Homework 3 Report

Christina Pavlopoulou
cpavl001@ucr.edu

Andres Calderon
acald013@ucr.edu

March 3, 2016

1 Data

EarthExplorer[1] is a website where the U.S. Geological Survey makes available a large repository of satellite imagery for scientific purposes. We downloaded a scene from a suburb of Riverside using the High Resolution Orthoimagery dataset. The initial scene was large, so we sample two images (figures 1 and 2) which collect enough and diverse types of objects. For this assignment, we decided to classify pools.

For the first image we collected a total of 100 points, 50 of them where pools and the remaining were not pools, as our training set (Listing 1).

Figure 1: First image used for training.

[1] http://earthexplorer.usgs.gov/

Figure 2: Second image used for validation.

2 Classification

For the classification part, we trained a `kNN` classifier using the training set. We used the `rminer`[2] package, provided by the R project statistical software (`https://www.r-project.org/`), using the default parameters.

3 Validation

For validation, we used the second image to collect a group of 50 points as our testing set. We created a grid of `50x50` pixels in which we checked how many of testing points belong to the same cell (figure 3). If more than one point belongs to the same cell, we keep just one of them. As a result, we kept 38 valid points.

Firstly, we used these 38 points as our ground truth. Specifically, we defined two classes (pools and no pools) and we found which of the cells belong to each class. Then, for each one of the points, we try to find in which cell it belongs, by flooring its coordinates. If it belongs to a pool cell then we label it as 1 (true positive). Otherwise, the label is 0 (false positive). Listing 2 shows the code we used.

Apart from the ground truth, we also, use the testing set as the validation set. We used the `kNN` classifier that we trained in section 2 and we classified the testing set to obtain the probability scores. After that, we plot the ROC curve (figure 4) by using the `plotroc` function of `Matlab`, with the ground truth and the scores as parameters. Finally, we calculated the `AUC` (0.8646) by using the `perfcurve` function of `Matlab` (Listing 3).

