BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE

PILANI

Assignment 2

BITS F464 – Machine Learning

Active learning and SOM

By

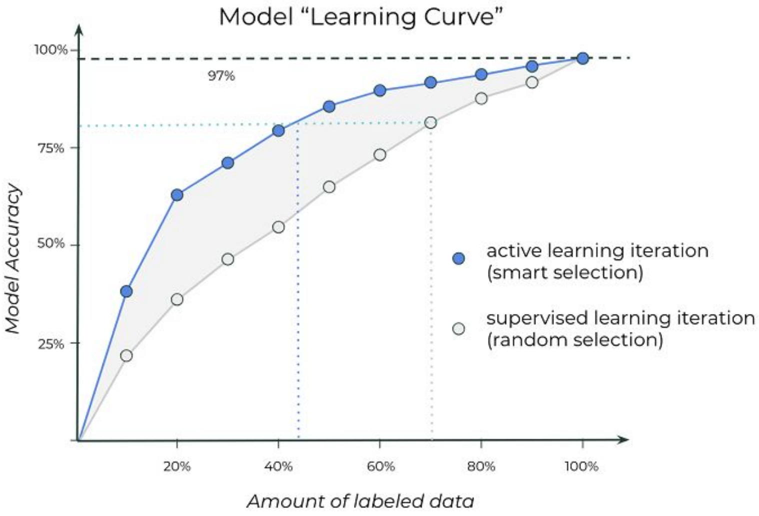
Kavya Gupta	2017A7PS0276P
Bhoomi Sawant	2017A7PS0001P
Prachi Agrawal	2018B5A70716P

Submitted to

Dr. Navneet Goyal

Active learning

It is a learning algorithm which can choose its training data points on its own. The active learning model actively selects the data points thus keeping the size of the training dataset as minimum.This is required because the annotation of unlabelled dataset is expensive and time consuming. We usually have an abundance of unlabelled data, but a small amount of labelled data. Obtaining unlabeled instances is essentially free nowadays but data labeling becomes a bottleneck. In active learning the algorithm is allowed to select a subset of training examples proactively to be labeled next. These types of algorithms are called active learners. They can dynamically pose the queries and ask the data points to be labeled by an oracle.



It is evident from the above figure that, as we increase the amount of training data, the accuracy increases in both active and passive learning. But keeping the data constant, smart selection outperforms the random selection.

To use the active learning on an unlabeled data set, first get a very small sample of this data labelled and train a model. After the model is built, make predictions for all the unlabeled data points. Using a score (explained in later sections), prioritize the labeling. Get the selected point(s) labelled and train a model. Repeat the steps iteratively to get a better and better model.

A major task involved in active learning is to decide whether or not query a data point, that is to decide if the gain from labelling a point is worth the cost of collecting that information. The two strategies used in active learning algorithms are

a) stream based sampling

b) pool based sampling

Stream based sampling

In this method, each training example is considered separately for making the decision of whether it would be sufficiently beneficial to query this data point or not. Similar to online learning, the model immediately decides whether to consider or reject the point. The data is not saved. No assumptions are made about the data distribution. The decision is made considering the informativeness of the example taken one at a time.

Pool based sampling

In this method, an informative measure is applied on the entire dataset and the instances to be labelled are chosen from the entire data pool. It evaluates the complete unlabeled dataset before making the decision of ignoring or choosing a point in the set of best queries.

Querying strategy

Active learning allows faster learning by incrementally and dynamically label the data using an informative measure.

The approach used to determine which training example(s) should be labeled next is known as querying strategy. The two common querying strategies are

a) Uncertainty sampling

b) Query-by-Committee (QBC)

These strategies are explained in the detail as follows:

Uncertainty Sampling

Least confident

In this method, the highest probability for each data point is chosen and sorted from smallest to largest. The example with lowest confidence level is chosen as a query point to be labeled next. The prioritization is done using least confidence based on formula:

$$s_{LC} = \underset{x}{\operatorname{argmax}} \left(1 - P(\hat{y}|x)\right)$$

Margin Sampling

This method incorporates the difference between highest and second highest probability. The training examples with least margin sampling score are chosen to be labeled first because these are most uncertain between the first and second most probable class label. The instance with the most narrow margin between the top two most likely classes is chosen. The formula used for prioritization is:

$$s_{MS} = \underset{x}{\operatorname{argmin}} \left(P(\hat{y}_{max}|x) - P(\hat{y}_{max-1}|x) \right)$$

Entropy-based

Entropy term means the degree of disorder in a system. The higher the entropy value, the more is the measure of disorder. This measure can be used to get an idea about certainty of the model. We prioritize the data points with high entropy compared to low entropy using the following formula:

$$s_E = \underset{x}{\operatorname{argmax}} \left(-\sum_i P(\hat{y}_i|x) \log P(\hat{y}_i|x) \right)$$

Query by Committee (QBC)

QBC is based on the idea that there is a committee of classifiers and the instance that the committee members disagree the most is queried. Bagging and boosting techniques can be used to create a committee. The disagreement among the committee classifiers can be measured using either XOR or some measure as vote entropy and KL divergence

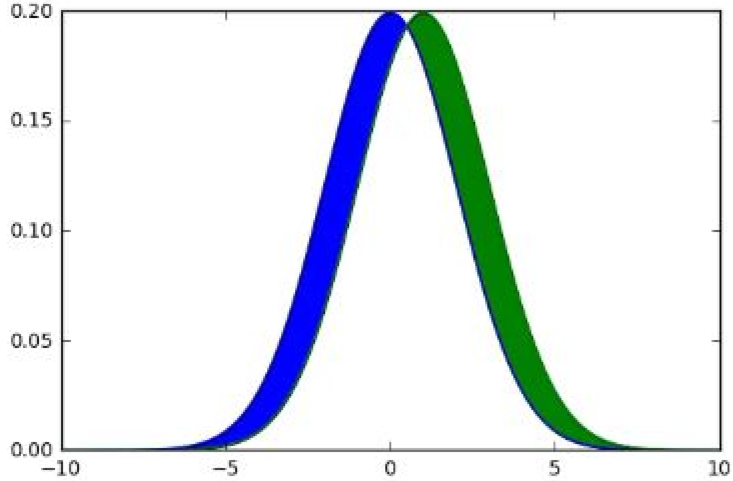
Vote Entropy

Let's say we have a committee with x classifiers and there are y instances to label. In order to calculate the vote entropy, first we let every classifier make the predictions. Each example will be associated with a probability distribution. Vote entropy selects the training example for which the entropy of this vote distribution is the largest.

KL divergence

The KL (Kullback-Leibler) divergence score quantifies the difference between two probability distributions commonly used in the form of cross entropy. The aim is to choose a data point to query that maximizes the KL divergence between posterior and prior. It is not symmetric but is useful in case of complex distributions.KL divergence can brief data scientists as to whether a distribution is good at approximating data or not. The lower the KL divergence is, more closer are the distributions to one another. Hence, we can certainly estimate the parameters of a Gaussian distribution using the value of its divergence with another known distribution. Generally, we try to minimize KL divergence using Gradient Descent and it considers the input data that have a sum of one as proper probability distributions.

In the graph, the areas where these two distributions do not overlap are shaded.



1. a. Creating a labelled dataset consisting of randomly chosen 10% points from the original dataset.

   Dataset Used: Digits Dataset
   Size of dataset: 1797 data points
   Number of classes in dataset: 10

   For implementation of active learning, we have used modAL, an active learning framework for Python3.

   Citation:
```
@article{modAL2018,
            title={mod{AL}: {A} modular active learning framework for {P}ython},
            author={Tivadar Danka and Peter Horvath},
           url={https://github.com/cosmic-cortex/modAL},
            note={available on arXiv at \url{https://arxiv.org/abs/1805.00979}}
        }
```

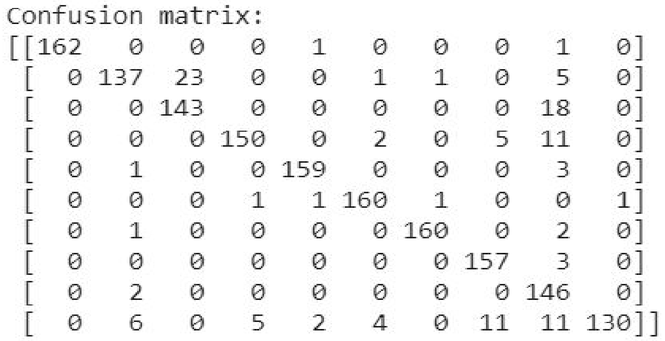
b. i) Uncertainty Sampling

Classifier used: K Nearest Neighbours for K=3

Pool based

A. Margin Sampling

We used the k-Nearest Neighbours classifier to train on the labelled dataset with 10% data points. The accuracy obtained on considering the remaining 90% data points as a test set was 92%.
The confusion matrix is shown below.



```
Confusion matrix:
[[162   0   0   0   1   0   0   0   1   0]
 [  0 137  23   0   0   1   1   0   5   0]
 [  0   0 143   0   0   0   0   0  18   0]
 [  0   0   0 150   0   2   0   5  11   0]
 [  0   1   0   0 159   0   0   0   3   0]
 [  0   0   0   1   1 160   1   0   0   1]
 [  0   1   0   0   0   0 160   0   2   0]
 [  0   0   0   0   0   0   0 157   3   0]
 [  0   2   0   0   0   0   0   0 146   0]
 [  0   6   0   5   2   4   0  11  11 130]]
```

