Satellite Images Classification

Simone Zagaria, Gennaro De Cicco, Petra Udovicic,
Ana Carina Branescu, Sai Swaroop Chittoor

December 27, 2023

1 Introduction

The aim of our project was to explore the 'Satellite Images' dataset and develop a model to classify its images by using and comparing two techniques: Convolutional Neural Network (CNN) and Logistic Regression. The goal was to recognize images from a satellite and identify four key elements: Clouds, Deserts, Green areas, and Water.

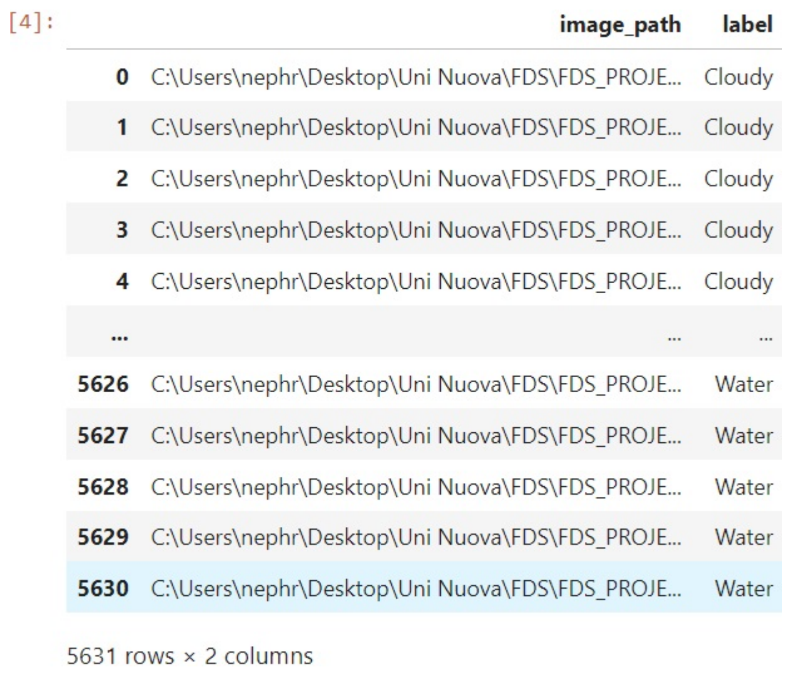
We thought that this model would be valuable because it could have plenty of applications, such as: biodiversity monitoring and conservation, climate change analysis and Urban planning.

2 Data preparation

Initially, we downloaded the satellite images folder from Kaggle and created a dataset in CSV format that includes all image paths and their corresponding categories as labels. The dataset consisted of the following columns:

'image_path' : The path in which the image file is stored

'label' : The class to which the image belongs



[4]:

	image_path	label
0	C:\Users\nephr\Desktop\Uni Nuova\FDS\FDS_PROJE...	Cloudy
1	C:\Users\nephr\Desktop\Uni Nuova\FDS\FDS_PROJE...	Cloudy
2	C:\Users\nephr\Desktop\Uni Nuova\FDS\FDS_PROJE...	Cloudy
3	C:\Users\nephr\Desktop\Uni Nuova\FDS\FDS_PROJE...	Cloudy
4	C:\Users\nephr\Desktop\Uni Nuova\FDS\FDS_PROJE...	Cloudy
...	...	...
5626	C:\Users\nephr\Desktop\Uni Nuova\FDS\FDS_PROJE...	Water
5627	C:\Users\nephr\Desktop\Uni Nuova\FDS\FDS_PROJE...	Water
5628	C:\Users\nephr\Desktop\Uni Nuova\FDS\FDS_PROJE...	Water
5629	C:\Users\nephr\Desktop\Uni Nuova\FDS\FDS_PROJE...	Water
5630	C:\Users\nephr\Desktop\Uni Nuova\FDS\FDS_PROJE...	Water

5631 rows × 2 columns

The dataset comprised 5631 rows distributed among four distinct classes. Subsequently, we calculated the distribution of images within each category and we visually represented the results through a bar graph:

