CASUAL EXPLANATIONS OF IMAGE MISCLASSIFICATIONS

Neural Networks

A.Y. 2020-2021

Matricola: 1926056

Name: Nicola De Siena

Chapter 1: Introduction

*In this project,we study the Convolutional Neural Networks(**CNN**) related to classification image problems. In particular, the misclassification of elements of the dataset and what cause it.*

*It's very importante if we consider this scenario:for example,a CNN algorithm is deployed to predict the presence of a tumor in a particulare area using computed axial tomography (**CAT**) in the outpatient of geriatric medicine. A false negative result means that a tumor is missed by the algorithm, this could lead to catastrophic outcomes for the patient. Naturally, under this circumstance, the first reaction is to ask: why does the algorithm fail to classify certain cases? Scenarios as such are commonly seen across different fields. However, an satisfying answer to this "why" question has rarely been adequately provided. Instead, we focus on striving to enhance the algorithm performance with a goal of increasing the prediction accuracy and to search the best method.*

Chapter 2: Method

To analyze the misclassification in an image classification problem,we used the CIFAR-10 dataset:it consists of 60000 32x32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images and the 10 classes are completely mutually exclusive.For the preprocessing,we splitted the dataset using the 80:20 ratio.

We created three diferrent convolutional neural networks,each one of these with different paramaters,to training the dataset.

After that,we used the python library Matplotlib to create graphs for accuracy,loss,precision and to show the confusion matrix for every experiments.

We chose the confusion matrix,because it compares the actual target values with those predicted by our NN models in the experiments,so we can clearly visualize the level of misclassification.

For the training,in all experiments,we used 30 epochs and for batch size the value 128.

We also add some random functions to check the right and wrong predictions in every model.

Chapter 3 : Experiments

For the first experiment,we created a CNN,called Nicknet1,with 5 convolutional layer and 3 dense layers. For all the convolutional layers,the kernel size was (15,15) and as optimizer we picked Adam. As padding value we chose value 'same' and softmax as activation function.

After the training,with Nicknet 1,we obtained this values represented in these graphs below.