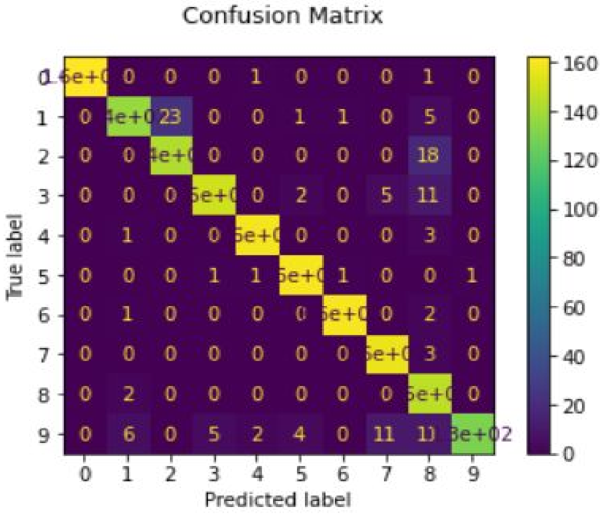


Fig.1 - Confusion matrix for test set (90% unlabelled data points)

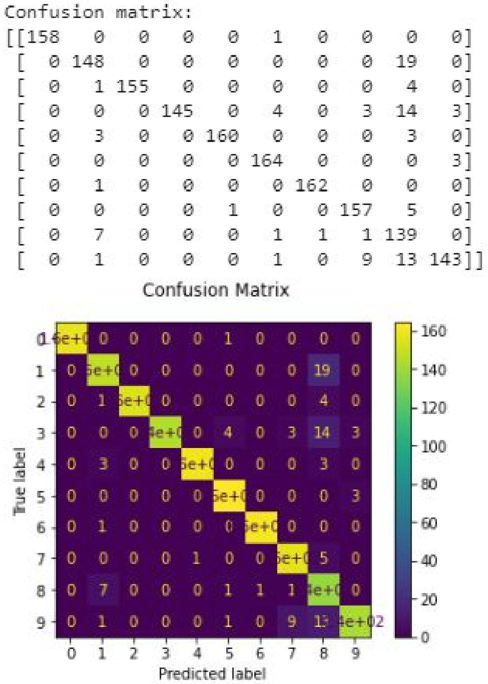
On querying data points from the pool we observe an improvement in accuracy which is tabulated below.

Queried Points	10% i.e. 179 points	20% i.e. 358 points	30% i.e. 537 points	40% i.e. 716 points
Accuracy (Initial: 92%)	98.55%	98.55%	98.94%	99.05%

B) Entropy based

We used the k-Nearest Neighbours classifier (k=3) to train on the labelled dataset with 10% data points. The accuracy obtained on considering the remaining 90% data points as a test set was 92.98%.

The confusion matrix is shown below.



```
Confusion matrix:
[[158   0   0   0   0   1   0   0   0   0]
 [  0 148   0   0   0   0   0   0  19   0]
 [  0   1 155   0   0   0   0   0   4   0]
 [  0   0   0 145   0   4   0   3  14   3]
 [  0   3   0   0 160   0   0   0   3   0]
 [  0   0   0   0   0 164   0   0   0   3]
 [  0   1   0   0   0   0 162   0   0   0]
 [  0   0   0   0   1   0   0 157   5   0]
 [  0   7   0   0   0   1   1   1 139   0]
 [  0   1   0   0   0   1   0   9  13 143]]
```

Fig. : Confusion matrix for test set (90% unlabelled data points)

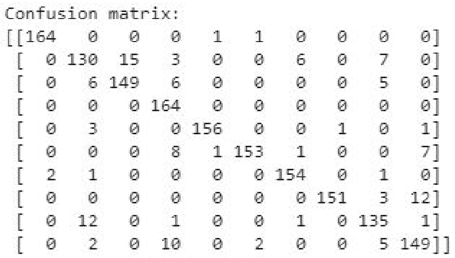
On querying data points from the pool we observe an improvement in accuracy which is tabulated below.

Queried Points	10% i.e. 179 points	20% i.e. 358 points	30% i.e. 537 points	40% i.e. 716 points
Accuracy (Initial: 92.98%)	98.66%	99.28%	99.28%	99.22%

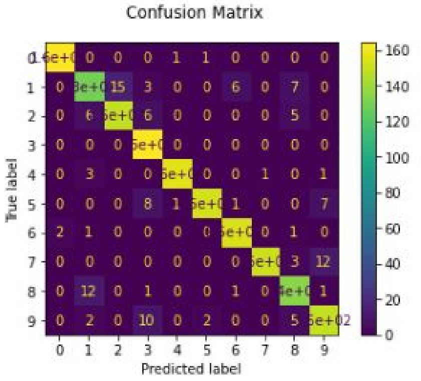
C) Least Confident sampling

We used the k-Nearest Neighbours classifier (k=3) to train on the labelled dataset with 10% data points. The accuracy obtained on considering the remaining 90% data points as a test set was 90.48%.

Confusion matrix is shown below.



```
Confusion matrix:
[[164   0   0   0   1   1   0   0   0   0]
 [  0 130  15   3   0   0   6   0   7   0]
 [  0   6 149   6   0   0   0   0   5   0]
 [  0   0   0 164   0   0   0   0   0   0]
 [  0   3   0   0 156   0   0   1   0   1]
 [  0   0   0   8   1 153   1   0   0   7]
 [  2   1   0   0   0   0 154   0   1   0]
 [  0   0   0   0   0   0   0 151   3  12]
 [  0  12   0   1   0   0   1   0 135   1]
 [  0   2   0  10   0   2   0   0   5 149]]
```



Confusion Matrix

Queried Points	10% i.e. 179 points	20% i.e. 358 points	30% i.e. 537 points	40% i.e. 716 points
Accuracy (Initial: 90.48%)	98.44%	99.11%	99.17%	99.22%

Comparison of different uncertainty measures namely margin sampling, entropy sampling and least confident sampling

Considering 10% additional points for querying, entropy based measure performs the best, followed by margin sampling and least confidence. When the number of additional queried data points is increased from 10% to 20%, the most improvement in accuracy is seen in the case of least confident whereas margin sampling doesn’t show any improvement. The values of accuracy for the three uncertainty sampling measures is almost similar. The accuracy increases when the number of additional labeled points is increased. Finally, with 40% additional data points, entropy based and least confident both shows highest, that is, 99.22% accuracy.

Stream based

A. Margin Sampling

Queried Points	10% i.e. 179 points	20% i.e. 358 points	30% i.e. 537 points	40% i.e. 716 points
Accuracy (Initial: 92.15%)	98.72%	98.88%	98.83%	99.1%

B. Entropy Sampling

Queried Points	10% i.e. 179 points	20% i.e. 358 points	30% i.e. 537 points	40% i.e. 716 points
Accuracy (Initial: 92.2%)	98.77%	98.88%	98.88%	99.16%

C. Least Confident

The initial accuracy was 93.37%
On querying points from the 90% unlabelled set with a threshold uncertainty value of 0.4, accuracy improved to 96.21%. This resulted in an addition of 48 points.

The threshold was reduced to 0.3 in order to add more points.
On querying 179 points i.e. 10% data points, the accuracy improved to 99.05%.
On addition of 20% data points, accuracy obtained was 98.99%.

Queried Points	10% i.e. 179 points	20% i.e. 358 points	30% i.e. 537 points	40% i.e. 716 points
Accuracy (Initial: 93.37%)	99.05%	98.99%	98.94%	99.1%

Comparison of different measures of uncertainty sampling
Initial addition of 10% data points resulted in a significant jump from ~92% to ~98.7% for margin sampling and entropy sampling and 93.37% to 99.05% in the case of least confident. The accuracy remained almost constant in the case of least confident measure whereas accuracy gradually increased in margin and entropy based. The final results for 40% additional labeled data points, all three measures showed similar values, that is, 99.1% for least confident and margin sampling and 99.16% for entropy based measure.

ii) QBC

A committee of 5 members was created using the following classifiers.

