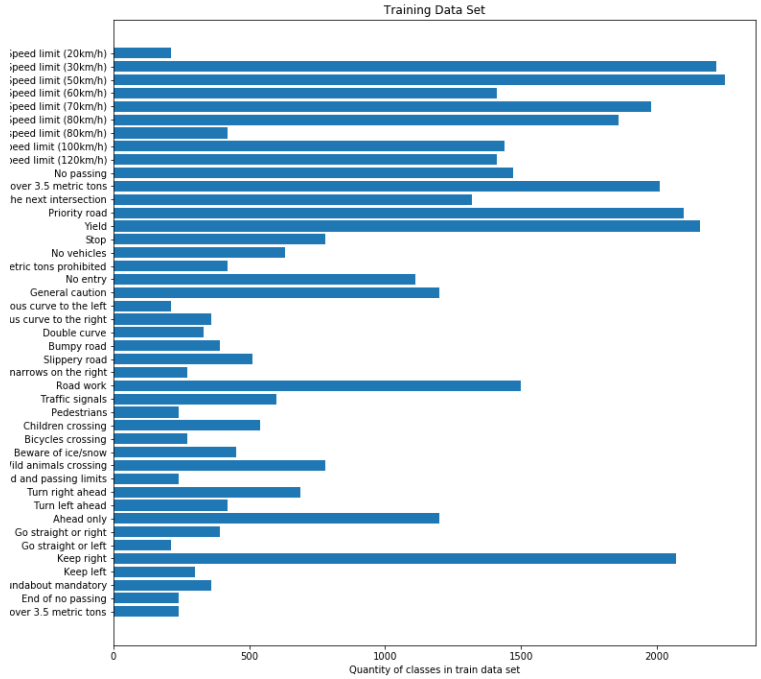
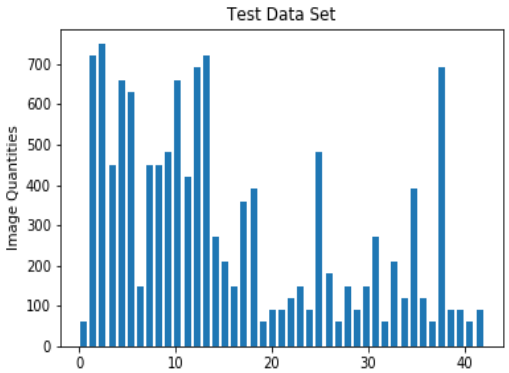
Data Set Summary and Exploration:

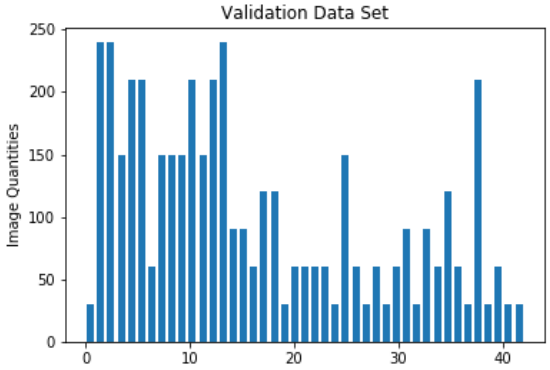
I used numpy to calculate summary statistics of the traffic signs data set:

1. Number of training examples: 34799
2. Number of Validation examples: 4410
3. Number of test examples: 12630
4. Shape of an image: (32,32,3)
5. Number of classes: 43

The Data sets are shown below:







DESIGN and Test Model Architecture

I decided to generate fake data. The Reason is that to prevent to be biased in certain classes, which they have far more images than the others. I took as minimum value 1750 in each class. There is no reason to take this number. But it is certain that takes much more time than training a DNN despite of using GPU.