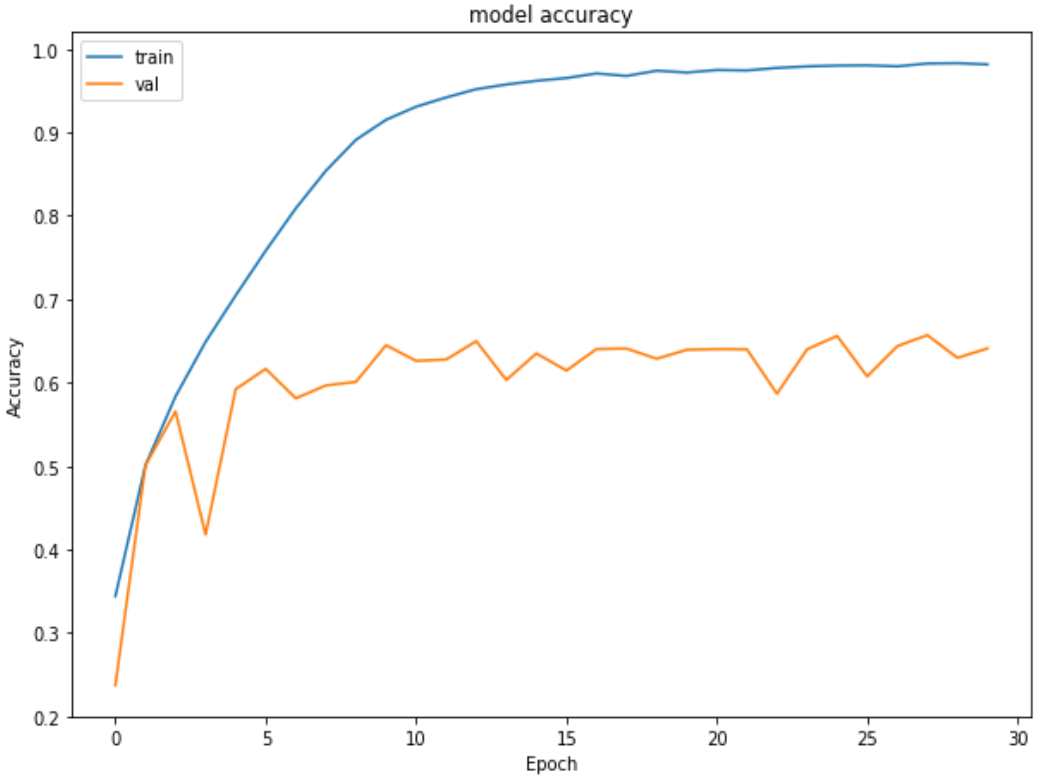
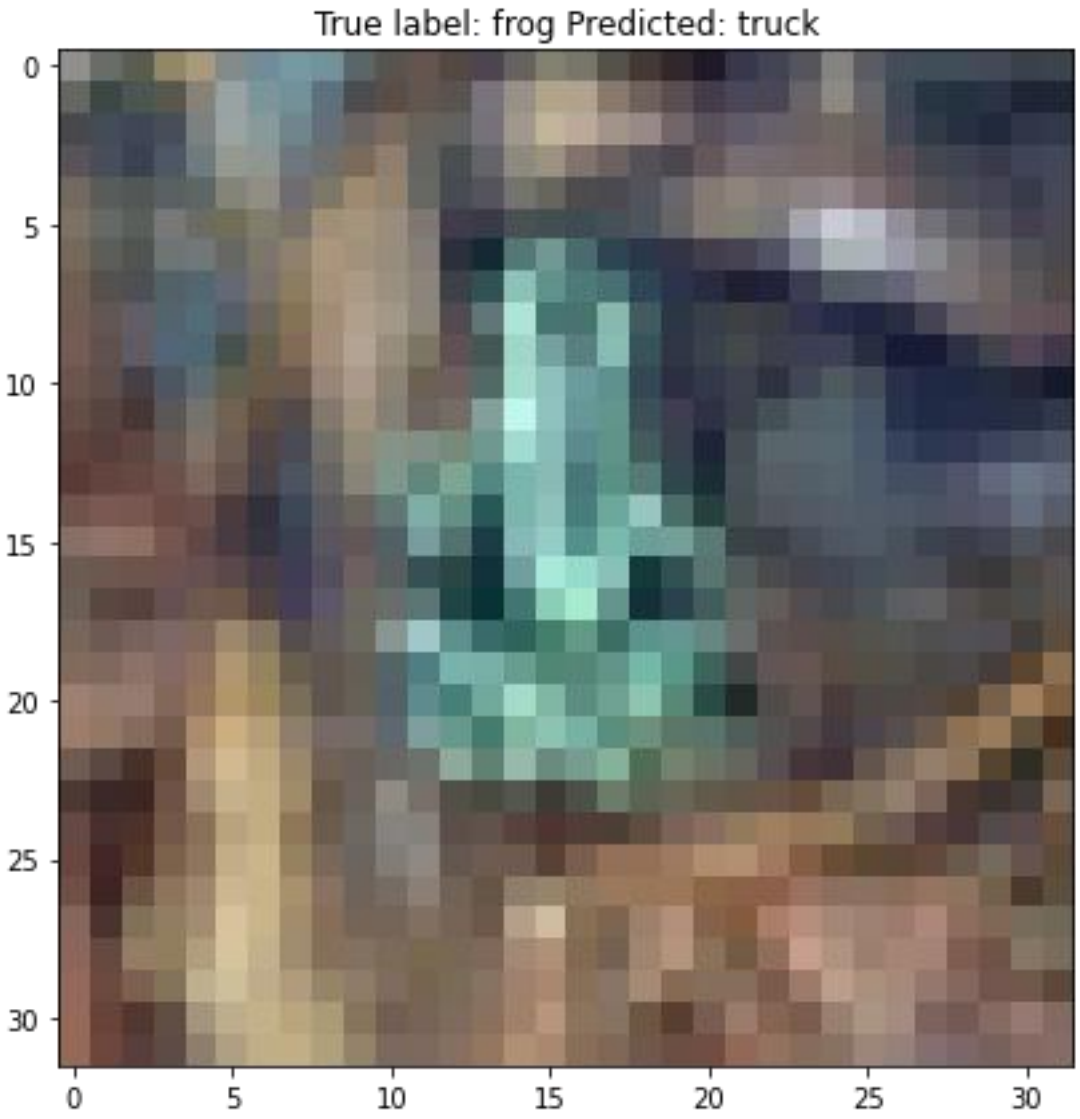
