

Conformational Clustering and Markov State Models

Conformational clustering

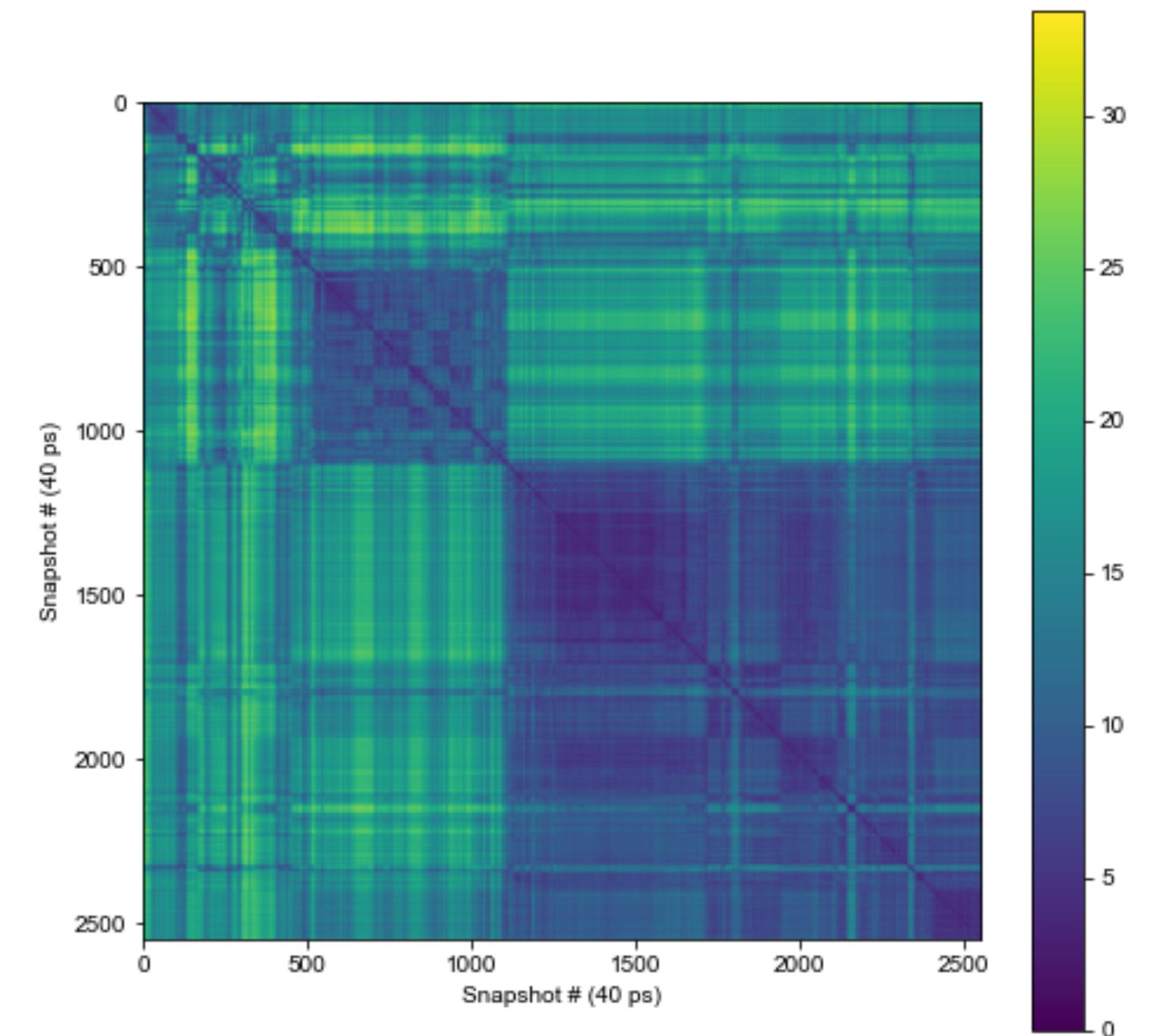
- This module will consist of a mini-lecture and exercise on conformational clustering
- At the end of this module, you should be able to answer the following questions:
 - What is clustering and why is it useful?
 - What distance matrices are there? How should they be selected?
 - How does agglomerative hierarchical clustering work? What is a linkage criterion?

