**Supervised Learning**

Supervised learning models are models that map inputs to outputs, and attempt to extrapolate patterns learned in past data on unseen data. Supervised learning models can be either regression models, where we try to predict a continuous variable, like stock prices—or classification models, where we try to predict a binary or multi-class variable, like whether a customer will churn or not. In the section below, we'll explain three popular types of supervised learning models: regression-only models, regression and classification models, and classification-only models.

**Regression Only Models**

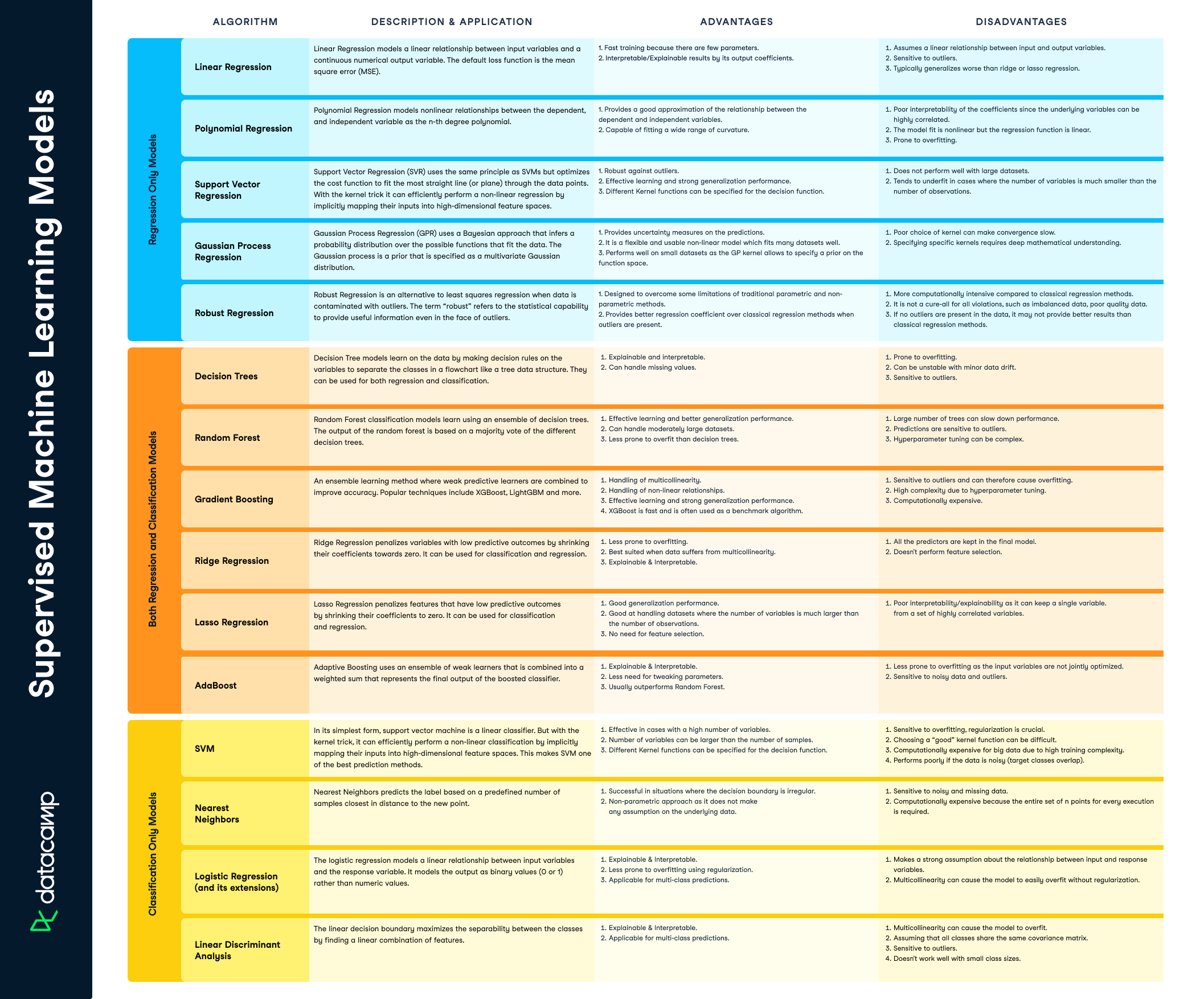
|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Description and Application** | **Advantages** | **Disadvantages** |
| **Linear Regression** | Linear Regression models a linear relationship between input variables and a continuous numerical output variable. The default loss function is the mean square error (MSE). | 1. Fast training because there are few parameters. 2. Interpretable/Explainable results by its output coefficients. | 1. Assumes a linear relationship between input and output variables. 2. Sensitive to outliers. 3. Typically generalizes worse than ridge or lasso regression. |
| **Polynomial Regression** | Polynomial Regression models nonlinear relationships between the dependent, and independent variable as the n-th degree polynomial. | 1. Provides a good approximation of the relationship between the dependent and independent variables. 2. Capable of fitting a wide range of curvature. | 1. Poor interpretability of the coefficients since the underlying variables can be   highly correlated. 2. The model fit is nonlinear but the regression function is linear. 3. Prone to overfitting. |
| **Support Vector  Regression** | Support Vector Regression (SVR) uses the same principle as SVMs but optimizes the cost function to fit the most straight line (or plane) through the data points. With the kernel trick it can efficiently perform a non-linear regression by implicitly mapping their inputs into high-dimensional feature spaces. | 1. Robust against outliers. 2. Effective learning and strong generalization performance. 3. Different Kernel functions can be specified for the decision function. | 1. Does not perform well with large datasets. 2. Tends to underfit in cases where the number of variables is much smaller than the number of observations. |
| **Gaussian Process  Regression** | Gaussian Process Regression (GPR) uses a Bayesian approach that infers a probability distribution over the possible functions that fit the data. The Gaussian process is a prior that is specified as a multivariate Gaussian distribution. | 1. Provides uncertainty measures on the predictions. 2. It is a flexible and usable non-linear model which fits many datasets well. 3. Performs well on small datasets as the GP kernel allows to specify a prior on the function space. | 1. Poor choice of kernel can make convergence slow. 2. Specifying specific kernels requires deep mathematical understanding. |
| **Robust Regression** | Robust Regression is an alternative to least squares regression when data is contaminated with outliers. The term “robust” refers to the statistical capability to provide useful information even in the face of outliers. | 1. Designed to overcome some limitations of traditional parametric and non-parametric methods. 2. Provides better regression coefficient over classical regression methods when outliers are present. | 1. More computationally intensive compared to classical regression methods. 2. It is not a cure-all for all violations, such as imbalanced data, poor quality data. 3. If no outliers are present in the data, it may not provide better results than  classical regression methods. Tree-based models |

