KUSHAGRA PUNDEER

Traffic Sign Recognition

Writeup - Build a Traffic Sign Recognition Project

The goals / steps of this project are the following:

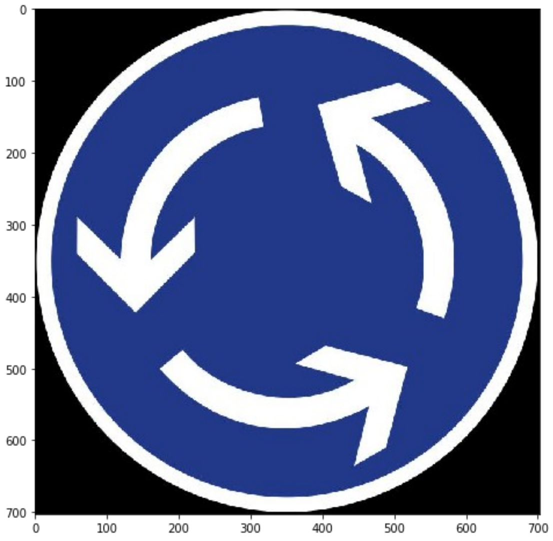
1. Load the data set (see below for links to the project data set)
2. Explore, summarize and visualize the data set
3. Design, train and test a model architecture
4. Use the model to make predictions on new images
5. Analyze the softmax probabilities of the new images
6. Summarize the results with a written report

(Image References)

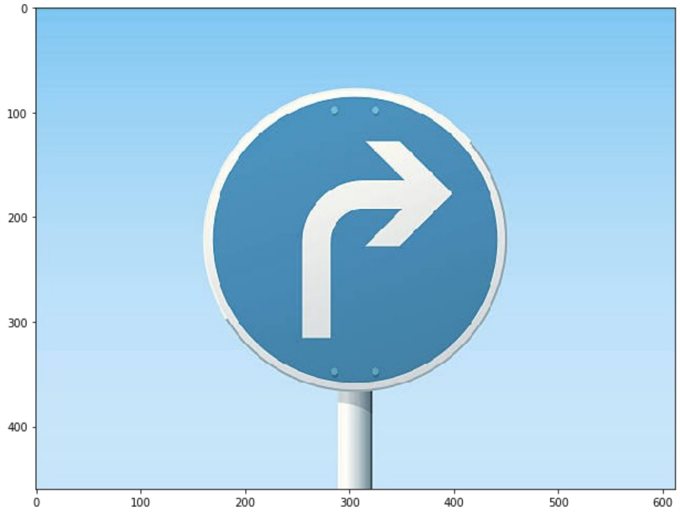
[image1]: ***Google image search – German traffic signs***

[image2]: Google image search – German traffic sign

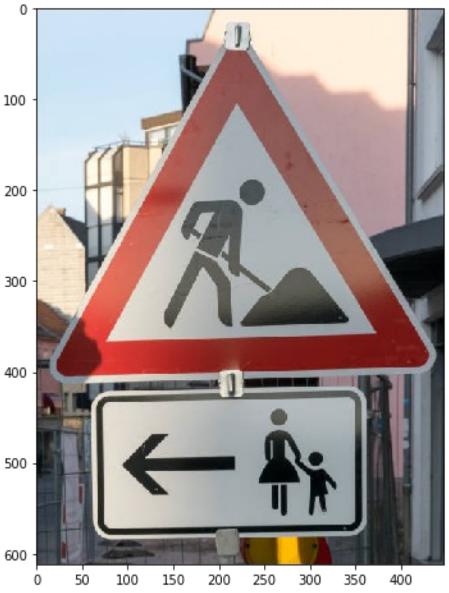
[image3]: *Google image search - German traffic signs*



[image4]: *Google image search - German traffic signs*



[image5]: Google Image search - *German traffic signs*



Writeup

1. I began by downloading the dataset into the project folder. The pickled loaded data is then used to form training, testing and validation data.
2. Next I calculated the no. of training examples, no. of testing examples, Image data shape and no. of classes. This gives basic summary of the data set.
3. 20 images were randomly selected from the dataset and plotted using matlabplot. To visualize the data set I used histograms to show distribution over training, testing and validation data.
4. Now the data is preprocessed. Image is converted to grayscale and normalized. Grayscale conversions makes the model faster over larger datasets.

Data Set Summary & Exploration

1. Provide a basic summary of the data set. In the code, the analysis should be done using python, numpy and/or pandas methods rather than hardcoding results manually.

