Title: Curse of dimensionality

URL: https://en.wikipedia.org/wiki/Curse_of_dimensionality

PageID: 787776

Categories: Category:Dimension, Category:Dynamic programming, Category:Machine learning, Category:Numerical analysis

-----

Supervised learning

Unsupervised learning

Semi-supervised learning

Self-supervised learning

Reinforcement learning

Meta-learning

Online learning

Batch learning

Curriculum learning

Rule-based learning

Neuro-symbolic AI

Neuromorphic engineering

Quantum machine learning

Classification

Generative modeling

Regression

Clustering

Dimensionality reduction

Density estimation

Anomaly detection

Data cleaning

AutoML

Association rules

Semantic analysis

Structured prediction

Feature engineering

Feature learning

Learning to rank

Grammar induction

Ontology learning

Multimodal learning

Apprenticeship learning

Decision trees

Ensembles Bagging Boosting Random forest

Bagging

Boosting

Random forest

k -NN

Linear regression

Naive Bayes

Artificial neural networks

Logistic regression

Perceptron

Relevance vector machine (RVM)

Support vector machine (SVM)

BIRCH

CURE

Hierarchical

k -means

Fuzzy

Expectation–maximization (EM)

DBSCAN

OPTICS

Mean shift

Factor analysis

CCA

ICA

LDA

NMF

PCA

PGD

t-SNE

SDL

Graphical models Bayes net Conditional random field Hidden Markov

Bayes net

Conditional random field

Hidden Markov

RANSAC

k -NN

Local outlier factor

Isolation forest

Autoencoder

Deep learning

Feedforward neural network

Recurrent neural network LSTM GRU ESN reservoir computing

LSTM

GRU

ESN

reservoir computing

Boltzmann machine Restricted

Restricted

GAN

Diffusion model

SOM

Convolutional neural network U-Net LeNet AlexNet DeepDream

U-Net

LeNet

AlexNet

DeepDream

Neural field Neural radiance field Physics-informed neural networks

Neural radiance field

Physics-informed neural networks

Transformer Vision

Vision

Mamba

Spiking neural network

Memtransistor

Electrochemical RAM (ECRAM)

Q-learning

Policy gradient

SARSA

Temporal difference (TD)

Multi-agent Self-play

Self-play

Active learning

Crowdsourcing

Human-in-the-loop

Mechanistic interpretability

RLHF

Coefficient of determination

Confusion matrix

Learning curve

ROC curve

Kernel machines

Bias–variance tradeoff

Computational learning theory

Empirical risk minimization

Occam learning

PAC learning

Statistical learning

VC theory

Topological deep learning

AAAI

ECML PKDD

NeurIPS

ICML

ICLR

IJCAI

ML

JMLR

Glossary of artificial intelligence

List of datasets for machine-learning research List of datasets in computer vision and image processing

List of datasets in computer vision and image processing

Outline of machine learning

v

t

e

The curse of dimensionality refers to various phenomena that arise when analyzing and organizing data in high-dimensional spaces that do not occur in low-dimensional settings such as the three-dimensional physical space of everyday experience. The expression was coined by Richard E. Bellman when considering problems in dynamic programming . The curse generally refers to issues that arise when the number of datapoints is small (in a suitably defined sense) relative to the intrinsic dimension of the data.

Dimensionally cursed phenomena occur in domains such as numerical analysis , sampling , combinatorics , machine learning , data mining and databases . The common theme of these problems is that when the dimensionality increases, the volume of the space increases so fast that the available data become sparse. In order to obtain a reliable result, the amount of data needed often grows exponentially with the dimensionality. Also, organizing and searching data often relies on detecting areas where objects form groups with similar properties; in high dimensional data, however, all objects appear to be sparse and dissimilar in many ways, which prevents common data organization strategies from being efficient.

Domains

Combinatorics

In some problems, each variable can take one of several discrete values, or the range of possible values is divided to give a finite number of possibilities. Taking the variables together, a huge number of combinations of values must be considered. This effect is also known as the combinatorial explosion . Even in the simplest case of d {\displaystyle d} binary variables , the number of possible combinations already is 2 d {\displaystyle 2^{d}} , exponential in the dimensionality. Naively, each additional dimension doubles the effort needed to try all combinations.

