SVC Classification Report

Papagrigoriou Vasileios Savvas
Auth

***Abstract*—This paper presents a comparative analysis of Support Vector Machine (SVM) models applied to two distinct datasets: CIFAR-10 and Letter Recognition. We evaluate the performance of different SVM kernels and parameters on these datasets and discuss the implications of using PCA (Principal Component Analysis) on model performance and training time.**

I. Introduction

SVMs are widely used in classification tasks due to their versatility and efficacy. In this study, we analyze their performance on the CIFAR-10 and Letter Recognition datasets, focusing on the impact of kernel choice, parameter tuning, and the use of PCA on model accuracy and training duration.

II. Methodology

The CIFAR-10 dataset consists of 50,000 32x32 color images in 10 classes, while the Letter Recognition dataset includes 20,000 instances of 26 capital letters in various fonts. We employed various SVM kernels: RBF, Polynomial (Poly), and Linear, with different settings for the hyperparameters C and gamma.

III. Data Specifications

A. CIFAR-10 Dataset

To investigate the effects of PCA and how the size of the training set influences the performance of different kernels, we performed our analysis on different subsets of the CIFAR-10 dataset. In particular, we made use of the subsequent data batches:

- A collection of 2.000 pictures: Understanding the models' performance with little data is essential for comprehending their capacity to learn from a smaller sample size, and this smaller dataset offered insights into this capacity.
- A collection of 4.000 thousand pictures: We were able to assess how the model's performance scaled with an increase in the quantity of training data thanks to this medium-sized dataset.
- A collection of 8.000 thousand pictures: With a larger batch size, the models' performance was evaluated in situations where a greater quantity of data was available, more indicative of real-world circumstances.

These varying sizes helped in comprehensively understanding the scalability and adaptability of the SVM models with respect to different volumes of data.

B. Letter Recognition Dataset

In the case of the Letter Recognition dataset, we extended our analysis to cover a wider range of data batch sizes to evaluate the performance of SVM kernels more thoroughly. The data batches used were:

- A batch of 2.000 instances: This size served to test the models' performance in a constrained data environment, which is often a realistic scenario in many applications.
- A batch of 4.000 instances: This provided a baseline for comparing the performance increment or decrement when the amount of data is doubled from the smallest batch size.
- A batch of 8.000 instances: This batch size was crucial to understand the model behavior with a moderately large dataset.
- A batch of 13.000 instances: This larger dataset size was used to observe the performance trends as the data volume starts to approach the full dataset.
- A batch of 20.000 instances: This size, being closest to the full dataset, was critical for assessing the maximum performance capability of the SVM models.

These data batches were instrumental in providing a comprehensive view of how each SVM kernel performs across different volumes of data and the impact of PCA on these varying sizes.

IV. Implications of Data Size on Model Performance

The varied data sizes in both datasets allowed us to discern critical insights into the performance scalability of the SVM models. It became evident that as the size of the training data increases, the models' ability to generalize improves, but this also comes with the trade-off of increased training times, particularly for certain kernels like RBF. The smaller batches were useful in highlighting the models' efficiency and quick adaptability, whereas the larger batches emphasized the models' robustness and generalization capabilities.

V. Detailed Analysis

A. CIFAR-10 Dataset Analysis

1) Kernel Performance: In the CIFAR-10 dataset, the Poly kernel outperformed the RBF and Linear kernels in terms of classification accuracy. This indicates that the Poly kernel's ability to model complex, non-linear relationships in image data is particularly effective for the diverse and intricate patterns present in CIFAR-10's image set. The RBF kernel, while generally robust across various datasets, may not capture

the specificities of image data as effectively as the Poly kernel in this context.

2) Overfitting Concerns: A significant observation was the disparity between high training accuracy and lower testing accuracy in many models. This suggests a tendency towards overfitting, where models are highly tuned to the training data and fail to generalize well to new, unseen data. This issue underscores the importance of implementing strategies to combat overfitting, such as regularization, cross-validation, or using more generalized models.

3) Impact of Hyperparameters: The performance variations with different settings of C and gamma parameters indicate the sensitivity of SVMs to these hyperparameters. Optimizing these parameters is crucial for achieving the best possible model performance. This suggests the need for a more rigorous hyperparameter tuning process, possibly employing grid search or randomized search methods.