-Random Forest Classifier with maximum depth 5
-Adaboost classifier
-Decision tree classifier with maximum depth 5
-K nearest neighbours with k=3
-Gaussian Process classifier

Pool based

Vote Entropy measure

Applying PCA on digits dataset to reduce dimensionality to 2 components.

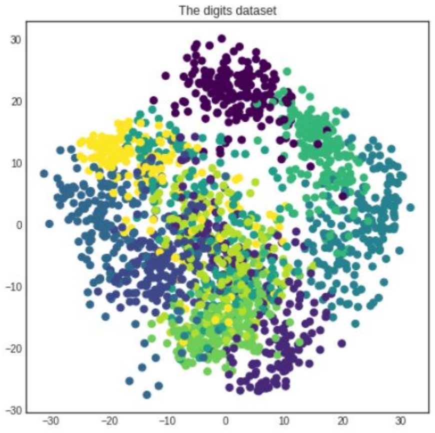


Fig 2. Digits dataset after applying PCA (number of components = 2)

Accuracy obtained on 90% unlabeled data using the committee of classifiers was 90%.

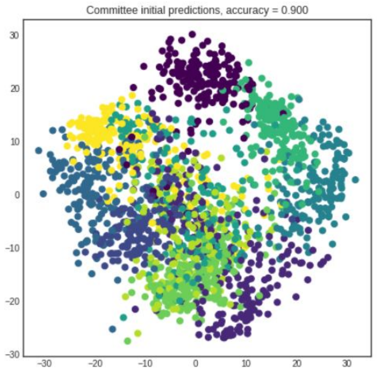


Fig.3 : Committee’s initial predictions on unlabelled data

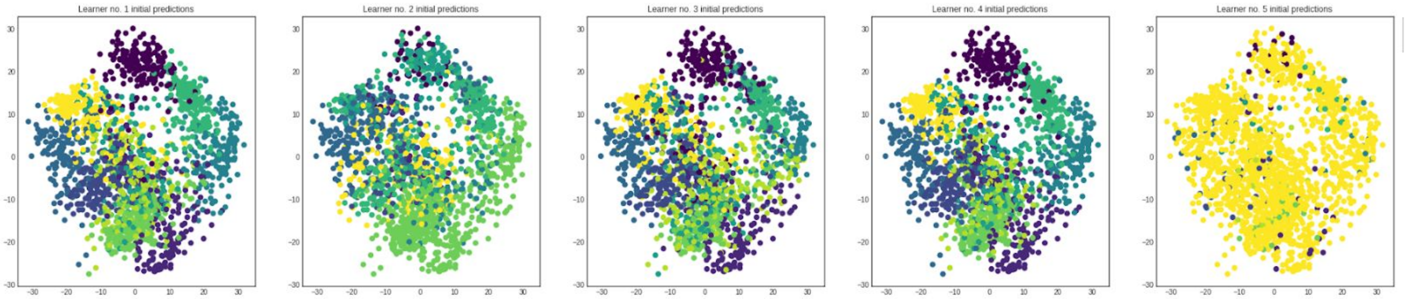


Fig.4 : Initial predictions of each member of committee on unlabelled data

Addition of 10% data points

After querying 10% data points the following results are obtained.
Accuracy observed after querying 179 data points from the pool i.e. 10% data points was 98.7%.

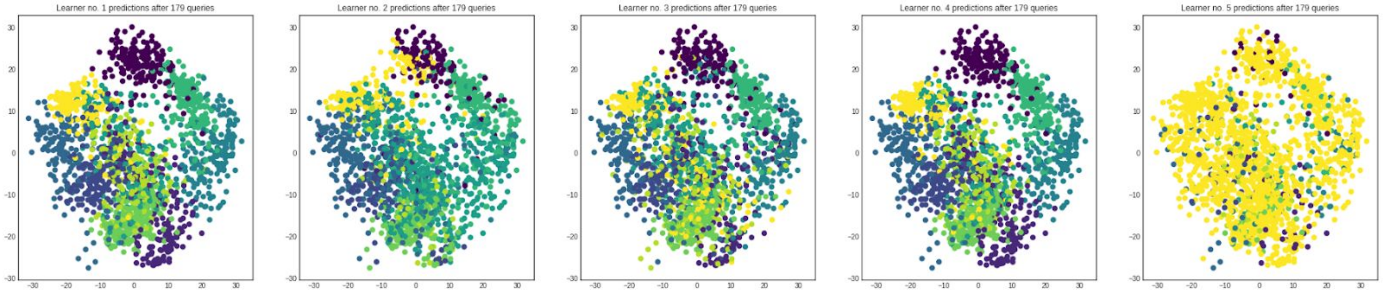


Fig 5: Predictions of each member of committee after querying 10% data points from pool.

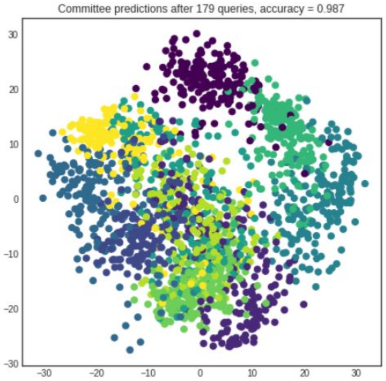


Fig.6 : Committee’s predictions after querying 10% data points from the pool.

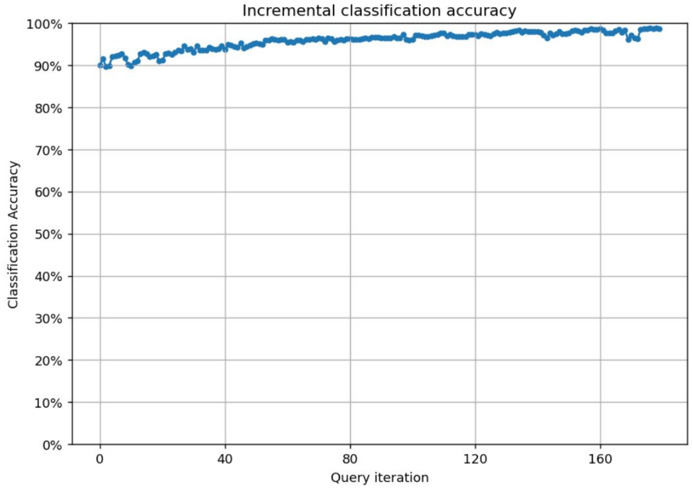


Fig. 7 : Plot of incremental classification accuracy after each query

Addition of 20% data points

After querying 20% data points the following results are obtained.
Accuracy observed after querying 358 data points from the pool i.e. 20% data points was 98.7%.

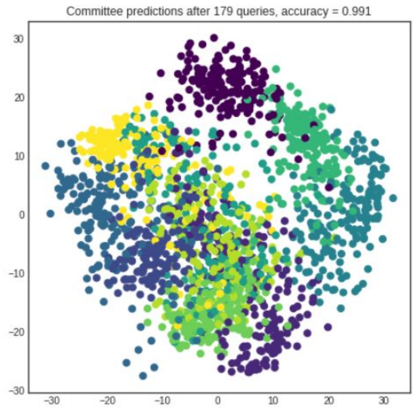


Fig.6 : Committee’s predictions after querying additional 10% i.e 20% data points from the pool.

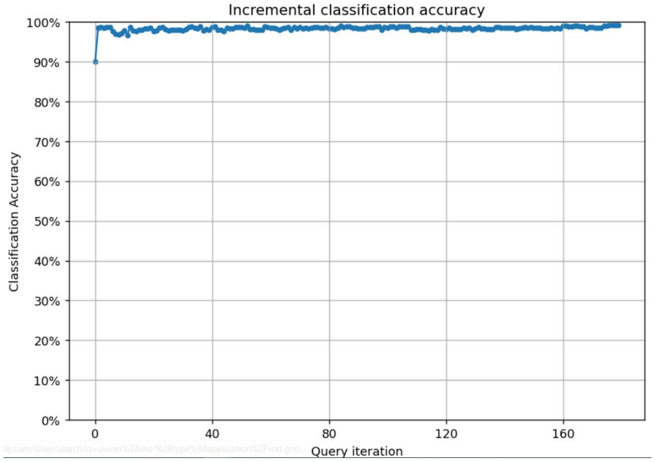


Fig. 7 : Plot of incremental classification accuracy after each query on addition of 10% more data points.

<u>Addition of 30% data points</u>

After querying 30% data points the following results are obtained.

Accuracy observed after querying 537 data points from the pool i.e. 30% data points was 98.9%.

