

feature vector

Author and editors

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

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

*Corresponding author: alex.jung@aalto.fi

Synonyms

covariate vector, input vector, predictor vector

Abstract

A feature vector is a list of numbers describing a data point, with one entry for each of its features. A feature is an attribute of the data point that is measured or computed without human supervision, so the length of the vector is the number of such attributes. A feature transformation computes the feature vector from the data point and delivers a point of the feature space, which a hypothesis map then reads to predict the label. Non-numeric features enter as numbers: a binary attribute as 0 or 1, a categorical one through one-hot encoding. Most machine learning (ML) methods use feature vectors in a Euclidean space whose dimension is the number of features. A feature vector can more generally be an element of a Hilbert space, which is how a kernel method uses one without ever forming it, reaching it only through inner products.

Definition

A house offered for sale can be described to a machine learning (ML) method by three numbers: 85 square meters of living area, 3 rooms, and 1974 as the year of construction. That list $(85, 3, 1974)$ is the feature vector of the house. In general, a feature vector gathers into one vector the features of a data point, those attributes of the data point that are measured or computed without human supervision (Duda et al., 2001). A list of d real numbers, for a natural number d , is an element of the Euclidean space $\mathbb{R}^d$ (Strang, 2016, Sect. 1.1).

The feature vector of a data point $\mathbf{z}$ is written $\mathbf{x} = (x_1, \dots, x_d)^\top$, and its entry x_j is one feature of that data point. It is obtained from the data point by a feature transformation $\Phi: \mathcal{Z} \rightarrow \mathcal{X}$, $\mathbf{x} = \Phi(\mathbf{z})$, which views the data point itself as its raw features and delivers a point of the feature space $\mathcal{X}$ (Fig. 1). A hypothesis map reads the feature vector and delivers a prediction of the label of the data point.

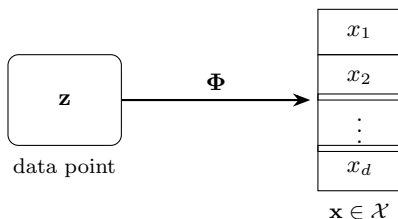


Figure 1: A feature vector $\mathbf{x}$ stacks the individual features $x_1, \dots, x_d$ of a data point $\mathbf{z}$, computed by the feature transformation Φ . It is a point in the feature space $\mathcal{X}$

Non-numeric features can also sit in the entries of a feature vector, once their values are encoded as real numbers (Hastie et al., 2009; Goodfellow et al., 2016). A binary feature, such as whether the house has a balcony, becomes 0 or 1. A categorical feature with K possible values, such as the heating type (gas, oil, or electric), is one-hot encoded: K entries of the feature vector carry the value 0 except for a single entry equal to 1, whose position names the value.

A feature transformation also reaches data points that are not records of numbers. A text becomes a feature vector by counting how often each entry of a fixed word list appears in it, which gives one entry per listed word. A digital image is an array of pixels, and the red, green and blue intensities of those pixels become one entry each. An audio recording becomes a feature vector through the values its signal takes at successive instants, one entry per instant. The feature transformation differs in each case, and so does the dimension d of the resulting feature space: the house above needs seven entries once its balcony and heating type are encoded, while an audio recording read at 256 instants needs 256 (Fig. 2).

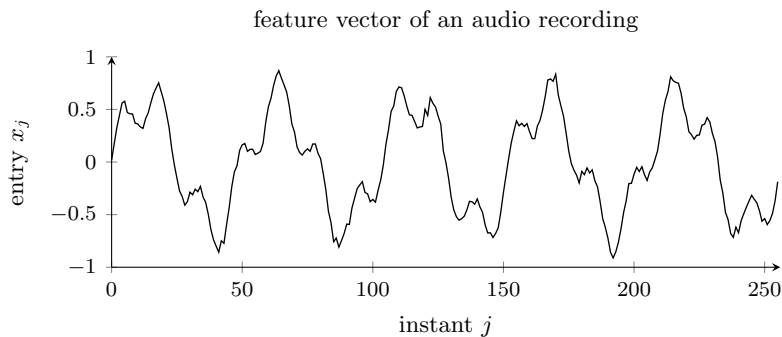


Figure 2: The feature vector of an audio recording read at 256 instants. Entry x_j is the value its signal takes at instant j , so this data point is a point of $\mathbb{R}^{256}$. Data generated by `pythondemos/featurevec.py`

Most ML methods use feature vectors in a finite-dimensional Euclidean space $\mathbb{R}^d$, whose dimension d is the number of features. More generally, a feature vector can be an element of an abstract Hilbert space $\mathcal{H}$. One way to build a feature vector of that kind starts from a kernel $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ on a given feature space $\mathcal{X}$. The kernel assigns to each $\mathbf{x} \in \mathcal{X}$ the function $k(\mathbf{x}, \cdot) : \mathcal{X} \rightarrow \mathbb{R}$, whose value at $\mathbf{x}' \in \mathcal{X}$ is $k(\mathbf{x}, \mathbf{x}')$. That function is itself a vector in a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$, a Hilbert space whose elements are functions on $\mathcal{X}$ (Aronszajn, 1950; Schölkopf and Smola, 2002). A kernel method uses these feature vectors without ever forming them explicitly: it reaches them only through their inner products, which the reproducing property reduces to kernel evaluations, $\langle k(\mathbf{x}, \cdot), k(\mathbf{x}', \cdot) \rangle_{\mathcal{H}} = k(\mathbf{x}, \mathbf{x}')$ (Schölkopf and Smola, 2002).

Cross References

feature, feature space, feature transformation, data point, vector, Euclidean space, kernel, RKHS, kernel method, Hilbert space

References

1. Duda et al. (2001). Pattern Classification. 2nd ed., Wiley, New York, NY, USA.
2. Goodfellow et al. (2016). Deep Learning. MIT Press, Cambridge, MA, USA.
3. Schölkopf and Smola (2002). Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge, MA, USA.
4. Strang (2016). Introduction to Linear Algebra. 5th ed., Wellesley-Cambridge Press, Wellesley, MA, USA.
5. Aronszajn (1950). Theory of Reproducing Kernels. Trans. Am. Math. Soc., 68(3), pp. 337–404.
6. Hastie et al. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed., Springer Science+Business Media, New York, NY, USA.