Active Learning Strategies DeepAL

Maymounah– 20190657
Ali Adel Ali – 20190337

5/5/2023

—

Selected Topics in AI_2

—

T.A/ Salah

Datasets

1- MNIST dataset:

The MNIST dataset contains 70,000 handwritten digits (0-9) that have been normalized and centered in a fixed-size image format of 28x28 pixels. The dataset is split into 60,000 training images and 10,000 test images. It is often used as a benchmark for image classification algorithms and is considered a relatively easy dataset to classify.

2- CIFAR10 dataset

The CIFAR-10 dataset consists of 60,000 color images in 10 classes, with 6,000 images per class. The images are 32x32 pixels in size, and each pixel is represented by three color channels (red, green, and blue). The dataset is split into 50,000 training images and 10,000 test images. The classes are mutually exclusive and include airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck.

3- MNIST dataset with Unbalanced

An unbalanced dataset is a dataset where the class distribution is not even or equal, meaning that one or more classes have significantly more instances than the others. This can cause problems for machine learning algorithms, as they may be biased towards the majority class andperform poorly on the minority class.

Strategies We Used:

1- Random sampling:

random sampling is a basic strategy for selecting data points to query the oracle. Random sampling in DeepAL involves randomly selecting a batch of unlabeled data points from the pool ofavailable data and presenting them to the oracle for labeling. The size of the batch is a parameter that can be tuned to balance the need for exploring the space of possible data points and exploiting the most informative ones.

2- Least Confidence:

Least Confidence is a type of uncertainty sampling used in the DeepAL (Modular Active Learning) framework. In uncertainty sampling, the goal is to select the samples that the model is least confident about for human annotation, so that the model can learn from the new labeled data.
In Least Confidence, the model is asked to predict the class probabilities for each sample, and the sample with the lowest maximum probability is selected for annotation. This means that the model is least confident about the most probable class for that sample.

3- Margin Sampling

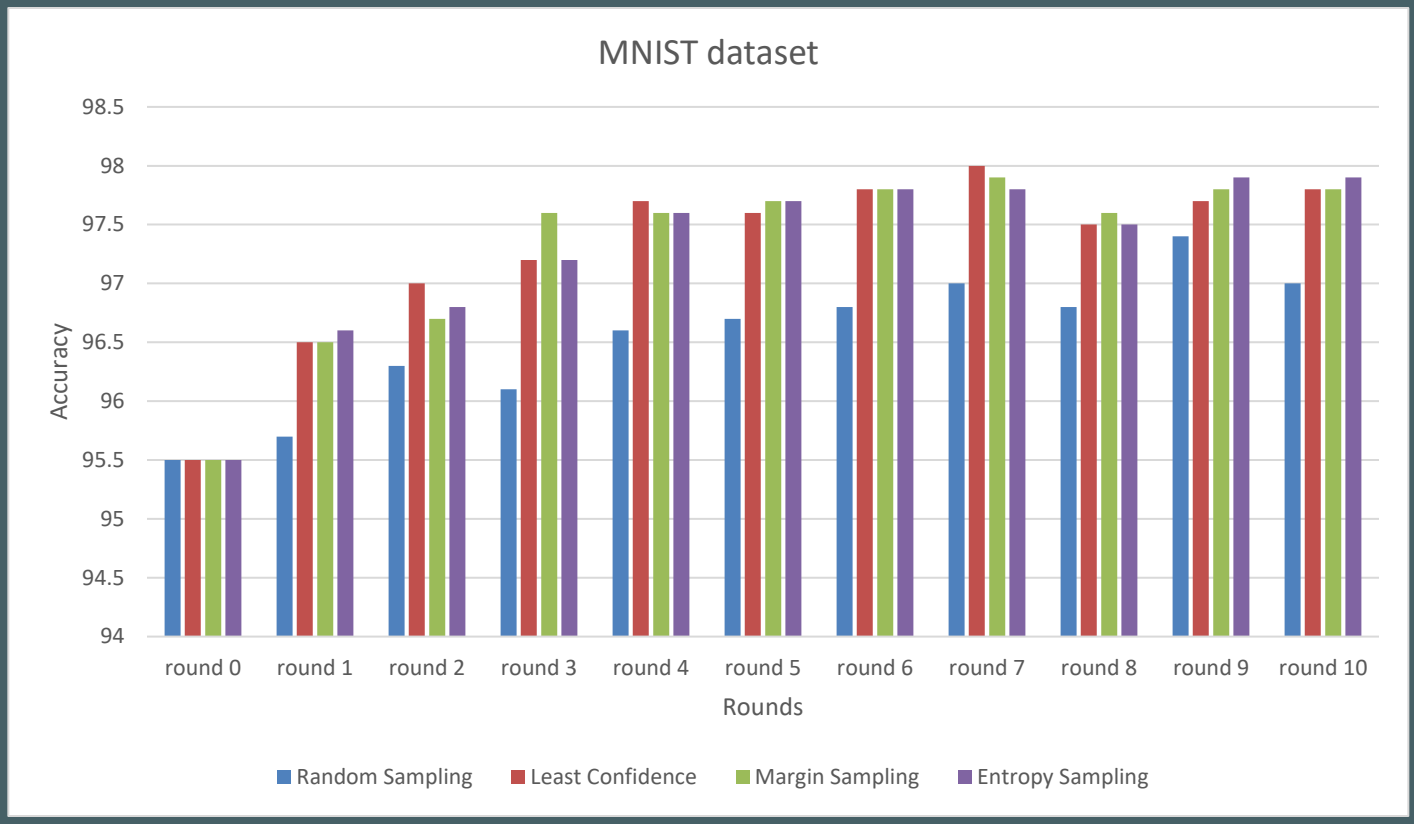
Margin sampling is an uncertainty-based active learning strategy that selects the data points that have the smallest difference between the top two predicted class probabilities. In other words, it selects the data points for which the model is most uncertain about which class they belong to.