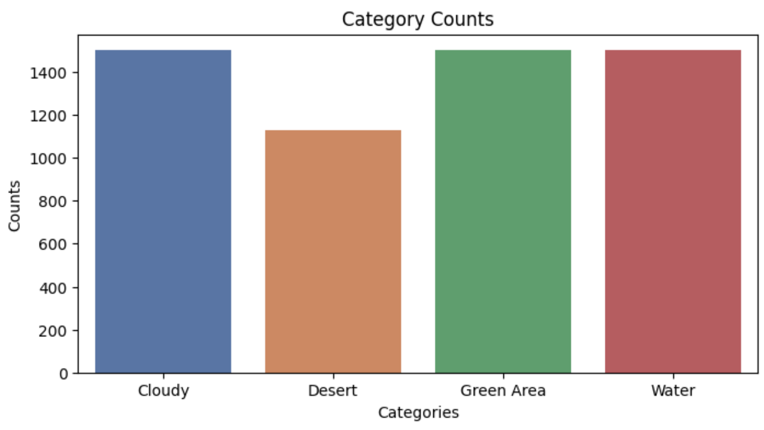


Figure 1: Bar graph of the distribution among categories

As we can see, The dataset appears to be evenly distributed across these categories, in fact the number of images for each category is:

1. Cloudy: 1500 images
2. Desert: 1131 images
3. Green Area: 1500 images
4. Water: 1500 images

3 Data Pre-processing

Next, we performed data splitting for training and testing, along with pre-processing. We opted for an 80-20 split for training and testing due to the relatively small size of the dataset.
In the pre-processing phase:

Both the images from the training and testing set were resized to 255 × 255 for normalization purposes.
Images from the training test were zoomed, sheared, flipped horizontally and vertically, and rotated. The purpose was to diversify the training data, allowing the model to learn more varying patterns and improve robustness.

below are some samples from the training set:

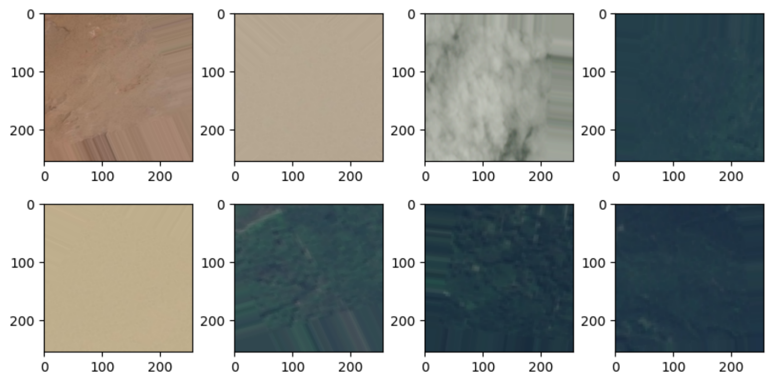


Figure 2: Sample images from the training set

On the other hand, images from the testing set weren't modified further because we aimed to test the model on real world images.
Below some examples from the testing set:

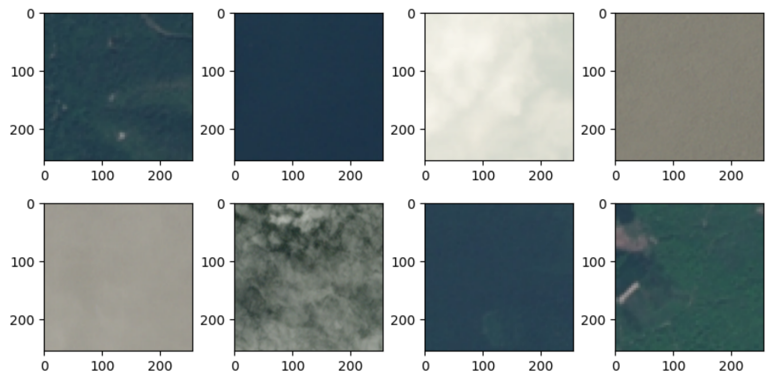


Figure 3: Sample images from the testing set

4 CNN

4.1 Model Creation and Training

The CNN model created contained in total 11 layers, including:

- 3 convolutional layers, where:
  - the first one was with 32 filters, each of size (2,2).
  - the second one was with 64 filters, each of size (2,2).
  - the third one was with 64 filters, each of size (3,3).
- 3 pooling layers:
  - the first one was applied after the first convolutional layer with a pool size of (2,2).
  - the second one was applied after the second convolutional layer with a pool size of (2,2).
  - the third one was applied after the third convolutional layer with a pool size of (2,2).
- 1 flatten layer to convert the images to 1D feature vectors.

