Infrared Image Cyclone Classification
University of Piraeus - NCSR Demokritos
Machine Learning

First Author
Stamatios Orfanos
mtn2211
stamatisorfanos99@gmail.com

Abstract

This project aims to explore the effectiveness of Support Vector Machines (SVM) for classifying cyclone categories from images. Specifically, two popular SVM methods, one over one and one over others, will be compared for their performance in this task. The project will involve preprocessing the image data to extract relevant features, such as color and texture, and then training and testing the SVM models on a dataset of labeled infrared cyclone images.

1 Introduction

The classification of weather conditions from infrared images has become increasingly important in various industries, including aerospace and aviation. Accurate and timely identification of weather patterns can help airlines, pilots, and air traffic control personnel make informed decisions about flight planning, routing, and safety.

In particular, the ability to classify weather conditions from infrared images provides valuable information about clouds, fog, and precipitation that are not easily detected by visible light cameras. Therefore, the purpose of this project is to explore the potential of Support Vector Machines (SVM) for classifying weather conditions from infrared images, with the aim of improving the accuracy and efficiency of weather-related decisions in the aerospace and aviation sectors.

Specifically, we will compare the performance of two SVM methods, one over one and one over others, to determine which method is more effective in this context. The findings of this study will provide insights into the usefulness of SVMs for automatically classifying weather conditions from infrared images, and how this technology can be applied in the field of aerospace and aviation.

2 Methodology

A rough outline of the methodology used to develop the infrared cyclone classification system includes the creation of two SVMs models. While training we have a validation split of 10%, which is enough given the size of the data set that we are using in this paper. Next we are going to save the model in order to predict the category of unknown image data.

2.1 Data

Initially the data set we are going to be using in this paper comes from the website Kaggle, which is found here Data. This data set contains all INSAT3D captured INFRARED and raw Cyclone Imagery over the Indian Ocean from 2012 to 2021 along with each Cyclone Image intensity in KNOTS. The Raw Data has been sourced from the MOSDAC server, where each image is accompanied by the number of knots. The data set was created to train a CNN-Convolutional Neural Network and predict the knots of each cyclone but using the information provided by Cyclone Scales we were able to classify each image to one of the five possible categories. An example of a category 3 cyclone is the following:

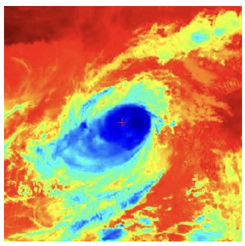

Figure 1: Category 3 Cyclone

2.2 Explanatory Data Analysis

Explanatory Data Analysis (EDA) is a crucial step in any machine learning project, including the classification of weather conditions from infrared images. EDA involves exploring and visualising the data to gain insights into its distribution, patterns, and relationships, and to identify potential issues, such as missing values and outliers. Through EDA, we can identify relevant features that are important for classification, such as texture and colour, and assess the quality of the data, including the presence of noise or artifacts.

EDA also helps in selecting an appropriate SVM method, as it provides insights into the characteristics of the data that may influence the performance of different SVM models. The first step in this procedure is to check the distribution of data for each category, where we can see that the distribution is fairly even among the categories with categories three, four and five having the most with thirty-two.

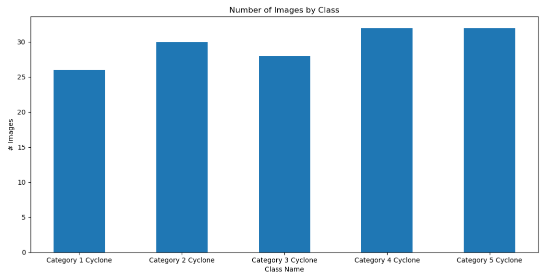

Figure 2: Distribution of Data

The second step is to get the height and width of all images, in order to understand the variation in size. This step is really important when using images as data since when using SVMs we need to have a certain size for each image that is going to be used as training data. The result of this step is the following:

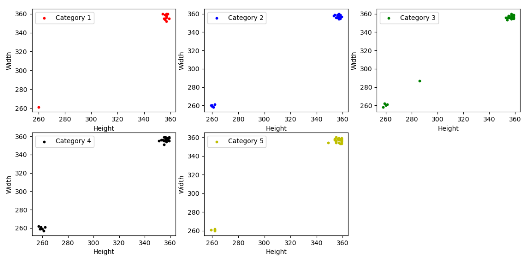

Figure 3: Images size for each Category

From this step of the analysis we found that the minimum height of an image was $h = 257$ and the minimum width was $w = 257$ giving as a image size of $(257, 257)$. The next step is to resize all the images in order to start the training process, so we create a new directory Data with each category.