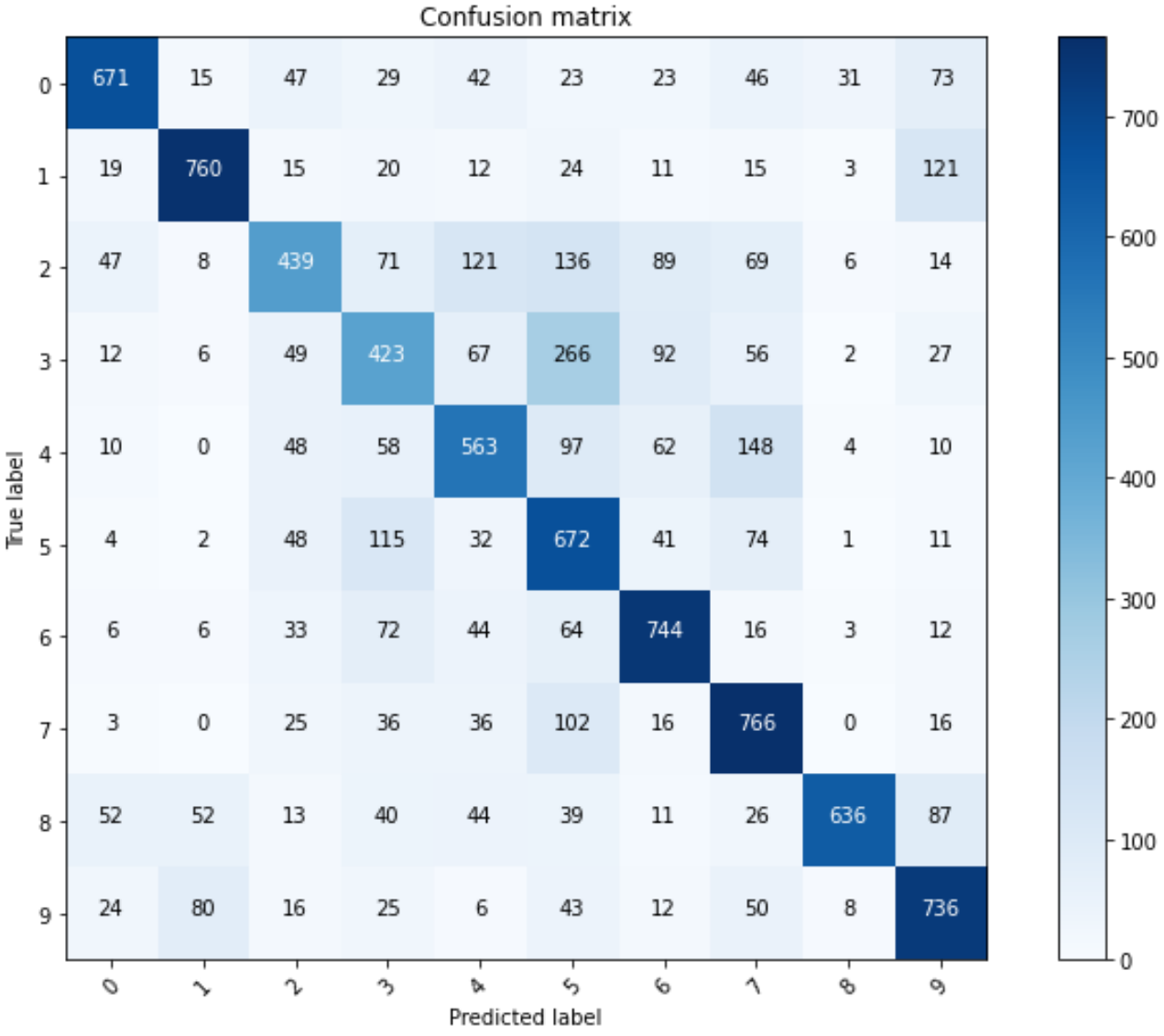