4) Training Times and Computational Efficiency: The CIFAR-10 dataset presented notable differences in training times across different models. Models with higher computational demands, although potentially more accurate, may not be feasible in time-sensitive or resource-constrained environments. This finding highlights the need for a balanced approach to model selection, considering both accuracy and computational efficiency.

B. Letter Recognition Dataset Analysis

1) Kernel Suitability: For the Letter Recognition dataset, the RBF kernel consistently delivered high performance, especially with larger data batches. This suggests that the RBF kernel's capability to handle non-linear patterns is well-suited to the task of recognizing different fonts and styles in letter images. The Poly kernel also showed promising results, particularly in larger datasets, indicating its potential applicability in similar text recognition tasks.

2) PCA's Role in Performance and Efficiency: As with the CIFAR-10 dataset, applying PCA on the Letter Recognition dataset reduced overall performance, indicating a loss of crucial information necessary for accurate classification. However, the reduced training times achieved through PCA might be beneficial in scenarios where rapid model deployment is more critical than achieving the highest possible accuracy.

3) Balancing Training Time and Model Performance: The trade-off between training time and model performance was evident in this dataset as well. While the RBF kernel achieved higher accuracy, its longer training times may not be practical for all applications. Conversely, the Poly kernel, with its quicker training times, could be a more feasible option for applications with limited computational resources.

VI. Letter Recognition Dataset Performance Analysis

A. Performance with PCA

TABLE I: SVM Performance on Letter Recognition with PCA (20000 Data Batch)

Kernel	C	Gamma	Train Acc	Test Acc	F1 Score	Time (min)
RBF	100	0.1	1.000	0.9325	0.932131	33.3404
Poly	0.1	0.1	0.952333	0.88875	0.888755	8.93297
Linear	0.1	auto	0.7475	0.7335	0.732209	14.439

TABLE II: SVM Performance on Letter Recognition with PCA (13000 of 20000 Data Batch)

Kernel	C	Gamma	Train Acc	Test Acc	F1 Score	Time (min)
RBF	100	auto	1.000	0.910962	0.910829	17.7529
Poly	0.1	0.1	0.956923	0.876154	0.876837	4.46801
Linear	0.1	auto	0.737051	0.731731	0.730541	6.032

TABLE III: SVM Performance on Letter Recognition with PCA (8000 of 20000 Data Batch)

Kernel	C	Gamma	Train Acc	Test Acc	F1 Score	Time (min)
Poly	0.1	0.1	0.974167	0.775	0.77749	0.198008
RBF	10	0.1	1.000	0.775	0.777131	0.595002
Linear	1	auto	0.823333	0.70625	0.703397	0.239998

TABLE IV: SVM Performance on Letter Recognition with PCA (2000 of 20000 Data Batch)

Kernel	C	Gamma	Train Acc	Test Acc	F1 Score	Time (min)
Poly	0.1	0.1	0.974167	0.775	0.77749	0.192997
RBF	10	0.1	1.000	0.775	0.777131	0.574992
Linear	1	auto	0.823333	0.70625	0.703397	0.249003

B. Performance Without PCA

TABLE V: SVM Performance on Letter Recognition without PCA (20000 Data Batch)

Kernel	C	Gamma	Train Acc	Test Acc	F1 Score	Time (min)
RBF	10	auto	1.000	0.971	0.970534	49.322
Poly	0.1	auto	0.999417	0.942875	0.942055	12.922
Linear	1	auto	0.879333	0.853625	0.851967	15.991

TABLE VI: SVM Performance on Letter Recognition without PCA (13000 of 20000 Data Batch)

Kernel	C	Gamma	Train Acc	Test Acc	F1 Score	Time (min)
RBF	10	auto	1.000	0.93625	0.936904	9.58105
Poly	0.1	auto	0.999792	0.896875	0.897533	2.47099
Linear	0.1	auto	0.889583	0.82125	0.820917	3.64601

TABLE VII: SVM Performance on Letter Recognition without PCA (8000 of 20000 Data Batch)

Kernel	C	Gamma	Train Acc	Test Acc	F1 Score	Time Taken (min)
RBF	10	auto	1.000	0.93625	0.936904	10.6901
Poly	1	auto	1.000	0.895312	0.896016	2.54
Linear	1	auto	0.906875	0.817187	0.816731	3.27099

