

SIMONE QUADRELLI



MACHINE LEARNING PROJECT

Academic year 2018 - 2019

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Dataset and Features</b>	<b>2</b>
2.1	RGB colour space . . . . .	2
2.2	Grayscale colour space . . . . .	3
<b>3</b>	<b>Data analysis</b>	<b>3</b>

## List of Tables

## List of Figures

1	Cumulative variance explainde by the components regarding the grayscale features . . . . .	4
2	Cumulative variance explainde by the components regarding the rgb features . . . . .	5
3	Training performances knn . . . . .	6

short abstract: what are you going to present in the report  
statement of the problem/goal of the analysis and description of the data set(s)  
list of three to five findings/keypoints  
the analysis with wise commentary  
(optional) theoretical background of the used methods  
conclusions (should include the findings/keypoints)  
the Appendix, containing all the R code

## 1 Introduction

The present project addresses the fruit and vegetable multiclass classification problem. Its objective of this project is to classify a variety of fruits and vegetables whose images are stored in a recently released dataset.

Fruit and vegetable classification is a challenging task, there are two different kind of problems that must be overcome: vegetables and fruits within the same class can differ in shape and colour due to maturity and growth. On the contrary, vegetables and fruits in different classes, such as red apples or grapes, may be very similar both in textures and in colours. Fruit and vegetable classification has a key role in many industries: it can be exploited to make agriculture more autonomous than ever before in history, but it can also be applied to all stages of the supply chain to assess the quality of the products from the producer to the consumer. The ability of harvesting robots can be enhanced by enforcing fruit and vegetables recognition, indeed it may allow them to recognize not only the type of fruit or vegetable but also its maturity or whether it is rotten. Moreover, autonomous classification provides more consistent quality assessment since it does not depend of the opinion of different workers.

The projects will analyze the classification performance of three machine learning algorithms on different fruits. The k-nearest neighbour (knn) algorithm was used to compute a baseline, a more sophisticated, the support vector machines (SVM) algorithm was later used for classification. The last approach was to construct a convolutional neural network because they are known to suit well the image classification. There are two different kind of features that were extracted to feed the algorithms: the grayscale and the rgb representation of the images.

The objective of the project are

- analyze the classification performance of the knn and svm using the grayscale on a variety of fruits
- analyze the classification performance of the knn and svm using the grayscale on a variety of fruits
- analyze the classification performance of the knn and svm using the rgb scale on a variety of fruits

- analyze the classification performance of the knn and svm using the rgb scale on a variety of fruits

Descrivere cosa si trova nelle sezioni successive

## 2 Dataset and Features

The analyzed dataset, known as *Fruits 360*, was originally published in 2017 by Horea Muresan and Mihai Oltean to address fruit and vegetable classification [1]. The existence of either too small datasets or too low-quality dataset was most compelling problem. On the contrary, this dataset provides more than 70000 images of 114 different fruits and vegetables. Each image has a definition of  $100 \times 100$  pixels. To increase the reliability of the results, it contains images of rotated fruits and vegetables. However those images make the training process much harder. As the authors explain, the dataset was produced as follows: fruits and vegetables were planted in the shaft of a low speed motor (3 rpm) and a short movie of 20 seconds was recorded. However, due to the variations in the lighting conditions, the background was not uniform and a dedicated algorithm was exploited to extract the fruit from the background. The images were of such high quality that no preprocessing was required. It is also worth noticing that the dataset was already splitted in train and test set, therefore no further splitting was required.

Fruits and vegetables have several distinct visual characteristics that can be extracted from the images. Colour, shape, size and texture are the most commonly used features<sup>1</sup> in image classification. Among all the possible features that can be extracted from an image, colours features play a key role in fruit and vegetable classification. Indeed, they seem to suit the domain and their computation is very fast and efficient [2]. The project focuses on the grayscale and rgb representation of the images.

The following lines introduce the reader to some concept of colour spaces and colour representation. Colour spaces are a specific organization of colors that allows to reproduce the representation of colours. Two different colour spaces were exploited in this project: grayscale and the rgb.

### 2.1 RGB colour space

The RGB colour model is an additive color model, meaning that red, green and blue are added together to reproduce an wide array of colors. A color in the RGB colour model is described by indicating how much of the red, green, and blue is included. The colour of a pixel is expressed as an RGB triplet (R,G,G). Each component can vary from 0 to 255, since each channel is represented by a byte.

---

<sup>1</sup>Features are the characteristics of an object that can distinguish it from other objects

## 2.2 Grayscale colour space

Since the dataset contains rgb images, it was necessary to extract a grayscale representation of the pixels to be used as features. In grayscale representation each pixel is a byte, meaning that the possible values it can be in range from 0 to 255.