**Both Regression and Classification Models**

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Description and Application** | **Advantages** | **Disadvantages** |
| **Decision Trees** | Decision Tree models learn on the data by making decision rules on the variables to separate the classes in a flowchart like a tree data structure. They can be used for both regression and classification. | 1. Explainable and interpretable. 2. Can handle missing values. | 1. Prone to overfitting. 2. Can be unstable with minor data drift. 3. Sensitive to outliers. |
| **Random Forest** | Random Forest classification models learn using an ensemble of decision trees. The output of the random forest is based on a majority vote of the different decision trees. | 1. Effective learning and better generalization performance. 2. Can handle moderately large datasets. 3. Less prone to overfit than decision trees. | 1. Large number of trees can slow down performance. 2. Predictions are sensitive to outliers. 3. Hyperparameter tuning can be complex. |
| **Gradient Boosting** | An ensemble learning method where weak predictive learners are combined to improve accuracy. Popular techniques include XGBoost, LightGBM and more. | 1. Handling of multicollinearity. 2. Handling of non-linear relationships. 3. Effective learning and strong generalization performance. 4. XGBoost is fast and is often used as a benchmark algorithm. | 1. Sensitive to outliers and can therefore cause overfitting. 2. High complexity due to hyperparameter tuning. 3. Computationally expensive. |
| **Ridge Regression** | Ridge Regression penalizes variables with low predictive outcomes by shrinking their coefficients towards zero. It can be used for classification and regression. | 1. Less prone to overfitting. 2. Best suited when data suffers from multicollinearity. 3. Explainable & Interpretable. | 1. All the predictors are kept in the final model. 2. Doesn't perform feature selection. |
| **Lasso Regression** | Lasso Regression penalizes features that have low predictive outcomes  by shrinking their coefficients to zero. It can be used for classification  and regression. | 1. Good generalization performance. 2. Good at handling datasets where the number of variables is much larger than the number of observations. 3. No need for feature selection. | 1. Poor interpretability/explainability as it can keep a single variable.  from a set of highly correlated variables. |
| **AdaBoost** | Adaptive Boosting uses an ensemble of weak learners that is combined into a weighted sum that represents the final output of the boosted classifier. | 1. Explainable & Interpretable. 2. Less need for tweaking parameters. 3. Usually outperforms Random Forest. | 1. Less prone to overfitting as the input variables are not jointly optimized. 2. Sensitive to noisy data and outliers. |