Sampling

There is an exponential increase in volume associated with adding extra dimensions to a mathematical space . For example, 10 2 = 100 evenly spaced sample points suffice to sample a unit interval (try to visualize a "1-dimensional" cube, i.e. a line) with no more than 10 –2 = 0.01 distance between points; an equivalent sampling of a 10-dimensional unit hypercube with a lattice that has a spacing of 10 –2 = 0.01 between adjacent points would require 10 20 = [(10 2) 10] sample points. In general, with a spacing distance of 10 – n the 10-dimensional hypercube appears to be a factor of 10 n (10–1) = [(10 n) 10 /(10 n)] "larger" than the 1-dimensional hypercube, which is the unit interval. In the above example n = 2: when using a sampling distance of 0.01 the 10-dimensional hypercube appears to be 10 18 "larger" than the unit interval. This effect is a combination of the combinatorics problems above and the distance function problems explained below.

Optimization

When solving dynamic optimization problems by numerical backward induction , the objective function must be computed for each combination of values. This is a significant obstacle when the dimension of the "state variable" is large.

Machine learning

In machine learning problems that involve learning a "state-of-nature" from a finite number of data samples in a high-dimensional feature space with each feature having a range of possible values, typically an enormous amount of training data is required to ensure that there are several samples with each combination of values. In an abstract sense, as the number of features or dimensions grows, the amount of data we need to generalize accurately grows exponentially.

A typical rule of thumb is that there should be at least 5 training examples for each dimension in the representation. In machine learning and insofar as predictive performance is concerned, the curse of dimensionality is used interchangeably with the peaking phenomenon , which is also known as Hughes phenomenon . This phenomenon states that with a fixed number of training samples, the average (expected) predictive power of a classifier or regressor first increases as the number of dimensions or features used is increased but beyond a certain dimensionality it starts deteriorating instead of improving steadily.

Nevertheless, in the context of a simple classifier (e.g., linear discriminant analysis in the multivariate Gaussian model under the assumption of a common known covariance matrix), Zollanvari, et al. , showed both analytically and empirically that as long as the relative cumulative efficacy of an additional feature set (with respect to features that are already part of the classifier) is greater (or less) than the size of this additional feature set, the expected error of the classifier constructed using these additional features will be less (or greater) than the expected error of the classifier constructed without them. In other words, both the size of additional features and their (relative) cumulative discriminatory effect are important in observing a decrease or increase in the average predictive power.

In metric learning , higher dimensions can sometimes allow a model to achieve better performance. After normalizing embeddings to the surface of a hypersphere, FaceNet achieves the best performance using 128 dimensions as opposed to 64, 256, or 512 dimensions in one ablation study. A loss function for unitary-invariant dissimilarity between word embeddings was found to be minimized in high dimensions.

Data mining

In data mining , the curse of dimensionality refers to a data set with too many features.

Consider the first table, which depicts 200 individuals and 2000 genes (features) with a 1 or 0 denoting whether or not they have a genetic mutation in that gene. A data mining application to this data set may be finding the correlation between specific genetic mutations and creating a classification algorithm such as a decision tree to determine whether an individual has cancer or not.

A common practice of data mining in this domain would be to create association rules between genetic mutations that lead to the development of cancers. To do this, one would have to loop through each genetic mutation of each individual and find other genetic mutations that occur over a desired threshold and create pairs. They would start with pairs of two, then three, then four until they result in an empty set of pairs. The complexity of this algorithm can lead to calculating all permutations of gene pairs for each individual or row. Given the formula for calculating the permutations of n items with a group size of r is: n ! (n – r) ! {\displaystyle {\frac {n!}{(n-r)!}}} , calculating the number of three pair permutations of any given individual would be 7 988 004 000 different pairs of genes to evaluate for each individual. The number of pairs created will grow by an order of factorial as the size of the pairs increase. The growth is depicted in the permutation table (see right).

As we can see from the permutation table above, one of the major problems data miners face regarding the curse of dimensionality is that the space of possible parameter values grows exponentially or factorially as the number of features in the data set grows. This problem critically affects both computational time and space when searching for associations or optimal features to consider.

Another problem data miners may face when dealing with too many features is that the number of false predictions or classifications tends to increase as the number of features grows in the data set. In terms of the classification problem discussed above, keeping every data point could lead to a higher number of false positives and false negatives in the model.

This may seem counterintuitive, but consider the genetic mutation table from above, depicting all genetic mutations for each individual. Each genetic mutation, whether they correlate with cancer or not, will have some input or weight in the model that guides the decision-making process of the algorithm. There may be mutations that are outliers or ones that dominate the overall distribution of genetic mutations when in fact they do not correlate with cancer. These features may be working against one's model, making it more difficult to obtain optimal results.

