usualsuspects

Release 0.1.1

May 04, 2020

Contents:

This is simply a personal collection of out-of-the box tools for making visualisations commonly found in ML papers.

Note: This is primarily a personal project with the aim of practicing python packaging and good software engineering practices whilst developing something useful for me personally. If it helps you that's great! However, there are much more fleshed out visualisation aides out there done by large teams and dedicated individuals.

Feel free to explore how the tools are used in the examples found in the Github repo.

CHAPTER 1

Installation

1.1 Installing usualsuspects

A stable version can be downloaded from PyPI

```
pip install usualsuspects
```

1.1.1 Installation from sources

The most recent version can be cloned from the Github repository and then built from the project root folder

```
pip install .
```

If you wish to modify some of the source code for your own task, you can make a developer install. It is slightly slower, but immediately takes on any changes into the source code.

```
pip install -e .
```

CHAPTER 2

License

CHAPTER 3

usualsuspects package

3.1 usualsuspects.tsne module

class usualsuspects.tsne.**Quick2DTSNE**(*X*, *y*, *perplexity*, *initialisation='pca'*, *n_jobs=-1*)

A quick tool for creating 2D T-SNE plots with input X, and optional true classifications y for coloring.

From Scikit Docs: t-Distributed Stochastic Neighbour Embedding is a tool for visualizing high-dimensional data. It converts similarities between data points to joint probabilities and tries to minimize the KL divergence between the joint probabilities of the low dimensional embedding and the high-dimensional data. T-SNE has a cost function that is not convex. i.e. with different initializations we are bound to have different results.

This tool is a shortcut to producing the T-SNE plot one finds so often in papers.

Parameters

- **X** (*array-like*) – asdfasdf
- **y** (*array-like*) – asdfasdf
- **perplexity** (*float*) – The perplexity is a hyperparameter that is related to the number of nearest neighbours to consider. Typically larger datasets need a higher perplexity to look good.
- **initialization** (*str (default="pca")*) – Possible values = ['pca', 'random'] describes the initialization.
- **n_jobs** (*int (default=-1)*) – The number of cores to use. Default is -1 which uses all available system cores

Returns

Return type *Quick2DTSNE*

plot_embedding(*title=None*, *save_to='tsne_plot.png'*)

Creates a matplotlib pyplot and saves it as a png image.

Parameters

- **title** (*str*) – Title for the plot

- **save_to** (*str*) – path and filename where the image will be saved. Extension is contextual but expects .png or .jpg

Returns Saves a PNG image of the plot to the path in *save_to*

Return type None

3.2 usualsuspects.pca module

class usualsuspects.pca.**Quick2DPCA**(*X*, *y*)

A quick tool for crating 2D PCA plots with input X, and optional true classification y for colouring

From SciKit Docs: Linear dimensionality reduction using SVD of the data to project it to a lower dimensional space. The input data is centered but not scaled for each feature before applying the SVD

This tool is a shortcut to producing PCA plots

plot_embedding(*title=None*, *save_to='pca_plot.png'*)

Creates and saves a matplotlib pyplot of the first 2 eigenvectors of the PCA operation on X

Parameters

- **title** (*str*) – Optional title for the plot
- **save_to** (*str*) – path and file_name where the image will be saved. Extension is contextual but expected behaviour is to use .png

Returns Saves a PNG image of the plot to the path in *save_to*

Return type None

Python Module Index

u

Index

P

Q

U