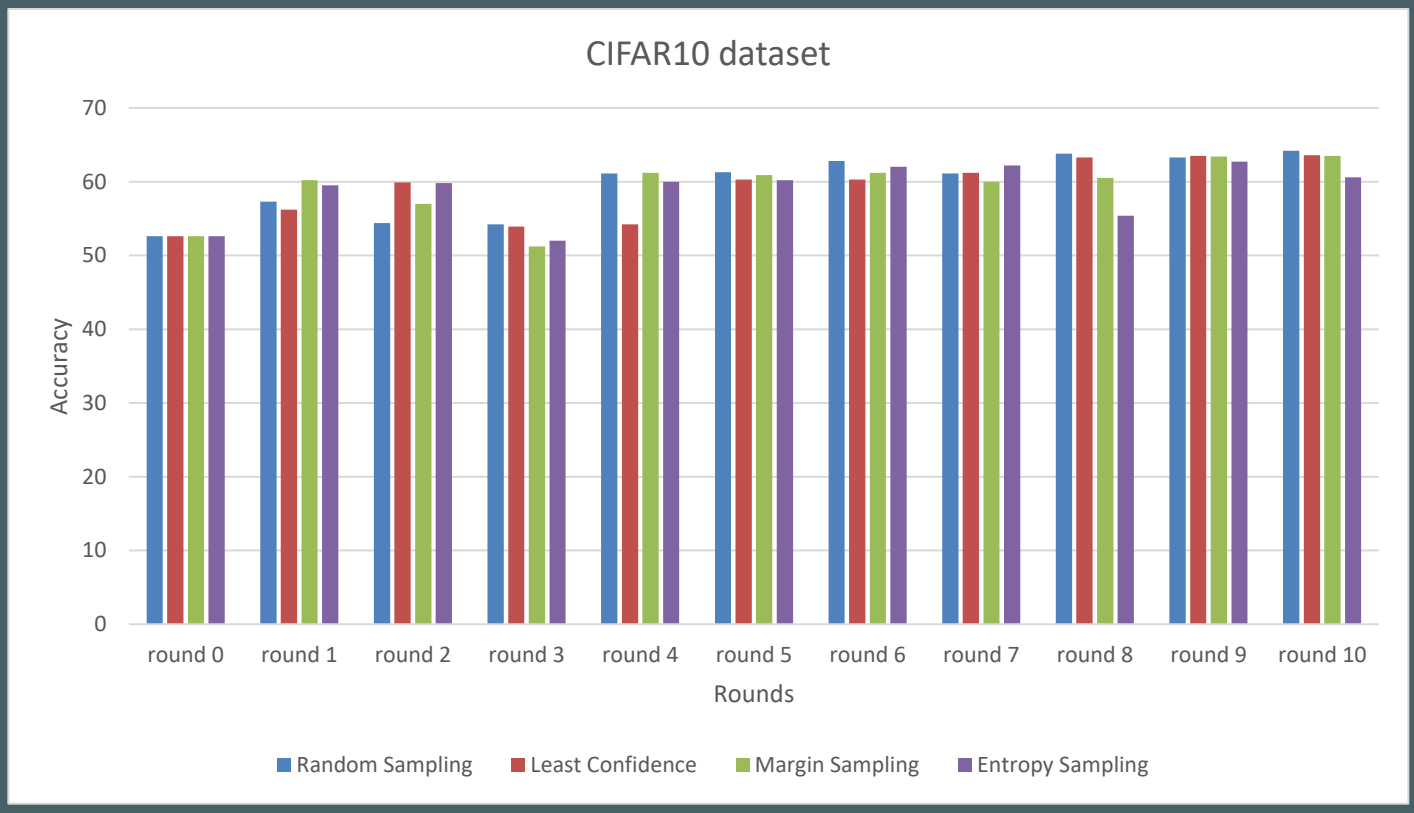
4- Entropy Sampling

Entropy sampling is an uncertainty-based active learning strategy that selects the data points with thehighest entropy

Steps For strategy:

1- **Read the dataset**
2- **Split the data to train/ validate/ test**
3- **Initialize an active learner object with specific strategy and a simple machine learning model**
4- **Find the performance for the model before apply querying**
5- **Apply active learning technique by querying on the validation dataset and save ourperformance after each query**.


MNIST dataset
98.5
98
97.5
97
96.5
96
95.5
95
94.5
94
Accuracy
round 0
round 1
round 2
round 3
round 4
round 5
round 6
round 7
round 8
round 9
round 10
Rounds
Random Sampling
Least Confidence
Margin Sampling
Entropy Sampling


CIFAR10 dataset
70
60
50
40
30
20
10
0
Accuracy
round 0
round 1
round 2
round 3
round 4
round 5
round 6
round 7
round 8
round 9
round 10
Rounds
Random Sampling
Least Confidence
Margin Sampling
Entropy Sampling

MNIST Dataset:

Strategy	n_round	# of queries	n_init_labeled	Final Accuracy
Random sampling	10	1000	10000	97.08%
Least Confidence	12	1000	10000	98.12%
Margin Sampling	15	1000	10000	98%
Entropy Sampling	15	1000	10000	97.98%

The best accuracy when we use the Least Confidence.

CIFAR10 Dataset:

Strategy	n_round	# of queries	n_init_labeled	Final Accuracy
Random sampling	10	1000	10000	64.26%
Least Confidence	10	1000	10000	63.61%
Margin Sampling	10	1000	10000	63.59%
Entropy Sampling	10	1000	10000	60.67%

Bonus Part:

