Data Selection Methodology on Spuriously Correlated Data

Yingjie Huang*
Department of Computer Science
University of California, Los Angeles
Los Angeles, United States
yingjieh512@g.ucla.edu

Vishal Koppuru*
Department of Computer Science
University of California, Los Angeles
Los Angeles, United States
vkoppuru@cs.ucla.edu

Md Aeinul Islam*
Department of Computer Science
University of California, Los Angeles
Los Angeles, United States
mdaeinulislam@g.ucla.edu

Jacob Leigh*
Department of Computer Science
University of California, Los Angeles
Los Angeles, United States
jlaurenceleigh@ucla.edu

***Abstract*— In this project, we will explore how different typical data selection heuristics will affect the worst group accuracy and average accuracy after reducing the dataset to a certain ratio on a model based on a dataset that is known to have spurious features. In PART II, based on these findings, we justify our new data selection inspired by SPARE, which can significantly improve worst-case accuracy as well as improve average accuracy across the board. Our results are tested on the SpuCo MNIST dataset with varying feature magnitude and correlation strengths.**

Keywords—spurious feature, data selection

I. Introduction

Real-world data is biased, and there will often be a huge shift between the train set and the test set. A model trained on a pruned dataset may yield better overall accuracy on the test set than the model trained with the whole dataset. The problem is: could we apply the same method to some datasets that are guaranteed to have spurious features included? We want to find such a data selection methodology that can keep a certain subset of the original datasets, and when we train our model on it, it can keep the average accuracy and increase the worst group accuracy, which means we manage to further mitigate the spurious features.

Keeping a lower ratio subset can help us to make the model avoid being too overfit to the train set and the shift between the test set. It also increases efficiency, which is also crucial since the real-world data is so huge that it takes a much longer time to train. Reducing the size of the training set will yield a significant reduction in time for training the model. Also, spurious features can cause serious problems in models that may be hard to detect. A model that depends on the wrong feature can cause serious problems. For instance, in autonomous car driving, the model should make a decision based on human and passenger features, but since the environment features: skies, trees and roads may be easier to learn for the model, the model will thus make a decision based on the unrelated concepts such as colors of roads, trees and skies and ignore the passing-by person, which may cause an unexpected car accident. As a result, identifying.

The naive approach by selecting a random subset of a certain ratio will fail since the random subset will have exactly the same distribution as the original dataset, which will also inherit the same spurious features from the original dataset. This leads to it maintaining both similar average group accuracy and worst group accuracy to the original dataset. Also, selecting data under spurious features is hard since we need to filter the data that is affected by the spurious feature and force the model to focus on the features we want. This gets harder under extreme cases when all data points contain spurious features, and a trivial data selection method will fail due to large portions of spurious feature data.

Although there are many studies conducted based on data selection or data pruning, they don’t specifically target the data selection methodology on a dataset that emphasizes spurious correlations between data points. For instance, in the forgetting score data selection heuristics (Toneva et al., 2019), the algorithm was based on CIFAR-10, MINST and permutedMINST. In our research, we will test all data selection methodologies under spucoMINST(SpuCo package, BigML-CS-UCLA, 2023), which is a data set emphasizing spurious correlations created by a large-scale machine learning group at UCLA.

Our approach is to do an ablation study evaluating different data selection heuristics under the same correlated strength, keep data ratio, and spurious feature difficulty and compare the results.

Based on these findings, we introduce a new dataset construction method. We were inspired by SPARE and its critical insight that spurious features get separated early in training. By the standards of large neural networks today, “early in training” may take a significant amount of wall-clock time to achieve. Therefore, one design metric that we wanted to explore was whether we could perform a similar partitioning along spurious features, even without having to train the model at all. In this way, it is simple, label-efficient, and easy to integrate into existing pipelines.

II. Related Works

2.1 SPARE:

SPARE is a data selection heuristic that can detect spurious correlations early in the training.(Yang et al. (2023)) They have proved with a theoretical base that the spurious feature examples can be removed during the early stage of training, since they will have significant loss decreases during early training, which makes separating them from other data points during early training possible. This will correspond to early clustering heuristic.

2.2 Forgetting Score:

For a certain example, if the label changes during two consecutive epochs of training, we count it as one forgetting event. The Forgetting Score is introduced by Toneva et al. (2019), which emphasizes the hardness of examples based on the forgetting events that happened during training. This heuristic will keep the examples with higher forgetting scores while discarding the rest.

2.3 GradNorm Score:

GradNorm Score is referenced by (Borsos et al., 2020) in selecting corsets for deep neural networks. For a certain example, we calculate its GradNorm score by adding up all the absolute values of the gradients and keeping the higher score examples.

2.4 Entropy Score:

Claude Shannon introduced the concept of data entropy in 1948, and Lewis and Gale (1994) showed that uncertainty sampling can reduce the required labeled data by orders of magnitude, which uses the concept of keeping subsets with higher entropy.

2.5 High Confidence Score:

Sohn et al. (2020) introduced a high confidence score in which the model may learn only from the spurious features if the data with high confidence dominates the learning process.

2.6 Margin Influence Score:

Koh and Liang (2017) showed that the influence of a data point can be evaluated given the gradient and Hessian. Our RMI heuristic is based on it by computing the gradient influence and dividing it by the distance to other centroids. By doing this, we keep the higher influential examples.