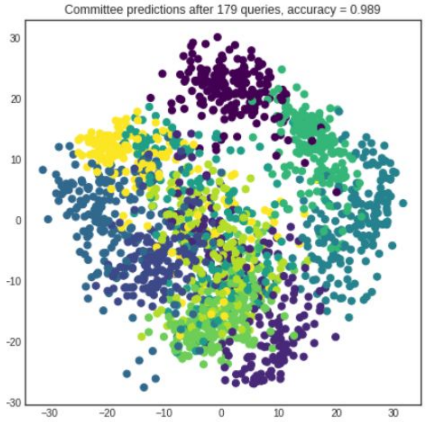


Fig.8 : Committee's predictions after querying additional 10% i.e 30% data points from the pool.

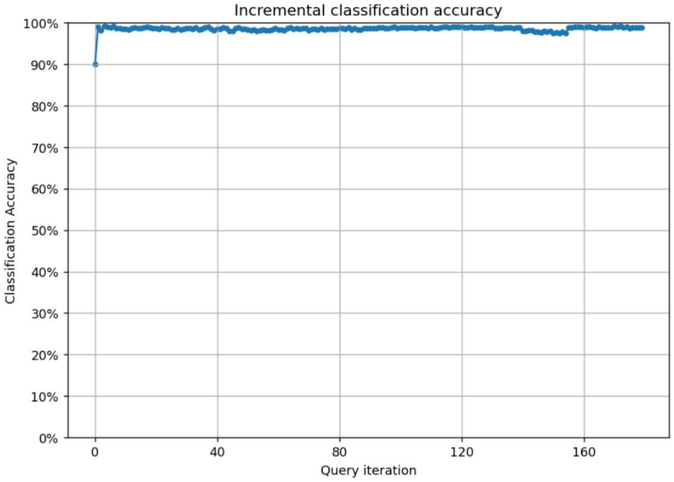


Fig. 9 : Plot of incremental classification accuracy after each query on addition of 10% more data points.

Addition of 40% data points

After querying 40% data points the following results are obtained.

Accuracy observed after querying 716 data points from the pool i.e. 40% data points was 99.1%.

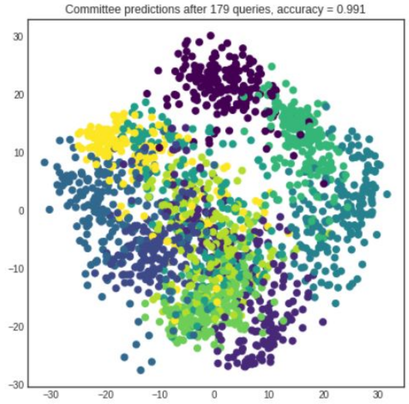


Fig.10 : Committee’s predictions after querying additional 10% i.e 40% data points from the pool.

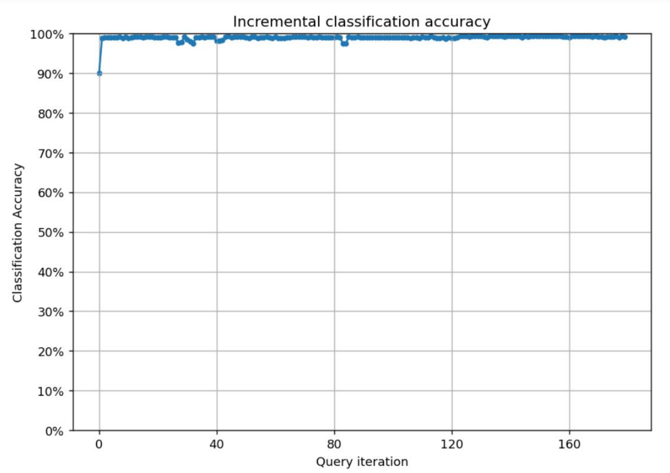


Fig. 11 : Plot of incremental classification accuracy after each query on addition of 10% more data points.

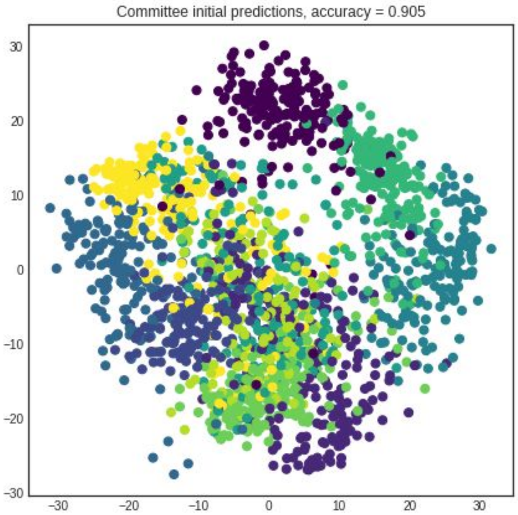
A summary of the above results is shown below.

Additional data points added	10% of original data points	20% of original data points	30% of original data points	40% of original data points
Accuracy (Initial accuracy 90%)	98.7%	99.1%	98.9%	99.1%

From the above results we conclude that on querying 10% data points we see a significant improvement in accuracy from 90% to 98.7%. On querying more points the accuracy shows little variation.

KL Divergence

The initial accuracy of the committee on training with 10% data points from the digits dataset was 90.5%.