**Classification Only Models**

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Description and Application** | **Advantages** | **Disadvantages** |
| **SVM** | In its simplest form, support vector machine is a linear classifier. But with the  kernel trick, it can efficiently perform a non-linear classification by implicitly  mapping their inputs into high-dimensional feature spaces. This makes SVM one  of the best prediction methods. | 1. Effective in cases with a high number of variables. 2. Number of variables can be larger than the number of samples. 3. Different Kernel functions can be specified for the decision function. | 1. Sensitive to overfitting, regularization is crucial. 2. Choosing a “good” kernel function can be difficult. 3. Computationally expensive for big data due to high training complexity. 4. Performs poorly if the data is noisy (target classes overlap). |
| **Nearest  Neighbors** | Nearest Neighbors predicts the label based on a predefined number of samples closest in distance to the new point. | 1. Successful in situations where the decision boundary is irregular. 2. Non-parametric approach as it does not make  any assumption on the underlying data. | 1. Sensitive to noisy and missing data. 2. Computationally expensive because the entire set of n points for every execution  is required. |
| **Logistic Regression  (and its extensions)** | The logistic regression models a linear relationship between input variables and the response variable. It models the output as binary values (0 or 1)  rather than numeric values. | 1. Explainable & Interpretable. 2. Less prone to overfitting using regularization. 3. Applicable for multi-class predictions. | 1. Makes a strong assumption about the relationship between input and response variables. 2. Multicollinearity can cause the model to easily overfit without regularization. |
| **Linear Discriminant  Analysis** | The linear decision boundary maximizes the separability between the classes by finding a linear combination of features. | 1. Explainable & Interpretable. 2. Applicable for multi-class predictions. | 1. Multicollinearity can cause the model to overfit. 2. Assuming that all classes share the same covariance matrix. 3. Sensitive to outliers. 4. Doesn't work well with small class sizes. |



**Unsupervised Learning**

Unsupervised learning is about discovering general patterns in data. The most popular example is clustering or segmenting customers and users. This type of segmentation is generalizable and can be applied broadly, such as to documents, companies, and genes. Unsupervised learning consists of clustering models that learn how to group similar data points together or association algorithms that group different data points based on pre-defined rules.

**Clustering Models**

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Description and Application** | **Advantages** | **Disadvantages** |
| **K-Means** | Most common clustering approach which assumes that the closer data points are to each other, the more similar they are. It determines K clusters based on Euclidean distances. | 1. Scales to large datasets. 2. Interpretable & explainable results 3. Can generate tight clusters | 1. Requires defining the expected number of clusters in advance. 2. Not suitable to identify clusters with non-convex shapes. |
| **DBSCAN** | Density-Based Spatial Clustering of Applications with Noise can handle non-linear cluster structures, purely based on density. It can differentiate and separate regions with varying degrees of density, thereby creating clusters. | 1. No assumption on the expected number of clusters. 2. Can handle noisy data and outliers 3. No assumptions on the shapes and sizes of the clusters 4. Can identify clusters with different densities | 1. Requires optimization of two parameters. 2. Can struggle in case of very high dimensional data |
| **HDBSCAN** | Family of the density-based algorithms and has roughly two steps: finding the core distance of each point, and expands clusters from them. It extends DBSCAN by converting it into a hierarchical clustering algorithm. | 1. No assumption on the expected number of clusters 2. Can handle noisy data and outliers. 3. No assumptions on the shapes and sizes of the clusters. 4. Can identify clusters with different densities | 1. Mapping of unseen objects in HDBSCAN is not straightforward. 2. Can be computationally expensive |
| **Agglomerative  Hierarchical Clustering** | Uses hierarchical clustering to determine the distance between samples based on the metric, and pairs are merged into clusters using the linkage type. | 1. There is no need to specify the number of clusters. 2. With the right linkage, it can be used for the detection of outliers. 3. Interpretable results using dendrograms. | 1. Specifying metric and linkages types requires good understanding of the statistical properties of the data 2. Not straightforward to optimize 3. Can be computationally expensive for large datasets |
| **OPTICS** | Family of the density-based algorithms where it finds core sample of high density and expands clusters from them. It operates with a core distance (ɛ) and reachability distance. | 1. No assumption on the expected number of clusters. 2. Can handle noisy data and outliers. 3. No assumptions on the shapes and sizes of the clusters. 4. Can identify clusters with different densities. 5. Not required to define a fixed radius as in DBSCAN. | 1. It only produces a cluster ordering. 2. Does not work well in case of very high dimensional data. 3. Slower than DBSCAN. |
| **Gaussian Mixture Models** | Gaussian Mixture Models (GMM) leverages probabilistic models to detect clusters using a mixture of normal (gaussian) distributions. | 1. Provides uncertainty measures for each observation 2. Can identify overlapping clusters | 1. Requires defining the expected number of clusters or mixture components in advance 2. The covariance type needs to be defined for the mixture of components |