2.7 Loss Score:

Bengio et al. (2009) introduce curriculum learning that indicates that the per-example losses will indicate the difficulty of that example being learned by the model. The loss score is based on the losses of the example, and we will keep the higher loss score examples using this heuristic.

2.8 Spuco Package:

Joshi et al. (2023) aim to provide a package that is useful and convenient to adjust for spurious correlation research. The Spuco Package was created as a result of this paper, which allows control of data spurious feature difficulties and data spurious correlation strength on MINST.

III. Problem Formulation

In this paper, we want to compare the performances of different data selection heuristics under three different variables: spurious feature difficulties, spurious correlation strengths, and data selection ratios. Suppose we have an original data set S, with a spurious feature difficult D, a data keeping ratio at k, and a spurious correlation strength C. We want to find a strict subset of S: S', in which if we train on that subset on the same model, we will obtain relatively the same average accuracy while having better worst group accuracy on the same given test sets. We will also propose a new method based on SPARE(Yang, Gan, Dziugaite, & Mirzasoleiman, 2024), which clusters the model output early-in-training in order to detect spurious correlations.

To first motivate our new method, we can think of this model early-in-training to be a less intelligent agent that is less capable of generalizing causal features across the dataset. Then, we can infer that there exists a smaller "proxy" model that is similarly capable of achieving relatively small loss by solely recognizing the spurious features. We further hypothesize that the outputs of the layers of this proxy model are similar to a translationally invariant feature vector of the dataset. Therefore, by clustering the logits of this model, we cluster the data points in the "spurious feature space." As a final step, in order to minimize the effects of spurious features on the main model, we select an equal number of features from each cluster. We call this **the SATIRE method (Separate After Training, Infer, then REsample).**

One implication is: assuming that the presence of spurious features in any set of examples is proportional to a given trainset, we can construct this proxy model once, and use it repeatedly on any trainset in order to produce a smaller set of exemplars that are more robust against spurious correlation. This method selects training examples strategically to reduce spurious dependencies while preserving predictive accuracy.

IV. Proposed Method

Our research will be done in two parts: in part 1, we will do ablation studies to compare different heuristic results, and in part 2, we will present our new approach.

For both methods, we make use of the LeNet CNN architecture. We chose this for its size and ease of use, being a relatively small model that was designed for the MNIST dataset, and model factory functions are also provided by the SpuCo library directly, making it an obvious choice.

PART I.

We evaluated several data-selection heuristics from class to see how trimming the training set affected worst-group and average accuracy.

The heuristics were: (1) random sampling (uniform subset), (2) loss-based (keep highest-loss examples from a trained model, assuming hard/minority points have higher loss), (3) gradnorm (keep highest gradient norms, again biasing toward hard examples), (4) confidence (keep low-confidence examples based on max softmax), (5) entropy (keep high-entropy examples), (6) forgetting (track forgetting counts during a warmup training and keep frequently forgotten points), (7) early-cluster (run a brief warmup, cluster early losses into two groups, keep from the high-loss cluster and top up from the low-loss cluster to hit the trim fraction), and (8) RMI (Representation-Margin Influence: per-example gradient influence divided by distance to other-class centroids, keep highest scores).

Using these heuristics, we ran a grid over three spurious-difficulty levels (MAGNITUDE_SMALL, MAGNITUDE_MEDIUM, MAGNITUDE_HIGH), three spurious strengths (0.9, 0.95, 0.995, i.e., the fraction of samples that carry the spurious feature), and five keep ratios (0.1, 0.3, 0.5, 0.7, 0.9, the retained fraction). For each setting and heuristic, we select a subset, then train a fresh model on that subset. We scale epochs inversely with the keep ratio so that each run gets roughly the same total training budget (small subsets get more epochs, large subsets get fewer).

PART II.

Our novel method SATIRE initializes two items: the tiny proxy model and a spuriously correlated trainset.

First, this trainset is downsampled (e.g. lower resolution for image inputs). This has two benefits: first, the spurious features are exaggerated in proportion to the rest of the image; second, this allows our model to be smaller with a shorter training/inference time. This tiny proxy model is trained on this dataset. We hypothesize that the proxy model will learn to predict labels solely from the spurious features.

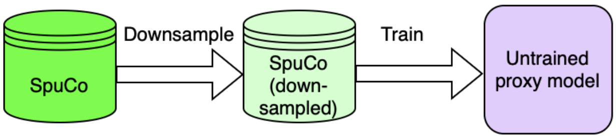


Figure 19. Initial Training of the Proxy Model.

Then, we obtain the main target dataset (which we suspect to contain spurious features). This main target dataset is fed as inputs to our proxy model. Then, we use the logits of this proxy model (i.e. the output of the penultimate linear layer before the classification) and cluster the dataset on these logits.

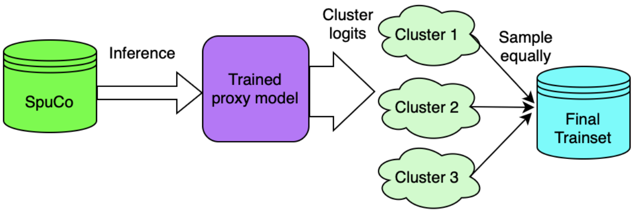


Figure 20. Subset selection with SATIRE.

One refinement we made is to append a one-hot encoding of the classes to the logits, multiplied by a small scalar number, so that the K-Means clustering algorithm also distinguishes between the true class labels.