[2] `https://cran.r-project.org/web/packages/rminer/index.html`

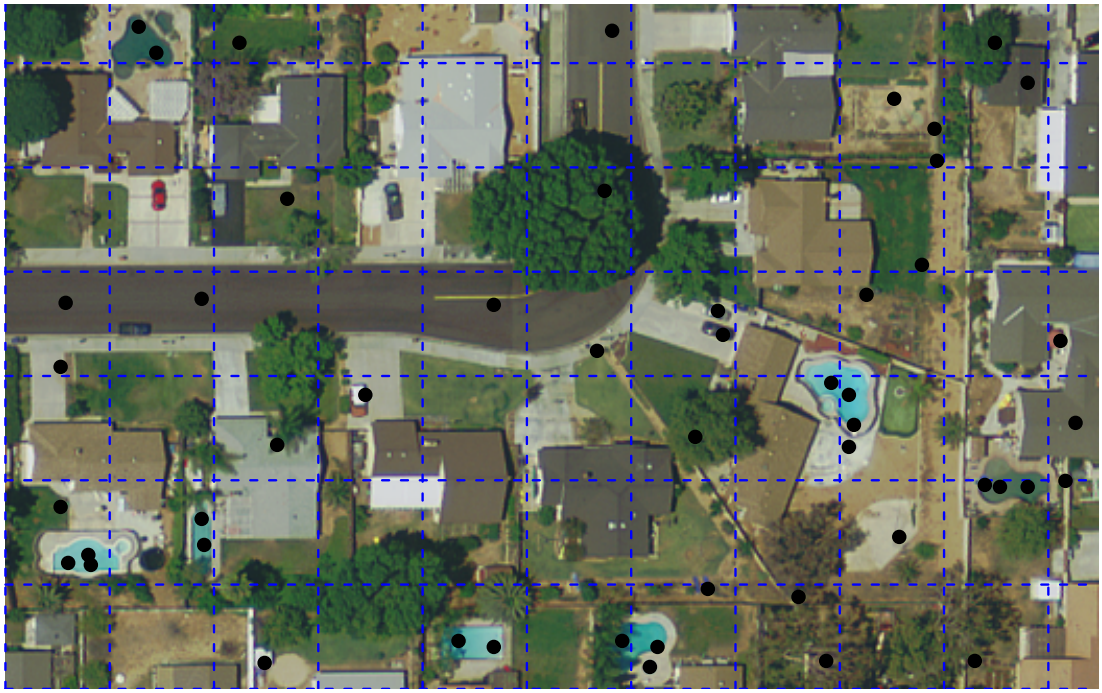

Figure 3: Locations for validation set in second image.

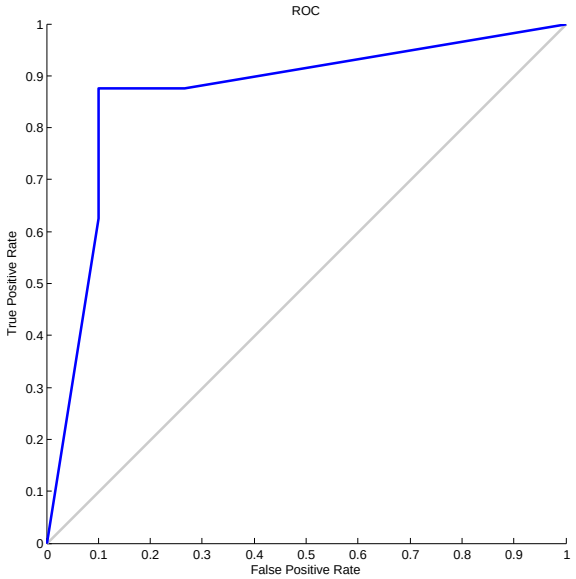

Figure 4: ROC curve as result of the validation. The `AUC` of the curve is 0.8646.

A Code

```
function hw3()
  im=imread('pools3b.tif');
  im=im(:,:,1:3);
  image(im)
  [x,y]=ginput(50)
  dlmwrite('coords.txt',horzcat(x,y))
  d = function1(x,y);
  display(d)
  dlmwrite('newvalpools2.txt',d)

  function[data]=function1(x,y)
    for i=1:numel(x)
      x1=x(i);
      y1=y(i);
      red(i)=im(round(y1),round(x1),1);
      green(i)=im(round(y1),round(x1),2);
      blue(i)=im(round(y1),round(x1),3);
      class(i)=1;
    end
    data=[red;green;blue;class];
    data=transpose(data);
  end
end
```

Listing 1: Code for point collection.

```
function validation()
  validateset=dlmread('valpools.csv');
  truegrid=dlmread('newpools_indeed.txt');
  truegridx=truegrid(:,1);
  truegridy=truegrid(:,2);
  x = modulo(validateset(:,1));
  y = modulo(validateset(:,2));
  validationgrids=horzcat(x,y);
  poolmap=containers.Map();

  for i=1:numel(x)
    flag=1;
    for j=1:numel(truegridx)
      if(x(i)==truegridx(j) && y(i)==truegridy(j))
        key=sprintf('%d,%d',x(i), y(i));
        poolmap(key)=1;
        flag=0;
        break;
      end
    end
    if(flag==1)
      key=sprintf('%d,%d',x(i), y(i));
      poolmap(key)=0;
    end
  end
  keys(poolmap)
  values(poolmap)

  function[x]=modulo(x)
    for i=1:numel(x)
      x(i)=x(i)-mod(x(i),50);
    end
  end
end
```

Listing 2: Code for collection of true or false positive points.

```
t=dlmread('truth.txt')
s=dlmread('scores.txt')
plotroc(t,s)
[X,Y,T,AUC] =perfcurve(t(1,:),s(1,:),1);
```

Listing 3: Code for `ROC` and `AUC` computing.