**Association Rules**

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Description and Application** | **Advantages** | **Disadvantages** |
| **Apriori algorithm** | The Apriori algorithm uses the join and prune step iteratively to identify the most frequent itemset in the given dataset. Prior knowledge (apriori) of frequent itemset properties is used in the process. | 1. Explainable & interpretable results. 2. Exhaustive approach based on the confidence and support. | 1. Requires defining the expected number of clusters or mixture components in advance 2. The covariance type needs to be defined for the mixture of component |
| **FP-growth algorithm** | Frequent Pattern growth (FP-growth) is an improvement on the Apriori algorithm for finding frequent itemsets. It generates a conditional FP-Tree for every item in the data. | 1. Explainable & interpretable results. 2. Smaller memory footprint than the Apriori algorithm | 1. More complex algorithm to build than Apriori 2. Can result in many (incremental) overlapping/trivial itemsets |
| **FP-Max Algorithm** | A variant of Frequent pattern growth that is focused on finding maximal itemsets. | 1. Explainable & Interpretable results. 2. Smaller memory footprint than the Apriori and FP-growth algorithms | 1. More complex algorithm to build than Apriori |
| **Eclat** | Equivalence Class Clustering and Bottom-Up Lattice Traversal (Eclat) applies a Depth-First Search of a graph procedure. This is a more efficient and scalable version of the Apriori algorithm. | 1. Explainable & interpretable results. 2. Computational faster compared to the Apriori algorithm | 1. Can provide only a subset of results in contrast to the Apriori algorithm and its variants |
| **Hypergeometric Networks** | HNet learns the Association from datasets with mixed data types (discrete and continuous variables) and with unknown functions. Associations are statistically tested using the hypergeometric distribution for finding frequent itemset. | 1. Explainable & Interpretable results 2. More robust against spurious associations as it uses statistical inferences 3. Can associate discrete (itemsets) in combination with continuous measurements 4. Can handle missing values | 1. Computationally intensive for very large datasets. |

**Dimensionality Reduction**

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Description and Application** | **Advantages** | **Disadvantages** |
| **PCA** | Principal Component Analysis (PCA) is a feature extraction approach that uses a linear function to reduce dimensionality in datasets by minimizing information loss. | 1. Explainable & Interpretable results. 2. New unseen datapoints can be mapped into the existing PCA space. 3. Can be used as a dimensionality reduction technique as a preliminary step to other machine learning tasks 4. Helps reduce overfitting 5. Helps remove correlated features | 1. Sensitive to outliers 2. Requires data standardization |
| **t-SNE** | t-distributed Stochastic Neighbor Embedding is a non-linear dimensionality reduction method that converts similarities between data points to joint probabilities using the Student t-distribution in the low-dimensional space | 1. Helps preserve the relationships seen in high dimensionality 2. Easy to visualise the structure of high-dimensional data in 2 or 3 dimensions 3. Very effective for visualizing clusters or groups of data points and their relative proximities | 1. The cost function is not convex; different initializations can get different results. 2. Computationally intensive for large datasets. 3. Default parameters do not always achieve the best results |
| **UMAP** | Uniform Manifold Approximation and Projection (UMAP) constructs a high-dimensional graph representation of the data then optimizes a low-dimensional graph to be as structurally similar as possible. | 1. It can be used as a general-purpose dimension reduction technique as a preliminary step to other machine learning tasks. 2. Can be very effective for visualizing clusters or groups of data points and their relative proximities. 3. Able to handle high dimensional sparse datasets | 1. Default parameters do not always achieve the best results |
| **ICA** | Independent Component Analysis (ICA) is a linear dimensionality reduction method that aims to separate a multivariate signal into additive subcomponents under the assumption that independent components are non-gaussian. Where PCA "compresses" the data, ICA "separates" the information. | 1. Can separate multivariate signals into its subcomponents. 2. Clear aim of the method; only applicable if there are multiple independent generators of information to uncover. 3. Can extract hidden factors in the data by transforming a set of variables to a new set that is maximally independent. | 1. Without any prior knowledge, determination of the number of independent components or sources can be difficult. 2. PCA is often required as a pre-processing step. |
| **PaCMAP** | Pairwise Controlled Manifold Approximation (PaCMAP) is a dimensionality reduction method that optimizes low-dimensional embeddings using three kinds of point pairs: neighbor pairs, mid-near pair, and further pairs. | 1. It can preserve both local and global structure of the data in original space. 2. Performance is relatively robust within reasonable parameter choices. | 1. Parameters have been tuned on smaller datasets and it is yet unknown how it behaves and extends to very high-dimensional datasets |

