Introduction to artificial intelligence
Application for Comparison of some classifiers in Python using Fruits dataset

Assyl Salah
Elif Ozdemir
Szymon Majorek

Supervised by: Jerzy Balicki

March 20, 2020

Contents

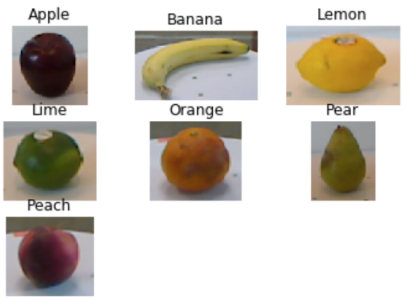
Chapter 1

Problem description

Nowadays precise classification of different kinds of fruit species, or food in general, is an important topic. It is not only a relevant topic in the field of academic research, but also in industrial applications. Many useful applications can be built based on such a classification system, for example it can be used in supermarkets to help cashiers. Cashiers need to identify not only the species of the fruit bought by the customer, but also its variety to determine the correct price. Such classification-based applications can identify the species purchased by the customer and match it automatically with the correct price. However, fruit classification based on computer vision remains difficult for the following reasons.

1. Shape, color, and texture similarity among numerous species of fruits, like the citrus genus, that contains oranges and grapefruits
2. Extremely high variation in a single class, which depends on the phase of fruit maturity, and the form in which the fruit is presented (e.g., fruits inside plastic bags, sliced on dishes, or unpicked).

Thus, we want to see how well can artificial intelligence complete the task of classifying them and which machine learning models of Scikit-learn library are the best in different comparison criteria.



Chapter 2

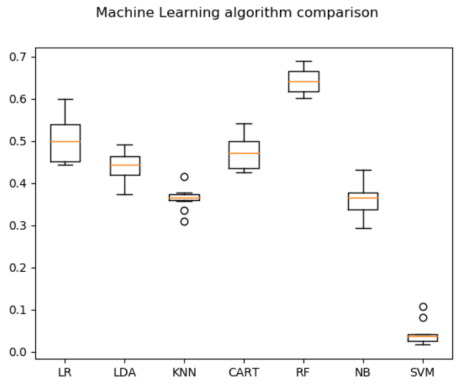
Problem analysis

In this project we compare Python Scikit-learn Classifiers depending on different criteria like accuracy, precision, mean absolute error, speed. The chosen Machine Learning models are the following:

- Logistic Regression
- Linear Discriminant Analysis
- K-Nearest Neighbors
- Decision Tree
- Random Forests
- Gaussian Naive Bayes
- Support Vector Machine
- Quadratic Discriminant Analysis
- Ada Boost classifier

The data is gathered from fruit images(28 thousand/44 species) by means of image processing. We extract global features like texture, color, shape and one local feature which is local binary pattern.
After to training and testing we get K-Fold Cross Validation results for all machine learning models. The data split is done by the train_test_split method from sklearn.model_selection.

The following is an example of comparison chart of classifier accuracies.



Chapter 3

Description of some existing solutions

The following page provides an example usage and comparison of different Python machine learning models on flower dataset. https://gogul.dev/software/image-classification-python
The above mentioned solution builds an intelligent system that's trained with massive dataset of flower/plant images The system predicts the label/class of the flower/plant using Computer Vision techniques and Machine Learning algorithms. It uses a simpler approach to produce a baseline accuracy for the problem. It uses the following three global feature descriptors:

- Color Histogram that quantifies color of the flower.
- Hu Moments that quantifies shape of the flower.
- Haralick Texture that quantifies texture of the flower.

The next step is extracting and gathering the features and labels. After which the solution trains and tests machine learning models using scikit-learn library. It uses train_test_split function provided by scikit-learn to split our training dataset into train_data and test_data. By this way, we train the models with the train_data and test the trained model with the unseen test_data. The split size is decided by the test_size parameter. It uses a validation technique called K-Fold Cross Validation. Finally, it chooses the best classifier depending on the results.
At the end of the article it suggests methods to improve the accuracy of classifiers. For example, gathering more data (in this case images) is a good step to get better results. And since image processing is an important part of this and our project, it mentions that extracting local features is also a very powerful tool to enhance the accuracy.

The next page provides a very simple overview of the classifiers. https://machinelearningmastery.com/compare-machine-learning-algorithms-python-scikit-learn/ The logic behind the algorithm is the same with the previous solution.

Chapter 4

Source code

Chapter 5

Results

Chapter 6

Conclusion

Bibliography

[1] https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_feature2d/py_sift_intro/py_sift_intro.html

[2] https://www.researchgate.net/publication/321475443_Fruit_recognition_from_images_using_deep_learning

[3] https://www.kaggle.com/moltean/fruits

[4] https://www.researchgate.net/publication/321475443_Fruit_recognition_from_images_using_deep_learning

[5] https://gogul.dev/software/image-classification-python