BigNmf Documentation

Release 1.0.0

Vaibhav Kulshrestha, Haran Rajkumar

Mar 15, 2022

Contents:

CHAPTER 1

Indices and tables

- genindex
- modindex
- search

CHAPTER 2

BigNmf

BigNmf (Big Data NMF) is a python package for performing single NMF and joint NMF algorithms. NMF (Non-negative matrix factorization) is a unsupervised classification algorithm.

2.1 Installation

This package is available on the PyPi repository. Therefore you can install, by running the following.

```
pip3 install bignmf
```

2.2 Usage

The following is an example code snippet for running the nmf.

2.2.1 1. Single NMF

```
from bignmf.datasets.datasets import Datasets
from bignmf.models.snmf.standard import StandardNmf

Datasets.list_all()
data=Datasets.read("SimulatedX1")
k = 3
iter =100
trials = 50

model = StandardNmf(data,k)
model.run(trials, iter, verbose=0)
print(model.error)
model.cluster_data()
```

(continues on next page)

(continued from previous page)

```
model.calc_consensus_matrices()
print(model.h_cluster)
```

2.2.2 2. Joint NMF

```
from bignmf.models.jnmf.integrative import IntegrativeJnmf
from bignmf.datasets.datasets import Datasets

Datasets.list_all()
data_dict = {}
data_dict["sim1"] = Datasets.read("SimulatedX1")
data_dict["sim2"] = Datasets.read("SimulatedX2")

k = 3
iter =100
trials = 50
lamb = 0.1

model = IntegrativeJnmf(data_dict, k, lamb)
model = StandardNmf(data,k)
model.run(trials, iter, verbose=0)
print(model.error)
model.cluster_data()
model.calc_consensus_matrices()
print(model.h_cluster)
```