The final step of explanatory data analysis includes finding the average image of each category in order to get a first idea about the differences between each category.

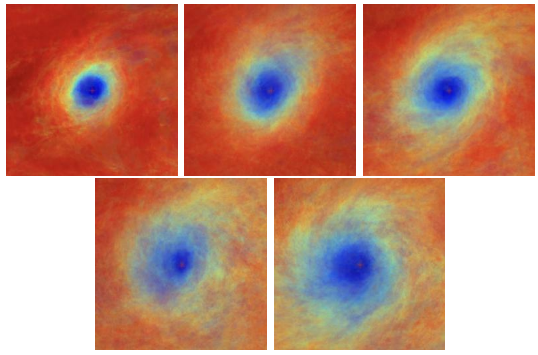

Figure 4: Cyclone Category from 1-5
(from right to left)

Given the information above we can understand that the differences between categories are not big, but follow a certain pattern. The main difference that we can observe is the fact the the centre of the cyclone is bigger in categories five and four and it gets smaller after that.

2.3 Support Vector Machines

SVM is a supervised machine learning algorithm that helps in classification or regression problems. It aims to find an optimal boundary between the possible outputs. SVM does complex data transformations depending on the selected kernel function and based on that transformations, it tries to maximise the separation boundaries between your data points depending on the labels or classes you have defined.

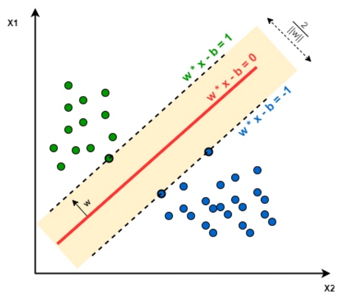

Figure 5: SVM Example

To generalise, the objective is to find a hyperplane that maximises the separation of the data points to their potential classes in an n-dimensional space. The data points with the minimum distance to the hyperplane (closest points) are called Support Vectors. In its most simple type, SVM does not support multi-class classification natively. It supports binary classification and separating data points into two classes. For multi-class classification, the same principle is utilised after breaking down the multi-classification problem into multiple binary classification problems.

The idea is to map data points to high dimensional space to gain mutual linear separation between every two classes. This is called a **One-to-One** approach, which breaks down the multi-class problem into multiple binary classification problems. A binary classifier per each pair of classes. In the One-to-Rest approach, the classifier can use $\frac{m(m-1)}{2}$ SVMs. In the One-to-One approach, we need a hyperplane to separate between every two classes, neglecting the points of the third class. This means the separation takes into account only the points of the two classes in the current split.

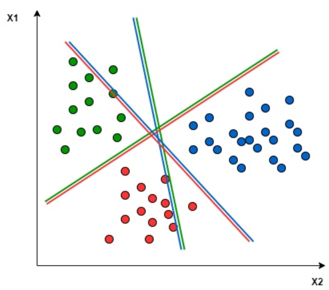

Figure 6: One-to-One SVM Example

Another approach one can use is **One-to-Rest**. In that approach, the breakdown is set to a binary classifier per each class. In the One-to-One approach, the classifier can use m SVMs. In the One-to-Rest approach, we need a hyperplane to separate between a class and all others at once. This means the separation takes all points into account, dividing them into two groups; a group for the class points and a group for all other points.

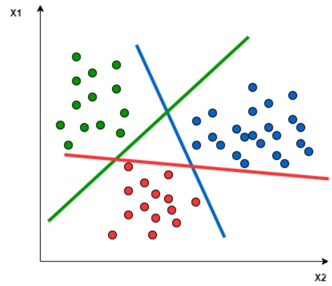

Figure 7: One-to-Rest SVM Example

3 Training

Testing the performance of the two SVM models, one over one and one over others, is a crucial step in this project. To evaluate the models, we will measure the accuracy, confusion matrix and F1-score of each model. These performance metrics will provide valuable information about the effectiveness of the models in classifying different cyclones based on severity category.