Clustering

- MD simulations yield configurations in continuous space
- Clustering methods group together similar configurations (or, in a more general data science context, observations)
- Clustering is useful
 - interpreting simulation results
 - calculating thermodynamic and kinetic properties
 - predicted populations of conformations
 - predicted rates of transitions (e.g. Markov state models [1, 2])
 - selecting representative configurations for molecular docking [3]

Distance matrices in clustering

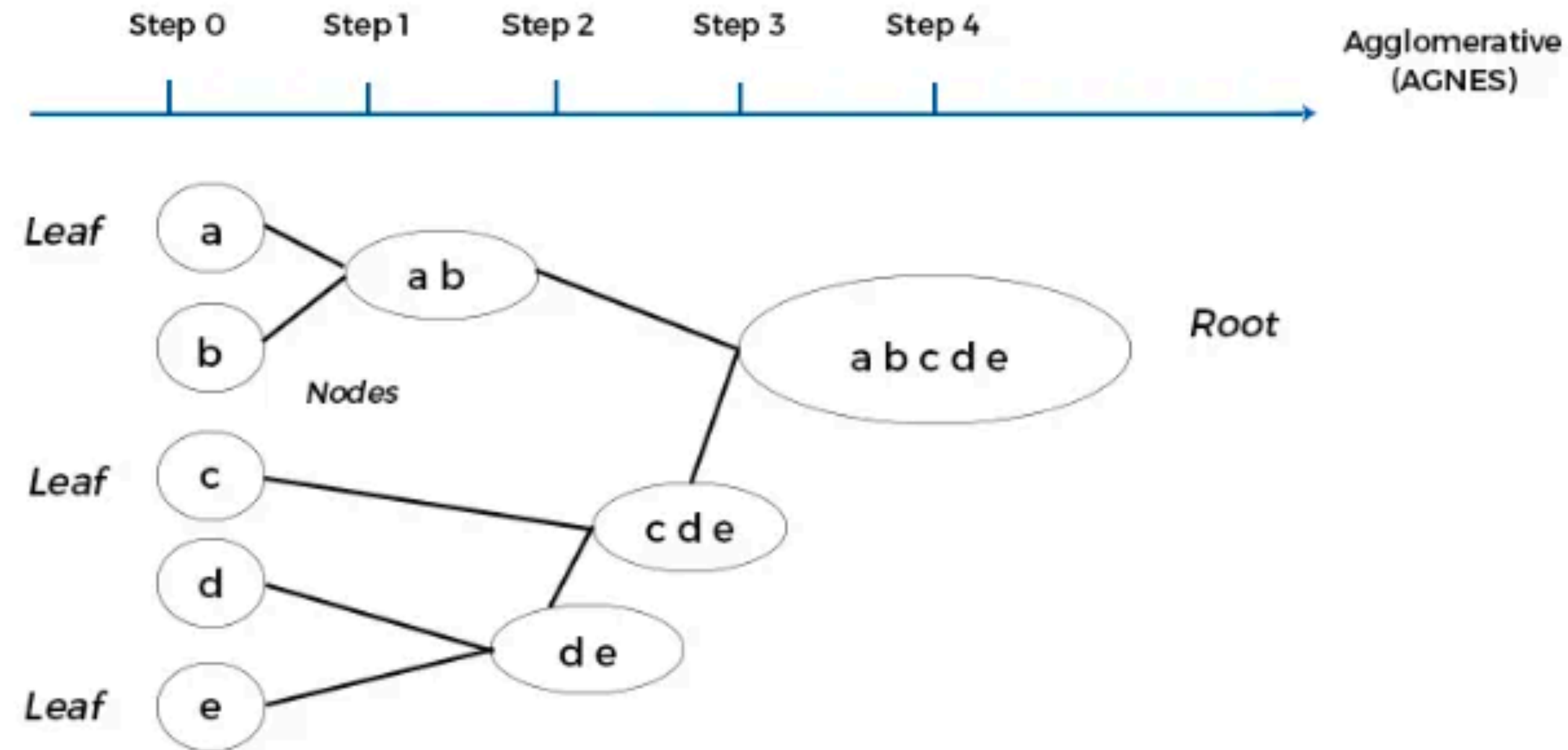
- Almost all clustering algorithms employ a distance matrix
- In a matrix **D**, D_{kl} denotes the distance between observation k and l
- Distance matrices include [3]
 - the RMSD
 - between alpha carbons/all heavy atoms
 - in a entire protein/in a region of the protein
 - Euclidean distance between principal components (like the RMSD, PCA can be based on different subsets of coordinates)
 - based on occupancy fingerprints
 - a 3D grid with zero or one depending whether a point is close to an atom
 - If M_{ab} is the number of points where one grid has a and the second b ,
 - the overlap is $M_{10} + M_{01}$
 - the Tanimoto similarity is $-\log_2[M_{11}/(M_{11} + M_{10} + M_{01})]$
 - the Jaccard distance is $[(M_{11} + M_{01})/(M_{11} + M_{10} + M_{01})]$



