Reading materials:

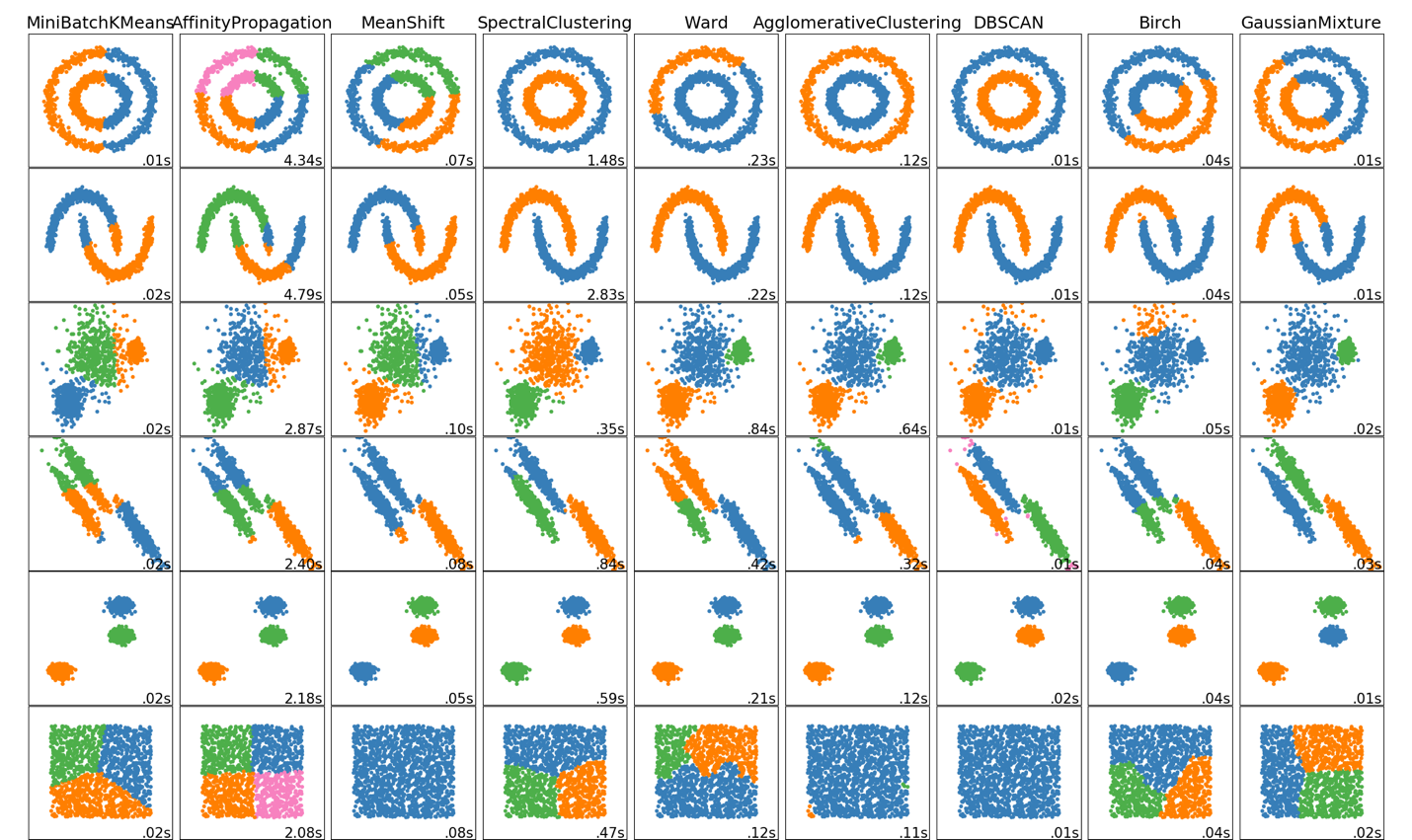
* [5 Clustering Algorithms](•%09https:/towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68)
* [Choosing the number of clusters:](https://link.medium.com/oP9nCiNws9)
* [Assessment metrics for Clustering.](https://medium.com/@ODSC/assessment-metrics-for-clustering-algorithms-4a902e00d92d)
* [Triplet loss](https://towardsdatascience.com/siamese-network-triplet-loss-b4ca82c1aec8)
* [Different kinds of scaling](https://scikit-learn.org/stable/auto_examples/preprocessing/plot_all_scaling.html)
* [Scikit\_learn clustering](https://scikit-learn.org/stable/modules/clustering.html#k-means)

Things to talk about with Ron:

1. Normalizing the data. (How to do that with large dataset?)
2. Use different distance metrics?
3. Normalization to use (some scalers have `partial\_fit` option)
4. Ask Ori how much RAM we have in our computers in order to load the whole dataset into memory

Possible Clustering algorithms:

1. K-median – Less prone to outliers
2. Mini-batch K-means – Choose a few images at random and send them to the algorithm
3. Mean Shift Clustering – No need to specify number of clusters. Do need to play with radius r
4. Heirercal clustering?
5. DBSCAN – dis.- find the right radius for non-noise



Deep clustering:

1. Take an existing network (let’s say one that identifies cancer) and take one layer as a relevant feature and perform some clustering on it.
2. Auto-encoders for dimensionality reduction