lazy learning, as in the K-nearest neighbors algorithm , used by online recommendation systems ("people who viewed/purchased/listened to this movie/item/tune also ...") is that the data set is continuously updated with new entries (e.g., new items for sale at Amazon, new movies to view at Netflix, new clips at YouTube, new music at Spotify or Pandora). Because of the continuous update, the "training data" would be rendered obsolete in a relatively short time especially in areas like books and movies, where new best-sellers or hit movies/music are published/released continuously. Therefore, one cannot really talk of a "training phase".

Lazy classifiers are most useful for large, continuously changing datasets with few attributes that are commonly queried. Specifically, even if a large set of attributes exist - for example, books have a year of publication, author/s, publisher, title, edition, ISBN, selling price, etc. - recommendation queries rely on far fewer attributes - e.g., purchase or viewing co-occurrence data, and user ratings of items purchased/viewed. [ 2 ]

#### Advantages

The main advantage gained in employing a lazy learning method is that the target function will be approximated locally, such as in the k-nearest neighbor algorithm . Because the target function is approximated locally for each query to the system, lazy learning systems can simultaneously solve multiple problems and deal successfully with changes in the problem domain. At the same time they can reuse a lot of theoretical and applied results from linear regression modelling (notably PRESS statistic ) and control. [ 3 ] It is said that the advantage of this system is achieved if the predictions using a single training set are only developed for few objects. [ 4 ] This can be demonstrated in the case of the k-NN technique, which is instance-based and function is only estimated locally. [ 5 ] [ 6 ]

#### Disadvantages

Theoretical disadvantages with lazy learning include:

The large space requirement to store the entire training dataset. In practice, this is not an issue because of advances in hardware and the relatively small number of attributes (e.g., as co-occurrence frequency) that need to be stored.

Particularly noisy training data increases the case base unnecessarily, because no abstraction is made during the training phase. In practice, as stated earlier, lazy learning is applied to situations where any learning performed in advance soon becomes obsolete because of changes in the data. Also, for the problems for which lazy learning is optimal, "noisy" data does not really occur - the purchaser of a book has either bought another book or hasn't.

Lazy learning methods are usually slower to evaluate. In practice, for very large databases with high concurrency loads, the queries are not postponed until actual query time, but recomputed in advance on a periodic basis - e.g., nightly, in anticipation of future queries, and the answers stored. This way, the next time new queries are asked about existing entries in the database, the answers are merely looked up rapidly instead of having to be computed on the fly, which would almost certainly bring a high-concurrency multi-user system to its knees.

Larger training data also entail increased cost. Particularly, there is the fixed amount of computational cost, where a processor can only process a limited amount of training data points. [ 7 ]

There are standard techniques to improve re-computation efficiency so that a particular answer is not recomputed unless the data that impact this answer has changed (e.g., new items, new purchases, new views). In other words, the stored answers are updated incrementally.

This approach, used by large e-commerce or media sites, has long been used in the Entrez portal of the National Center for Biotechnology Information (NCBI) to precompute similarities between the different items in its large datasets: biological sequences, 3-D protein structures, published-article abstracts, etc. Because "find similar" queries are asked so frequently, the NCBI uses highly parallel hardware to perform nightly recomputation. The recomputation is performed only for new entries in the datasets against each other and against existing entries: the similarity between two existing entries need not be recomputed.

#### Examples of Lazy Learning Methods

K-nearest neighbors , which is a special case of instance-based learning.

Local regression .

Lazy naive Bayes rules, which are extensively used in commercial spam detection software. Here, the spammers keep getting smarter and revising their spamming strategies, and therefore the learning rules must also be continually updated.

#### References

##### Further reading

lazy: Lazy Learning for Local Regression , R package with reference manual

"The Lazy Learning Package" . Archived from the original on 16 February 2012.

Webb G.I. (2011) Lazy Learning. In: Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning. Springer, Boston, MA

David W. Aha: Lazy learning. Kluwer Academic Publishers, Norwell 1997, ISBN 0-7923-4584-3.

Atkeson, Christopher G.; Moore, Andrew W.; Schaal, Stefan (1 February 1997). "Locally Weighted Learning for Control". *Artificial Intelligence Review* . 11 (1): 75– 113. doi : 10.1023/A:1006511328852 . S2CID 3694612 .

Bontempi, Gianluca; Birattari, Mauro; Bersini, Hugues (1999). "Lazy Learning for Local Modeling and Control Design" . *International Journal of Control* . 72 (7): 643– 658. Bibcode : 1999IJC....72..643B . doi : 10.1080/002071799220830 .

Aha, David W.; Kibler, Dennis; Albert, Marc K. (1 January 1991). "Instance-based learning algorithms" . *Machine Learning* . 6 (1): 37– 66. doi : 10.1007/BF00153759 .