### Dimensionality Reduction

### By-Krashna Gurme



### ****Definition****

Dimensionality Reduction aka dimension reduction is the transformation of data from a high-dimensional space into a low-dimensional space. So that low-dimensional representation contains some meaningful properties of the original data.

**Types of Dimensional Dataset**

1. **Low Dimensional Dataset:** The number of dimensions/features you have for each sample. Example - Iris flower dataset contains only 4 features which are Sepal length, Sepal width, Petal length, and Petal width so considered as a low dimensional dataset.
2. **High Dimensional Dataset:** The number of dimensions is staggeringly high. Example - Data on the health status of patients can be high-dimensional (blood status, immune system status, alcohol level, nutrition, treatments, any other drug consumption, diagnosed diseases, etc. The more the number of dimensions then more the calculation.

Basically, there are two approaches to reduce the dimensionality of the data.

 A. Linear Approach and

 B. Non-Linear Approach

These approaches further get divided into two categories,

1. Feature Selection
2. Feature Extraction aka Feature Projection

### ****Feature Selection (Attribute Selection)****

“The process of selecting subset of relevant features for processing, without any transformation”. In more general words feature selection is a technique where you manually or automatically select those features, which can significantly contribute to model building process or predicting a variable/output.

These [Feature selection](https://en.wikipedia.org/wiki/Feature_selection) approaches try to find a “subset of the input variables” (aka features or attributes) for model building. Feature Selection techniques often used in domains where there are many features and comparatively few samples or data points.

E.g. Analysis of written texts and DNA microarray data. Where plenty’s of features but very few (10 to 100) samples present.

Now, we will discuss three strategies of feature selection.

#### a. Filter Strategy-

 “Filter methods measure the importance of features by their correlation with dependent variable i.e. Predicting Variable.”

Here, features with higher variance may get selected assuming they may contain useful data.

The filter methods include the following examples,

* **Information Gain** — This gives the information for the attributes given in a set so that we can discriminate between the different classes of features.
* **Chi-square Test**— It is used for testing the independence of two events, by obtaining the observed value and the expected value. Further measuring how these two events deviate from each other.
* **Correlation Coefficient**— It’s a statistical measure of the strength of the relationship b/w the relative movements of two variables. The value ranges from -1.0 to +1.0.
* **Variance Threshold**— It removes the feature which having variation below a certain threshold.

#### b. Wrapper Strategy-

 “Wrapper methods measure the usefulness of a subset of feature by actually training a model on it.”

This is a search guided by the accuracy method. Meaning, new features are getting added to boost the performance of the model.

These include,

* **Recursive Feature Elimination (RPE)** — It fits a model and removes the weakest feature unless the specified number of features are satisfied. Each time when features are checked and discarded, there is a ranking of features happens according to the elimination process.
* **Sequential Feature Selection Algorithms** — It includes two methods, one is Sequential Forward Selection(SFS) and other Sequential Backward Selection(SBS). Forward selection starts with no feature at the start and keeps on adding a feature so as to improve the model performance. backward selection does exactly the opposite of what FS does. It removes the most irrelevant features and checks the model performance at every iteration.

#### c. Embedded Strategy-

 “Selected features are added or removed while building the model based on prediction errors”.

Embedded methods learn which features best contribute to the accuracy of the model. These methods possess inbuilt variable selection methods.

Examples,

* **L1 (Least Absolute Shrinkage and Selection Operator (LASSO)) Regularization aka Lasso Regression**— It is used to add a penalty term against the complexity for reducing the problem of overfitting. In general, regularization is adding information in order to solve the problem of overfitting.
* **L2 (Ridge) Regularization aka Ridge Regression —**It calculates the least-squares error of the magnitudes of coefficient, but is more sensitive to outliers.
* **Elastic Net Regularization/Regression**— it’s a combination of L1+L2.
* Decision Tree

### ****Feature Extraction****

Feature extraction involves reducing the number of resources required to describe a large set of data.

**Why Feature Extraction?**

While performing the analysis of a large number of variables, one requires large computation power and large memory. Also, it may cause the classification algorithm to [overfit](https://en.wikipedia.org/wiki/Overfitting) the training samples. This leads to the model which we build is not a good generalizer to new samples or data. Overfitting rises when the Model learns data rather than the pattern which hides in the data. Reducing the dimension of feature, we have fewer relationship b/w variables to consider and we less likely to overfit the model. It can also reduce the amount of redundant data for a given analysis, which saves computation time for an algorithm.

While Feature Elimination eliminates the least important features, and increases the simplicity of the model while maintaining the interpretability. But disadvantage is loss of information, which might be useful in building the model.

1. **Principal Component Analysis (PCA)**

The most commonly or widely used dimension reduction technique for feature extraction is PCA.

The main ‘linear dimensionality reduction’ technique, which performs linear mapping of a data to a lower dimensional space. It’s an [unsupervised](https://en.wikipedia.org/wiki/Unsupervised_learning) machine learning algorithm, which leads PCA to ignore the class labels and just concentrates on the variance in a data. Which can sometime lead into misclassification of data. The goal is to find the directions that maximizes the variance in the dataset.

**2. Linear Discriminant Analysis (LDA)**

In contrast to PCA , LDA is [supervised](https://en.wikipedia.org/wiki/Supervised_learning)machine learning algorithm. LDA is a generalization of [Fisher’s Linear Discriminant](https://en.wikipedia.org/wiki/Linear_discriminant_analysis#Fisher%27s_linear_discriminant)**,** “a method used in Statistics, Pattern Recognition, and Machine Learning to find a linear combination of features that characterizes or separates two or more classes of objects /events.”

The dimensionality reduction algorithm reduces the number of dimensions from original to C-1 number of features. Where ‘C’ is the number of classes.

For example, suppose we have 3 classes and 18 features, then LDA will reduce the feature from 18 to only 2 (C-1=3–1=2) features.

#### Catch — [LDA](https://sebastianraschka.com/Articles/2014_python_lda.html) approach is very similar to PCA, but in addition to maximizing the variance in data (PCA), we are additionally interested in axes that maximize the separation between multiple classes.

#### Conclusion

I hope you found this article helpful! Let me know what you think, especially if there are suggestions for improvement. Please check my other blogs and cite.!  :)

<https://medium.com/@krashnagurme/hypothesis-testing-statistics-probability-4e3d3f9a7e85>

<https://medium.com/@krashnagurme/importing-google-drive-files-csv-into-pandas-dataframe-for-google-colaboratory-b5f716ddb2c4>

  !!! HappY LearninG!!!