- 2 dropout layers, where:
  - the first one was with a dropout rate of 0.1, applied after the Flatten layer.
  - the second one was with a dropout rate of 0.1, applied after the first Dense layer.
- 2 dense layers, where
  - the first one was with 512 neurons and ReLU activation.
  - the second one is the final output layer with 4 neurons and softmax activation.

The model created was subsequently trained using the training set. We chose 11 epochs in the training as is it generally considered an optimal number of iterations for training a model. A smaller number might have resulted in underfitting, as the training process relies on iterations of gradient descent. On the other hand, too many epochs can cause the model to overfit.

4.2 Accuracy Testing and Results

We then proceeded to create a graph using the values for accuracy and loss using the data acquired from the training. In the accuracy graph, we observe a gradual, non-uniform increase in accuracy, while in the second graph, there is a gradual, non-uniform decrease in loss.

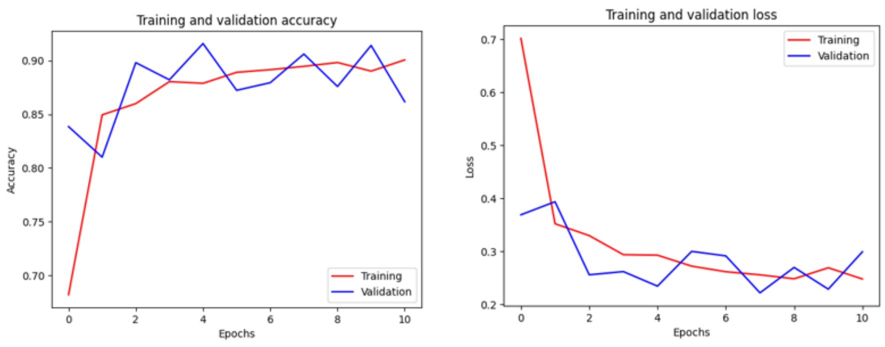
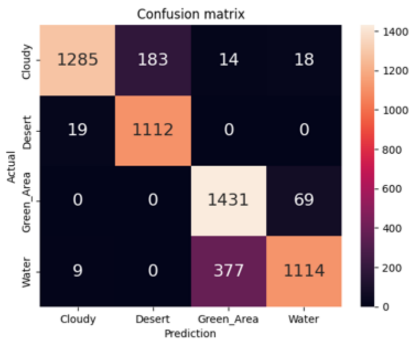
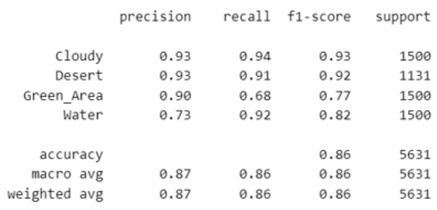


Figure 4: Accuracy and loss graphs for CNN model

Finally, we measured the accuracy of the predictions of the trained CNN model on the entire dataset using a confusion matrix:





```
              precision    recall  f1-score   support

      Cloudy       0.93      0.94      0.93      1500
      Desert       0.93      0.91      0.92      1131
  Green_Area       0.90      0.68      0.77      1500
       Water       0.73      0.92      0.82      1500

    accuracy                           0.86      5631
   macro avg       0.87      0.86      0.86      5631
weighted avg       0.87      0.86      0.86      5631
```

Figure 5: Confusion matrix and statistics in CNN model predictions

As we can see from the statistics on the right of the image, the CNN model demonstrates solid performance with an average precision of 0.87, indicating a high accuracy in correctly identifying positive instances, a recall of 0.86, suggesting that the model effectively captures a great portion of the true positive instances, and a F1-score of 0.86, demonstrating overall robust classification capabilities.