Heat map of Euclidean distances between top 20 principal components in a simulation of ubiquitin

Agglomerative hierarchical clustering

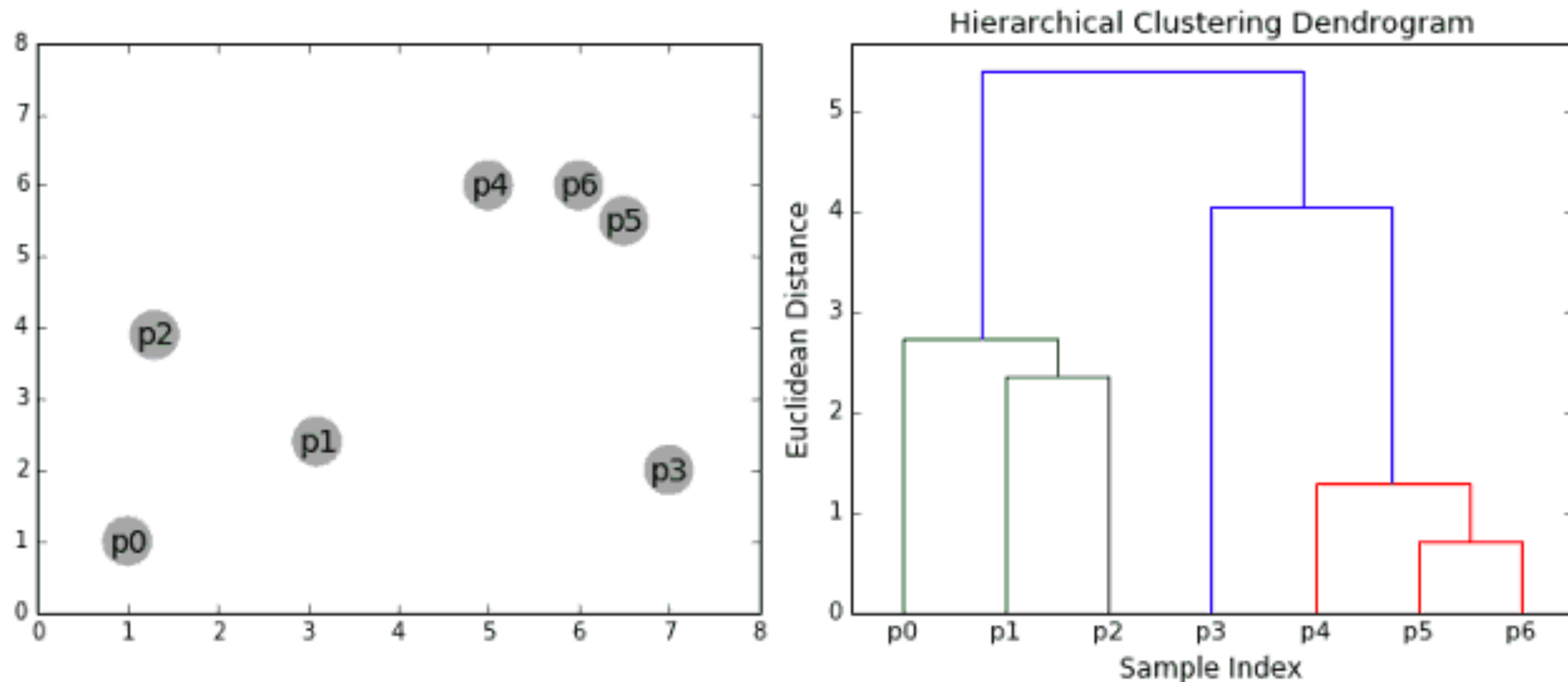
- Closest pair of observations (or clusters) are grouped together until all observations are in groups