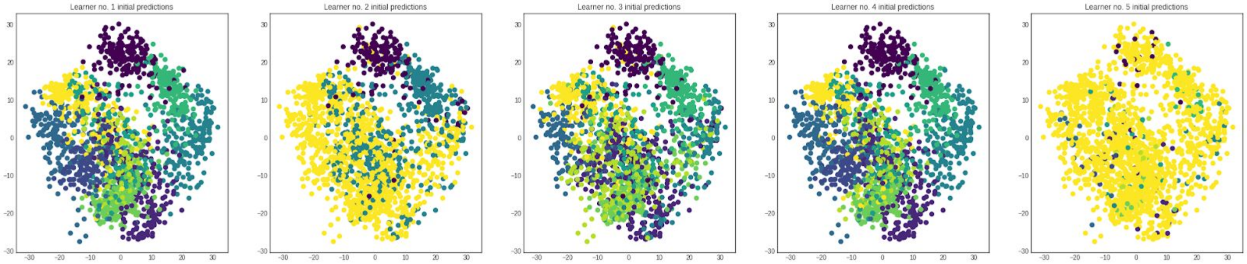


Fig 12 : Initial predictions of each learner

<u>Addition of 10% data points</u>

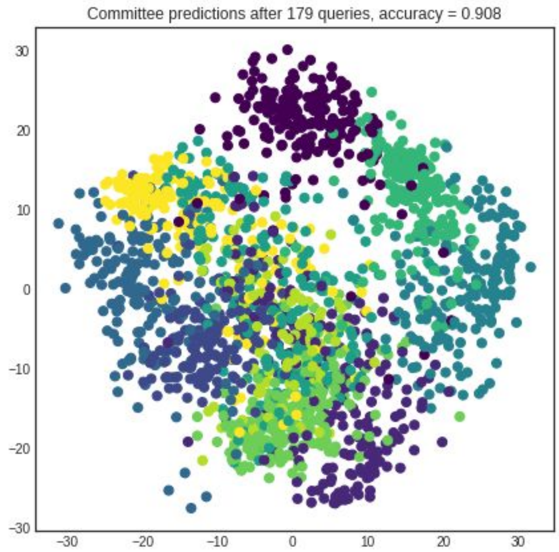


Fig 13

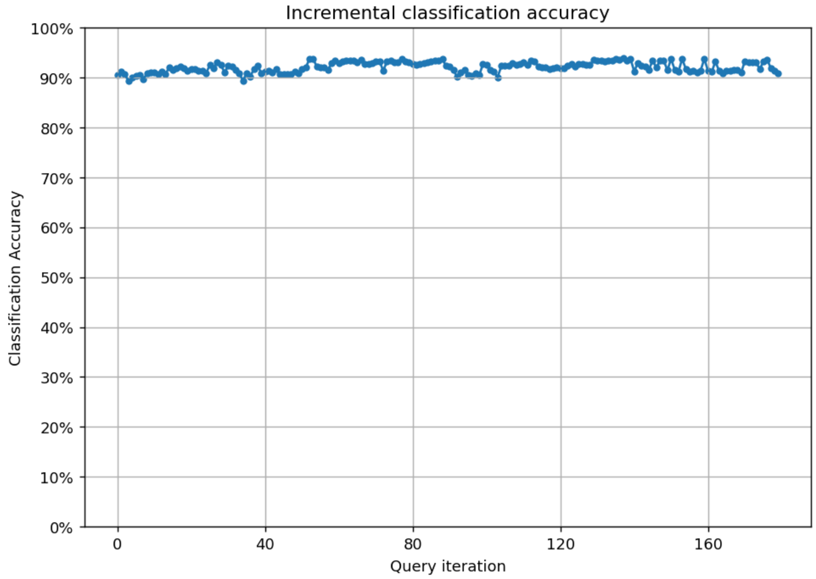


Fig 14

<u>Addition of 20% data points</u>

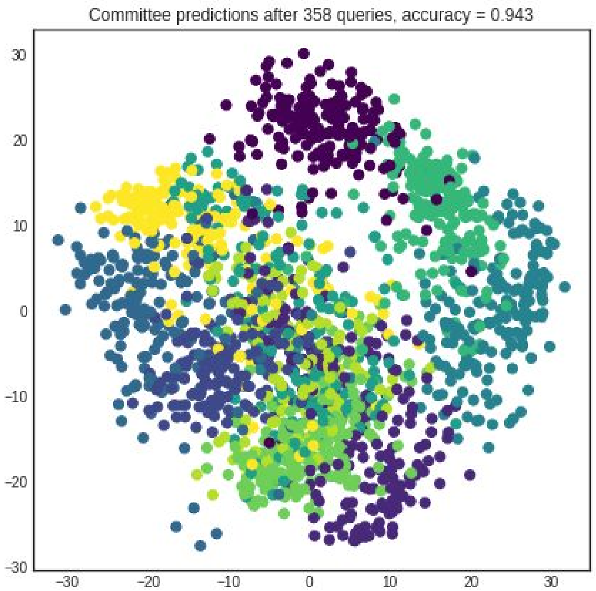


Fig 15

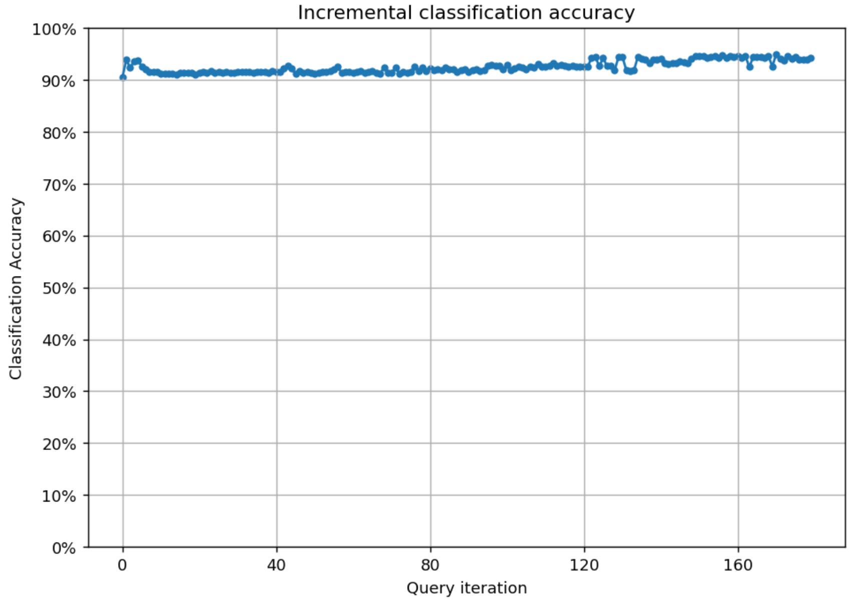


Fig 16

<u>Addition of 30% data points</u>

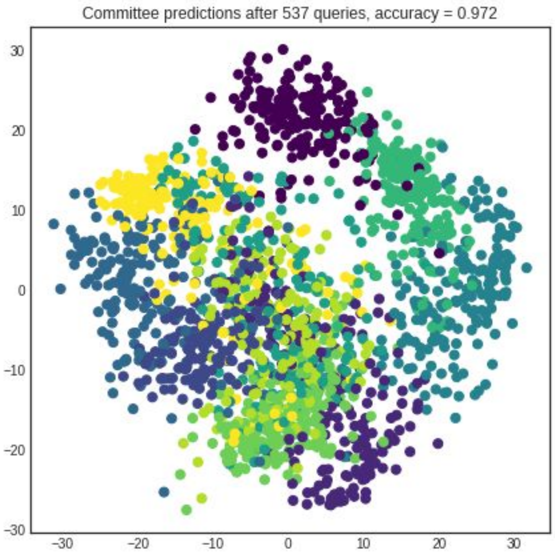


Fig 17

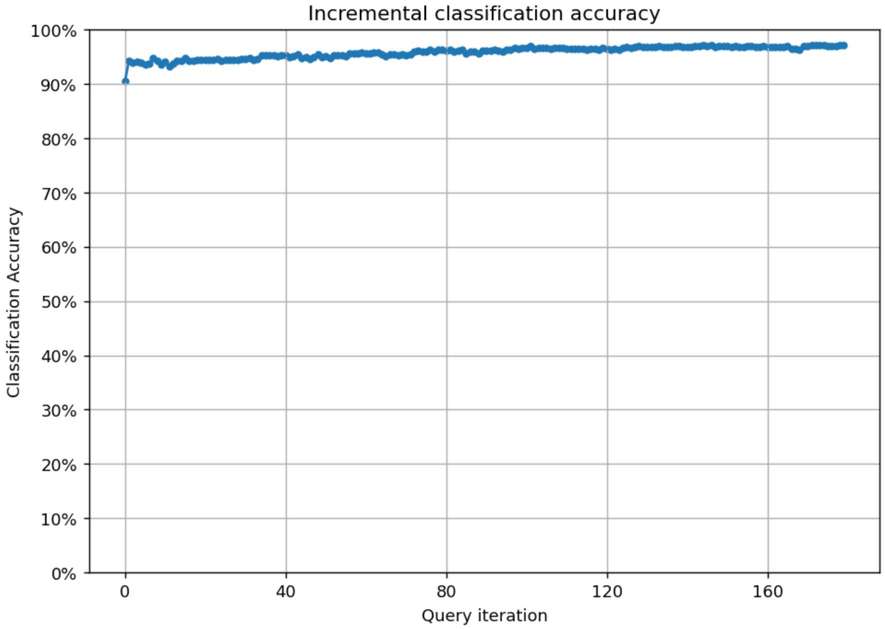


Fig 18

<u>Addition of 40% data points</u>

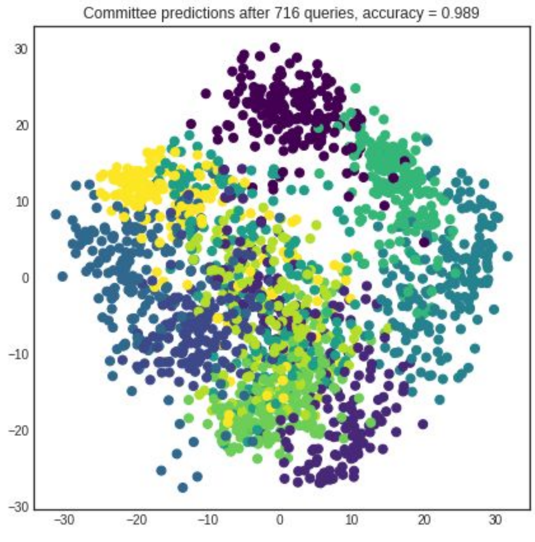


Fig 19

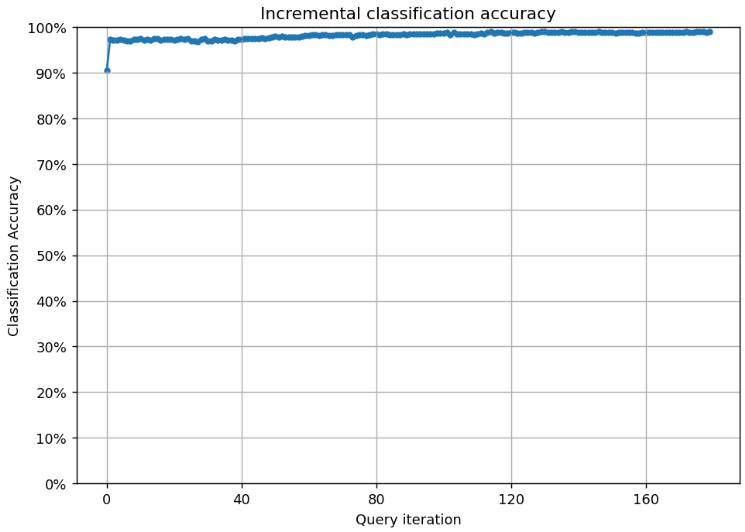


Fig 20

A summary of the above results is shown below.

Additional data points added	10% of original data points	20% of original data points	30% of original data points	40% of original data points
Accuracy (Initial accuracy 90.5%)	90.8%	94.3%	97.2%	98.9%

Comparison of results obtained using Vote Entropy and KL Divergence for Query By committee using Pool based sampling

When vote entropy is used as a disagreement measure for querying, 10% addition of data points gave a significant jump in accuracy from 90% to 98.7% after which addition of data points showed slight variations in accuracy. However, when KL divergence was used as a disagreement measure,a more gradual increase in accuracy was observed on addition of data points. Hence, an observation that we make is that using vote entropy as a measure for disagreement for querying helps in improving accuracy of the committee with lesser number of queries.

