Introduction to Deep Learning

TP R 3 - Convolutional Neural Networks
5th year Statistics and Data Science

Ayoub Asri

02 November 2024

all data of this TP is available in this link :

https://drive.google.com/drive/folders/1ls07T82766Fg1tpsi_bB9zmvfdLSRSbf

1. The concept of convolution

using magick

Q1. read the museum picture provided using magick tools

Q2. create a horizontal filter and flip it to become a vertical filter

Q3. apply these two filters to the image

Q4. try other filters :

```
matrix1 <- matrix(c(0,1,1,0,1,1,0,1,1), nrow = 3, ncol = 3)
edge = matrix(c(0,0,0,0,0,0,1,1,1), 3, 3)
sharpen  = matrix(c(0,-1,0, -1,6,-1, 0,-1,0),3, 3)
embos  = matrix(c(-2,0,0, 0,1,0, 0,0,2),3, 3)
edge2 = matrix(c(-1,-1,-1, -1,8,-1, -1,-1,-1),3, 3)
edge3 = matrix(c(0,1,0, 1,-4,1, 0,1,0), 3, 3)
```

using a predefined function for images with one channel

Q5. complete this function to create a convolution operation function

```
# Define a 2D convolution function
convolve2d <- function(image, kernel) {
  # Get dimensions of image and kernel


  # Output image placeholder
  output <- matrix(0, nrow = image_height, ncol = image_width)

  # Calculate padding
```

```
  # Apply convolution

    }
  }
  return(output)
}
```

Q6. read the image using this code

```
image <- readImage(system.file("images", "sample-color.png", package = "EBImage"))
```

Q7. display the image

Q8. define the same filters as part 1 plus a gaussian blur filter

Q9. apply all these filters to the provided image

Q10. **(advanced)** try to modify this function to deal with images with 3 channel

2. MNIST with convolution

Use a convolutional model (of your choice) to estimate the numbers of MNIST dataset.

3. CNN for time series :

toy example

Q1. Load the libraries

Q2. Simulate a simple sine wave as time series data (use a seed of 123 to insure the reproduction of the data)

Q3. define a function to prepare the time series for a CNN model. The function need to have a parameter : look_back

Q4. use a look_back to 10 and prepare the data

Q5. use a 80% rows of the data as training set

Q6. define a CNN model and compile it

Q7. fit the model by using 20% as a validation

Q8. predict the model on testing set

Q9. plot the predicted vs actual data

real data

Q10. read the data set "2014-12.csv" choose the variable **INDPRO** . Do the necessary transformations and answer all the previous questions

Q11. try other time series

4. Data Augmentaion in images

The goal of this exercise is to predict if a picture contains a cat or a dog using the data set provided "cats_and_dogs_filtered.zip"

Q1. How many images are there in the train and test (validation) folder

Q2. plot some random images of cats and dogs (you can use the EBImage package)

Q3. define a CNN model compile it and summarize it

Q4. normalize the data by multiplying by a rescaling factor

Q5. flow the training images using a 20 images batches

Q6. do the same for the testing set images

Q7. fit the model using the information from Q4-Q6.

Q8. evaluate the model. Is the model performant ?

To increase the performance of the model, we will use data augmentation technique

Q9. define data augmentation by transforming images using rotation, shifting, shearing, zooming,...

Q10. take a sample image and apply these transformations to it. Plot the results

Q11. define a CNN model and fit it on the augmented data

Q12. is there any improvement ?

Q13. use a CNN model including dropout

Q14. comment on the use of data augmentation

5. Cifar dataset

Q1. load the dataset CIFAR (if you don't have it you can use Fashion MNIST)

Q2. what are the dimension of images in the dataset

Q3. plot some random images of the dataset

Q4. get the training and testing data into right formatting format

Q5. define an appropriate CNN model

Q6. compile the model using predefined learning rate = 0.0001 and a weight decay 1e-6

Q7. fit the model on data and use a validation set

Q8. predict the data on the testing set

Q9. refit the model using an appropriate format of data augmentation

Q10. compare the results.

6. Advanced ideas : Convolution ships

Read the data set containing ships images (defined in the JSON) or read it directly.

link to the dataset ;

https://drive.google.com/drive/folders/1ls07T82766Fg1tpsi_bB9zmvfdLSRSbf

Create a CNN model to predict if the image contain a ship or not.

7. Advanced ideas : Exploring convolutional layers

Q1. read the MNIST data set and create the training and testing images

Q2. reshape the training and testing images :

- convert it to a 4-dimensional array (1 in the 4th array)
- convert the grey scale to 0-1 scale

Q3. visualize some random images from the test set. (for example image 35)

Q4. create a CNN with 2 convolutional layers and a two layers MLP

Q5. compile the model

Q6. using the compiled model get all the outputs the layers of the convolutional part of the model

Q7. create a model for this convolutional part only (definition input and output)

Q8. predict this model on the image from question 3

Q9. get the result of each layer separately.

Q10. (many intermediate steps) plot all the channels for each layer.

Q11. repeat question 10 with the image of the museum from exercise 1