[http://primo.ai/index.php?title=Hierarchical_Clustering;_Agglomerative_\(HAC\)_%26_Divisive_\(HDC\)](http://primo.ai/index.php?title=Hierarchical_Clustering;_Agglomerative_(HAC)_%26_Divisive_(HDC))

Agglomerative hierarchical clustering

- Closest pair of observations (or clusters) are grouped together until all observations are in groups



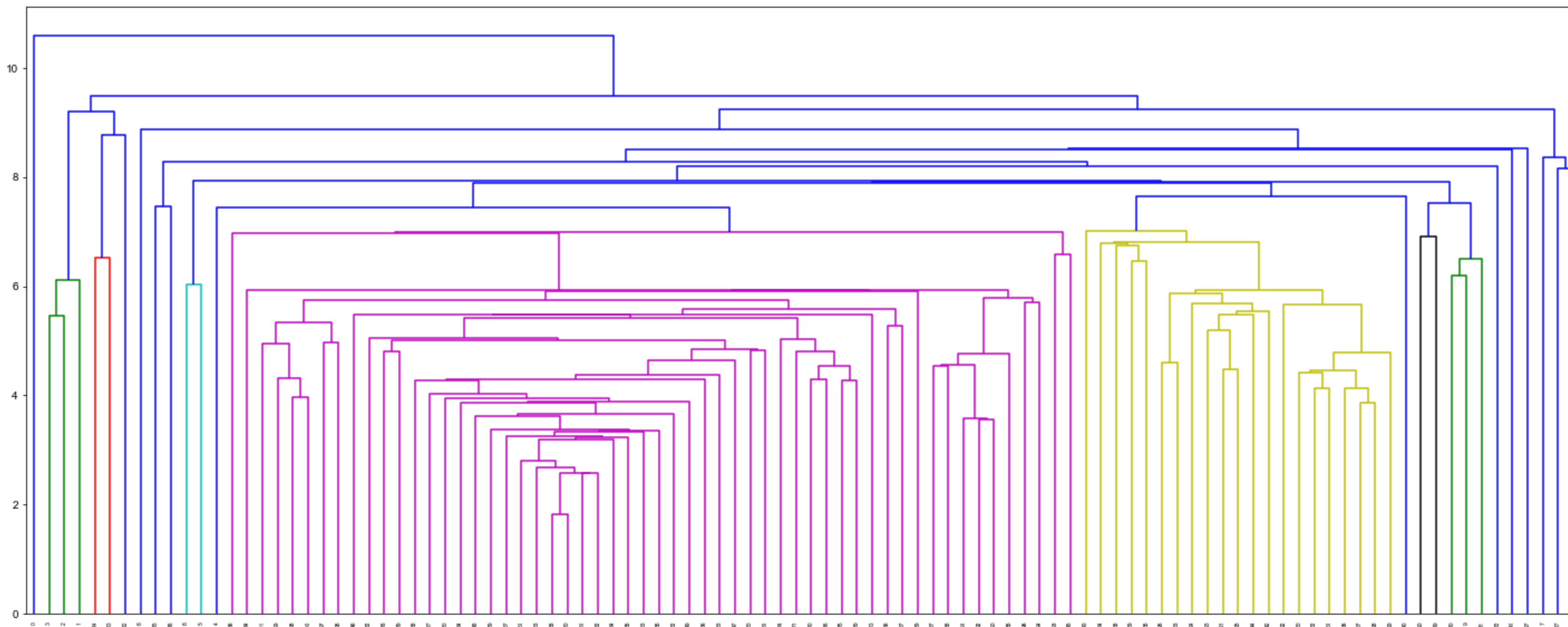
[http://primo.ai/index.php?title=Hierarchical_Clustering;_Agglomerative_\(HAC\)_%26_Divisive_\(HDC\)](http://primo.ai/index.php?title=Hierarchical_Clustering;_Agglomerative_(HAC)_%26_Divisive_(HDC))