Stream based

Vote Entropy

Using the committee of classifiers defined earlier, the initial accuracy obtained was 90.8%.

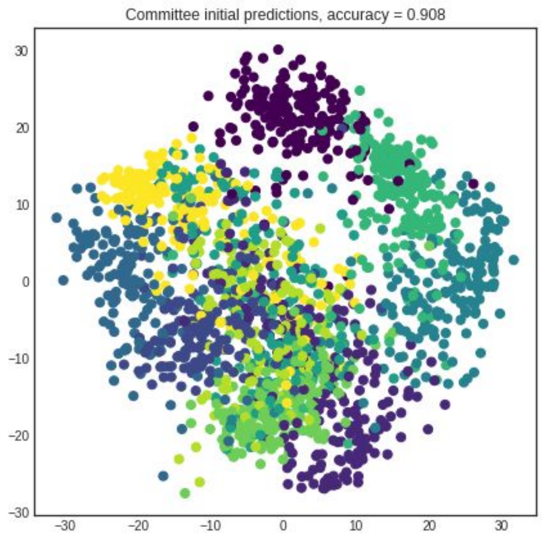
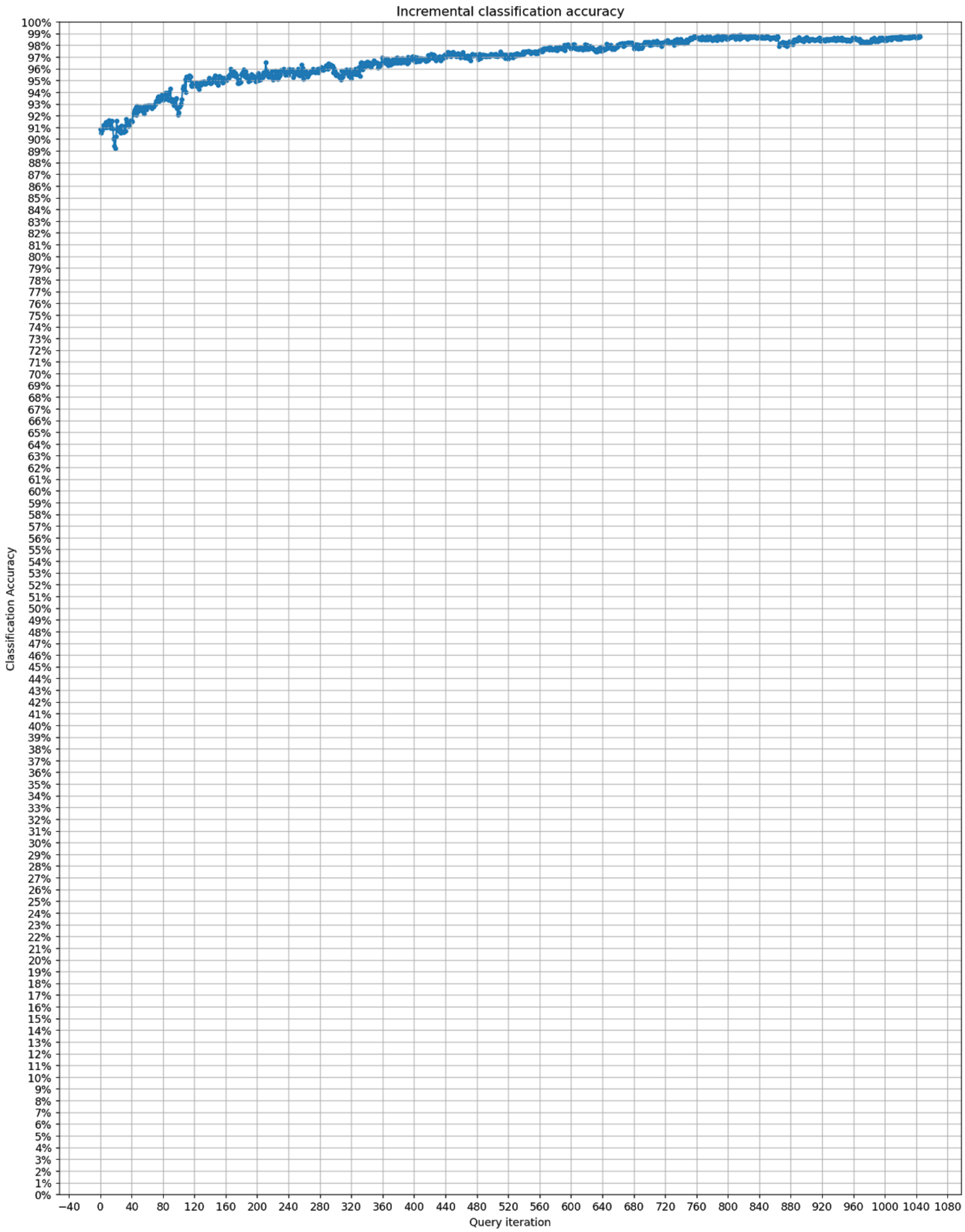


Fig. 21 : Committee’s initial predictions

Using the vote entropy disagreement measure, data points having disagreement value greater than 0.5 were queried and the committee learned from those queries.



Incremental classification accuracy
Classification Accuracy
Query iteration
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
−40
0
40
80
120
160
200
240
280
320
360
400
440
480
520
560
600
640
680
720
760
800
840
880
920
960
1000
1040
1080

Fig 22: Representation of improvement in accuracy with each query using stream based sampling

Additional data points added	10% of original data points	20% of original data points	30% of original data points	40% of original data points
Accuracy (Initial: 90.8%)	95%	97%	97.5%	97.5%

We observe that on addition of 20% data points a significant improvement in accuracy is observed from 90.8% to 97%.

KL Divergence

Using the committee of classifiers defined earlier, the initial accuracy obtained was 90.15%.

