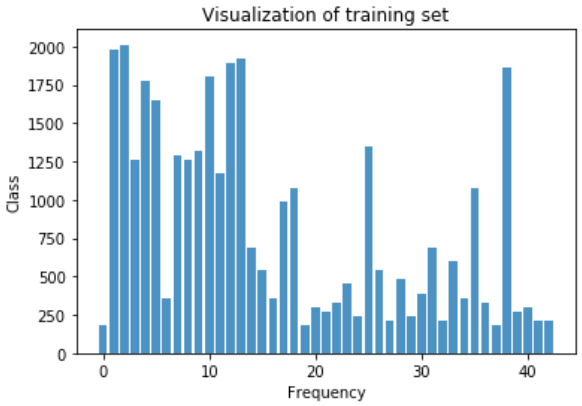
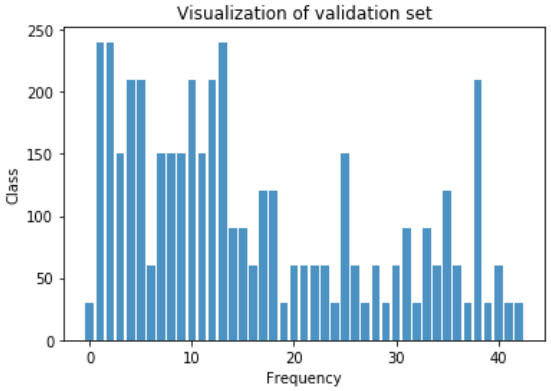
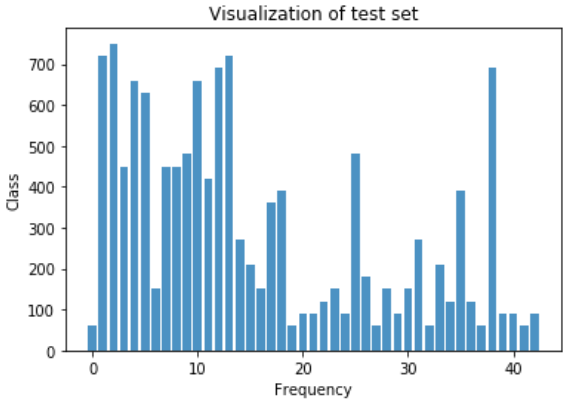
I address the rubric points as follows:

- 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 hard coding results manually.
    - I converted the datasets into numpy arrays and calculated the size as follows.
      Size of training set: 34799
      Size of testing set: 12630
      Size of validation set: 4410
      Shape of traffic sign image: 32*32*3
      Number of classes: 43
- Visualization of the datasets"
  The following bar graphs show the number of examples for each class in the three datasets.




- Preprocessing of images.
  I preprocessed the images in the following two steps
    1. Converted the pixel values to zero mean and equal variance by subtracting 128 and dividing by 128.

2. Converted into grayscale.

- Model Architecture
  Input: 32*32*1 grayscale image
  Convolution layer 1, 1*1 stride, valid padding, outputs: 28*28*6
  Relu activation
  Max pool layer, 2*2 stride, ksize=2*2, outputs 14*14*6
  Convolution layer 2: 1*1 stride, valid padding, outputs: 10*10*16
  Relu activation
  Max pool layer: 2*2 stride, ksize=2*2, outputs: 5*5*16
  Flatten layer: output: 400
  Fully connected layer: output 120
  Dropout, keep probability=0.5
  Relu activation
  Fully connected: output 84
  Relu activation
  Fully connected: output 43(the number of classes)

- Training the model.
  Optimizer used: Adam's optimizer
  Hyperparameters
  - Weights were initialized from a normal distribution with mean zero and standard deviation 0.1
  - Number of epoch: 50
  - Batch size: 50
  - Learning rate: 0.001

- Final results
  - Validation set accuracy: 93.9%
  - Test set accuracy: 91.99%
  - Train set accuracy: 99.06%

Discussion: I used the Lenet architecture, which was giving me an accuracy of about 85% with the number of epochs as 10 and batch size as 128. I changed these parameters to 50 and 50 respectively to get accuracy value above 93%.

- Testing the model on new images.
  I used the following five images from the web.
  
  My model identified 1/5 images correctly. Hence, the accuracy was 20%.