Linkage

- The distance matrix provides distances between observations
- What is the distance between clusters?
- Different linkage algorithms are available in scipy: <https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.linkage.html>. In different algorithms, the distance between clusters is the
 - single - minimum distance between observations
 - complete - maximum distance between observations
 - centroid - distance between centroids (the mean of the cluster)
- Which linkage algorithm will yield the smallest and largest apparent distance between clusters?

Agglomerative hierarchical clustering

- There are
 - Different definitions of distances between observations and clusters
 - Different ways to go from linkage matrix to clusters
- See [Clustering.ipynb](#), which shows clustering analysis for a simulation of Mpro



Dendrogram of hierarchical clustering for every 1 ns for a simulation of ubiquitin

Review

- What is clustering and why is it useful?
- What distance matrices are there? How should they be selected?
- How does agglomerative hierarchical clustering work? What is a linkage criterion?

Markov State Models

- This module will consist of
 - an explanation of Markov State Models in the analysis of biological MD
 - a tutorial on pyemma
- At the end of this module, you should be able to
 - answer the following questions:
 - what is a transition matrix and why is it useful?
 - what is a MSM?
 - build a MSM using pyemma

Molecular simulations as Markov chains

- Molecular simulations can be thought of as Markov chains - where the future is only dependent on the present, not the past
 - Always true for MCMC
 - After a sufficient amount of lag time, true for molecular dynamics
- A Markov chain is a *single realization* of a Markov process

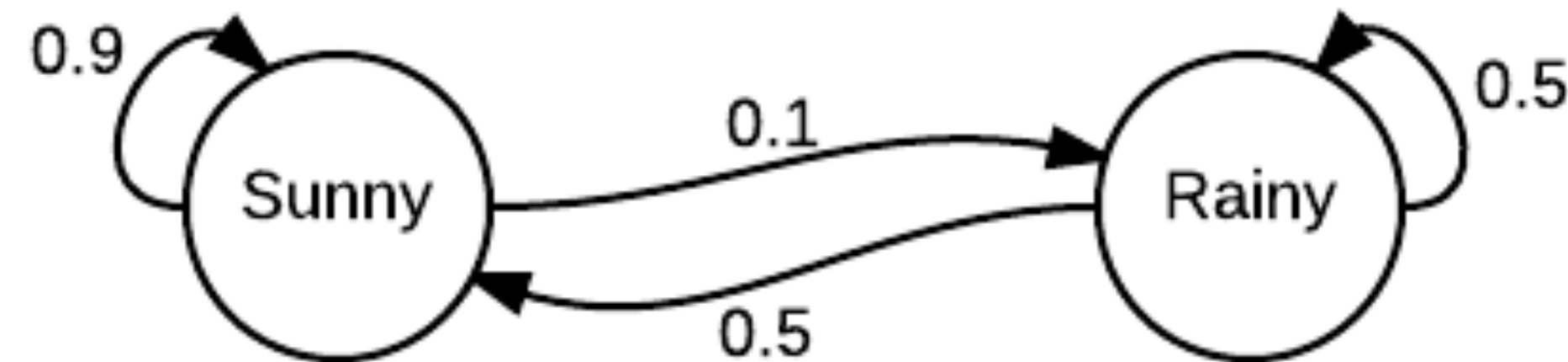
Transition Matrices

- The *complete description* of the dynamics of a Markov system is contained in a transition matrix.
- If the **probability** of moving from i to j in one time step is $Pr(j | i) = P_{i,j}$, the transition matrix P is given by,

