



Cheat Sheet: Building Unsupervised Learning Models

Unsupervised learning models

Model Name	Brief Description	Code Syntax
UMAP	<p>UMAP (Uniform Manifold Approximation and Projection) is used for dimensionality reduction.</p> <p>Pros: High performance, preserves global structure.</p> <p>Cons: Sensitive to parameters.</p> <p>Applications: Data visualization, feature extraction.</p> <p>Key hyperparameters:</p> <ul style="list-style-type: none">• n_neighbors: Controls the local neighborhood size (default = 15).• min_dist: Controls the minimum distance between points in the embedded space (default = 0.1).• n_components: The dimensionality of the embedding (default = 2).	<div><div>1</div><div>from umap.umap_ import UMAP</div></div> <div><div>2</div><div>umap = UMAP(n_neighbors=15, min_dist=0.1, n_components=2)</div></div> <div></div>
t-SNE	<p>t-SNE (t-Distributed Stochastic Neighbor Embedding) is a nonlinear dimensionality reduction technique.</p> <p>Pros: Good for visualizing high-dimensional data.</p> <p>Cons: Computationally expensive, prone to overfitting.</p> <p>Applications: Data visualization, anomaly detection.</p> <p>Key hyperparameters:</p> <ul style="list-style-type: none">• n_components: The number of dimensions for the output (default = 2).• perplexity: Balances attention between local and global aspects of the data (default = 30).• learning_rate: Controls the step size during optimization (default = 200).	<div><div>1</div><div>from sklearn.manifold import TSNE</div></div> <div><div>2</div><div>tsne = TSNE(n_components=2, perplexity=30, learning_rate=200)</div></div> <div></div>
PCA	<p>PCA (principal component analysis) is used for linear dimensionality reduction.</p> <p>Pros: Easy to interpret, reduces noise.</p> <p>Cons: Linear, may lose information in nonlinear data.</p> <p>Applications: Feature extraction, compression.</p> <p>Key hyperparameters:</p> <ul style="list-style-type: none">• n_components: Number of principal components to retain (default = 2).• whiten: Whether to scale the components (default = False).• svd_solver: The algorithm to compute the components (default = 'auto').	<div><div>1</div><div>from sklearn.decomposition import PCA</div></div> <div><div>2</div><div>pca = PCA(n_components=2)</div></div> <div></div>
DBSCAN	<p>DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a density-based clustering algorithm.</p> <p>Pros: Identifies outliers, does not require the number of clusters.</p> <p>Cons: Difficult with varying density clusters.</p> <p>Applications: Anomaly detection, spatial data clustering.</p> <p>Key hyperparameters:</p> <ul style="list-style-type: none">• eps: The maximum distance between two points to be considered neighbors (default = 0.5).• min_samples: Minimum number of samples in a neighborhood to form a cluster (default = 5).	<div><div>1</div><div>from sklearn.cluster import DBSCAN</div></div> <div><div>2</div><div>dbscan = DBSCAN(eps=0.5, min_samples=5)</div></div> <div></div>
HDBSCAN	<p>HDBSCAN (Hierarchical DBSCAN) improves on DBSCAN by handling varying density clusters.</p> <p>Pros: Better handling of varying densities.</p> <p>Cons: Can be slower than DBSCAN.</p> <p>Applications: Large datasets, complex clustering problems.</p> <p>Key hyperparameters:</p> <ul style="list-style-type: none">• min_cluster_size: The minimum size of clusters (default = 5).• min_samples: Minimum number of samples to form a cluster (default = 10).	<div><div>1</div><div>import hdbscan</div></div> <div><div>2</div><div>clusterer = hdbscan.HDBSCAN(min_cluster_size=5)</div></div> <div></div>
K-Means clustering	<p>K-Means is a centroid-based clustering algorithm that groups data into k clusters.</p> <p>Pros: Efficient, simple to implement.</p> <p>Cons: Sensitive to initial cluster centroids.</p> <p>Applications: Customer segmentation, pattern recognition.</p> <p>Key hyperparameters:</p> <ul style="list-style-type: none">• n_clusters: Number of clusters (default = 8).• init: Method for initializing the centroids ('k-means++' or 'random', default = 'k-means++').• n_init: Number of times the algorithm will run with different centroid seeds (default = 10).	<div><div>1</div><div>from sklearn.cluster import KMeans</div></div> <div><div>2</div><div>kmeans = KMeans(n_clusters=3)</div></div> <div></div>

Associated fuctions used

Method	Brief Description	Code Syntax
make_blobs	Generates isotropic Gaussian blobs for clustering.	<div><div>1</div><div>from sklearn.datasets import make_blobs</div><div>2</div><div>X, y = make_blobs(n_samples=100, centers=2, random_state=42)</div><div></div></div>
multivariate_normal	Generates samples from a multivariate normal distribution.	<div><div>1</div><div>from numpy.random import multivariate_normal</div><div>2</div><div>samples = multivariate_normal(mean=[0, 0], cov=[[1, 0], [0, 1]], size=100)</div><div></div></div>
plotly.express.scatter_3d	Creates a 3D scatter plot using Plotly Express.	<div><div>1</div><div>import plotly.express as px</div><div>2</div><div>fig = px.scatter_3d(df, x='x', y='y', z='z')</div><div>3</div><div>fig.show()</div><div></div></div>
geopandas.GeoDataFrame	Creates a GeoDataFrame from a Pandas DataFrame.	<div><div>1</div><div>import geopandas as gpd</div><div>2</div><div>gdf = gpd.GeoDataFrame(df, geometry='geometry')</div><div></div></div>
geopandas.to_crs	Transforms the coordinate reference system of a GeoDataFrame.	<div><div>1</div><div>gdf = gdf.to_crs(epsg=3857)</div><div></div></div>
contextily.add_basemap	Adds a basemap to a GeoDataFrame plot for context.	<div><div>1</div><div>import contextily as ctx</div><div>2</div><div>ax = gdf.plot(figsize=(10, 10))</div><div>3</div><div>ctx.add_basemap(ax)</div><div></div></div>
pca.explained_variance_ratio_	Returns the proportion of variance explained by each principal component.	<div><div>1</div><div>from sklearn.decomposition import PCA</div><div>2</div><div>pca = PCA(n_components=2)</div><div>3</div><div>pca.fit(X)</div><div>4</div><div>variance_ratio = pca.explained_variance_ratio_</div><div></div></div>

Author

Jeff Grossman
Abhishek Gagneja