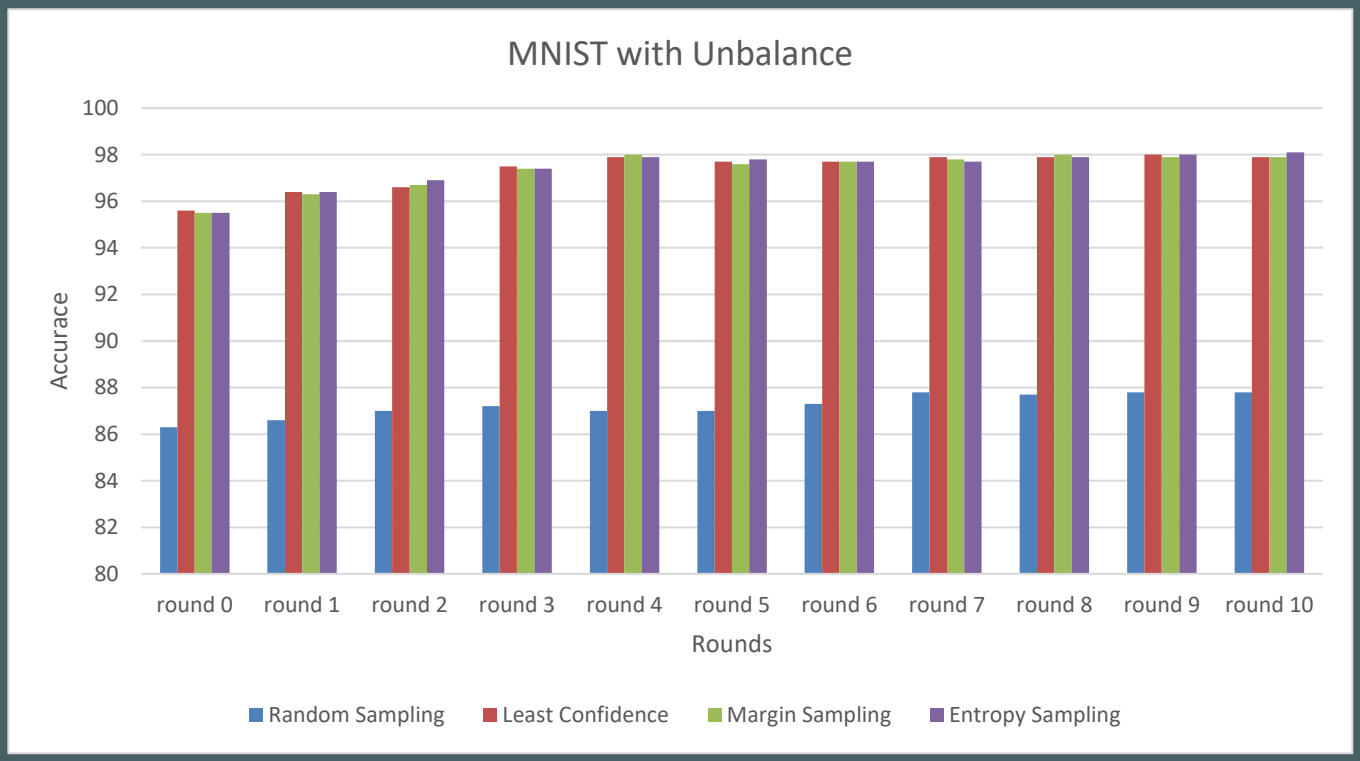
MNIST dataset

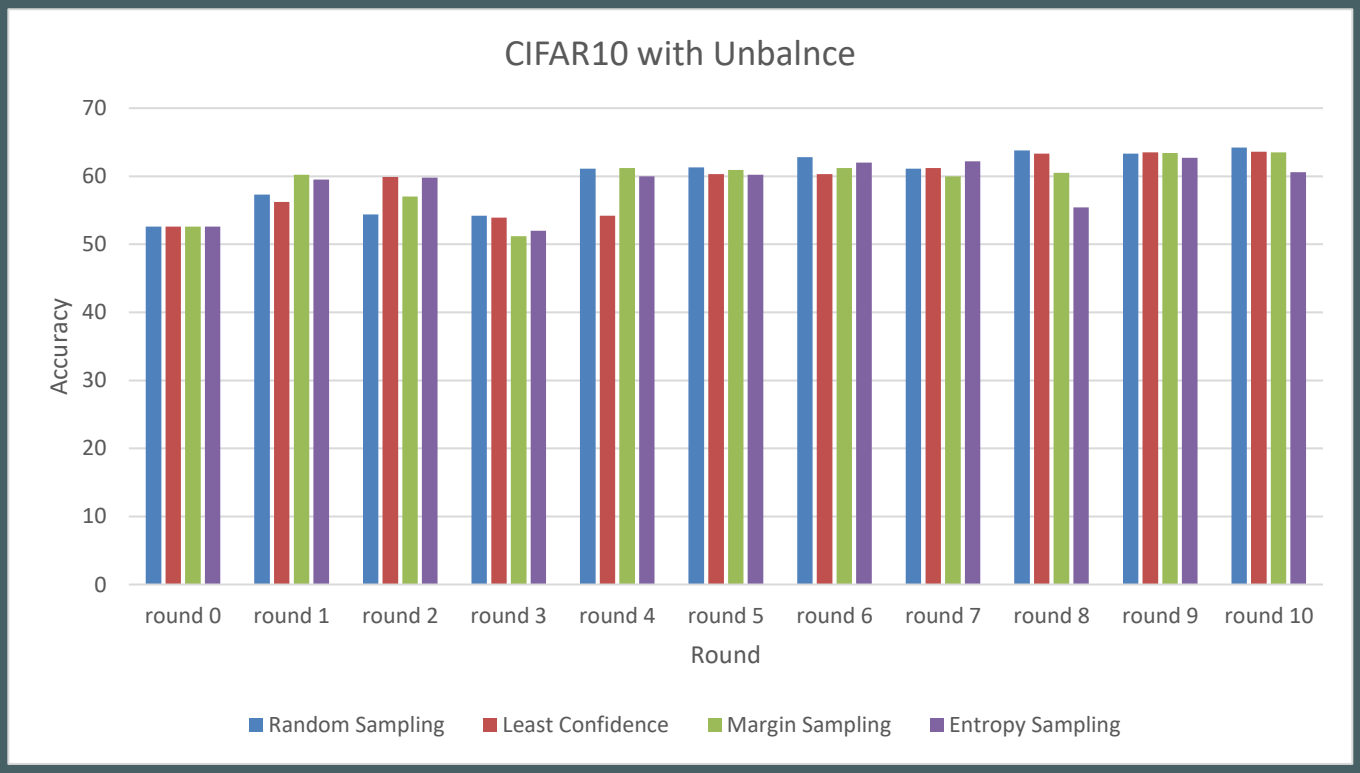
```
def get_MNIST(handler):

    raw_train = datasets.MNIST('./data/MNIST', train=True, download=True)
    raw_test = datasets.MNIST('./data/MNIST', train=False, download=True)
    class0_indices = np.where(raw_train.targets == 0)[0]
    class0_indices_to_remove = class0_indices[:3000]
    raw_train.data = np.delete(raw_train.data, class0_indices_to_remove, axis=0)
    raw_train.targets = np.delete(raw_train.targets, class0_indices_to_remove)
    return Data(raw_train.data[:40000], raw_train.targets[:40000], raw_test.data[:40000], raw_test.targets[:40000], handler)
```

CIFAR10 dataset

```
def get_CIFAR10(handler):
    data_train = datasets.CIFAR10('./data/CIFAR10', train=True, download=True)
    data_test = datasets.CIFAR10('./data/CIFAR10', train=False, download=True)
    class0_indices = np.where(data_train.targets == 0)[0]
    class0_indices_to_remove = class0_indices[:5000]
    data_train.data = np.delete(data_train.data, class0_indices_to_remove, axis=0)
    data_train.targets = np.delete(data_train.targets, class0_indices_to_remove)
    return Data(data_train.data[:40000], torch.LongTensor(data_train.targets)[:40000], data_test.data[:40000], torch.LongTensor(
```


MNIST with Unbalance
100
98
96
94
92
90
88
86
84
82
80
Accurace
round 0
round 1
round 2
round 3
round 4
round 5
round 6
round 7
round 8
round 9
round 10
Rounds
Random Sampling
Least Confidence
Margin Sampling
Entropy Sampling


CIFAR10 with Unbalnce
70
60
50
40
30
20
10
0
Accuracy
round 0
round 1
round 2
round 3
round 4
round 5
round 6
round 7
round 8
round 9
round 10
Round
Random Sampling
Least Confidence
Margin Sampling
Entropy Sampling