TABLE VIII: SVM Performance on Letter Recognition without PCA (2000 of 20000 Data Batch)

Kernel	C	Gamma	Train Acc	Test Acc	F1 Score	Time (min)
RBF	10	auto	1.000	0.93625	0.936904	10.4032
Poly	0.1	auto	0.999792	0.896875	0.897533	2.70734
Linear	0.1	auto	0.889583	0.82125	0.820917	3.53565

TABLE IX: Optimal Performance of Different Kernels on CIFAR-10 Dataset

Kernel	C	Gamma	Train Acc	Test Acc	F1 Score	Time (s)	Data Size
Poly	10.0	auto	1.0	0.407	0.409109	537.393	8000
Poly	1.0	0.0001	1.0	0.3757	0.377222	315.936	4000
RBF	100.0	1.00E-10	0.681	0.3303	0.326977	978.653	2000
Linear	10.0	0.0001	1.0	0.3073	0.308494	414.259	4000

VII. Classifier Performance Comparison

In this analysis, we examine the performance of K-Nearest Neighbors (KNN), Nearest Class Centroid (NCC), and Support Vector Machine (SVM) classifiers, focusing on a CIFAR-10 dataset subset of 4,000 data points. The performance metrics include training accuracy, testing accuracy, F1 score, and computational time.

The SVM models, employing a polynomial kernel with varying parameters for C and gamma, consistently achieved perfect training accuracy (1.0) and a testing accuracy of 0.3757, with an F1 score of approximately 0.3772. Notably, these results were consistent across different configurations of the hyperparameters C (1, 10, 100) and gamma (auto, 0.0001, 0.001), with processing times ranging from approximately 288 to 389 seconds. This consistency indicates a robustness in the SVM model's performance against variations in these parameters, at least within the tested ranges.

In contrast, the KNN classifier, even when trained on a significantly larger dataset of 50,000 data points, achieved a lower testing accuracy of 0.3398 and an F1 score of 0.326017, albeit with a higher processing time of 279.211 seconds. The NCC classifier, known for its speed, demonstrated a stable performance across different dataset sizes, with a best F1 score of 0.255551 on a 4,000 data point dataset and a remarkably quick processing time of 0.605985 seconds.

This comparison underscores the SVM's superior efficacy in terms of testing accuracy and F1 score, even when trained on a smaller dataset. The KNN and NCC classifiers, while beneficial in scenarios requiring faster processing times or larger datasets, fall short in achieving the accuracy levels of the SVM models. These findings highlight the critical importance of selecting an appropriate classifier based on the specific requirements of accuracy, computational efficiency, and dataset size.

VIII. Impact of PCA on SVM Classifier Performance

This section delves into the effects of Principal Component Analysis (PCA) on the performance of Support Vector Machine (SVM) classifiers, specifically focusing on polynomial and radial basis function (RBF) kernels. The analysis is based on a CIFAR-10 dataset subset comprising 4,000 data points, with metrics including training accuracy, testing accuracy, F1 score, and computational time.

For SVMs with a polynomial kernel, regardless of the hyperparameters C (1, 10, 100) and gamma (auto, 0.0001, 0.001), the models consistently achieved a perfect training accuracy of 1.0. However, the testing accuracy plateaued at approximately 0.3459, with an F1 score around 0.3467. Notably, the incorporation of PCA significantly reduced the computational time, with times ranging from approximately 8.9 to 9.0 seconds, a drastic reduction compared to non-PCA times.

In contrast, SVMs employing the RBF kernel, across similar ranges of C and gamma, also reached perfect training accuracies but exhibited markedly lower testing accuracies and F1 scores (around 0.1 and 0.0182, respectively). The processing times for these models were notably higher, averaging around 29 seconds. This disparity in performance indicates that while PCA contributes to computational efficiency, its impact on model accuracy is highly dependent on the choice of the kernel.

An interesting observation was made with the linear kernel SVMs. Despite the expectation of computational efficiency with PCA, the linear kernel models did not yield results even after an extended duration of approximately 300 minutes. This outcome suggests a potential limitation or inefficiency in the application of PCA when combined with the linear kernel in SVMs, especially for complex datasets like CIFAR-10.