Figura 1.1 Accuracy Nicknet1

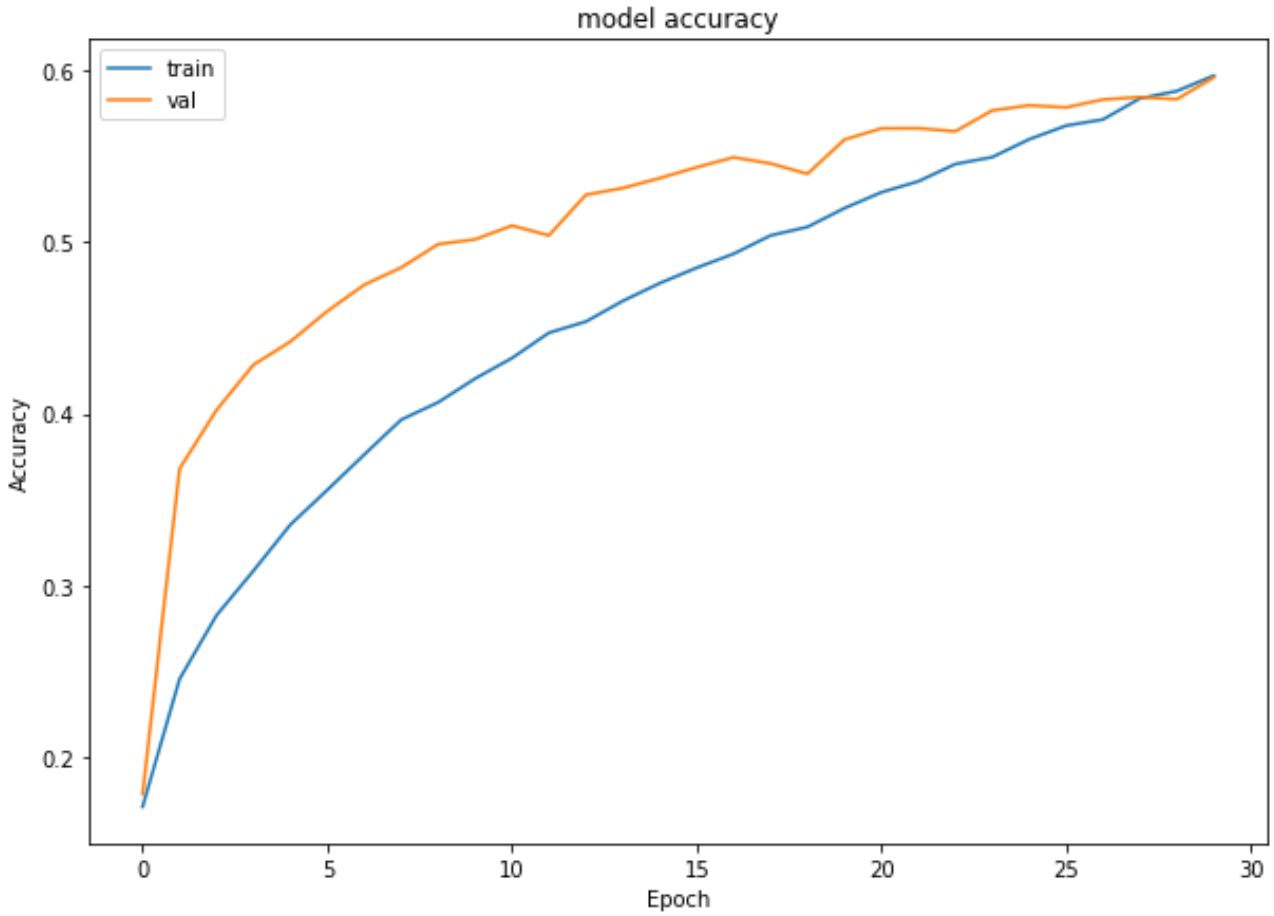


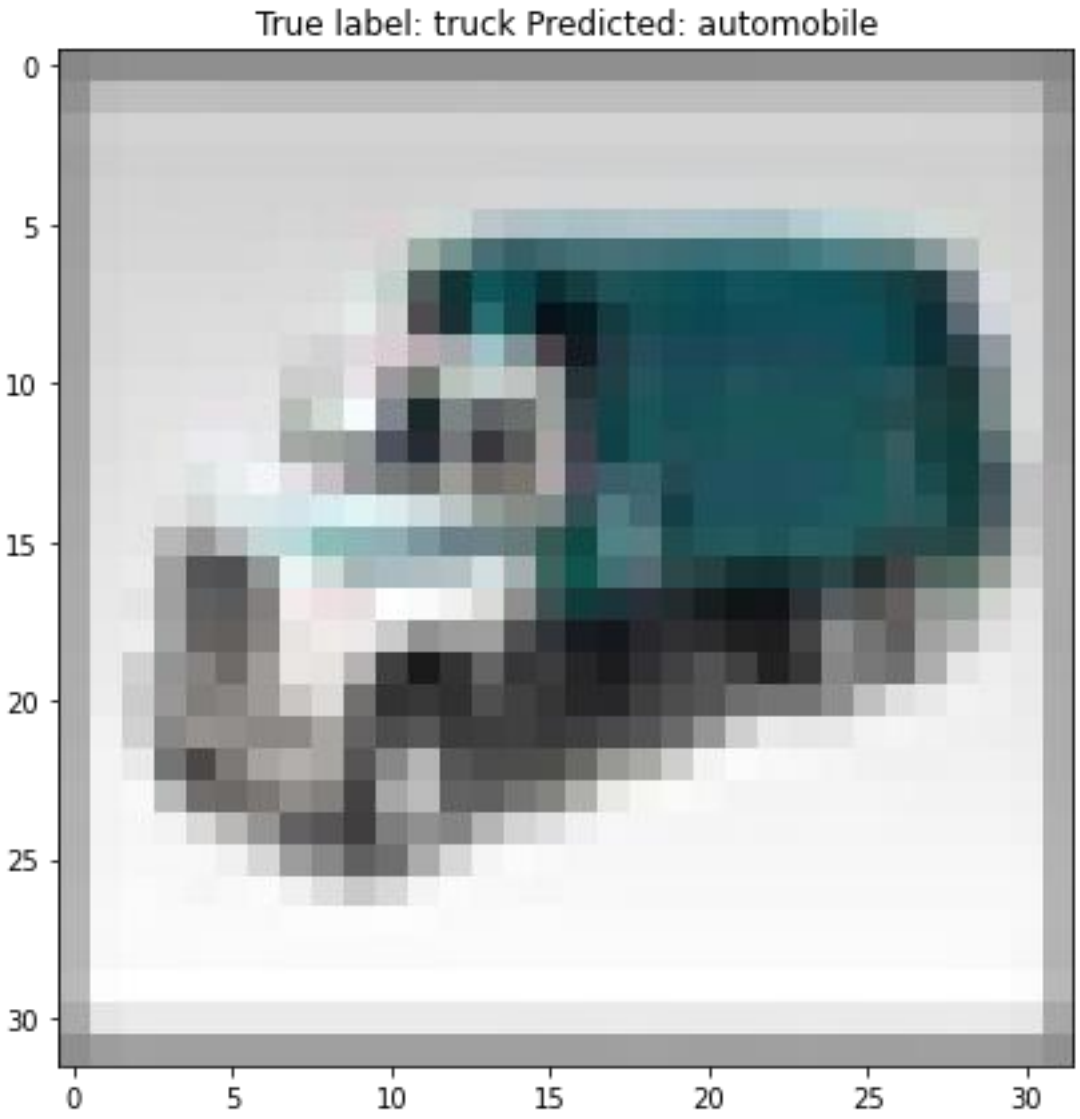
Example of misclassified picture in Nicknet1