Lastly, we keep an equal number of data points from each cluster. This number is proportional to the smallest cluster from the previous step. We also confirm that the smallest cluster is reasonably small (since it must be the minority within the class). Through these steps, we have created a subset of the target dataset that is significantly smaller and also robust against spurious correlations.

As a validation step, we create a real model that trains on the final output dataset. This model is tested on a new testset which confirms that worst-case accuracy for each class has improved significantly.

V. Experiments and ablation studies

(Refer to appendix for detailed graphs across various heuristics and spurious correlation settings.)

For the baseline model, which trains on the whole dataset, it has both high average and worst-group accuracy, which are both around 90% when the spurious feature difficulty is SMALL, regardless of spurious correlation strengths, meaning that the model still manages to detect sufficient information for determining the core features of the data without being affected by spurious features. But when the spurious feature difficulty increases to MEDIUM, under high spurious correlations, the worst group accuracy drops drastically. For instance, under MEDIUM difficulty and 0.995 spurious correlation strength, the worst group accuracy drops to around 18%. The model begins struggling learning about the core features. Besides, in more extreme cases when spurious feature difficulty becomes large, the baseline model begins struggling for worst group accuracy and it becomes 0 under extreme cases when spurious correlation = 0.995, meaning almost all data points are affected by spurious features. We can say the magnitude of the spurious feature will dominate the performance of the baseline model worst group accuracy.

Random heuristic follows from the baseline model as it selects a random subset of the whole dataset, meaning it will have the same distribution as the original dataset. It's also stable across different spurious feature difficulties. However, the average group accuracy is low under extremely small keeping ratio of the data under the large feature difficulty as it will struggle to learn meaning the random heuristic is more dependent on dataset under high difficulty of spurious feature.

Loss heuristic, though not stable and oscillating a lot across different spurious feature difficulty, it manages to maintain a relatively above average performance on both worst group and average group accuracy. It performs really well under Medium difficulty. The huge oscillation is due to the selection being based on the sum of absolute values of gradients which is not robust to small gradient differences across the data set. That results in a slightly worse performance than other heuristics under small feature difficulty. But when the feature difficulty increases, the loss heuristic will successfully separate the majority and minority group using the loss' magnitude as the threshold to determine hard/easy groups.

GradNorm will underperform the baseline and random under small feature difficulty and improve at medium feature difficulty. It finally outperforms loss heuristic under large feature difficulty. This is due to instead of computing losses, we compute the gradients as the criterion for selecting examples, which is more robust to the change in the data sets. However, when spurious feature difficulty is low, the GradNorm heuristic will pick examples from majority groups as they dominate the whole data set and with low spurious feature difficulty, the spurious feature will act like random noise to the dataset. Many examples from majority groups will still yield high gradients due to the effects of noises. But under large spurious feature difficulty, the majority group will yield small gradients as they are now easier to detect and the heuristic will pick from minority groups which will have larger gradients, which yields a better performance than before.

Confidence score heuristic does the same role as loss and gradnorm which are both trying to retrieve hard examples from the dataset to achieve learning more from the minority groups. In small and medium difficulty, confidence score maintains low variance across different spurious correlations. Even under large spurious feature difficulty, the average and worst-group accuracy degrades slowly compared to other oscillating heuristics. This robust property is due to the softmax function being smooth and removing high confidence examples will not affect the training drastically. The structure and components of the remaining dataset are more stable compared to other heuristics which yield its robustness across different difficulties.

Entropy as a heuristic performs well when the spurious cue is more distinct. At MEDIUM and LARGE runs, it outperforms baseline significantly with small keep fractions, and also remains relatively stable across keeps, decaying mildly at higher keeps. But, at SMALL difficulty, it becomes very volatile: low keep ratios result in significantly lower worst-group and even average accuracies, and even at higher keep ratios the accuracies recover unevenly. This suggests that when spurious cues are weak, high entropy no longer cleanly targets minority/hard examples (by spurious feature), instead adding noise.

With the forgetting score heuristic, worst-group and average accuracies generally match or trail the baseline across keep ratios, with smaller keep ratios hurting the accuracy more. There are occasional isolated gains (eg. some of the LARGE difficulty or 0.95 strength runs), but there aren't obvious improvements across the board. This might be because some hard/minority examples never actually get counted as "forgotten", so they are underselected especially with too aggressive trimming.

Early clustering performs very well when the spurious feature is strong and separable (LARGE difficulty), providing very clear improvements in worst-group and average accuracy compared to baseline and other heuristics, especially at low keep ratios. In these settings, the high-loss cluster likely aligns best with minority groups, resulting in better performance overall when training on those examples. However, at MEDIUM difficulty, low keep ratios actually underperform, and only higher keep ratios move back towards baseline. At SMALL difficulty, behavior is much more volatile, and aggressive trimming can greatly tank both average and worst-group accuracy. When the spurious feature difficulty isn't high, the loss clusters aren't as distinct from each other, which may explain why early trimming is more noisy in these settings.

RMI performs consistently strong and often above baseline. With high strength and difficulty settings, it outperforms most other heuristics, especially with low keep ratios. But, at SMALL difficulty, low keeps can hurt accuracy, and higher keeps recover or exceed baseline. When the spurious cues are weak, RMI benefits from retaining more data to preserve diversity, but when these cues are more clear, trimming most of the data allows it to capture minority groups best.