The conversion algorithm implemented by the library *imager* is the Luma algorithm. Each rgb channel  $R, G, B$  of a pixel are passed to a function  $\Gamma(x)$  to obtain its transformation, respectively  $R', G', B'$ .

The  $\Gamma$  function is defined as:

$$x' = \Gamma(x) = Ax^\gamma \quad (1)$$

where  $x$  is the pixel intensity,  $A$  is a scalar whose value is often 1 and  $\gamma$  is usually  $\frac{1}{2.2}$ .

To obtain the final grayscale representation  $y$  of a pixel  $x$ , the following linear transformation is applied [3]

$$y = 0.21R' + 0.71G' + 0.07B' \quad (2)$$

## 3 Data analysis

The procedure described in the following is the same for grayscale features and rgb features. The first step in the analysis was the data extraction and the creation of the data matrix and the vector of the labels. The images have 100x100 pixels, but to limit the computational complexity of the following algorithms i decided to resize them to 50x50 pixels. The feature vector are extracted using the functions of the library *imager*. Once the data vectors are extracted, a data matrix is created. Each row of the matrix contains the linearized representation of the image. A separate vector of labels was created, one for each image.

---

**Algorithm 1** Schema of the analysis

---

- 1: feature extraction
  - 2: Dimensionality reduction
  - 3: Extraction of the most significant features
  - 4: Construction of a baseline using knns
  - 5: Evaluation of svm
  - 6: CNN
- 

Most learning algorithms are exponentially slow in the number of the feature or at least significantly slow down as the number of feature increases. Furthermore, it is reasonable to believe that many of the pixels are useless since they are just part of the background. As a result i choose to reduce the dimensionality of the data using the Principal component analysis (PCA). It is one of the most common technique to reduce the dimensionality of the data since it is very

efficient. Once all the principal component are computed we can choose a subset of the that cna explain most of the variance.

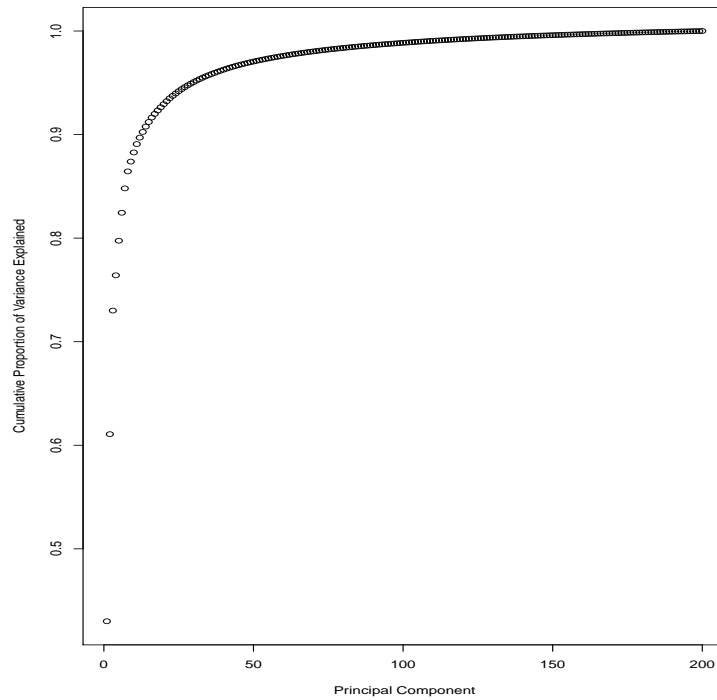


Figure 1: Cumulative variance explainde by the components regarding the grayscale features

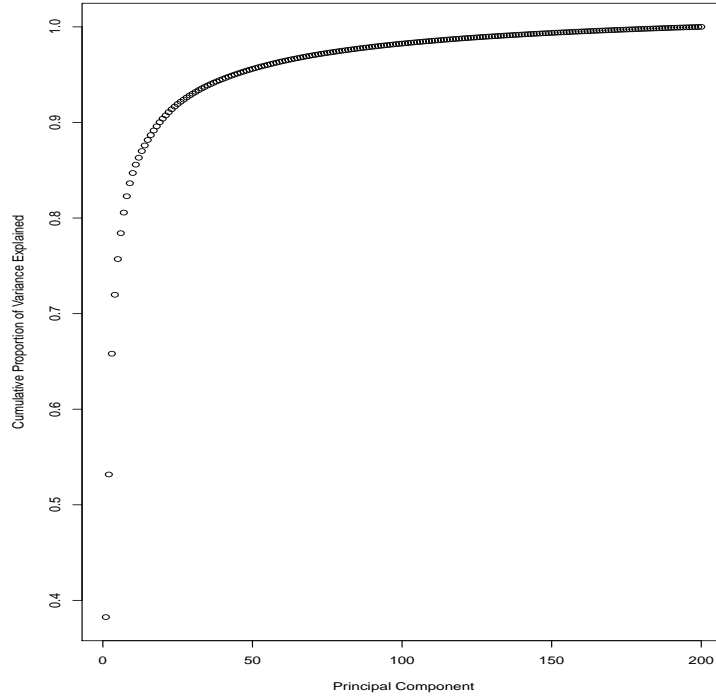


Figure 2: Cumulative variance explained by the components regarding the rgb features