This problem is up to the data miner to solve, and there is no universal solution. The first step any data miner should take is to explore the data, in an attempt to gain an understanding of how it can be used to solve the problem. One must first understand what the data means, and what they are trying to discover before they can decide if anything must be removed from the data set. Then they can create or use a feature selection or dimensionality reduction algorithm to remove samples or features from the data set if they deem it necessary. One example of such methods is the interquartile range method, used to remove outliers in a data set by calculating the standard deviation of a feature or occurrence.

Distance function

When a measure such as a Euclidean distance is defined using many coordinates, there is little difference in the distances between different pairs of points.

One way to illustrate the "vastness" of high-dimensional Euclidean space is to compare the proportion of an inscribed hypersphere with radius r {\displaystyle r} and dimension d {\displaystyle d} , to that of a hypercube with edges of length 2 r . {\displaystyle 2r.} The volume of such a sphere is 2 r d π d / 2 d Γ (d / 2) {\displaystyle {\frac {2r^{d}\pi ^{d/2}}{d\;\Gamma (d/2)}}} , where Γ {\displaystyle \Gamma } is the gamma function , while the volume of the cube is (2 r) d {\displaystyle (2r)^{d}} .

As the dimension d {\displaystyle d} of the space increases, the hypersphere becomes an insignificant volume relative to that of the hypercube. This can clearly be seen by comparing the proportions as the dimension d {\displaystyle d} goes to infinity:

Furthermore, the distance between the center and the corners is r d {\displaystyle r{\sqrt {d}}} , which increases without bound for fixed r.

In this sense when points are uniformly generated in a high-dimensional hypercube, almost all points are much farther than r {\displaystyle r} units away from the center. In high dimensions, the volume of the d -dimensional unit hypercube (with coordinates of the vertices ± 1 {\displaystyle \pm 1}) is concentrated near a sphere with the radius d / 3 {\displaystyle {\sqrt {d}}/{\sqrt {3}}} for large dimension d . Indeed, for each coordinate x i {\displaystyle x_{i}} the average value of x i 2 {\displaystyle x_{i}^{2}} in the cube is

The variance of x i 2 {\displaystyle x_{i}^{2}} for uniform distribution in the cube is

Therefore, the squared distance from the origin, r 2 = ∑ i x i 2 {\textstyle r^{2}=\sum _{i}x_{i}^{2}} has the average value d /3 and variance 4 d /45. For large d , distribution of r 2 / d {\displaystyle r^{2}/d} is close to the normal distribution with the mean 1/3 and the standard deviation 2 / 45 d {\displaystyle 2/{\sqrt {45d}}} according to the central limit theorem . Thus, when uniformly generating points in high dimensions, both the "middle" of the hypercube, and the corners are empty, and all the volume is concentrated near the surface of a sphere of "intermediate" radius d / 3 {\textstyle {\sqrt {d/3}}} .

This also helps to understand the chi-squared distribution . Indeed, the (non-central) chi-squared distribution associated to a random point in the interval [-1, 1] is the same as the distribution of the length-squared of a random point in the d -cube. By the law of large numbers, this distribution concentrates itself in a narrow band around d times the standard deviation squared (σ 2) of the original derivation. This illuminates the chi-squared distribution and also illustrates that most of the volume of the d -cube concentrates near the boundary of a sphere of radius σ d {\displaystyle \sigma {\sqrt {d}}} .

A further development of this phenomenon is as follows. Any fixed distribution on the real numbers induces a product distribution on points in R d {\displaystyle \mathbb {R} ^{d}} . For any fixed n , it turns out that the difference between the minimum and the maximum distance between a random reference point Q and a list of n random data points P 1 ,..., P n become indiscernible compared to the minimum distance:

This is often cited as distance functions losing their usefulness (for the nearest-neighbor criterion in feature-comparison algorithms, for example) in high dimensions. However, recent research has shown this to only hold in the artificial scenario when the one-dimensional distributions R {\displaystyle \mathbb {R} } are independent and identically distributed . When attributes are correlated, data can become easier and provide higher distance contrast and the signal-to-noise ratio was found to play an important role, thus feature selection should be used.

