Midterm Report

Introduction:

Project Title:

Geometric Partition Entropy for Identifying Optimal Training Sets for Classification Tasks

Motivation/background/business problem:

In machine learning classification tasks, obtaining sufficient labeled data to train models is often crucial for models to achieve accurate prediction results. However, due to various constraints such as time limitations or hardware restrictions in the real world, acquiring enough labeled data can be challenging. In such cases, it becomes imperative for us to be able to still maintain a decent prediction accuracy of the machine learning model despite limitations on sample size we could use to train them. In order to achieve this goal, we often need to optimize the distribution of clustered data within a high-dimensional feature space given the quantity of data is limited. Even though entropy-based measures offer a natural approach for handling such distributions, existing estimators have encountered difficulties, particularly in dealing with the curse of dimensionality, sparse samples, and the presence of outliers etc.

This project aims to address these challenges by introducing a data-efficient learning approach, specifically a novel form of entropy quantification called Boltzmann-Shannon Interaction Entropy used for datasets in high-dimensional feature spaces. The project will explore the relationship between entropy measures of training sets and the performance of models trained on those sets. The ultimate goal is to provide insights and guidance on optimal sub-sampling or data collection strategies for creating minimally effective training sets. The study will focus on well-known datasets, namely CIFAR-10 and CIFAR-100, which are extensively studied and widely available.

The computation of the entropy metric itself is straightforward, but the project's main challenge lies in feature extraction and other preprocessing tasks essential for robust classification tasks.

This capstone project offers a comprehensive introduction to image classification tasks using highly regarded datasets, complemented by more practical benefits (using limited amount of training data to train machine learning models efficiently and effectively).

Additionally, it provides a unique perspective for further analysis in the field of machine learning and information theory.

Problem statement:

Identify smaller subsets from extensive training datasets (such as CIFAR-10) that yield machine learning classifiers' performance nearly equivalent to that achieved when trained on the complete original dataset.

Existing work or literature review:

1. Geometric Partition Entropy: Coarse-Graining a Continuous State Space by Christopher Tyler Diggans and Abd AlRahman R. AlMomani. This literature mentions that entropy serves as a measure of uncertainty and ignorance in predicting continuous phenomena. However, the traditional approaches, based on thermodynamics and Shannon's theory, are fundamentally discrete. This means they are not ideally suited for continuous phenomena. Therefore, the literature proposes a new approach named Geometric Partition Entropy to quantify uncertainty in predicting continuous phenomena using entropy, which is more suitable, consistent, and informative compared to traditional methods, especially in cases of complex distributions or limited sampling.
2. Boltzmann–Shannon interaction entropy: A normalized measure for continuous variables with an application as a subsample quality metric by Christopher Tyler Diggans and Abd AlRahman R. AlMomani. It introduces the concept of Boltzmann–Shannon interaction entropy (BSIE) as a parameter-free measure for quantifying uncertainty in continuous probability distributions. It highlights its advantages over traditional entropy estimators (in handling sparse samples and extreme outliers) and demonstrates its utility in evaluating subsampling quality in regression tasks. Furthermore, unlike the Geometric Partition Entropy approach, this method is normalized and unbiased, so the entropy values won't be impacted by the amount of variance due to the distinction in sample size.

Overall approach:

We currently are using the CIFAR10 dataset and a sample CNN from Kaggle to do the followings:

1. Train the sample CNN with training set in CIFAR10 and make sure the classification accuracy on the test set is good enough (around 80 percent). We save this trained

model which serves as our embedding function to extract feature space of CIFAR10 images.

2. Map all the images in the CIFAR10 training set into the pretrained model to attain the intermediate feature space (just before the final classification layer of the CNN model) and therefore have a matrix with the dimension of number of features times number of training images.

3. Apply Singular Vector Decomposition (SVD) to this matrix:

    i) Will have one matrix U vectored in the feature space.

    ii) Will have a matrix of singular values (the middle diagonal matrix). These values are the variances associated with the data in the direction of the singular vector. Singular values are arranged in descending order along the diagonal.

    iii) Will have one matrix V^T vectored in the data space. (images)

4. Randomly subsample an equal number of images from each category in the training data and combine them together to form one subsample. Repeat this process multiple times.

5. Apply SVD to these subsamples of the CIFAR-10 dataset. So, we will have singular vectors with respect to the subsamples as well as the original singular vector with respect to the whole dataset.