PART II

With the following set of experiments, we wanted to test the robustness of our new data selection method, SATIRE. Since the size of the selected subset is proportional to the smallest cluster determined by the proxy model, we did not parametrize this problem in terms of keep_ratio. However, we compared it to keep_ratio = 1 with the rest of the problem, since in the worst case of equally sized clusters it keeps all examples.

We train the final model the same for 5 epochs. However, for datasets with a very high correlation strength (0.995), SATIRE tended to keep an extremely low number of data points, resulting in severe underfitting. Therefore, the number of epochs was increased to around 20 for high correlation strengths.

We will begin our discussion with where SATIRE shines and where it does not. To begin, SATIRE's performance with small feature magnitude and weak correlation strength is comparable to algorithms from Part I, slightly underperforming at times. However, we can attribute this to the fact that SATIRE tends to keep a very low number of data points, and the model is trained for a fixed number of epochs at 5. When we increased the number of epochs, the performance improved. Therefore, we conclude that this small discrepancy is a result of underfitting.

At larger feature magnitudes of MEDIUM and LARGE and higher correlation strengths, SATIRE begins to outperform most other metrics. In particular, we want to draw your attention to MAGNITUDE_MEDIUM and LARGE at strength 0.995. Both the average and worst-case group accuracies are generally higher than the baseline comparisons. This is a considerably more significant result; it shows our approach's robustness against spurious features under more difficult constraints, and ability to methodically resample clusters based on spurious features.

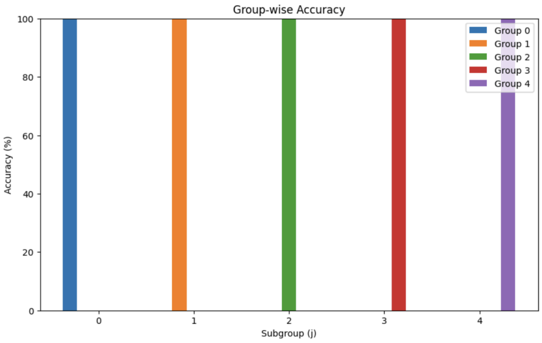
The reason for this behavior will be studied slightly more in depth in the following paragraphs, but briefly: the dataset SATIRE selects tries to balance the presence of *features* within a class. The original SpuCo dataset creates the majority in a class to exhibit a specific feature, but SATIRE forces other variations of this feature to be expressed more prominently across the dataset by downsizing the majority. This explains why SATIRE shows more robustness in terms of worst-case accuracy, no matter how skewed the dataset is.

Furthermore, the methods analyzed in Part I cannot detect the presence of spurious correlation until after training. SATIRE, however, is able to recognize them through the initial validation of the proxy model, by detecting unbalanced accuracy within a class.

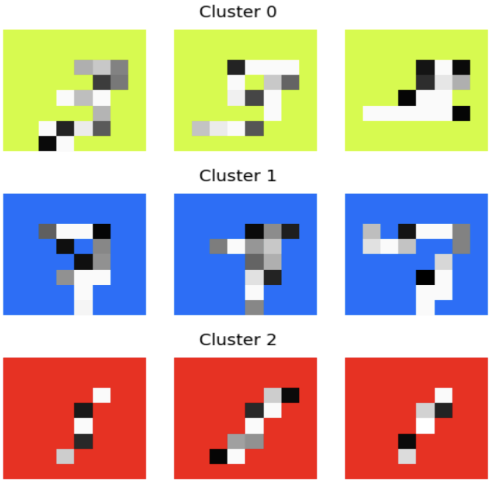
Speaking objectively however, we were not able to completely avoid failure on the most difficult dataset, which is the SpuCo MNIST with LARGE_MAGNITUDE and correlation coefficient of 0.995. The lowest class accuracy was 36.77%.

Since the average is 65.55%, the model was able to generalize some minority classes, but failed to do so for certain others. We provide an investigation through an ablation and case study.

First of all, we investigate the performance of the proxy model:



Here's a visual example of some clusters from the MAGNITUDE_LARGE dataset, with correlation = 0.995, k=25.Unsurprisingly, the proxy model performs poorly, with minority groups having 0 accuracy. This is as expected; the proxy model is designed to infer the class solely based on the spurious features.

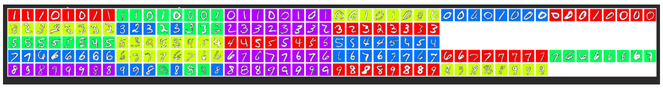


If you visually extrapolate the digit from the low-res images, you can see a very strong tendency for this method to cluster based on the label and the background color. This supports our hypothesis which claims that the logits of the proxy model can be used as a substitute for the data point's feature vector.

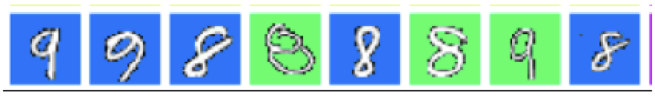
However, we must acknowledge that the quality of this clustering is reliant on some prior knowledge about the spurious features present in the dataset. We were aware that there were 5 classes total in this dataset, with 4 minority groups and 1 majority group each. This comes out to 25 groups, which we targeted by setting the number of clusters to k=25. To prove this, we conducted an isolated study where we reduced the cluster count to k=10. In this case, the model performs similarly to other algorithms, dropping the worst-case accuracy to 0 on the MAGNITUDE_LARGE dataset, with correlation = 0.995.