As the plot above shows, in both cases, using 150 is enough to explain all the variance in the data. To compare and analyze the performance of the learning algorithms I choose to use the following set of principal components  $\{3, 10, 50, 100, 150\}$

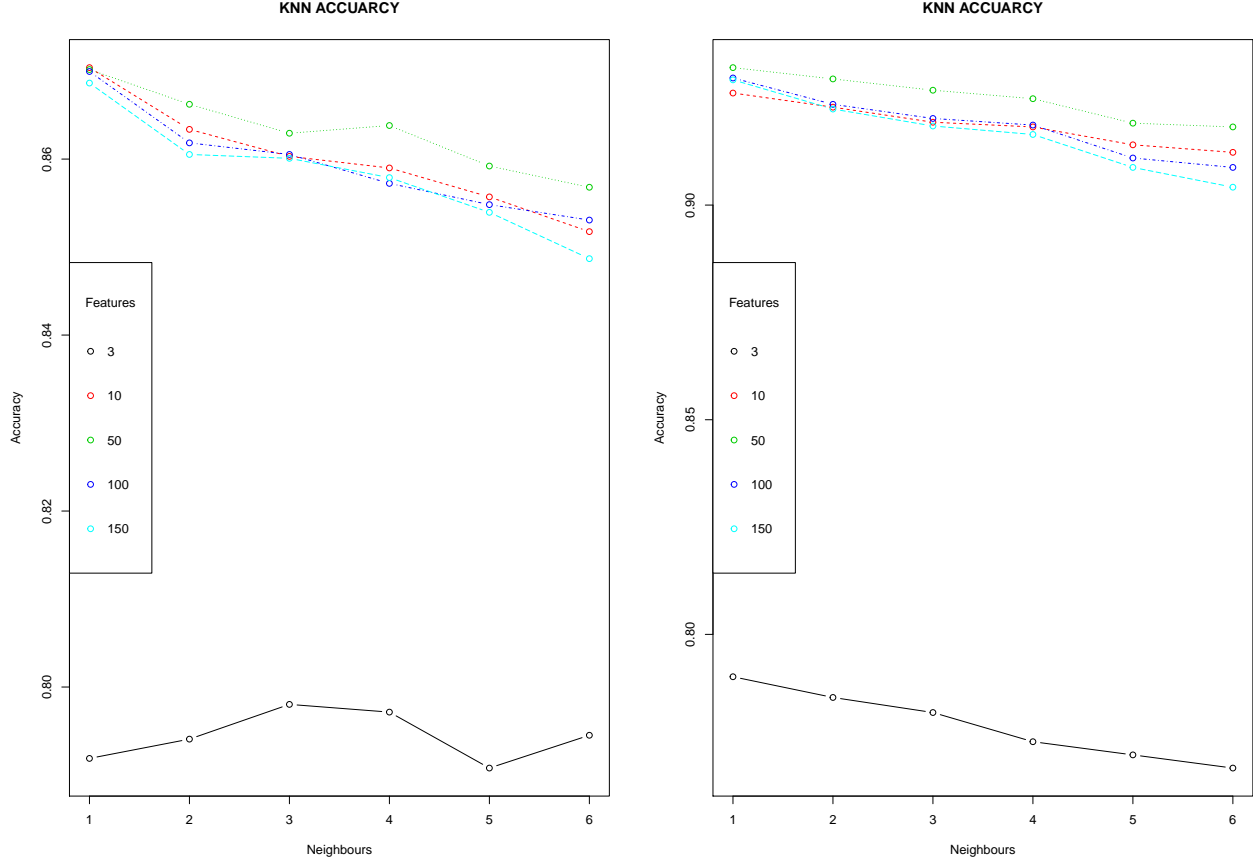


Figure 3: Training performances knn

## References

- [1] Horea Muresan, Mihai Oltean, *Fruit recognition from images using deep learning*, Acta Univ. Sapientiae, Informatica Vol. 10, Issue 1, pp. 26-42, 2018.
- [2] Khurram Hameed, Douglas Chai, Alexander Rassau, *A comprehensive review of fruit and vegetable classification techniques*. Image and Vision Computing 80 (2018) 22-44.
- [3] Samuel Macedo, Givanio Melo, Judith Kelner, *A comparative study of grayscale conversion techniques applied to SIFT descriptors*, SBC Journal on Interactive Systems, volume 6, number 2, 2015.