6. Figure out whether the singular vectors of these subsamples are pointing in the same direction as the singular vector of the CIFAR-10 dataset (can use dot product etc.)

7. If they share similar directions as the whole dataset one, then figure out whether they are similar to the whole dataset in geometric distribution by inputting the singular vectors of both subsamples and the whole dataset into Geometric Partition Entropy/Boltzmann Shannon interaction Entropy and compare their output values.

8. If they are similar to the output value of the singular vector of the whole training set, then we can expect these subsamples to have similarly good training performance on machine learning models as the entire training dataset of CIFAR-10.

Methods:

Data (if applicable):

CIFAR-10 dataset. The CIFAR-10 dataset is a widely used benchmark dataset in the domain of machine learning and computer vision. It is frequently employed for tasks such as image classification and object recognition. In this project, we will use it to train and further evaluate the performance of sample Convolutional Neural Networks from Kaggle in the image classification task. We will also extract the internal feature space of these images to do deeper analysis.

Overview:

- CIFAR-10 stands for "Canadian Institute For Advanced Research" and "10" signifies that it contains 10 classes.
- The dataset consists of 60,000 32x32 color images for 10 classes, with 6,000 images per class.
- These classes include common objects and animals: airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks.
- The dataset is divided into training and testing sets, with 50,000 images in the training set and 10,000 images in the test set.
- It is a balanced dataset, meaning each class contains an equal number of images.
- The CIFAR-10 dataset is freely available for academic and research purposes, and it can be accessed through various platforms and repositories, including the official CIFAR website or the TensorFlow and PyTorch libraries.

Note: we could also use CIFAR-100 dataset for this project if time permits.

Processing:

We shuffle the original 50,000 training images and divide them into two parts: training set with 45,000 images and validation set of 5,000 images. The remaining 10,000 images belong to the test set (to evaluate and compare the classification accuracy of CNN models we will train with the whole training set and with subsamples).

Analytic Methods:

- After extracting the feature spaces of both subsamples and whole training set of CIFAR-10, we will then attain their respective singular vectors via Singular Vector Decomposition (SVD) operation and followed by Boltzmann Shannon Interaction Entropy to get their corresponding output values. (The appropriate inputs to the BSIE function might subject to change and will be determined later after the discussion with my mentor)

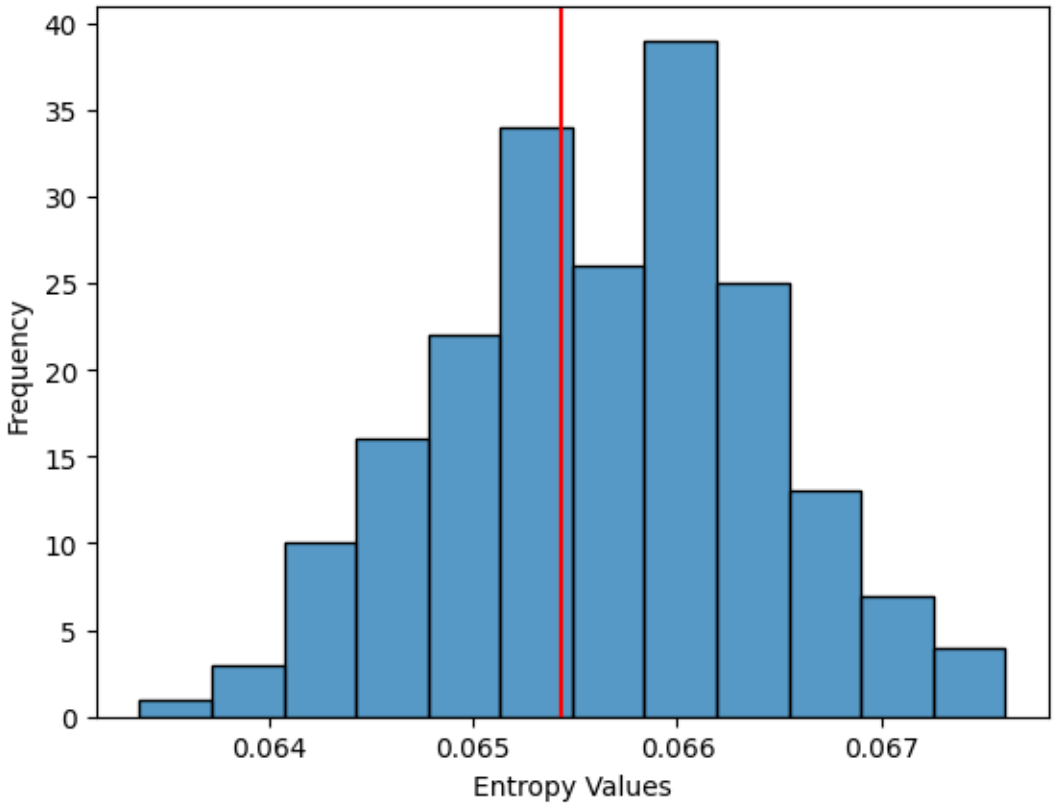
- We will then plot these entropy values of subsamples as well as the entropy value of the CIFAR-10 training set. Through comparison, we can discover which subsample entropy values are closest to the entropy value of the entire training set.
- Regarding these closest entropy values, we will retrieve their associated subsamples and use these subsamples to train new Convolutional Neural Networks respectively.
- Compare the performance (image classification accuracy) between these newly trained networks with the original network.
- More detailed analytic procedures will be included after the whole project is completed.

