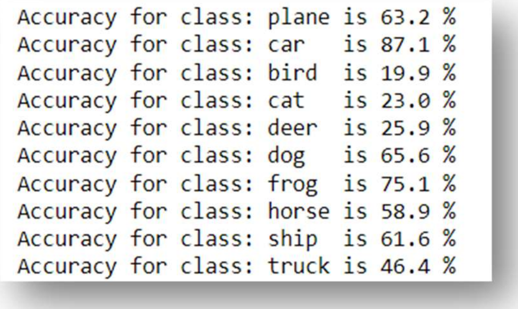
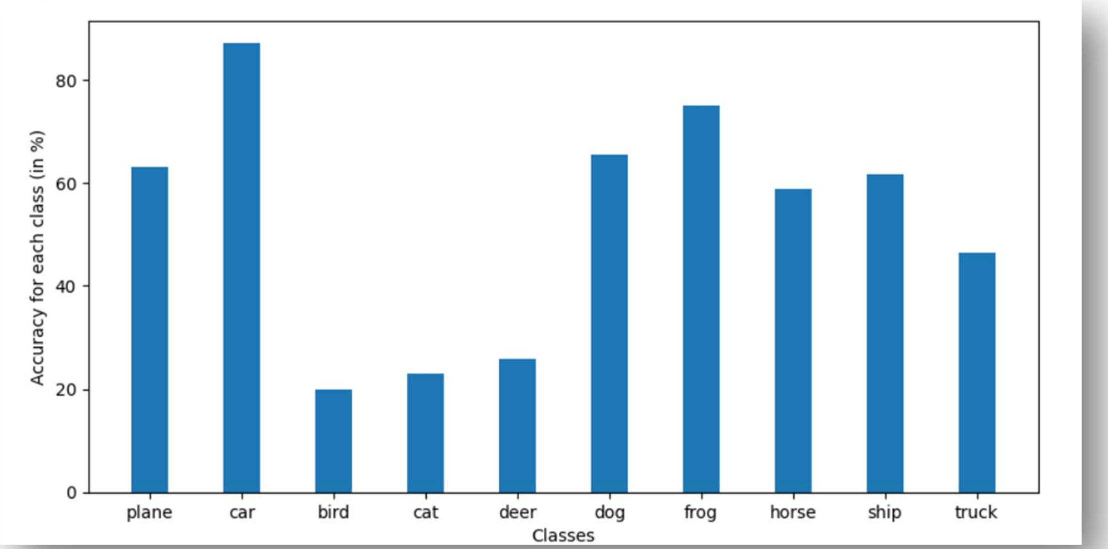
Section 1

1. Performances of our classification:

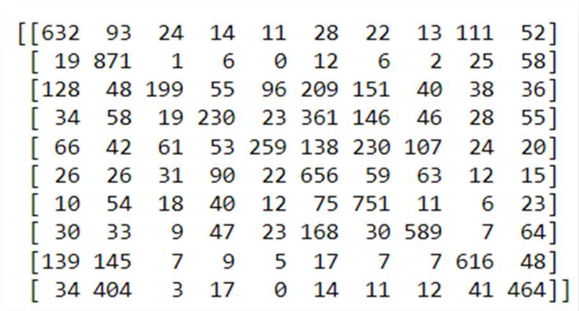
Accuracy level for each class:



```
Accuracy for class: plane is 63.2 %
Accuracy for class: car   is 87.1 %
Accuracy for class: bird  is 19.9 %
Accuracy for class: cat   is 23.0 %
Accuracy for class: deer  is 25.9 %
Accuracy for class: dog   is 65.6 %
Accuracy for class: frog  is 75.1 %
Accuracy for class: horse is 58.9 %
Accuracy for class: ship  is 61.6 %
Accuracy for class: truck is 46.4 %
```



2. Calculating the confusion matrix:



```
[[632  93  24  14  11  28  22  13 111  52]
 [ 19 871   1   6   0  12   6   2  25  58]
 [128  48 199  55  96 209 151  40  38  36]
 [ 34  58  19 230  23 361 146  46  28  55]
 [ 66  42  61  53 259 138 230 107  24  20]
 [ 26  26  31  90  22 656  59  63  12  15]
 [ 10  54  18  40  12  75 751  11   6  23]
 [ 30  33   9  47  23 168  30 589   7  64]
 [139 145   7   9   5  17   7   7 616  48]
 [ 34 404   3  17   0  14  11  12  41 464]]
```

3. What is Autograd?

Autograd is a Pytorch feature that is used to automatically train neural networks. Generally speaking, Autograd can be considered as an engine for computing vector-Jacobian product.

It is used for its ability to differentiate tensors and therefore perform gradient-based optimization for training neural networks and other machine learning models.

Section 2

1. What is RNN?

RNN (Recurrent Neural Network) is a type of neural network designed to process sequence of data instead of fixed-sized data like traditional networks. An RNN is able for example to take as an input a sequence of words and output a prediction.

2. Why do we use RNN when we are working with text?

The RNN are used for working with text as they allow to take a sequence a sequence as an input. They also can take as input sequency of variables size which is often the case when working with text. And finally, RNN is useful with text as they allow to model sequences, capture context, temporalities and dependencies which can be our goal when working with neural network.

3. In your opinion, how well does the text generation work?

In my opinion, text generation using RNN for text general has made a lot of progress over the last few years. Nevertheless, this kind of neural networks can be very tricky to train and to get good results with. These networks seem to be particularly sensitive to vanishing gradient during training or overfitting.

4. Name three other domains where RNNs are suitable model types for regression/classification.

RNN could also be suitable for other applications like:

- Speech Recognition
- Generating Image Descriptions
- Image recognition