In this project we import the **Scikit-learn** package and use the **sklearn.svm** module. From this module we get the SVC and LinearSVC methods using the following parameters:

1. Regularisation parameter C : 0.85

2. Kernel: $poly - Polynomial$

3. Kernel coefficient: γ : $auto$

Finally based on the each SVM approach we get the following:

1. Decision Function Shape parameter ovo in one versus one

2. Decision Function Shape parameter ovr in one versus rest

3.1 Accuracy

In this problem we examined five different categories, which means that the random choice would have 20% accuracy. In this paper we were able to produce two models that classify data with accuracy:

1. One versus One Accuracy: 73%

2. One versus Rest Accuracy: 73%

The main problem we faced after many iterations is that shuffling of the data and the train and test split are very important to the success of the classifier. In the worst case we got accuracy of 43%, while in the best scenario we got accuracy of 80% and the most common accuracy among the number of experiments was 73%. The problem in this case is the lack of data, because it reduces the diversity and complexity of the training set, leading to over-fitting and poor generalisation to new data.

3.2 Confusion Matrix

The confusion matrix for the One versus One model is the following:

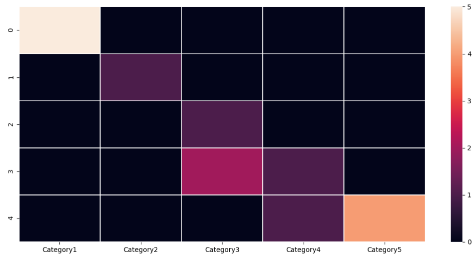

Figure 8: One-to-One Confusion Matrix

The confusion matrix for the One versus Rest model is the following:

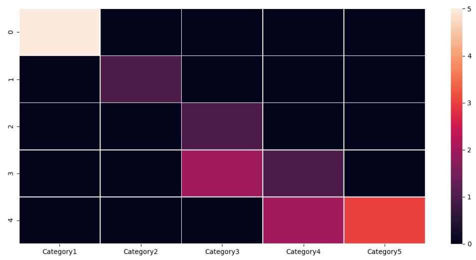

Figure 9: One-to-Rest Confusion Matrix

In both cases we can see that the models make mistakes. The positive in this case is the fact that the difference between the prediction and the actual result is not great. We understand that the models predicted a category five cyclone as a category four in both cases, which is a wrong but not as bad as predicting a cclone of category one or two.

3.3 F1 Score

The macro-F1 score of the models produced in this paper are:

1. One versus One macro-F1: 75%

2. One versus Rest macro-F1: 73%

4 Conclusion

Although we achieved very good results in classifying the infrared images of cyclones with an average accuracy of 73%, we faced the problems of a small data-set.When a model is trained on a small data-set, it may memorise the training examples rather than learning the underlying patterns and relationships between the input features and the output labels. In order to achieve a more stable accuracy we have to overcome the negative effects of a lack of data, it is important to use appropriate data augmentation techniques, such as rotation, flipping to artificially increase the size and diversity of the training set.

In conclusion, machine learning can be a powerful tool for image classification tasks, but it has its limitations. One of the main drawbacks of traditional machine learning models, such as SVMs, is that they rely on hand-crafted features and may not be able to capture the full complexity of image data. Moreover, the performance of traditional machine learning models can be limited by the size and diversity of the training set. Deep learning, on the other hand, offers a more powerful and flexible approach to image classification by using neural networks to automatically learn the relevant features and relationships between the input data and the output labels. Deep learning models, such as convolutional neural networks (CNNs), can extract hierarchical and abstract features from images, which can lead to better accuracy and generalisation on complex and diverse data-sets.

In conclusion, while traditional machine learning models such as SVMs can be useful for image classification tasks, deep learning offers a more powerful and flexible approach that can better handle the complexity and diversity of image data. However, it is important to carefully select the appropriate approach based on the specific requirements and characteristics of the task at hand

References

SVM Programming Basics - Scikit-learn
Support Vector Machine — Introduction to Machine Learning Algorithms
MIT Support Vector Machine