$$P = \begin{bmatrix} P_{1,1} & P_{1,2} & \dots & P_{1,j} & \dots & P_{1,S} \\ P_{2,1} & P_{2,2} & \dots & P_{2,j} & \dots & P_{2,S} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ P_{i,1} & P_{i,2} & \dots & P_{i,j} & \dots & P_{i,S} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ P_{S,1} & P_{S,2} & \dots & P_{S,j} & \dots & P_{S,S} \end{bmatrix}.$$

Exercise: Transition Matrices

- Suppose that weather is a Markov process and it is either sunny or rainy.



- Given these transition probabilities, write a transition matrix for the process. Remember that the **probability** of moving from i to j in one time step is $Pr(j | i) = P_{i,j}$.

$$\begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix}$$

Exercise: Transition Matrix Powers

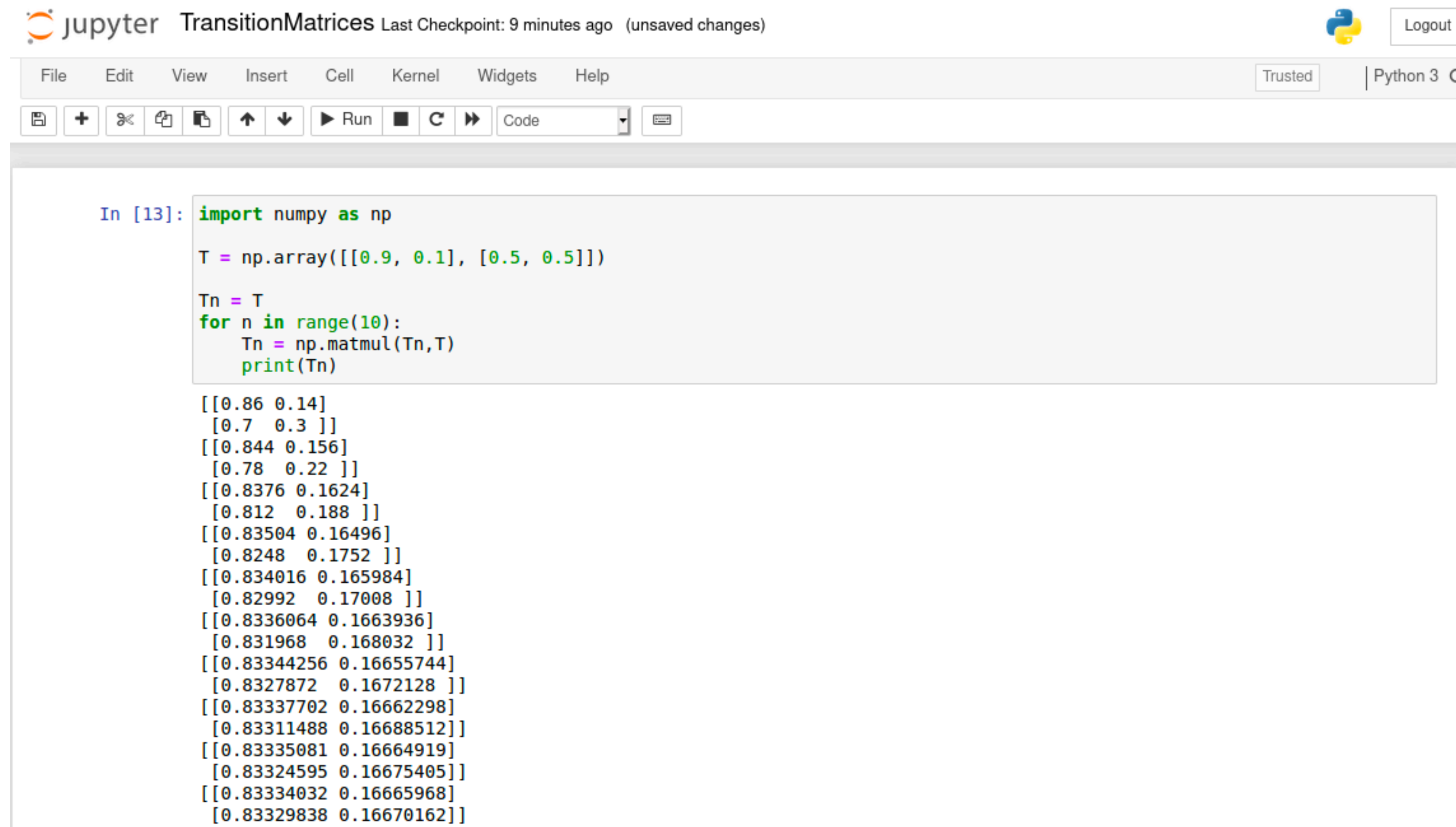
- Given this transition matrix,

$$\begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix},$$

use a Jupyter Notebook to compute the probability after 2, ..., 10 steps.

- Hints:
 - It is helpful to define the matrix as a numpy array
 - ``import numpy as np``
 - ``T = np.array([[0.9, 0.1], [0.5, 0.5]])``
 - use `np.matmul` instead of the `*` operator.

Solution: Transition Matrix Powers



```
jupyter TransitionMatrices Last Checkpoint: 9 minutes ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

In [13]: import numpy as np

T = np.array([[0.9, 0.1], [0.5, 0.5]])

Tn = T
for n in range(10):
    Tn = np.matmul(Tn, T)
    print(Tn)

[[0.86 0.14]
 [0.7 0.3 ]]
[[0.844 0.156]
 [0.78 0.22 ]]
[[0.8376 0.1624]
 [0.812 0.188 ]]
[[0.83504 0.16496]
 [0.8248 0.1752 ]]
[[0.834016 0.165984]
 [0.82992 0.17008 ]]
[[0.8336064 0.1663936]
 [0.831968 0.168032 ]]
[[0.83344256 0.16655744]
 [0.8327872 0.1672128 ]]
[[0.83337702 0.16662298]
