Assignment L05 Image Classification with SVM

Houston Community College

Mary Balemba

ITAI 1378 Computer Vision

Professor Anna Devarakonda

February 22nd, 2025

Assignment L05 Image Classification with SVM

Reflective Journal: SVM for Image Classification

1. Reflection on Learning

Support Vector Machines (SVM) is a supervised learning algorithm that is widely used for classification tasks. At its core, SVM aims to find the best possible boundary (or hyperplane) that separates different categories of data points with the greatest possible margin. This approach is particularly effective for handling high-dimensional data, which makes it a good candidate for image classification tasks.

In the case of image classification, each image is transformed into a feature vector that represents its pixel values. The SVM model then determines the optimal way to separate images into different categories. One of its key advantages is the use of a **kernel trick**, which allows it to classify data that isn’t linearly separable by transforming it into a higher-dimensional space.

For this project, we applied SVM to the **CIFAR-10 dataset**, which consists of **60,000 images** divided into **10 categories**. Each image was represented as a **32×32×3 (RGB) array**, meaning that each had **3,072 features** when flattened into a one-dimensional vector. The SVM model processed these features and classified the images based on learned patterns.

Before training the SVM model, several preprocessing steps were applied to the CIFAR-10 dataset to optimize performance. The dataset was first explored by visualizing sample images to understand category variations. Since SVM requires numerical vector inputs, images were reshaped from 32×32×3 into 1D vectors. To improve efficiency, an optional grayscale conversion was tested to reduce computational complexity while preserving essential patterns. Additionally, pixel values were normalized to a [0,1] range to prevent numerical instability and ensure balanced feature contributions. For model training, selecting the right kernel was crucial. A linear kernel was initially chosen for its simplicity, but it struggled with overlapping classes like cats and dogs. The model was trained on 5,000 images and tested on the CIFAR-10 set, achieving 35%-40% accuracy—better than random guessing but lower than deep learning models. The results highlighted SVM’s strengths in recognizing patterns but also its limitations with complex image variations. To improve performance, advanced techniques such as using an RBF kernel, applying Principal Component Analysis (PCA) for dimensionality reduction, and feature engineering with Histogram of Oriented Gradients (HOG) were considered. Despite some effectiveness, deep learning models remain the preferred choice for large-scale image classification.

2. Responses to Lab Questions

1. **What makes SVM a good choice for image classification?**
   1. SVM is particularly effective for datasets with **small to moderate sizes**.
   2. It works well in **high-dimensional spaces** and **does not require large amounts of data** to perform well.
2. **What challenges did you encounter?**
   1. **Computational cost**: Training was slow, especially with larger datasets.
   2. **Feature limitations**: Unlike CNNs, SVM does not automatically learn hierarchical patterns in images.