More recently, it has been suggested that there may be a conceptual flaw in the argument that contrast-loss creates a curse in high dimensions. Machine learning can be understood as the problem of assigning instances to their respective generative process of origin, with class labels acting as symbolic representations of individual generative processes. The curse's derivation assumes all instances are independent, identical outcomes of a single high dimensional generative process. If there is only one generative process, there would exist only one (naturally occurring) class and machine learning would be conceptually ill-defined in both high and low dimensions. Thus, the traditional argument that contrast-loss creates a curse, may be fundamentally inappropriate. In addition, it has been shown that when the generative model is modified to accommodate multiple generative processes, contrast-loss can morph from a curse to a blessing, as it ensures that the nearest-neighbor of an instance is almost-surely its most closely related instance. From this perspective, contrast-loss makes high dimensional distances especially meaningful and not especially non-meaningful as is often argued.

Nearest neighbor search

The effect complicates nearest neighbor search in high dimensional space. It is not possible to quickly reject candidates by using the difference in one coordinate as a lower bound for a distance based on all the dimensions.

However, it has recently been observed that the mere number of dimensions does not necessarily result in difficulties, since relevant additional dimensions can also increase the contrast. In addition, for the resulting ranking it remains useful to discern close and far neighbors. Irrelevant ("noise") dimensions, however, reduce the contrast in the manner described above. In time series analysis , where the data are inherently high-dimensional, distance functions also work reliably as long as the signal-to-noise ratio is high enough.

k -nearest neighbor classification

Another effect of high dimensionality on distance functions concerns k -nearest neighbor (k -NN) graphs constructed from a data set using a distance function. As the dimension increases, the indegree distribution of the k -NN digraph becomes skewed with a peak on the right because of the emergence of a disproportionate number of hubs , that is, data-points that appear in many more k -NN lists of other data-points than the average. This phenomenon can have a considerable impact on various techniques for classification (including the k -NN classifier), semi-supervised learning , and clustering , and it also affects information retrieval .

Anomaly detection

In a 2012 survey, Zimek et al. identified the following problems when searching for anomalies in high-dimensional data:

Concentration of scores and distances: derived values such as distances become numerically similar

Irrelevant attributes: in high dimensional data, a significant number of attributes may be irrelevant

Definition of reference sets: for local methods, reference sets are often nearest-neighbor based

Incomparable scores for different dimensionalities: different subspaces produce incomparable scores

Interpretability of scores: the scores often no longer convey a semantic meaning

Exponential search space: the search space can no longer be systematically scanned

Data snooping bias: given the large search space, for every desired significance a hypothesis can be found

Hubness: certain objects occur more frequently in neighbor lists than others.

Many of the analyzed specialized methods tackle one or another of these problems, but there remain many open research questions.

Blessing of dimensionality

Despite the expected "curse of dimensionality" difficulties, common-sense heuristics based on the most straightforward methods "can yield results which are almost surely optimal" for high-dimensional problems. The term "blessing of dimensionality" was introduced in the late 1990s. Donoho in his "Millennium manifesto" explained why he thinks the "blessing of dimensionality" will form a basis of future data mining. The effects of the blessing of dimensionality were discovered in many applications and found their foundation in the concentration of measure phenomena . One example of the blessing of dimensionality phenomenon is linear separability of a random point from a large finite random set with high probability even if this set is exponentially large: the number of elements in this random set can grow exponentially with dimension. Moreover, this linear functional can be selected in the form of the simplest linear Fisher discriminant . This separability theorem was proven for a wide class of probability distributions: general uniformly log-concave distributions, product distributions in a cube and many other families (reviewed recently in).

"The blessing of dimensionality and the curse of dimensionality are two sides of the same coin." For example, the typical property of essentially high-dimensional probability distributions in a high-dimensional space is: the squared distance of random points to a selected point is, with high probability, close to the average (or median) squared distance. This property significantly simplifies the expected geometry of data and indexing of high-dimensional data (blessing), but, at the same time, it makes the similarity search in high dimensions difficult and even useless (curse).

Zimek et al. noted that while the typical formalizations of the curse of dimensionality affect i.i.d. data, having data that is separated in each attribute becomes easier even in high dimensions, and argued that the signal-to-noise ratio matters: data becomes easier with each attribute that adds signal, and harder with attributes that only add noise (irrelevant error) to the data. In particular for unsupervised data analysis this effect is known as swamping.

See also

Bellman equation

Clustering high-dimensional data

Concentration of measure

Dimensionality reduction

Dynamic programming

Fourier-related transforms

Grand Tour

Linear least squares

Model order reduction

Multilinear PCA

Multilinear subspace learning

Principal component analysis

Singular value decomposition

References