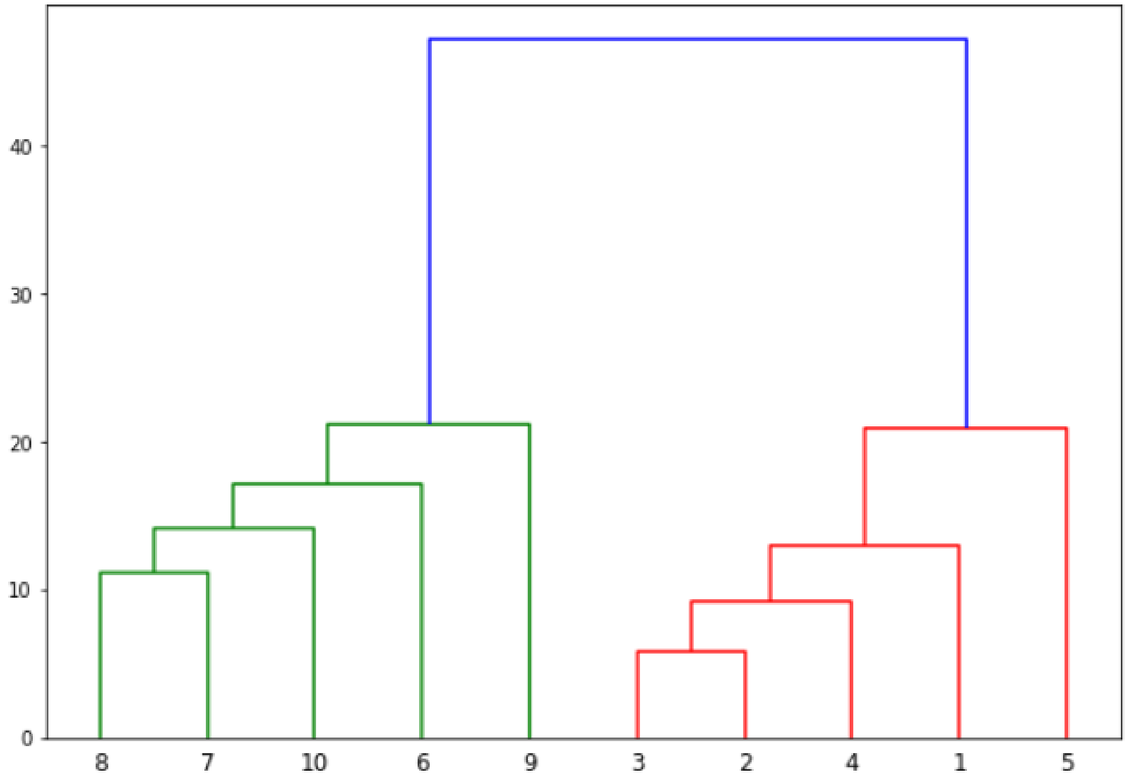
Practice 5

Hierarchical Clustering

Problem

- **Construct spark environment in your local computer and use Bisecting K-Means**



- Use predefined function in :
  - pyspark.**ml.clustering**
  - pyspark.**ml.linalg**

https://stackabuse.com/hierarchical-clustering-with-python-and-scikit-learn/

Dataset

➢ Digits data set

- Each datapoint has an image of a digit with 8x8 pixels.

➢ 5 Statistic Features

Classes	10
Samples per class	~180
Samples total	1797
Dimensionality	64
Features	Integers 0-16

* Reference

https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_digits.html

Practice 5

1. **Use predefined classes in *pyspark.ml.clustering : BisectingKMeans*, *and in pyspark.mllib.linalg : Vectors***

2. **First, preprocess the data using "sort_by_target" function(See next page).**

3. **Second, train Bisecting K-Means model with training data(we don't use label of training data).**

4. **After training the models, calculate *NMI* score of test data points.**

   Parameters for Bisecting K-Means

   - K = 10, minDivisibleClustersize = 1.0

https://spark.apache.org/docs/latest/ml-clustering.html
https://spark.apache.org/docs/2.1.3/api/python/_modules/pyspark/mllib/linalg.html

Practice 5

5. You can sort the data by target like this:

```
nTrain = 1500

def sort_by_target(digits):
    try:
        Data = digits[:,:-1]
        Target = digits[:,-1]

        reorder_train = np.array(sorted([(target, i) for i, target
                                in enumerate(Target[:nTrain])]))[:,1]
        reorder_test = np.array(sorted([(target, i) for i, target
                               in enumerate(Target[nTrain:])]))[:,1]
        Data[:nTrain] = Data[reorder_train.astype(np.int64).tolist()]
        Target[:nTrain] = Target[reorder_train.astype(np.int64).tolist()]
        Data[nTrain:] = Data[(reorder_test + nTrain).astype(np.int64).tolist()]
        Target[nTrain:] = Target[(reorder_test + nTrain).astype(np.int64).tolist()]

        digits = np.concatenate((Data,Target.reshape(-1,1)), axis = 1)

        return digits[:nTrain], digits[nTrain:]
    except:
        return None
```

6. *Call function like this:*

```
trainData, testData = sort_by_target(data_label)
```

Submission

1. You must submit “**result.txt**” file on I-campus
2. In your **result.txt** file, there must be ***NMI score*** of hierarchical clustering result for digit dataset.
3. ***NMI*** means ***normalized mutual information*** which is a metric to measure some clustering results.
4. Your result.txt file must be like following:

Windows

```
NMI of hierarchical clustering
0.6470
```

Linux

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.normalized_mutual_info_score.html