# Reflective Journal: Image Classification with SVM and CIFAR-10

## Introduction

This reflective journal documents my learning experience while working on image classification using the Support Vector Machine (SVM) algorithm with the CIFAR-10 dataset. The goal of this task was to load and preprocess image data, train an SVM model, and evaluate its performance in classifying images into different categories.

## Reflection on Learning

### Understanding SVM and Its Application

Support Vector Machine (SVM) is a supervised learning algorithm commonly used for classification tasks. It works by finding an optimal hyperplane that maximizes the margin between different classes in a dataset. For image classification, SVM can be used with feature-extracted images to classify them into predefined categories.

### Data Preparation Steps

The data preparation process involved several important steps:  
- Loading the CIFAR-10 dataset using TensorFlow.  
- Converting the images from RGB to grayscale to simplify the feature extraction.  
- Flattening the grayscale images into 1D arrays for SVM compatibility.  
- Splitting the dataset into training and testing sets.

### Model Training and Evaluation

After preparing the data, an SVM classifier with a linear kernel was trained on the extracted features. The model was evaluated using accuracy metrics and classification reports, which provided insights into its performance. The accuracy obtained during the training and evaluation process was satisfactory.

### Challenges Faced and Solutions

Some challenges encountered during the lab included:  
- Issues with missing image files or incorrect data paths.  
- Insufficient dataset size causing errors in train-test splitting.  
- Model overfitting when working with a small dataset.  
  
These challenges were resolved by carefully verifying the dataset structure, ensuring enough samples were available, and debugging the code where necessary.

## Responses to Lab Questions

1. The CIFAR-10 dataset consists of images from 10 different categories. For this experiment, a subset of images (e.g., cats and dogs) was used for classification.  
2. The images were converted to grayscale to reduce complexity and improve feature extraction.  
3. The SVM model was trained with a linear kernel, achieving a high classification accuracy.  
4. The evaluation metrics, including precision, recall, and F1-score, indicated good model performance.  
5. If given a larger dataset, a deep learning approach (e.g., Convolutional Neural Networks) might provide even better results.

## Visualizations and Results

The images below represent the original, grayscale, and normalized versions of the dataset, along with classification results.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

## Conclusion

This exercise provided a valuable introduction to image classification using classical machine learning techniques. Through this lab, I learned the importance of data preprocessing, the strengths and limitations of SVM for image classification, and practical debugging techniques. Future improvements could include using deep learning models to achieve even higher classification accuracy.

**References**

* Cortes, C., & Vapnik, V. (1995). *Support-vector networks*. Machine Learning, **20**, 273-297.
* Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, **12**, 2825-2830.
* Krizhevsky, A. (2009). *Learning Multiple Layers of Features from Tiny Images*. CIFAR-10 Dataset.
* Scikit-Learn Documentation. (2024). *Support Vector Machines*. Retrieved from: https://scikit-learn.org/stable/modules/svm.html
* TensorFlow Documentation. (2024). *CIFAR-10 Dataset*. Retrieved from: https://www.tensorflow.org/datasets/catalog/cifar10