Confusion matrix
True label
Predicted label
0 671 15 47 29 42 23 23 46 31 73
1 19 760 15 20 12 24 11 15 3 121
2 47 8 439 71 121 136 89 69 6 14
3 12 6 49 423 67 266 92 56 2 27
4 10 0 48 58 563 97 62 148 4 10
5 4 2 48 115 32 672 41 74 1 11
6 6 6 33 72 44 64 744 16 3 12
7 3 0 25 36 36 102 16 766 0 16
8 52 52 13 40 44 39 11 26 636 87
9 24 80 16 25 6 43 12 50 8 736
0 1 2 3 4 5 6 7 8 9
700 600 500 400 300 200 100 0

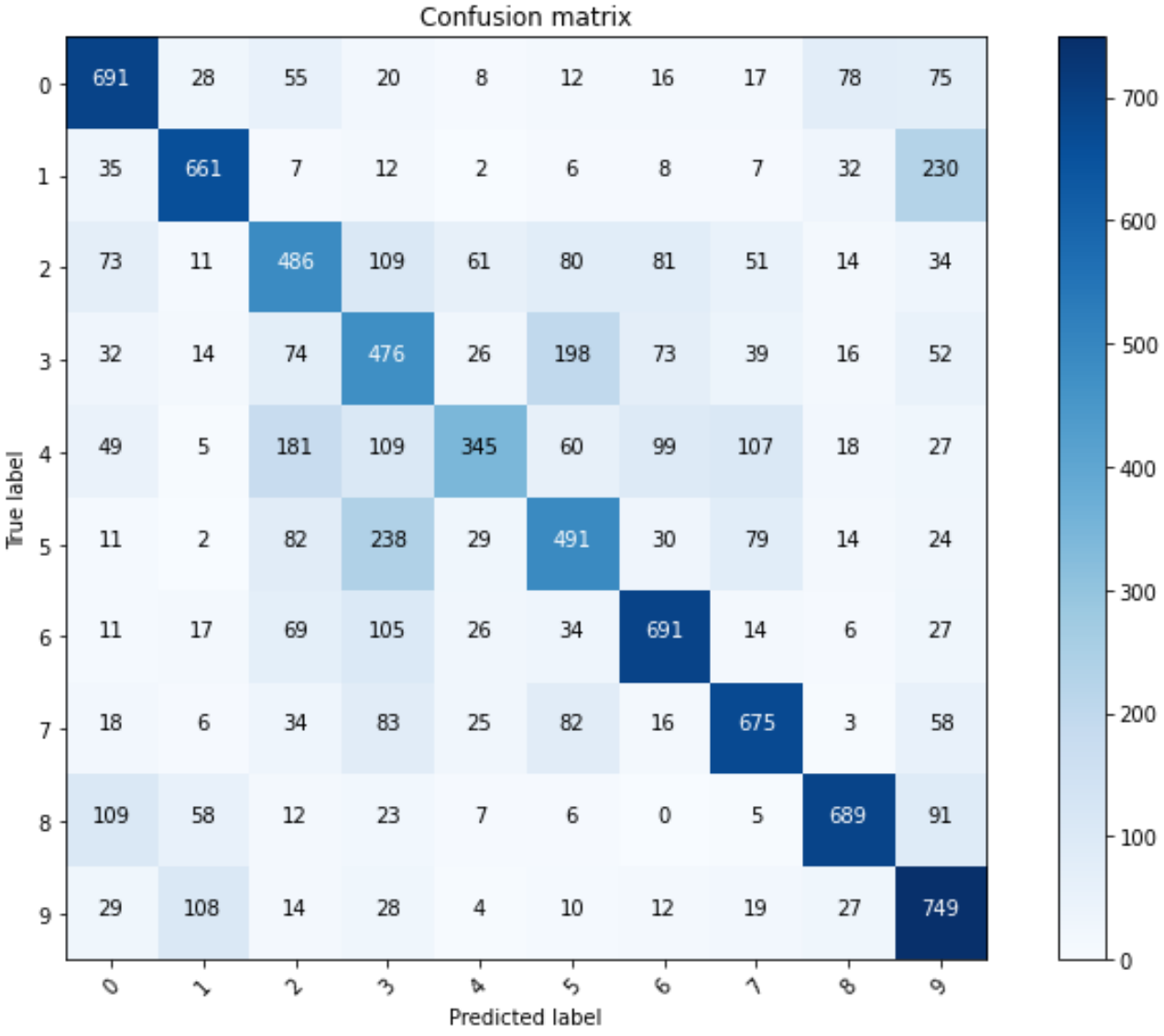
For the second experiment,we created a CNN,called Nicknet2,with 5 convolutional layer and 3 dense layers. For all the convolutional layers,the kernel size was (15,15) and as optimizer we picked SGD. As padding value we chose value 'same' and softmax as activation function.