Finally, let us visualize the selected dataset, and see if it gives a reasonable mix of backgrounds:

Note that all images in a row come from the same class. Every 8 contiguous elements are from the same cluster found by the clustering algorithm.



You can pinpoint the the exact place where the dataset lacks diversity:



This is taken from class 4 (category for digits 8 and 9), which shares the same background with the majority of class 2 (category for digits 4 and 5). Visually, one can notice that there are only 3 examples from this minority group, and in the results, (4, 2) is indeed the lowest accuracy group of 36.77%!

Therefore, the main weakness in SATIRE comes from the fact that the clustering algorithm is not as predictable in how minority groups are clustered. As seen above, some clusters contained different but similar spurious features; in this case, since blue is close to green in terms of RGB, they were clustered together for class 4. However, it does significantly alleviate the strength of spurious correlation compared to the original dataset, from 0.995.

VI. Conclusions

This study systematically evaluated eight data selection heuristics on the SpuCoMNIST dataset (Joshi et al., 2023) across 45 experimental configurations varying spurious feature difficulty (MAGNITUDE_SMALL, MAGNITUDE_MEDIUM, MAGNITUDE_LARGE), correlation strength (0.9, 0.95, 0.995), and data retention ratio (0.1, 0.3, 0.5, 0.7, 0.9). The experimental results demonstrate that data selection heuristics exhibit regime-dependent effectiveness, with early clustering showing superior performance at MAGNITUDE_LARGE and RMI demonstrating consistent performance across multiple experimental conditions.

For baseline models trained on the complete dataset, worst-group accuracy remained approximately 90% for SMALL difficulties across all correlation strengths, but degraded substantially at MAGNITUDE_LARGE, reaching 0% when correlation strength approached 0.995. This catastrophic baseline failure at extreme correlation levels motivated our development of alternative selection strategies.

Building upon these findings from Part I, we proposed SATIRE (Separate After Training, Infer, then REsample), a proxy-model-based selection method that clusters training examples in the logit space of a model trained on downsampled data. SATIRE's performance at MAGNITUDE_MEDIUM and MAGNITUDE_LARGE with correlation strength 0.995 mostly exceeds baseline comparisons in both average and worst-case group accuracy. While the methodology is sensitive to parameters such as the cluster count, with k=25 achieving 36.77% worst-case accuracy and k=10 resulted in 0% worst-case accuracy on the most challenging configuration (MAGNITUDE_LARGE, correlation 0.995), it is robust against spurious correlation across a wide range of settings. This validation demonstrates that SATIRE's clustering approach successfully partitions the spurious feature space, though it may suffer from a large variation in performance based on the clustering mechanism.

Future research directions include extending evaluation to real-world datasets with diverse spurious correlation patterns, exploring adaptive selection strategies, and investigating ensemble methods that combine multiple heuristics to achieve robust performance across varying difficulty regimes.

Github repo: https://github.com/jel221/SATIREandSpuCo

References

[1] Yang, Y., Gan, E., Dziugaite, G. K., & Mirzasoleiman, B. (2023). Identifying Spurious Biases Early in Training through the Lens of Simplicity Bias. *ArXiv*. https://arxiv.org/abs/2305.18761

[2] Joshi, S., Yang, Y., Xue, Y., Yang, W., & Mirzasoleiman, B. (2023). Challenges and Opportunities in Improving Worst-Group Generalization in Presence of Spurious Features. *ArXiv*. https://arxiv.org/abs/2306.11957

[3] Toneva, M., Sordoni, A., Combes, R. T., Trischler, A., Bengio, Y., & Gordon, G. J. (2018). An Empirical Study of Example Forgetting during Deep Neural Network Learning. *ArXiv*. https://arxiv.org/abs/1812.05159

[4] Borsos, Z., Mutný, M., & Krause, A. (2020). Coresets via Bilevel Optimization for Continual Learning and Streaming. *ArXiv*. https://arxiv.org/abs/2006.03875

[5] D. Lewis and W. Gale. A sequential algorithm for training text classifiers. In Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, pages 3–12. ACM/Springer, 1994.

[6] C.E. Shannon. A mathematical theory of communication. Bell System Technical Journal, 27:379–423,623–656, 1948

[7] Sohn, K., Berthelot, D., Li, C., Zhang, Z., Carlini, N., Cubuk, E. D., Kurakin, A., Zhang, H., & Raffel, C. (2020). FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence. *ArXiv*. https://arxiv.org/abs/2001.07685

[8] Koh, P. W., & Liang, P. (2017). Understanding Black-box Predictions via Influence Functions. *ArXiv*. https://arxiv.org/abs/1703.04730

[9] Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009). Curriculum learning. International Conference on Machine Learning.

VII. Gen AI Usage

We used generative AI (eg. ChatGPT, Cursor IDE) for parts of the notebooks, in particular with debugging and code refactoring for the Part 1 heuristics analysis. In particular, to go from testing heuristics with a single setting, to a more streamlined grid of settings, we made use of AI for that code refactoring. We also used it for graph plotting in both parts, by describing what kinds of plots we wanted and using it to adjust how the line graphs are displayed.