Then I put train and validation data together and from them generated fake data. By generating fake data following process performed:

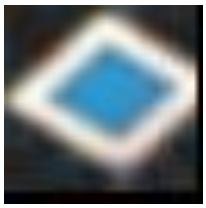
0) Original Image:

1) Image filtering: Gaussian Kernel --> 3x3 Kernel with depth 20

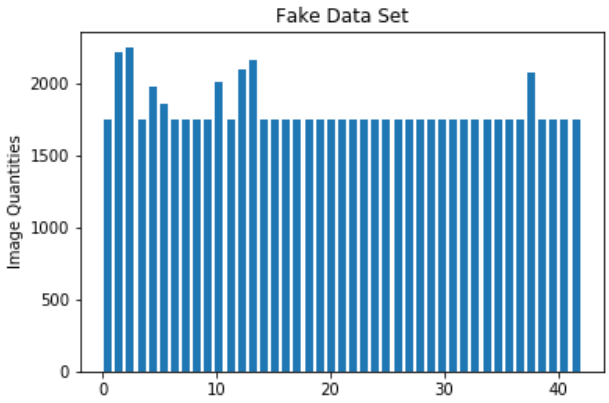
2) Random Rotating: Angle between -15° and 15°

3) Random Scaling: Scale between -2 and 2

4) Affline Transformation



I normalized then all images to values between 0 and 1 and additionally reduced all to grayscaled images. So the data sets are ready to feed into model. Before that the new training data set is shown below.

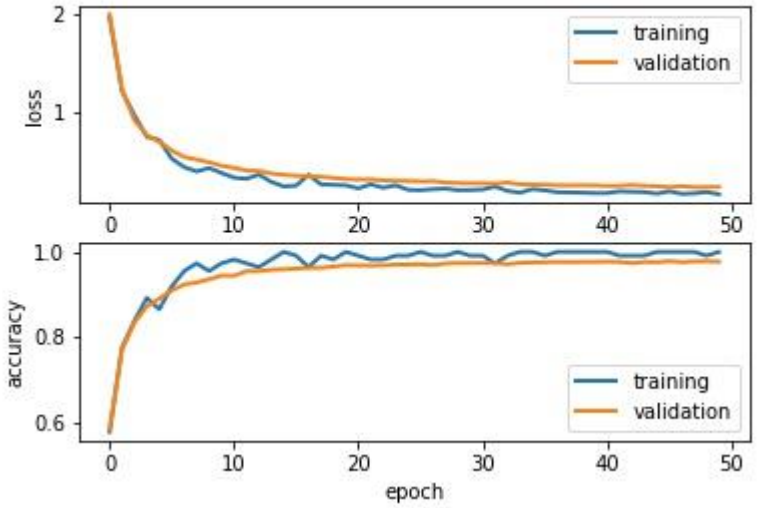


Final Model:

Layer	Description
Input	32x32x1
Convolution 5x5	2x2 stride, valid padding, outputs 28x28x20
ReLu	
Max Pooling	2x2 stride, outputs 14x14x20
Convolution 5x5	2x2 stride, valid padding, outputs 10x10x50
ReLu	
Max Pooling	2x2 stride, outputs 5x5x50
Flatten	Inputs 5x5x50, outputs 1250
Fully connected	Inputs 1250, outputs 250
ReLu	
Dropout	50%
Fully connected	Inputs 250, outputs 100
ReLu	
Dropout	50%
Fully connected	Inputs 100, outputs 43

- I used a similar architecture (LeNet) offered as a startpoint.
- The first problem was actually lack of data for some classes and as a result fewer knowledge at the end point. After I had generated fake data, actually I got good results. But I wanted to play with algorithm to learn about the effect of depth of the output volume of each layer and it is a hyperparameter too.
- Dropouts are added just the fully connected layers. The purpose was again to prevent overfitting.

- As I learned in lessons and recommended the Adam Optimizer was chosen. The learning rate is 0.001 with 50 Epochs and batch size 256. Discovering right Epoch, learning rate and batch size depends on trial and error.
- After settled these Parameter was additional L2 Regularization with lambda 5e-4 performed for weights of fully connected layers to prevent overfitting. It was hard to find a right value. One can be too low and do nothing actually, another one can be too high and cause underfit. Below are the graphs of loss and accuracy of my final Model:
    - Train Accuracy: 1.00
    - Validation Accuracy: 0.977
    - Test Accuracy: 0.955

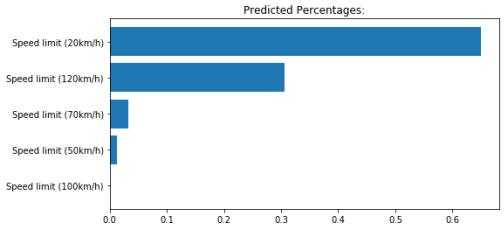


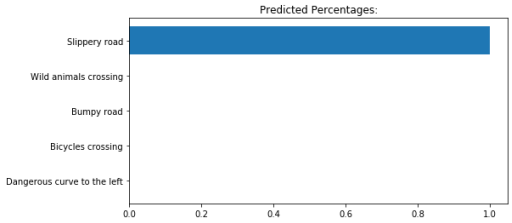
TEST a Model on New Images

Below is the 7 random selected images and their representation. My model has correct predicted 5 image out of 7. Accuracy is about 71 %. As you can see in charts in next page the 2 Speed Sign has wrong predicted, at least my model is confident that both is speed sign.