Evaluation:

We are trying to figure out if subsamples (given they are good representation of the original dataset in terms of data geometric distribution) could also be used to train Convolutional Neural Network and achieve decent performance. Hence, the evaluation is to observe the classification accuracy of these newly trained models on the CIFAR-10 test set and compare them with the performance of the original Convolutional Neural Network trained with the whole training set.

Results:

Histogram of BSIE values of 200 subsamples and the BSIE value of original training dataset



As we can see, the entropy value of the whole training dataset (red line) is almost like a median value with respect to entropy values for other subsamples. Since the project is not complete yet, the final result is not available currently. However, the next step would be to use each subsample to train the Convolutional Neural Network respectively, test their classification accuracy on the test set, and compare the results with the performance of Convolutional Neural Network trained with the original entire training set.

In fact, we can expect that if a certain subsample possesses an entropy value close to the whole dataset one, then the new model trained with this subsample could have a similarly good classification performance like the original model even though such classification performance is not entirely guaranteed.

Therefore, the next visualization would be a scatterplot showing the relationship between entropy values (of subsamples and whole dataset) and corresponding classification accuracy.

In the end (if time permits), we can try different approaches to construct our embedding functions (which is just the machine learning model we train and used for extracting feature space) or change the way to draw subsamples etc. and these approaches will yield distinctive model performance. We will use tables to summarize the results once finished.

App or Tool (if applicable):

The machine learning architecture we chose for the AFRL project is one sample Convolutional Neural Network implemented with PyTorch from Kaggle. This model is deployed to conduct image classification tasks upon training. The robustness of this architecture can be guaranteed in a sense that it follows the conventional structure (which proves to be good) to build Convolutional Neural Network (Convolution layer, Activation function, Pooling layer, Fully Connected layer etc.) and it consistently maintains a classification accuracy above 75 percent on the unseen testing images.

Discussion:

Summary:

This project aims to employ different data-efficient learning approaches to select representative subsamples to train machine learning models (Convolutional Neural Network in this case). The resulting models maintain a similarly good performance just like the model trained with an entire dataset.

Conclusion/Take home messages:

- There exist multiple ways to figure out if a subset of data shares a similar structure as the original dataset. Using entropy method is simply one of them.
- If a subset of data is a good structural representation of the original dataset, the machine learning model trained on it could also display decent performance as if trained on the original dataset.

Future work:

- Use pretrained models from PyTorch torchvision class to extract feature spaces of images.
- When constructing the embedding function, we can also try to use a subsample (rather than the whole training dataset) to do so even if this subsample may not be a perfect representation of the whole training dataset.
- Still use the whole training dataset to construct the embedding function, but this time extract the feature space (and do SVD and BSIE) class by class. After we attain a good interpretation on the data geometric distribution regarding each class, we

can find good data representation from each class and combine them together to form subsamples. We will test the model performance on these subsamples.

Ethical considerations:

None and all the data we used is publicly available.

Other considerations:

The goal of the project is to use less data to train ML models in an efficient way by analyzing the internal feature space and geometric distribution of subsamples, but to extract such things from subsamples, we have to train the model first on the whole dataset anyway.

This project is a solo project, and the only group member is me (Xiao Wen, UNI: xw2943). Currently, I have done the work up to (but not included) point 8 in the overall approach section above.