VIII. Team Contributions

Jacob Leigh: Handled Part II (SATIRE) solo. Came up with the idea for SATIRE. Performed system design/proposal, and implemented SATIRE from start to finish on Colab. Refer to Github for the work done. Iteratively improved on the results through augmentations on the data or the design itself, utilizing both literature and office hours. Conducted all experiments for SATIRE, coalesced data, and evaluated them. Completed all sections of the report and presentation for Part II, including the introduction, methodology, evaluation, and ablation/case study.
+ Diagrams. I think they're well made and proud of them.

Yingjie Huang: Came up with the data selection heuristics comparison idea. Built the skeleton code of part I. Collect the preliminary results for slides. Contribute to the slides write-up and presentation. For the report, did the intro, abstract, related works, and problem formulation. Aided in writing the proposed method in Part I. Search up all the related papers for references. Running experiments and contributing to the reasoning of experiments and ablation studies.

Vishal Koppuru: Aided in ideation for Part I. Worked (with MD) off of skeleton Yingjie built to analyze heuristics across grid of settings, fixed some heuristics code, and wrote plotting code. Contributed to slides/presentation and report (Part I of proposed methods, experiments and ablation studies for some of part I heuristics).

MD Aeinul Islam: Aided in ideation for Part I. Implement and debug heuristics across grids of settings and results analysis (with Vishal). Plotted graphs from all heuristics from generated results. Contribute to Slides/Presentations and Report (Part I). Worked on running hour long experiments and analysing results; drawing and writing conclusions in the report as well as summarizing charts in report. Also worked on the Appendix in the report. Worked on some of the experiments and ablation studies for some of the part I heuristics. Explored alternative Heuristics like CREST & CRAIG.

IX. Appendix

Graphs presenting average and worst-group accuracies versus data retention ratio, with each line representing a data selection method, across different spurious feature difficulties (SMALL, MEDIUM, LARGE) and correlation strengths (0.9, 0.95, 0.995)
(Graphs start on the next page)

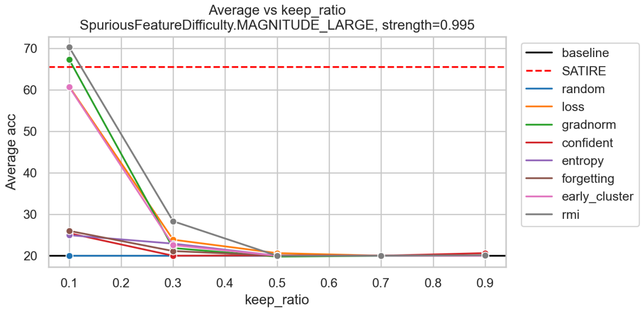


Figure 1. Average accuracy: [MAGNITUDE_LARGE, 0.995]

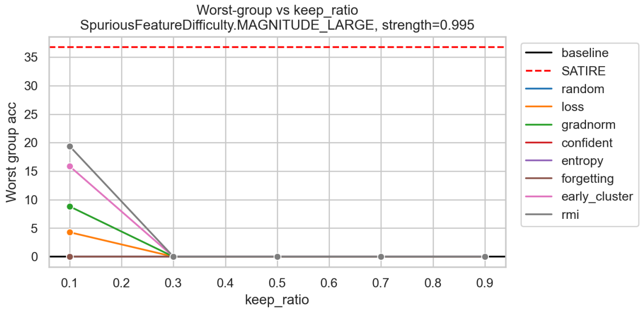


Figure 2. Worst-group accuracy: [MAGNITUDE_LARGE, 0.995]

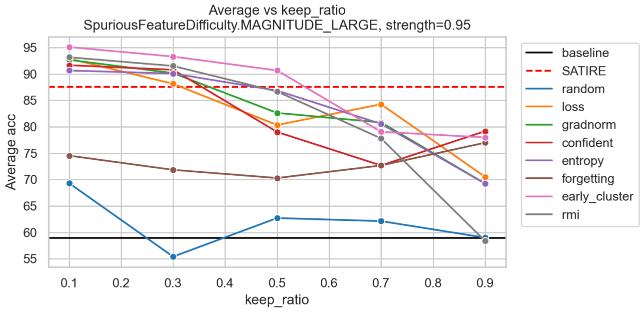


Figure 3. Average accuracy: [MAGNITUDE_LARGE, 0.95]

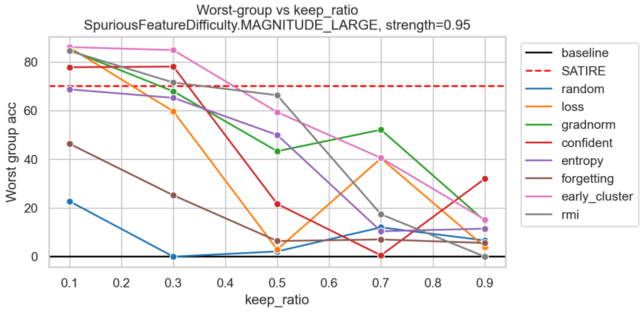


Figure 4. Worst-group accuracy: [MAGNITUDE_LARGE, 0.95]

