WIKIPEDIA
The Free Encyclopedia

Feature scaling

Feature scaling is a method used to normalize the range of independent variables or features of data. In data processing, it is also known as data normalization and is generally performed during the data preprocessing step.

Motivation

Since the range of values of raw data varies widely, in some machine learning algorithms, objective functions will not work properly without normalization. For example, many classifiers calculate the distance between two points by the Euclidean distance. If one of the features has a broad range of values, the distance will be governed by this particular feature. Therefore, the range of all features should be normalized so that each feature contributes approximately proportionately to the final distance.

Another reason why feature scaling is applied is that gradient descent converges much faster with feature scaling than without it.[1]

It's also important to apply feature scaling if regularization is used as part of the loss function (so that coefficients are penalized appropriately).

Methods

Rescaling (min-max normalization)

Also known as min-max scaling or min-max normalization, rescaling is the simplest method and consists in rescaling the range of features to scale the range in [0, 1] or [−1, 1]. Selecting the target range depends on the nature of the data. The general formula for a min-max of [0, 1] is given as:[2]

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

where x is an original value, x' is the normalized value. For example, suppose that we have the students' weight data, and the students' weights span [160 pounds, 200 pounds]. To rescale this data, we first subtract 160 from each student's weight and divide the result by 40 (the difference between the maximum and minimum weights).

To rescale a range between an arbitrary set of values [a, b], the formula becomes:

$$x' = a + \frac{(x - \min(x))(b - a)}{\max(x) - \min(x)}$$

where a, b are the min-max values.

Mean normalization

$$x' = \frac{x - \bar{x}}{\text{max}(x) - \text{min}(x)}$$

where x is an original value, x' is the normalized value, $\bar{x} = \text{average}(x)$ is the mean of that feature vector. There is another form of the means normalization which divides by the standard deviation which is also called standardization.

Standardization (Z-score Normalization)

In machine learning, we can handle various types of data, e.g. audio signals and pixel values for image data, and this data can include multiple dimensions. Feature standardization makes the values of each feature in the data have zero-mean (when subtracting the mean in the numerator) and unit-variance. This method is widely used for normalization in many machine learning algorithms (e.g., support vector machines, logistic regression, and artificial neural networks).[3] The general method of calculation is to determine the distribution mean and standard deviation for each feature. Next we subtract the mean from each feature. Then we divide the values (mean is already subtracted) of each feature by its standard deviation.

$$x' = \frac{x - \bar{x}}{\sigma}$$

Where x is the original feature vector, $\bar{x} = \text{average}(x)$ is the mean of that feature vector, and σ is its standard deviation.

Scaling to unit length

Another option that is widely used in machine-learning is to scale the components of a feature vector such that the complete vector has length one. This usually means dividing each component by the Euclidean length of the vector:

$$x' = \frac{x}{\|x\|}$$

In some applications (e.g., histogram features) it can be more practical to use the L_1 norm (i.e., taxicab geometry) of the feature vector. This is especially important if in the following learning steps the scalar metric is used as a distance measure. Note that this only works for $x \neq \mathbf{0}$.

Application

In stochastic gradient descent, feature scaling can sometimes improve the convergence speed of the algorithm.[4] In support vector machines,[5] it can reduce the time to find support vectors. Note that feature scaling changes the SVM result.

See also

- Normalization (statistics)
- Standard score
- fMLLR, Feature space Maximum Likelihood Linear Regression

References

1. Ioffe, Sergey; Christian Szegedy (2015). "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". arXiv:1502.03167 (https://arxiv.org/abs/1502.03167) [cs.LG (https://arxiv.org/archive/cs.LG)].
2. "Min Max normalization" (https://ml-concepts.com/2021/10/08/min-max-normalization/). *ml-concepts.com*.
3. Grus, Joel (2015). *Data Science from Scratch*. Sebastopol, CA: O'Reilly. pp. 99, 100. ISBN 978-1-491-90142-7.
4. "Gradient Descent, the Learning Rate, and the importance of Feature Scaling" (https://towardsdatascience.com/gradient-descent-the-learning-rate-and-the-importance-of-feature-scaling-6c0b416596e1).
5. Juszczak, P.; D. M. J. Tax; R. P. W. Dui (2002). "Feature scaling in support vector data descriptions". *Proc. 8th Annu. Conf. Adv. School Comput. Imaging*: 25–30. CiteSeerX 10.1.1.100.2524 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.100.2524).

Further reading

- Han, Jiawei; Kamber, Micheline; Pei, Jian (2011). "Data Transformation and Data Discretization" (https://books.google.com/books?id=pQws07tdpjoC&pg=PA111). *Data Mining: Concepts and Techniques*. Elsevier. pp. 111–118. ISBN 9780123814807.

External links

- Lecture by Andrew Ng on feature scaling (http://openclassroom.stanford.edu/MainFolder/VideoPage.php?course=MachineLearning&video=03.1-LinearRegressionII-FeatureScaling&speed=100/)

Retrieved from "https://en.wikipedia.org/w/index.php?title=Feature_scaling&oldid=1145780045"