After the training,with Nicknet 2,we obtained this values represented in these graphs below.



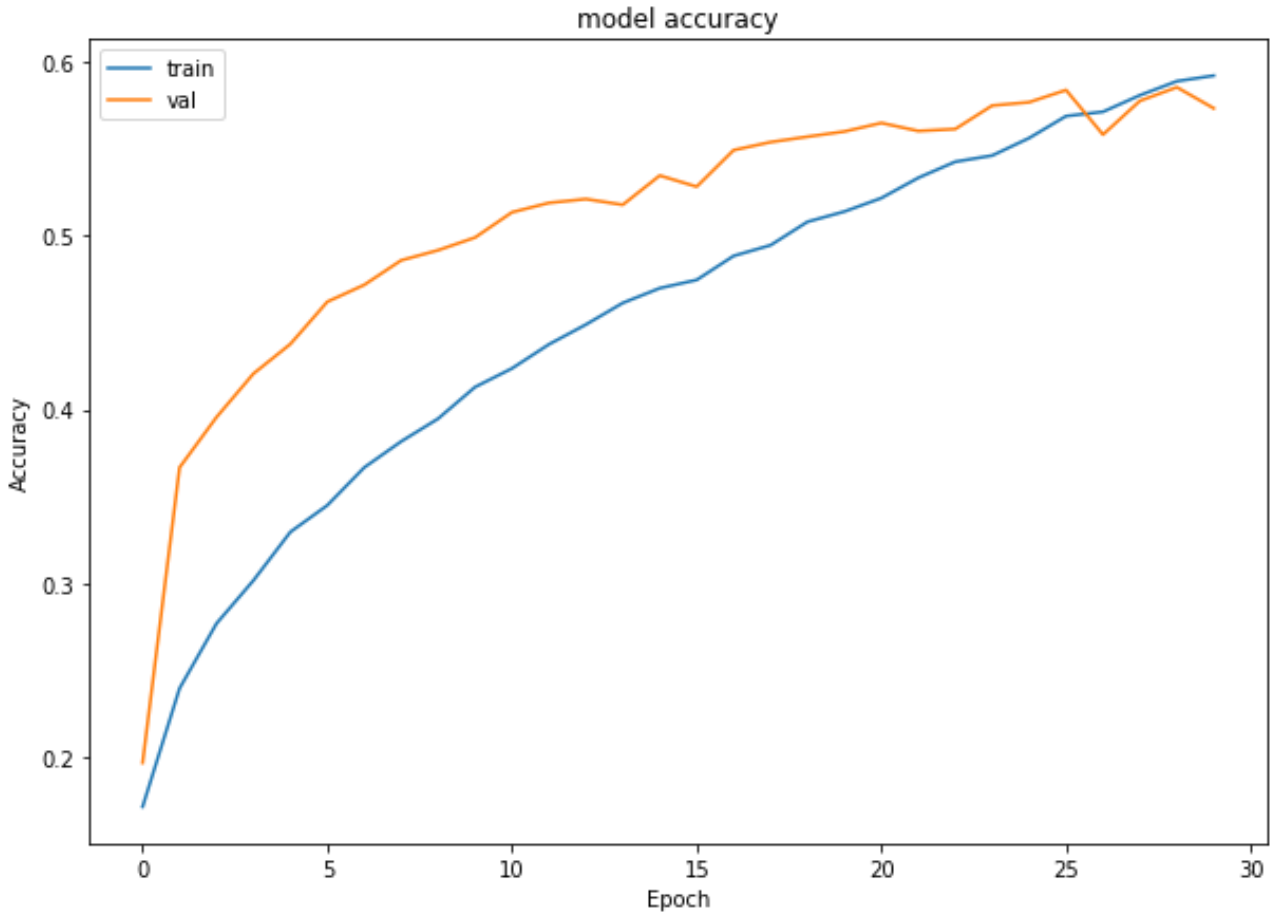


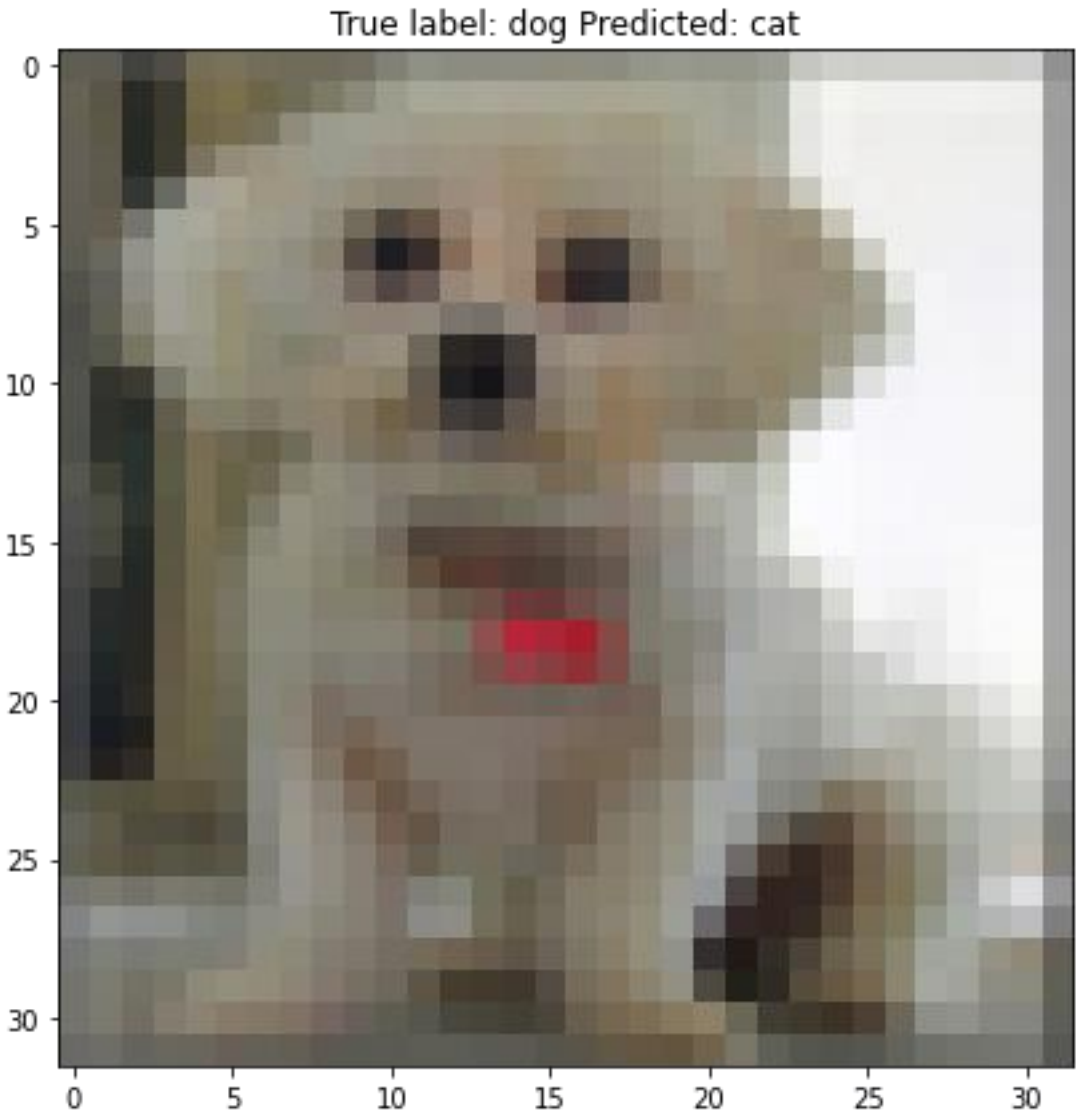
Example of misclassified picture in Nicknet2


Confusion matrix
True label
Predicted label
0 691 28 55 20 8 12 16 17 78 75
1 35 661 7 12 2 6 8 7 32 230
2 73 11 486 109 61 80 81 51 14 34
3 32 14 74 476 26 198 73 39 16 52
4 49 5 181 109 345 60 99 107 18 27
5 11 2 82 238 29 491 30 79 14 24
6 11 17 69 105 26 34 691 14 6 27
7 18 6 34 83 25 82 16 675 3 58
8 109 58 12 23 7 6 0 5 689 91
9 29 108 14 28 4 10 12 19 27 749
0 1 2 3 4 5 6 7 8 9
700 600 500 400 300 200 100 0