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

The size of training set is ?

```
(34799, 32, 32, 3)
```

The size of the validation set is ?

```
(4410, 32, 32, 3)
```

The size of test set is ?

```
(12630, 32, 32, 3)
```

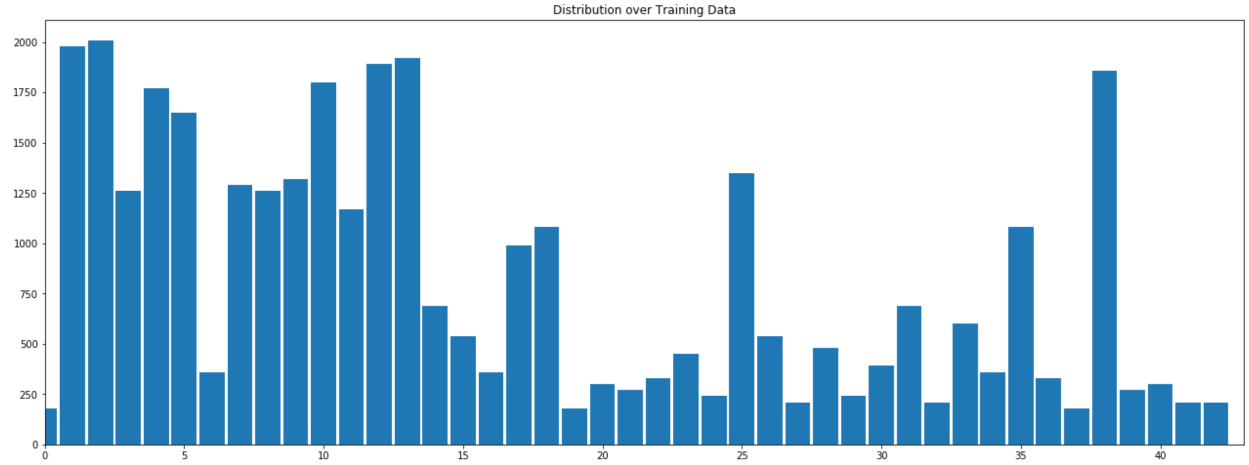
The shape of a traffic sign image is ?

```
(32, 32, 3)
```

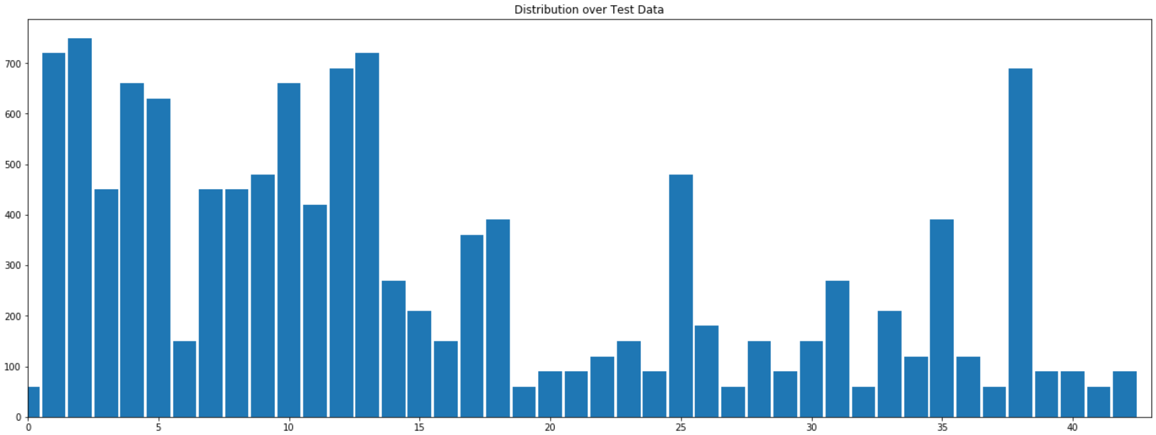
The number of unique classes/labels in the data set is ?
43

