

scanpy.tl.tsne

```
scanpy.tl.tsne(adata, n_pcs=None, use_rep=None, perplexity=30, early_exaggeration=12,
learning_rate=1000, random_state=0, use_fast_tsne=False, n_jobs=None, copy=False, *,
metric='euclidean')
```

t-SNE [Maaten08] [Amir13] [Pedregosa11].

t-distributed stochastic neighborhood embedding (tSNE) [Maaten08] has been proposed for visualizing single-cell data by [Amir13]. Here, by default, we use the implementation of *scikit-learn* [Pedregosa11]. You can achieve a huge speedup and better convergence if you install *Multicore-tSNE* by [Ulyanov16], which will be automatically detected by Scanpy.

Parameters:

adata : `AnnData`

Annotated data matrix.

n_pcs : `Optional[int]` (default: `None`)

Use this many PCs. If `n_pcs==0` use `.X` if `use_rep` is `None`.

use_rep : `Optional[str]` (default: `None`)

Use the indicated representation. `'X'` or any key for `.obs` is valid. If `None`, the representation is chosen automatically: For `.n_vars < 50`, `.X` is used, otherwise `'X_pca'` is used. If `'X_pca'` is not present, it's computed with default parameters.

perplexity : `Union[float , int]` (default: `30`)

The perplexity is related to the number of nearest neighbors that is used in other manifold learning algorithms. Larger datasets usually require a larger perplexity. Consider selecting a value between 5 and 50. The choice is not extremely critical since t-SNE is quite insensitive to this parameter.

metric : `str` (default: `'euclidean'`)

Distance metric calculate neighbors on.

early_exaggeration : `Union[float , int]` (default: `12`)

Controls how tight natural clusters in the original space are in the embedded space and how much space will be between them. For larger values, the space between natural clusters will be larger in the embedded space. Again, the choice of this parameter is not very critical. If the cost function increases during initial optimization, the early exaggeration factor or the learning rate might be too high.

learning_rate : `Union [float , int]` (default: `1000`)

Note that the R-package “Rtsne” uses a default of 200. The learning rate can be a critical parameter. It should be between 100 and 1000. If the cost function increases during initial optimization, the early exaggeration factor or the learning rate might be too high. If the cost function gets stuck in a bad local minimum increasing the learning rate helps sometimes.

random_state : `Union [None , int , RandomState]` (default: `0`)

Change this to use different initial states for the optimization. If `None` , the initial state is not reproducible.

n_jobs : `optional [int]` (default: `None`)

Number of jobs for parallel computation. `None` means using `scanpy._settings.ScanpyConfig.n_jobs` .

copy : `bool` (default: `False`)

Return a copy instead of writing to `adata` .

Return type:

`Optional [AnnData]`

Returns:

: Depending on `copy` , returns or updates `adata` with the following fields.

X_tsne : `np.ndarray` (`adata.obs` , dtype `float`)

tSNE coordinates of data.