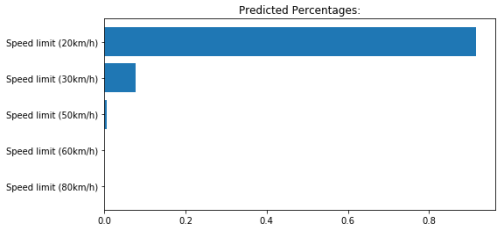
As improvement at this point I suggest myself, colored images as training data can be used. Second improvement can be deeper network with more fake data. Because my training data is still unbalanced.


70

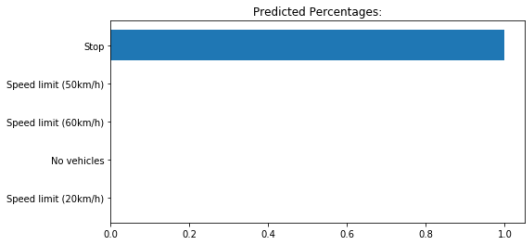

Predicted Percentages:
Speed limit (20km/h)
Speed limit (120km/h)
Speed limit (70km/h)
Speed limit (50km/h)
Speed limit (100km/h)
0.0
0.1
0.2
0.3
0.4
0.5
0.6


Predicted Percentages:
Slippery road
Wild animals crossing
Bumpy road
Bicycles crossing
Dangerous curve to the left
0.0
0.2
0.4
0.6
0.8
1.0

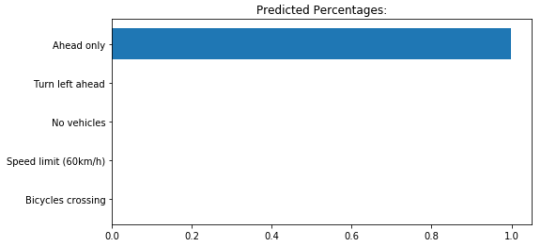

30

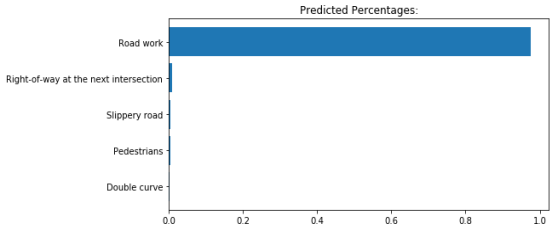

Predicted Percentages:
Speed limit (20km/h)
Speed limit (30km/h)
Speed limit (50km/h)
Speed limit (60km/h)
Speed limit (80km/h)
0.0
0.2
0.4
0.6
0.8


STOP

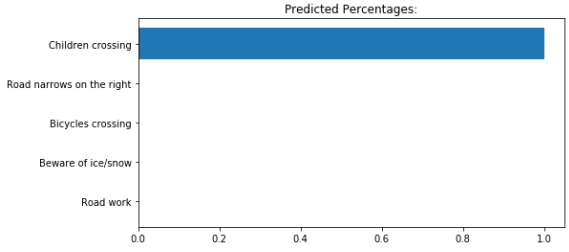

Predicted Percentages:
Stop
Speed limit (50km/h)
Speed limit (60km/h)
No vehicles
Speed limit (20km/h)
0.0
0.2
0.4
0.6
0.8
1.0


alamy stock photo


Predicted Percentages:
Ahead only
Turn left ahead
No vehicles
Speed limit (60km/h)
Bicycles crossing
0.0
0.2
0.4
0.6
0.8
1.0


Predicted Percentages:
Road work
Right-of-way at the next intersection
Slippery road
Pedestrians
Double curve
0.0
0.2
0.4
0.6
0.8
1.0


www.alamy.com - J2HW8H


Predicted Percentages:
Children crossing
Road narrows on the right
Bicycles crossing
Beware of ice/snow
Road work
0.0
0.2
0.4
0.6
0.8
1.0