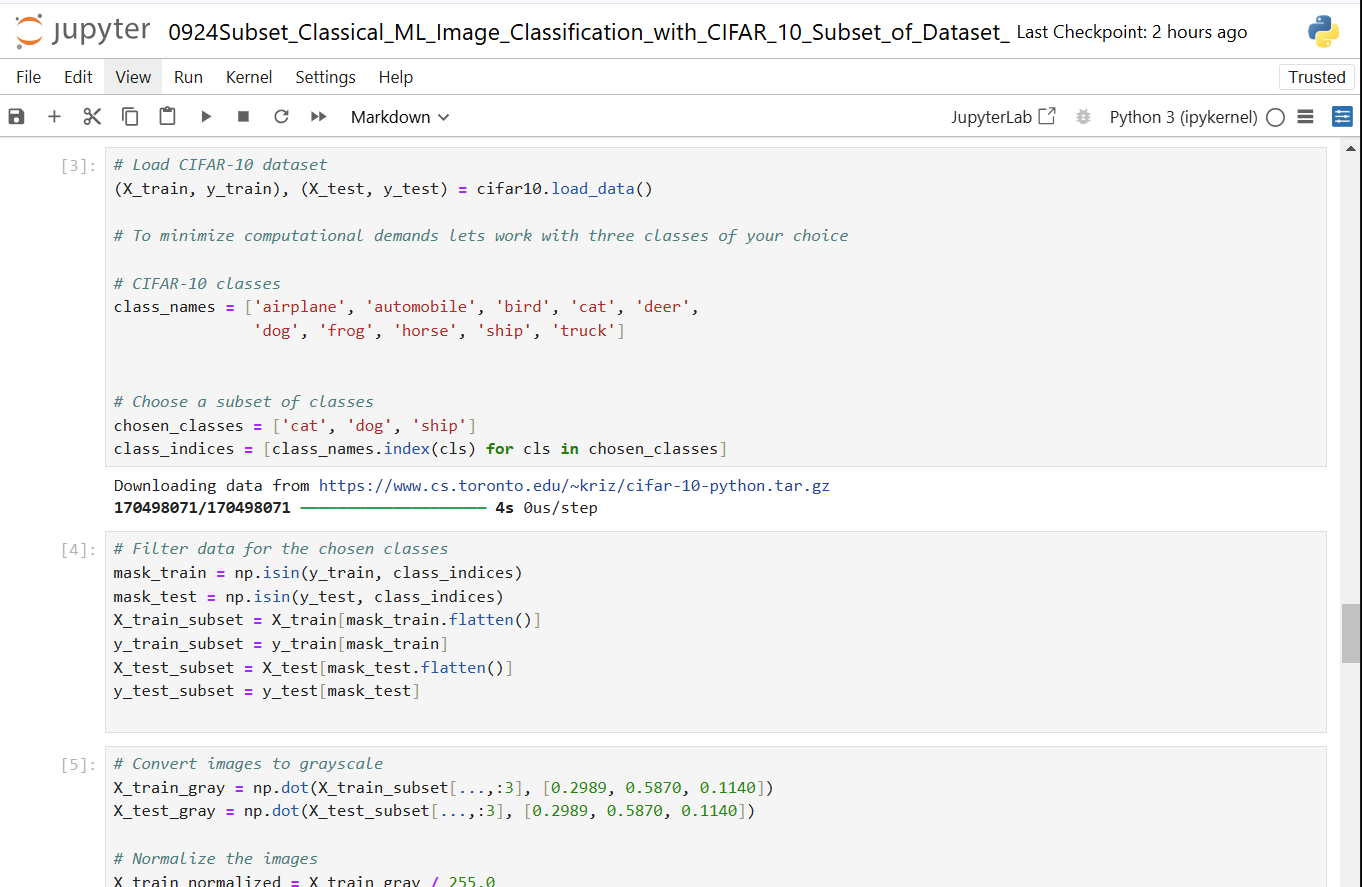
3. Inclusion of Visuals

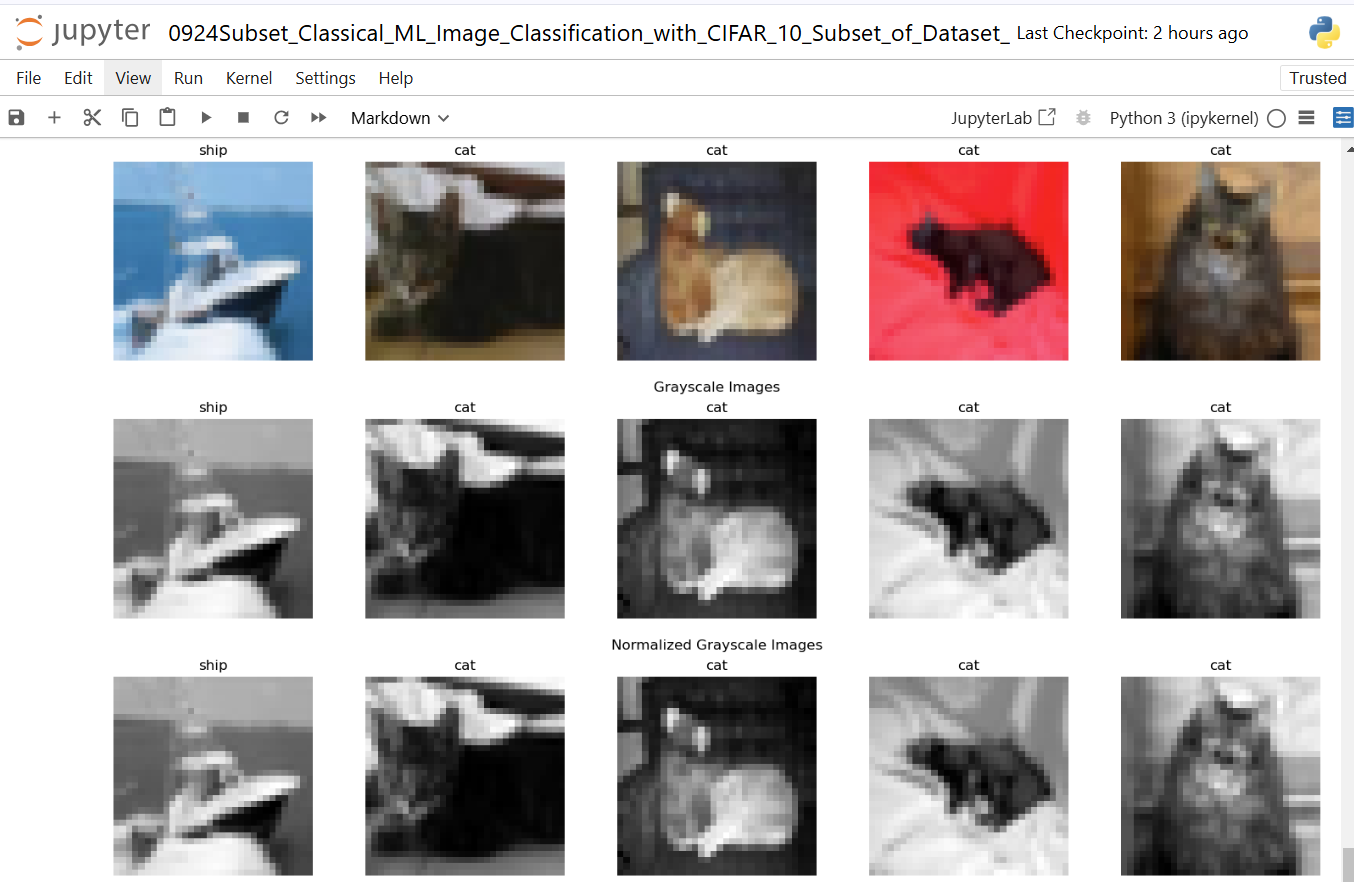
The following images were included to illustrate the process:

Sample dataset images to show raw input.

Grayscale-converted images (if used).

Classification results comparing true vs. predicted labels.





4. Critical Analysis: SVM vs. Other Models

Strengths and Weaknesses of SVM in Image Classification

**Strengths:**

* Works well for **small datasets** where deep learning models might overfit.
* Effective in **high-dimensional spaces**.
* The **kernel trick** enables classification of complex data.

**Weaknesses**:

* **Computationally expensive** for large datasets.
* Does not **automatically extract features**, unlike CNNs.
* Performance is highly dependent on **choosing the right kernel**.

Comparing SVM to CNNs

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **SVM** | **CNNs** |  |
| Feature Extraction | Manual (raw pixels) | Automatic (deep layers) |  |
| Performance | Moderate (~35-40%) | High (>90% for CIFAR-10) |  |
| Scalability | Poor for large datasets | Optimized for big data |  |
| Training Speed | Slow (quadratic scaling) | Faster (optimized for GPUs) |  |

**Conclusion**

SVM remains a powerful tool for classification, but it is less suited for large-scale image tasks compared to CNNs. A hybrid approach, where CNNs extract features that are then classified using SVM, could provide the best of both worlds.

**References**

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE, 86*(11), 2278–2324.

Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology (TIST), 2*(3), 1–27.

Schölkopf, B., & Smola, A. J. (2002). Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press.

Krizhevsky, A. (2009). Learning Multiple Layers of Features from Tiny Images. *University of Toronto*.

Cortes, C., & Vapnik, V. (1995). Support-Vector Networks. *Machine Learning, 20*(3), 273–297.

Hsu, C.-W., Chang, C.-C., & Lin, C.-J. (2003). A Practical Guide to Support Vector Classification.