 [0.83311488 0.16688512]]
[[0.83335081 0.16664919]
 [0.83324595 0.16675405]]
[[0.83334032 0.16665968]
 [0.83329838 0.16670162]]
```

What do you notice about how the transition matrix changes as the number of steps increases?

Uses of Transition Matrices

- the rates of transitions between any pair of states
- the most probable pathways between any pair of states
- the probability of a transition after n steps is P^n .
- the stationary probability of any state

Markov State Models

Markov State Models

- MSMs
 - Markov states = conformational clusters known as *micro states*
 - stationary probability = Boltzmann distribution
 - are similar to chemical kinetics models, but more states and based on molecular simulation opposed to curve fitting
- MSMs are useful because they
 - can combine information from short MD trajectories
 - piece together local equilibria into a global picture
- Introduction to Markov state models by Frank Noe: https://youtu.be/YXppP_QTut8?list=PLych0HcnzSQLi1CQmxiZig9frLGidF70K

References

- [1] Pande, V. S.; Beauchamp, K. A.; Bowman, G. R. Everything You Wanted to Know about Markov State Models but Were Afraid to Ask. *Methods* (San Diego, Calif.) 2010, 52 (1), 99–105. <https://doi.org/10.1016/j.ymeth.2010.06.002>.
- [2] Chodera, J. D.; Noé, F. Markov State Models of Biomolecular Conformational Dynamics. *Current Opinion in Structural Biology* 2014, 25, 135–144. <https://doi.org/10.1016/j.sbi.2014.04.002>.

Some software

- For MD analysis
 - MDTraj: <http://mdtraj.org/1.9.3/index.html>
 - ProDy: http://prody.csb.pitt.edu/tutorials/trajectory_analysis/trajectory.html
- For Markov State Models
 - MSMBuilder: <http://msmbuilder.org/3.8.0/>
 - enspara: <https://github.com/bowman-lab/enspara>