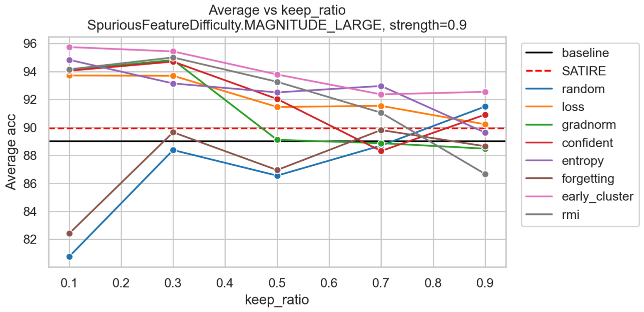


Figure 5. Average accuracy: [MAGNITUDE_LARGE, 0.9]

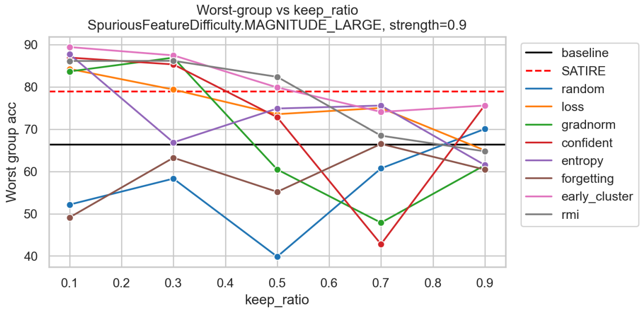


Figure 6. Worst-group accuracy: [MAGNITUDE_LARGE, 0.9]

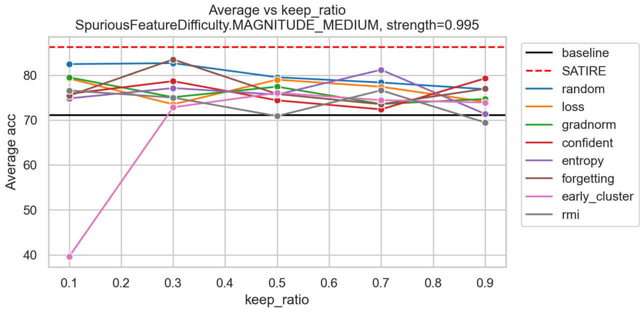


Figure 7. Average accuracy: [MAGNITUDE_MEDIUM, 0.995]

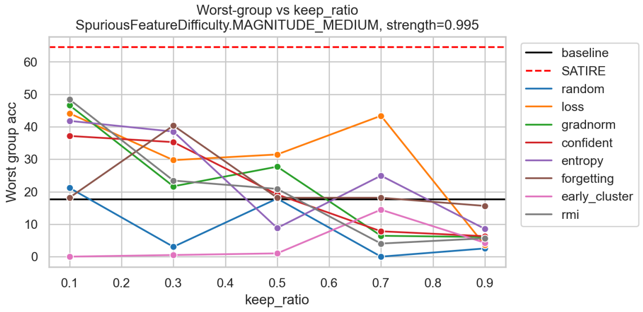


Figure 8. Worst-group accuracy: [MAGNITUDE_MEDIUM, 0.995]

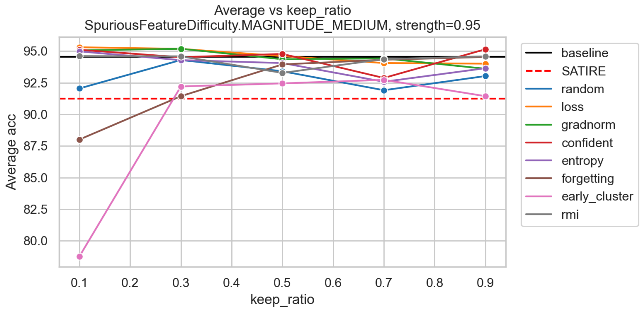


Figure 9. Average accuracy: [MAGNITUDE_MEDIUM, 0.95]

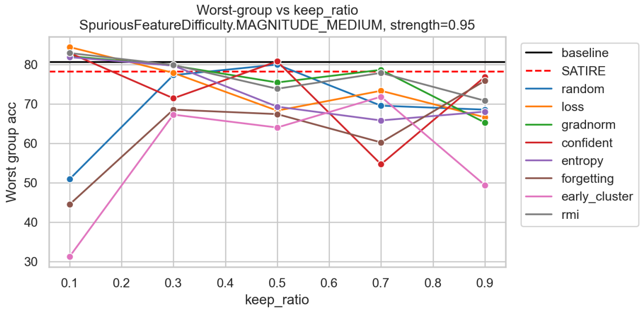


Figure 10. Worst-group accuracy: [MAGNITUDE_MEDIUM, 0.95]

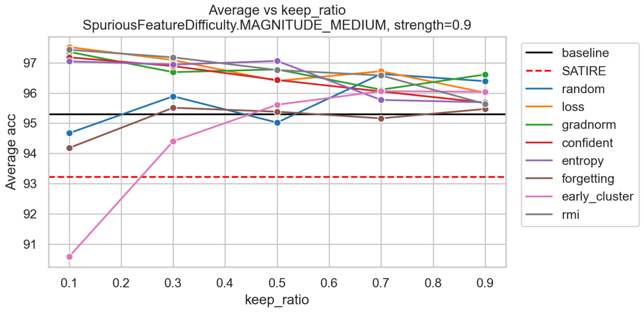


Figure 11. Average accuracy: [MAGNITUDE_MEDIUM, 0.9]

