Image classification

1. Introduction

The following report shows the results of our application of simple classification algorithms to Sentinel-2 satellite images in JPG format with three spectral bands (RGB). The aim is to demonstrate the feasibility of this application.

2. Description of the Process

2.1 Data

The dataset provided contains 27,000 images with 13 spectral bands representing 10 different land classes. To simplify the task for us, we use the images in JPG format with three spectral bands (RGB).

2.2 Data Loading

We imported all the necessary libraries and loaded all the files into the folders to determine the number of images available in each folder. In addition, we defined a different folder name for each class and stored the images along with their labels, to provide easier and more organized access to the image data.

2.3 Dataset Preparation

The dataset was divided into training and test sets to effectively evaluate the models. The training set contains (80%) and a test set (20%). Using all three classifiers (DecisionTreeClassifier, LogisticRegression, KNeighborsClassifier) we generated models to extract insights from the images.

2.4 Descriptive data Analyze

In our descriptive data analysis, we conducted some analyses to gain insight into the dataset. First, we quantified the number of images in each folder, which provides an understanding of the balance or imbalance between different land classes. Next, we examined the dimensions of each image to ensure consistency and compatibility for subsequent analyses. We also visualized five random images from each folder, this allowed us to explore the diversity of images and landscapes within the dataset. Finally, we computed histograms of the red, green,

and blue channels for each image, allowing us to visualize the distribution of pixel intensities and understand the color composition and intensity variations across the dataset.

2.5 Model Configuration

We implemented three classification models: Decision Trees, Logistic Regression, and k-NN. These models were trained on the training dataset and evaluated on the test dataset.

DecisionTreeClassifier():
"Decision Trees can be used to solve both classification and regression problems. The algorithm can be thought of as a graphical tree-like structure that uses various tuned parameters to predict the results. The decision trees apply a top-down approach to the dataset that is fed during training." - Towards Data Science

LogisticRegression():
"Logistic Regression is one of the most simple and commonly used Machine Learning algorithms for two-class classification. It is easy to implement and can be used as the baseline for any binary classification problem. Its basic fundamental concepts are also constructive in deep learning." - Datacamp

KNeighborsClassifier():
"KNeighborsClassifier is an algorithm that effectively categorizes data points according to the trends found in that said point's nearest data points or neighbors." – educative

3. Experiments and Results

In our experiments, we evaluated the classification performance using metrics such as accuracy, precision, recall and F1 score. In addition, we generated confusion matrices to gain deeper insight into the behavior of the classifiers. To enhance the clarity of our analysis, we visualized the confusion matrices for each classifier.

Accuracy can be calculated globally.
Precision, sensitivity, specificity and F1 score are calculated for each class.
Accuracy calculates the proportion of correct predictions out of the total number of predictions. Precision measures how many of the model's positive predictions are correct. Sensitivity measures how well the model identifies true positive cases.

And the F1 score calculates a mean between precision and recall, which provides a balanced measure of the model's accuracy.

One problem we encountered was the "ITERATIONS REACHED LIMIT" error, which typically occurs when the volume of data exceeds the processing capacity. This could be due to an overwhelming amount of data being processed, leading to computational constraints.

4. Discussion and Conclusion

DecisionTreeClassifier():
The results of the decision tree classifier show a globally accuracy of 47%, this means it correctly classifies around 47%. The other classifications such as precision, recall and F1 score show similar values in each reflection. The land classes 'Forest' and 'Sealake' have high values, there the classifier is able to perform well. For 'Highway' and 'PermanentCrop' the reflections have lower values.
On closer inspection, random examples of 'Forest' and 'SeaLake' reveal their relatively monotonous nature, with no significant variation in appearance. This homogeneity probably contributes to the higher metric performance observed in these classes.

LogisticRegression():
Logistic regression calculates a "STOP: TOTAL NO. OF ITERATIONS REACHED LIMIT", which means that the amount of data exceeds the processing capacity. We still get a calculation of accuracy, precision, recall and F1 score, but not with all the data.
The global bounded accuracy of logistic regression is 39%, it classifies about 39% correctly. Overall, the classification values are generally about 10-20% lower than for the DecisionTree Classifier, considering that not the whole dataset was used. Nevertheless, it shows similar results in each reflection and the higher metric values for the 'Forest' and 'SeaLake' classes.

KNeighborsClassifier():
The classification report of the KNeighbors classifier shows a global accuracy of 34%. The computed values such as precision, recall and F1 score show different values for the same classes, this indicates inconsistencies in the performance of the classifier for this class. For example, 'Resendtials' contains a high precision and a low recall value, this indicates that the classifier is correctly predicting the classes to which they belong but is missing the true cases.

The confusion matrix also contains a lot of zeros, which means that the classifier did not make any predictions for certain classes. This can happen because the nearest neighbor classifier is computationally and sensitive to the choice of distance metric and the value of k.

In conclusion, it can be said that the DecisionTreeClassifier produces the best results.
The LogisticRegression is limited in its dataset and the KNeighbors computes inconsistencies.
The DecisionTreeClassifier has a global accuracy of 47%, which means that it correctly classifies about 47% correctly. For a professional classification, less than half could be considered a suboptimal classification. Nevertheless, in our case it proves that the classification is possible.

5. Reference

An Exhaustive Guide to Decision Tree Classification in Python 3.x - Oct 27, 2021, Towards Data Science
https://towardsdatascience.com/an-exhaustive-guide-to-classification-using-decision-trees-8d472e77223f

Understanding Logistic Regression in Python Tutorial - Dec 2019, Datacamp
https://www.datacamp.com/tutorial/understanding-logistic-regression-python

KNeighborsClassifier in scikit-learn - 2024, educative
https://www.educative.io/answers/kneighborsclassifier-in-scikit-learn

PowerPoint: 'Descriptive Analyze'
PowerPoint: 'Predictive Analyze'
Example Code: 'Image Classification Day and night'