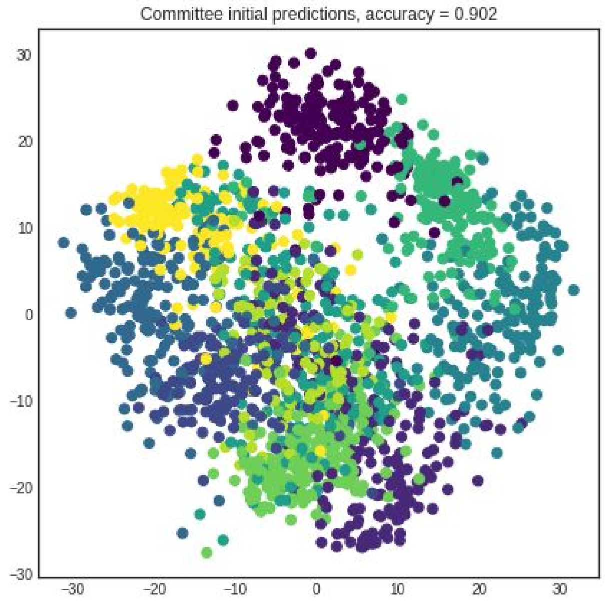


Fig. 23 : Committee's initial predictions

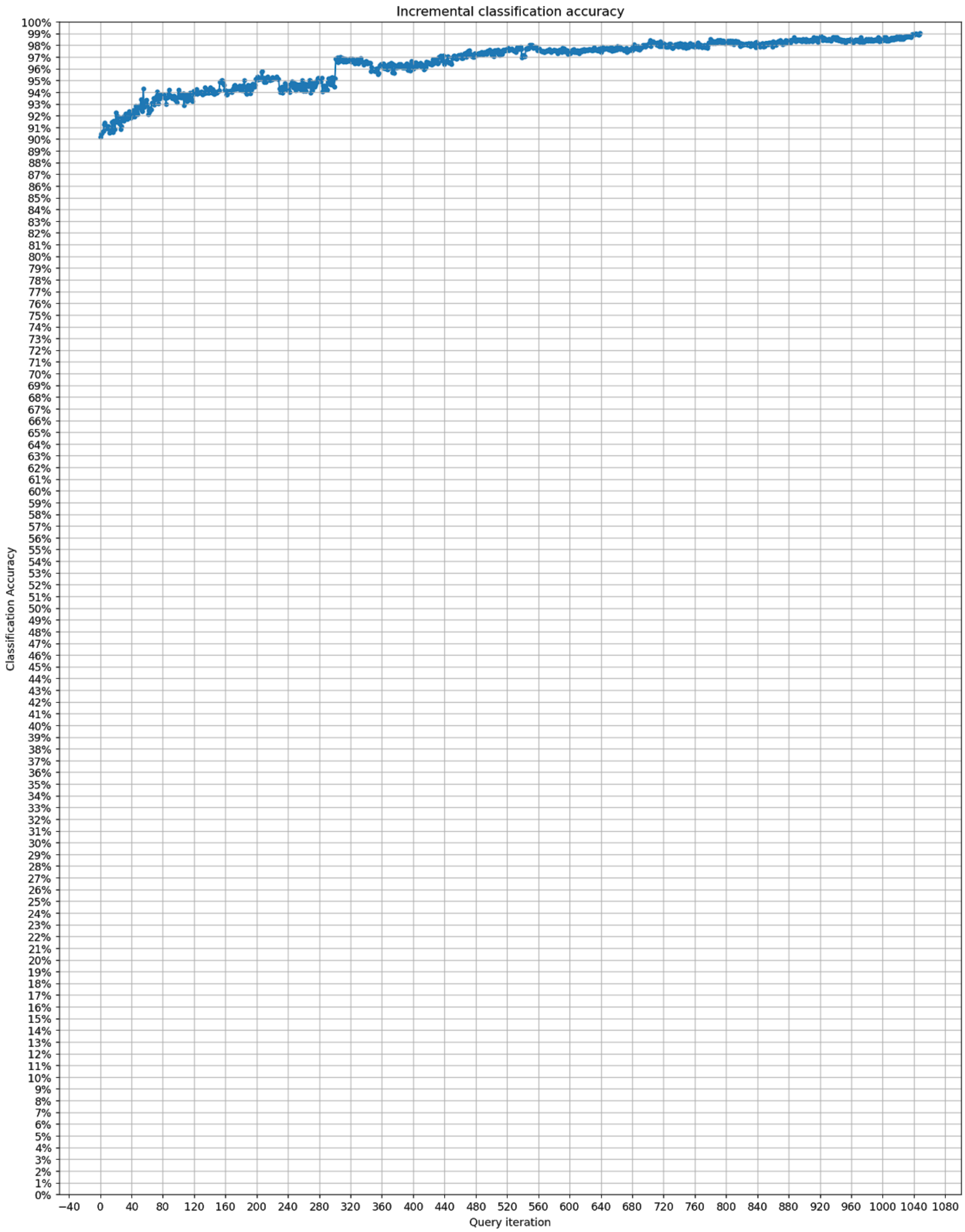


Fig 24: Representation of improvement in accuracy with each query using stream based sampling

Additional data points added	10% of original data points	20% of original data points	30% of original data points	40% of original data points
Accuracy (Initial: 90.15%)	94%	96.5%	98%	98%

Comparison for KL divergence and vote entropy measure

When vote entropy is used as a disagreement measure for querying, 10% addition of data points gave a jump from 90.8% to 95% accuracy. When KL divergence was used as a disagreement measure, a jump from 90.15% to 94% accuracy was seen by adding 10% data points. However, at 40% additional labelled data points, almost same results were observed, 98% in KL divergence and 97.5% in vote entropy.

iii) Version space

Version space consists of the set of classifiers that are consistent with the labelled examples. The points with larger disagreement have a larger contribution in reducing the version space on querying.

Vote entropy was considered as a disagreement measure to calculate the number of points in version space. The number of points in the version space are 1752.

The order points to be chosen to label in order to reduce the version space by maximum are shown with reference to their indices in the digits dataset here.

iv) Comparison of active and passive learning

1) Choosing the best uncertainty sampling pool based model (Entropy based, 40% additionally labeled points), accuracy = 99.22%
Randomly selected 40% points to label, accuracy = 97.94%

2) Choosing the best uncertainty sampling stream based model (Entropy based, 40% additionally labeled points), accuracy = 99.16%
Randomly selected 40% points to label, accuracy = 97.94%

3) Choosing the best QBC pool based model (Vote entropy based, 40% additionally labeled points), accuracy = 99.1%
Randomly selected 40% points to label, accuracy = 97.44%

4) Choosing the best QBC stream based model (KL divergence based, 40% additionally labeled points), accuracy = 98%
Randomly selected 40% points to label, accuracy = 97.44%

Conclusion: Keeping the number of labeled instances the same, active learning outperforms passive learning, that is, choosing the data points to be queried for labelling is a smart choice.

v) K means clustering

K means is a clustering algorithm where 'K' refers to the desired number of clusters and 'means' refers to finding out the centroid using averaging. In this algorithm, k random points are selected as centroids and every other point is assigned to one of these clusters depending upon the nearest centroid. After the clusters are formed, new centroids are calculated and the process is repeated till a specific number of iterations or till the centroids don't change much.

Results

90% data points were chosen randomly from the entire dataset and were unlabelled. From these 90% data points, 40% points were randomly chosen.

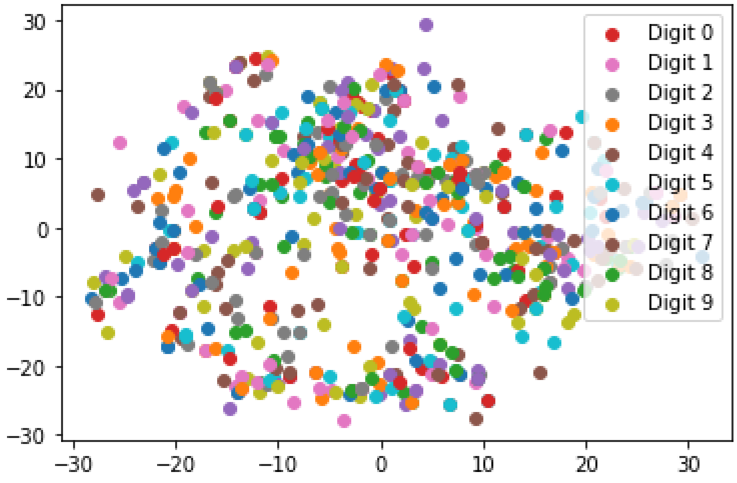


Figure 25: Dimensionally reduced representation of data before clustering

Then, K-means clustering was used to form clusters.

20% points of each cluster were labelled. Based on the maximum occurring label for a particular cluster, all the points of the cluster were labelled.
Results are shown below.

Cluster Number	Number of points in cluster	Label
0	81	7
1	54	6
2	73	8
3	33	5

4	63	4
5	104	3
6	76	0
7	60	2
8	39	9
9	67	1

Accuracy obtained was 71.38%.

How much saving it results in if each label costs you Rs. 100 and each labelling takes one hour?

40% of 90% of total points = 650
By using the clustering method, we have only queried 20% of 650 points, that is, 130 points. If we had not used this technique, we would have queried all 650 points.
Using the above method, saved querying of 650-130 = 520 examples

2. Self-organizing map (SOM)

SOM was introduced by Finnish professor Teuvo Kohonen in the 1980s, thus it is also called a Kohonen map. A self-organizing map is a type of artificial neural network, based on competitive learning, trained using unsupervised learning, generally used to represent a high-dimensional dataset to a low dimensional representation called a map.

SOM uses a neighborhood function to perform topology preserving mapping to preserve the relative distance between data points. SOM produces a map where similar samples are mapped close to each other. SOM network consists of input and output layers. When input is fed to the network, output layer units compete and the winning neuron is the one whose connection weights are similar to the input data point. The winning neuron and sometimes the neighboring neurons get the chance to get its connection weight adjusted. Initially, random weights are assigned and slowly output units align themselves. This allows maps to grow into different forms and shapes, most commonly a square, rectangle, hexagon, or L shape in 2D space.

SOM RESULTS-

Dataset used for clustering -Wine Dataset

The Wine dataset used is observational and is obtained from the chemical analysis of wines grown in a particular region of Italy but cultivated by multiple cultivators. This dataset is used by various industries to group similar wines together and accordingly determine the various types(number of clusters) of wines supported by a region’s climatic factors such as temperatures and humidity. The results can also be used for recommending wines and segmenting wine drinkers, to judge as to what cluster of wine (and thereafter what proportions of chemicals) is the most popular.

The attributes are-

- Alcohol
- Malic acid
- Ash
- Alcalinity of ash
- Magnesium
- Total phenols
- Flavanoids
- Nonflavanoid phenols
- Proanthocyanins

- Color intensity
- Hue
- OD280/OD315 of diluted wines
- Proline

Citation of the library used-

Federico Comitani, 2019. fcomitani/SimpSOM: v1.3.4. doi:10.5281/zenodo.2621560

After activating periodic boundary conditions(with weights being initialised randomly), we trained the SOM network for 10000 epochs with an initial learning rate of 0.01 .