2. Include an exploratory visualization of the dataset.

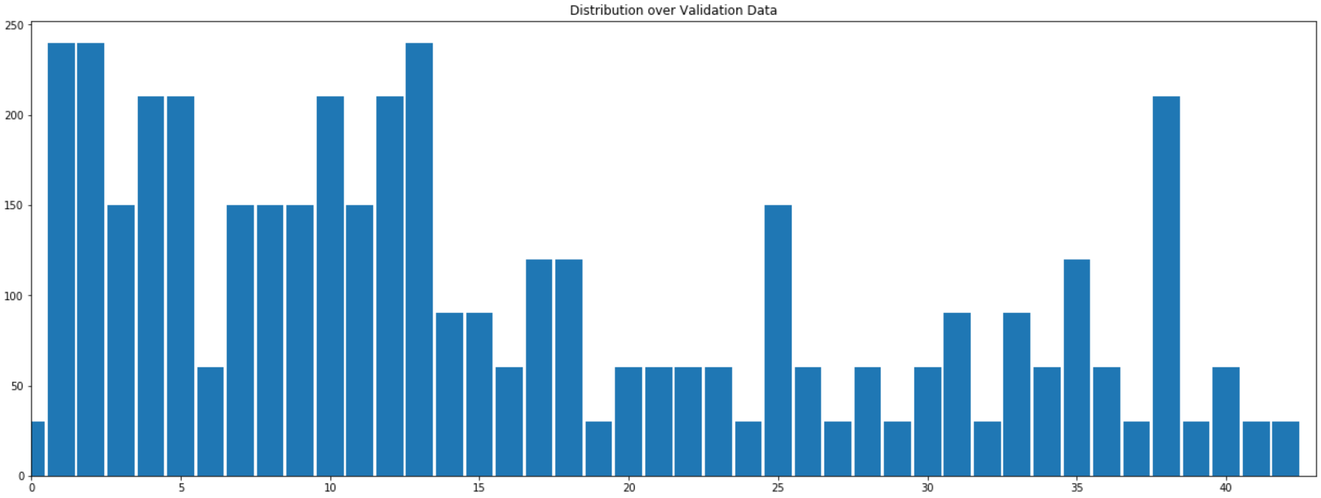
A. Distribution over Training Data



B. Distribution over Testing Data



C. Distribution using Validation data



Design and Test a Model Architecture

1. Describe how you preprocessed the image data.

As a first step, I decided to convert the images to grayscale because it the model runs faster when trained with grayscale conversion.I had to train the model using floydhub as I had AWS issues which are now resolved by Udacity support which had limited gpu usage and hence I decided to go with grayscale conversion which would be faster. I will be using AWS going forward for next projects

The grayscale image is then normalized between probabilities 0.1 and 0.9

2. Describe what your final model architecture looks like including model type, layers, layer sizes, connectivity, etc
My final model consisted of the following layers:

Modified the LeNet architecture to take grayscale image and for this project.

Layer	Description
Input	32x32x1 Grayscale image
Convolutional	*Output = 10x10x16*
Fully Connected	*Input = 400. Output = 120.*
Fully Connected.	Input = 120. Output = 84 (Used Dropout)
Fully Connected.	Input = 84. Output = 43

3. Describe how you trained your model. The discussion can include the type of optimizer, the batch size, number of epochs and any hyperparameters such as learning rate.

To train the model, I used Adam optimizer, with learning rate of ~0.04

```
Hyperparameters were:
EPOCHS = 22 (Approximate epochs needed)
BATCH_SIZE = 128 (chosen based on system)
```

Final model results were:
validation set accuracy of 0.934
test set accuracy of 0.89

It sometimes gives better results, reducing epochs and fine-tuning the parameters further will improve accuracy

Test a Model on New Images

1. Five traffic signs were chosen from the google image search “German Traffic Signs”

The new images are shown in the document above along with the references.

Image 1: No entry sign looks old, Color of the ambulance has poor contrast to the white on the sign.
Image 5: Has both construction and child crossing. This is an interesting example as the image has two traffic signs. I recognizes the child crossing correctly. I reran a few times to check if it was random selection or if it chooses the Child crossing. The classifier picks Child crossing every time. The Construction sign appears to have some shadow and bright areas and hence not as uniform in contrast.
Image2: Is slightly blurred, with some small scuff marks showing wear.

These image features would have made it harder for them to qualify.

2. Discuss the model's predictions on these new traffic signs and compare the results to predicting on the test set. At a minimum, discuss what the predictions were, the accuracy on these new predictions, and compare the accuracy to the accuracy on the test set

Accuracy was 100% but it was over only 5 images. With larger test set accuracy can be established better .

Here are the results of the predictions

1. Prediction : `['No entry']   | Correct`
2. Prediction : `[General caution] | Correct`
3. Prediction : `['Roundabout mandatory'] | Correct`
4. Prediction : `['Turn right ahead'] | Correct`
5. Prediction : `['Children crossing'] | Correct for the bottom sign. Pretty good considering there are two traffic signs.`

The model was able to correctly guess 5 of the 5 traffic signs, which gives an accuracy of 100%.

3. Describe how certain the model is when predicting on each of the five new images by looking at the softmax probabilities for each prediction. Provide the top 5 softmax probabilities for each image along with the sign type of each probability.

The top five soft max probabilities were:

IMAGE 1

```
Probabilities: [  1.00000000e+00   8.63367644e-22   3.59253618e-22
2.14769100e-29   5.34920835e-30 ]
Classes      : [17 40  1 21 12]
Prediction   : ['No entry']
```

IMAGE 2

```
Probabilities : [  1.00000000e+00   5.79947515e-21   8.71544930e-28
5.80826671e-28  2.32200514e-28]
Classes       : [18 27 11 12  1]
Prediction    : ['General caution']
```

IMAGE 3

```
Probabilities: [  9.99540091e-01   3.88932211e-04   4.60137744e-05
2.34943418e-05   7.85439397e-07]
Classes      : [40 11  7 28 16]
Prediction   : ['Roundabout mandatory']
```

IMAGE 4

```
Probabilities: [  9.98104572e-01   1.42472470e-03   2.80004897e-04
9.24372216e-05   2.43815120e-05]
Classes      : [33  3  5  1  2]
Prediction   : ['Turn right ahead']
```

IMAGE 5

```
Probabilities: [ 0.25616032  0.12824868  0.09078123  0.07743439  0.0626541 ]
Classes      : [28 12 11 40 32]
Prediction   : ['Children crossing']
```

i