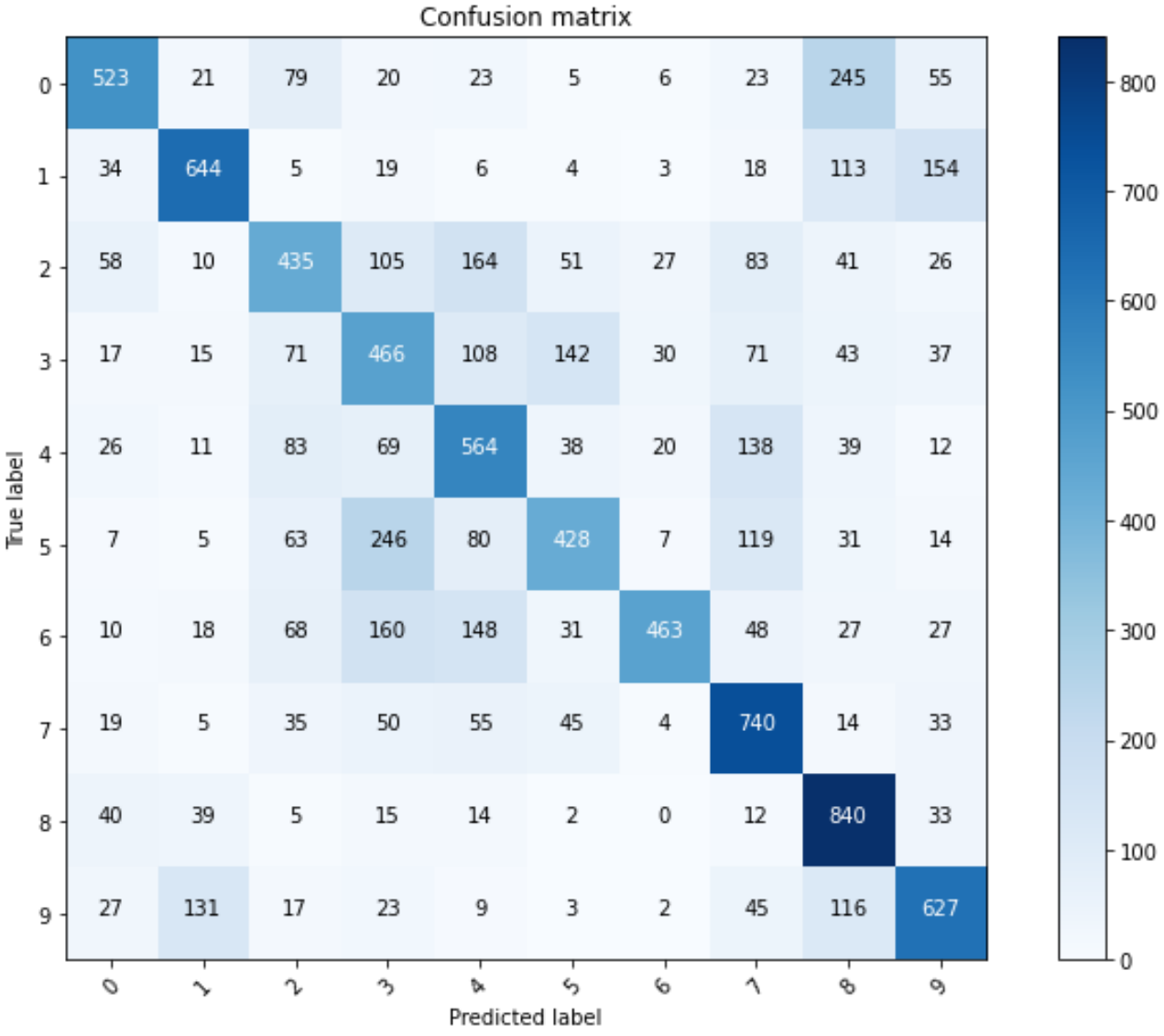
For the third experiment,we created a CNN,called Nicknet3,with 5 convolutional layer and 3 dense layers. For all the convolutional layers,the kernel size was (10,10) and as optimizer we picked SGD. As padding value we chose value 'same' and softmax as activation function.

After the training,with Nicknet 3,we obtained this values represented in these graphs below.





Example of misclassified picture in Nicknet3


Confusion matrix
True label
Predicted label
0 1 2 3 4 5 6 7 8 9
523 21 79 20 23 5 6 23 245 55
34 644 5 19 6 4 3 18 113 154
58 10 435 105 164 51 27 83 41 26
17 15 71 466 108 142 30 71 43 37
26 11 83 69 564 38 20 138 39 12
7 5 63 246 80 428 7 119 31 14
10 18 68 160 148 31 463 48 27 27
19 5 35 50 55 45 4 740 14 33
40 39 5 15 14 2 0 12 840 33
27 131 17 23 9 3 2 45 116 627
800 700 600 500 400 300 200 100 0

Chapter 4 : Results and conclusions

For the finale results,we showed with library matplotlyb the metrics of every experiments in the previous graphs. In the first experiment,we have the highest value in accurancy,which is 0.9, and in the confusion matrix. While in the other two experiments,the accuracy value is 0.6 and is clear that in these cases the misclassification level is higher. The main difference between the first experiment and the other two is the choice of the optimizer:in Nicknet 1 we picked Adam,while in Nicknet 2 and Nicknet3 we picked SGD.SGD is a variant of gradient descent and instead of performing computations on the whole dataset,only computes on a small subset or random selection of data examples.While Adam is an algorithm fro gradient-based optimization of stochastic objective functions and computes indvidual adaptive learning rates for different parameters. Regarding the confusion matrices,there is a common feature:the classes 'bird' and 'cat' have a low level of accurancy,it's below 0.5,indeed using the random functions,the 90% of misclassified pictures were related to these two classes,while for labels 'ship' and 'truck' the accuracy is high in every matrix.In the end we can say that the misclassification regarding classes 'bird' and 'cat' is a problem closely related to the dataset used in

these experiments. The pictures of these two classes must have morphological similitaries that prevent the three model networks from classifying them properly. So, the thing that most affects misclassification is the dataset we choose for our classification problem.