5 Softmax Logistic Regression

5.1 Model Creation and Training

In general, Softmax logistic regression can be viewed as simpler model than CNN with a single fully connected layer instead of multiple convolutional and pooling layers. In fact, the Softmax Logistic Regression model we created contained 2 layers:

- one flatten layer to convert the images to 1D feature vectors.
- one dense layer as the final output layer with 4 neurons and softmax activation.

The Logistic Regression model was then trained using 11 epochs.

5.2 Accuracy Testing and Results

We then proceeded to create a graph using the values for accuracy and loss using the data acquired from the training. In the accuracy graph, we observe a mediocre increase in accuracy, while in the second graph, there is a mediocre decrease in loss.

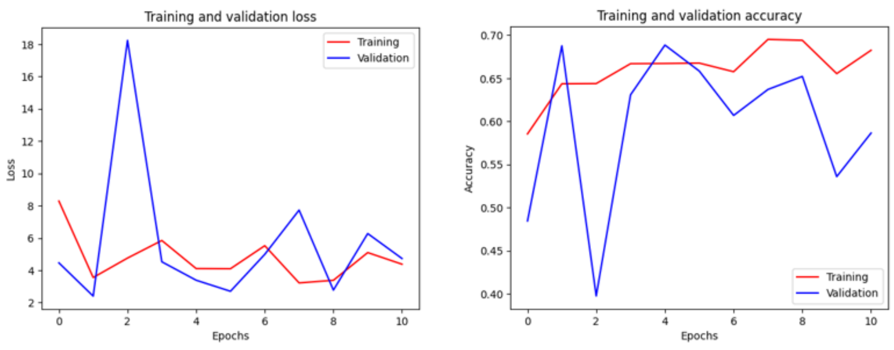
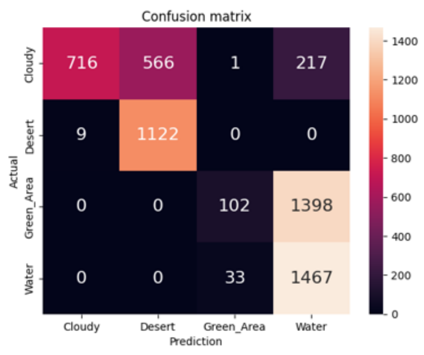


Figure 6: Accuracy and loss graphs for Logistic Regression model

And, to measure the accuracy, we predicted on all the images on the dataset using the trained Logistic Regression model and built a confusion matrix:



```
              precision    recall  f1-score   support

      Cloudy       0.99      0.48      0.64      1500
      Desert       0.66      0.99      0.80      1131
  Green_Area       0.75      0.07      0.12      1500
       Water       0.48      0.98      0.64      1500

    accuracy                           0.61      5631
   macro avg       0.72      0.63      0.55      5631
weighted avg       0.72      0.61      0.54      5631
```

Figure 7: Accuracy and loss graphs for Logistic Regression model

As indicated by the statistics on the right side of the image, the Logistic Regression model achieves rather modest scores compared to the CNN model, with an average precision of 0.72, recall of 0.63, and an F1-score of 0.55.

6 Conclusions

In conclusion, this article presented the construction and comparison of two models for satellite image classification: a Convolutional Neural Network (CNN) and a Logistic Regression. As expected, the CNN model demonstrated robust performance. On the other hand, Logistic regression struggled with complex image tasks due to its simple decision boundaries and limited ability to learn detailed features. These results demonstrate CNN model's effectiveness in handling the complexity of the classification task compared to the Logistic Regression model.

7 Group Organization

Initially, Simone Zagaria worked on the planning and divided the tasks to be done. In particular, the Dataset exploration, data preparation and pre-processing phase was done by Simone Zagaria and Gennaro De Cicco, the model creation and training and the accuracy graphs was done by Ana Carina Branescu and Sai Swaroop Chittoor and was revised by Simone Zagaria, whilethe confusion matrix and the predictions on random images was created by Petra Udovicic. The Softmax Logistic Regression model was done by Simone Zagaria and revised by Petra Udovicic. Finally, Simone Zagaria and Gennaro De Cicco took care of the final report.

8 References

The Dataset we used for this work is available here, moreover, the works we used as reference are available here and here.