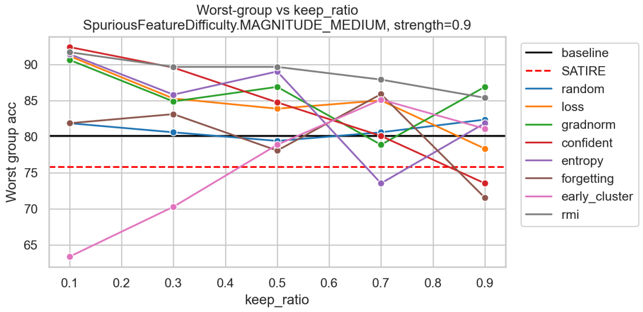


Figure 12. Worst-group accuracy: [MAGNITUDE_MEDIUM, 0.9]

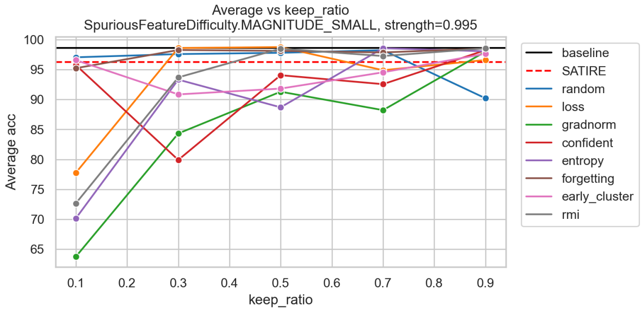


Figure 13. Average accuracy: [MAGNITUDE_SMALL, 0.995]

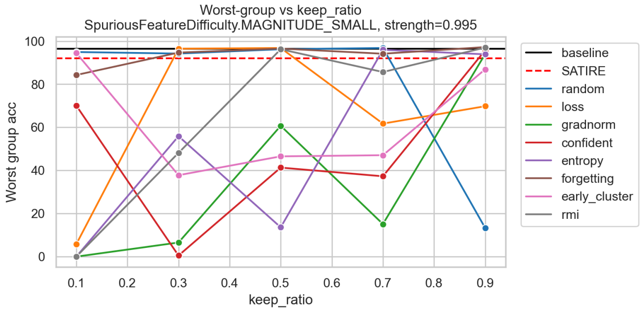


Figure 14. Worst-group accuracy: [MAGNITUDE_SMALL, 0.995]

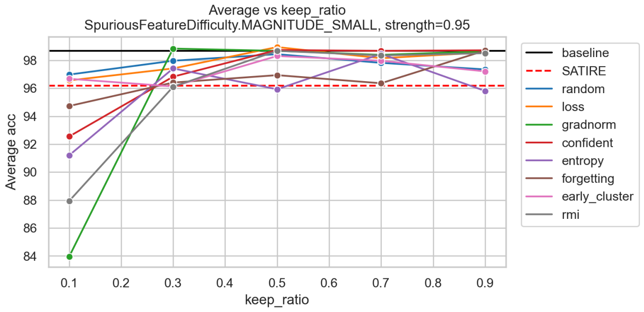


Figure 15. Average accuracy: [MAGNITUDE_SMALL, 0.95]

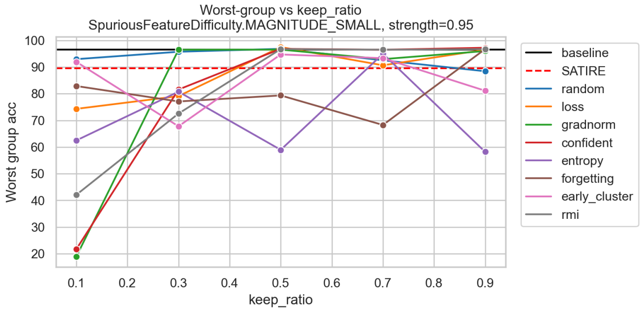


Figure 16. Worst-group accuracy: [MAGNITUDE_SMALL, 0.95]

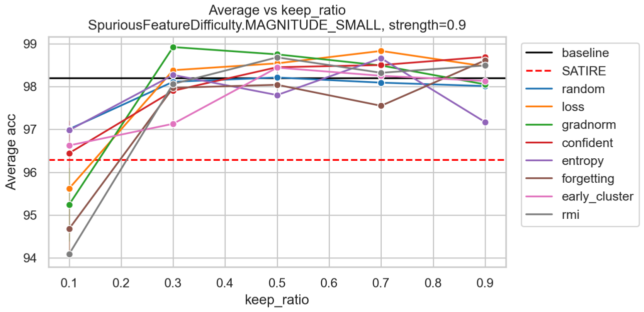


Figure 17. Average accuracy: [MAGNITUDE_SMALL, 0.9]

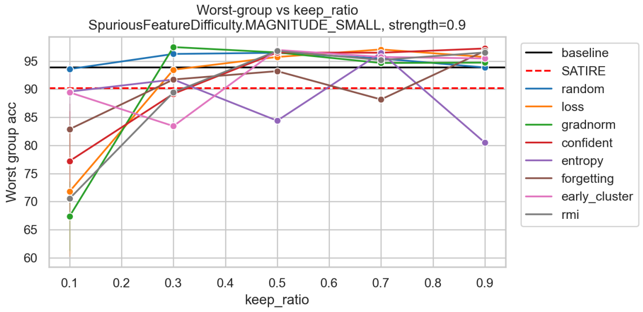


Figure 18. Worst-group accuracy: [MAGNITUDE_SMALL, 0.9]