Map of the network nodes obtained is as shown below (weights on the axis correspond to column 0 values i.e. Alcohol)-

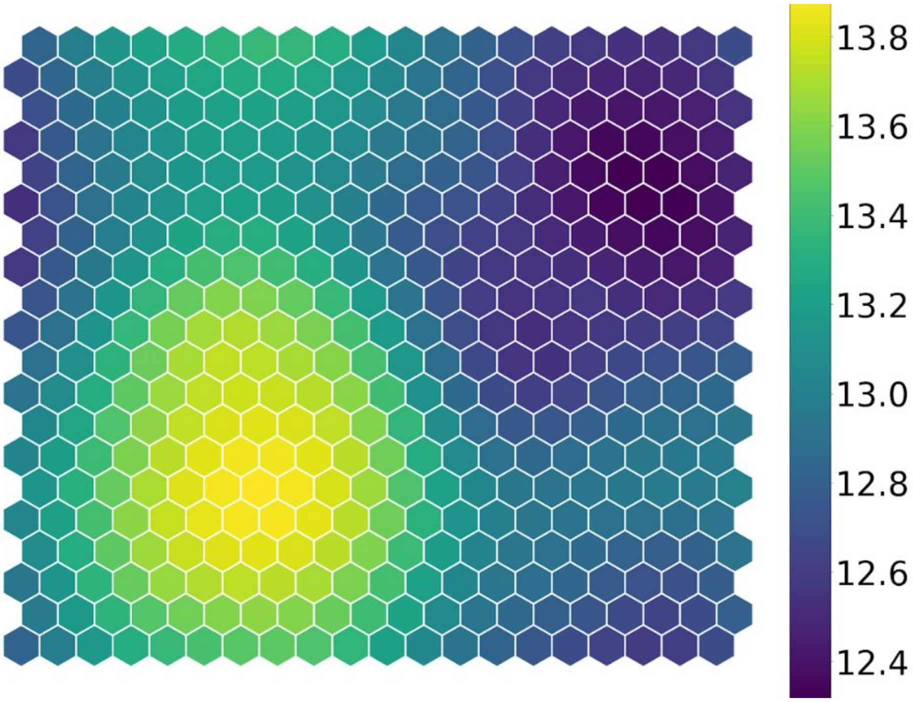


Fig 26

Here similar wines are mapped close together while other distinct ones are mapped relatively farther.

The accuracy of this mapping can be further improved by changing the map parameters and by training the SOM network for more epochs (and by also adjusting the learning rate).

Map of distance between each node and its respective neighbours is as illustrated below (with axis representing the weights difference)-

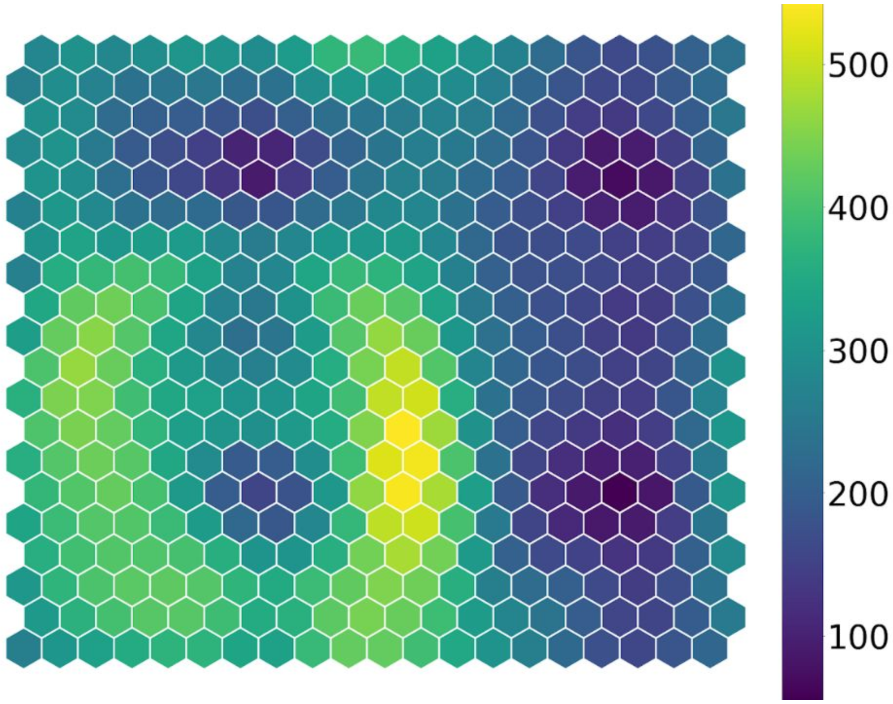
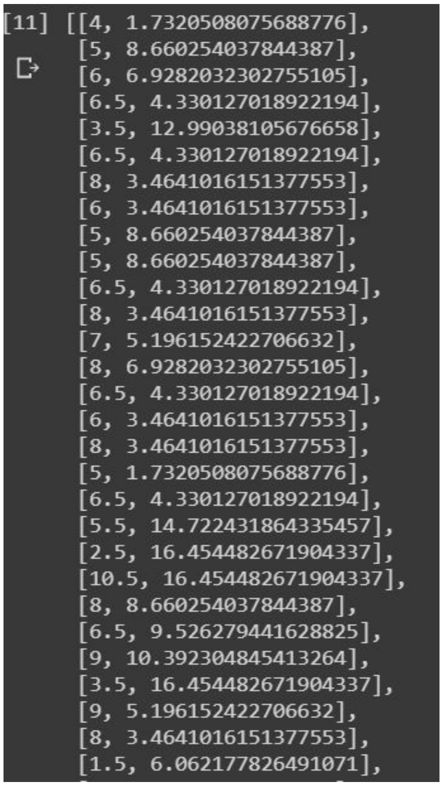


Fig 27

We then projected the data points on a two dimensional network frame. The snippet of acquired points is as shown-



```
[11] [[4, 1.7320508075688776],
      [5, 8.660254037844387],
      [6, 6.9282032302755105],
      [6.5, 4.330127018922194],
      [3.5, 12.99038105676658],
      [6.5, 4.330127018922194],
      [8, 3.4641016151377553],
      [6, 3.4641016151377553],
      [5, 8.660254037844387],
      [5, 8.660254037844387],
      [6.5, 4.330127018922194],
      [8, 3.4641016151377553],
      [7, 5.196152422706632],
      [8, 6.9282032302755105],
      [6.5, 4.330127018922194],
      [6, 3.4641016151377553],
      [8, 3.4641016151377553],
      [5, 1.7320508075688776],
      [6.5, 4.330127018922194],
      [5.5, 14.722431864335457],
      [2.5, 16.454482671904337],
      [10.5, 16.454482671904337],
      [8, 8.660254037844387],
      [6.5, 9.526279441628825],
      [9, 10.392304845413264],
      [3.5, 16.454482671904337],
      [9, 5.196152422706632],
      [8, 3.4641016151377553],
      [1.5, 6.062177826491071],
```

The data points have then been clustered according to their similarities in the origin space using the Quality Threshold Algorithm. The results have been presented below.

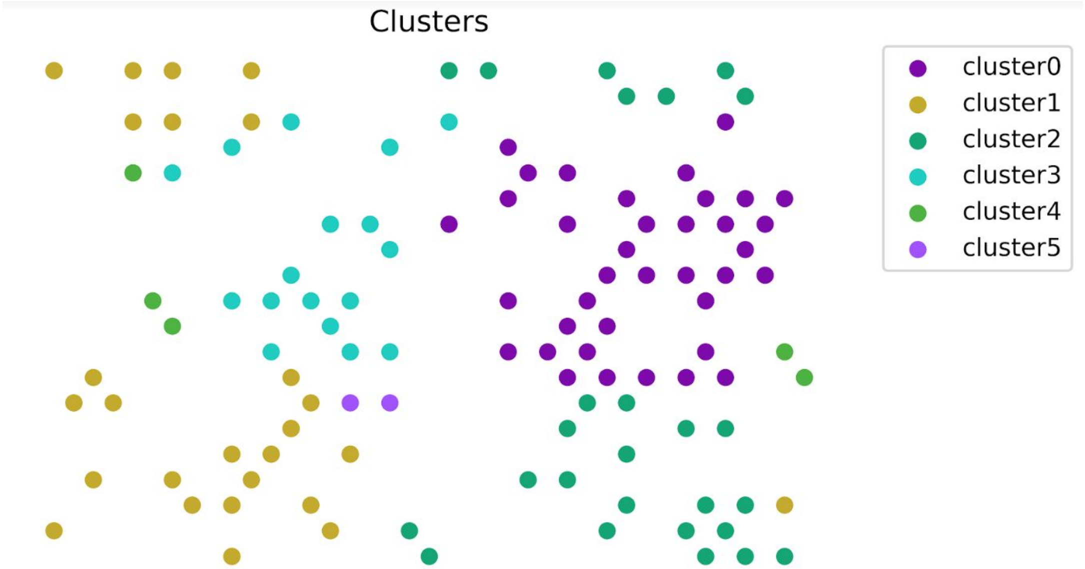


Fig 28