**The Different Layer Configurations**

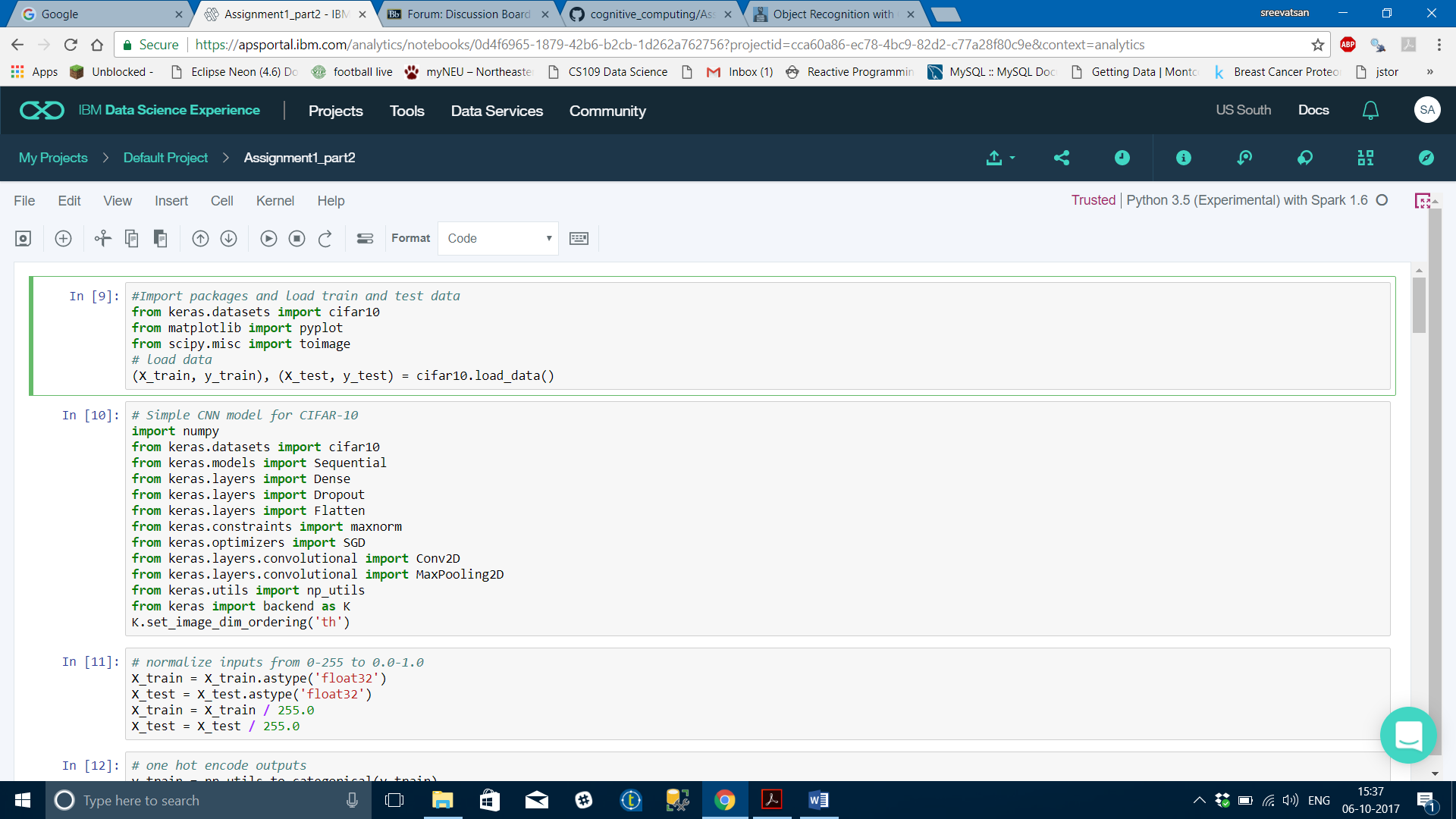
**Dataset:**

The [CIFAR-10 dataset](http://www.cs.toronto.edu/~kriz/cifar.html) consists of 60,000 photos divided into 10 classes (hence the name CIFAR-10). Classes include common objects such as airplanes, automobiles, birds, cats and so on. The dataset is split in a standard way, where 50,000 images are used for training a model and the remaining 10,000 for evaluating its performance. The photos are in color with red, green and blue components, but are small measuring 32 by 32 pixel squares.

**Importing Packages:**

As a first step all the required packages like keras, numpy, keras layers,optimizers ,convolutional layers are imported.

The cifar 10 Dataset is loaded and split into training and test data.

Then the loaded data is normalized in values between 0 to 1.

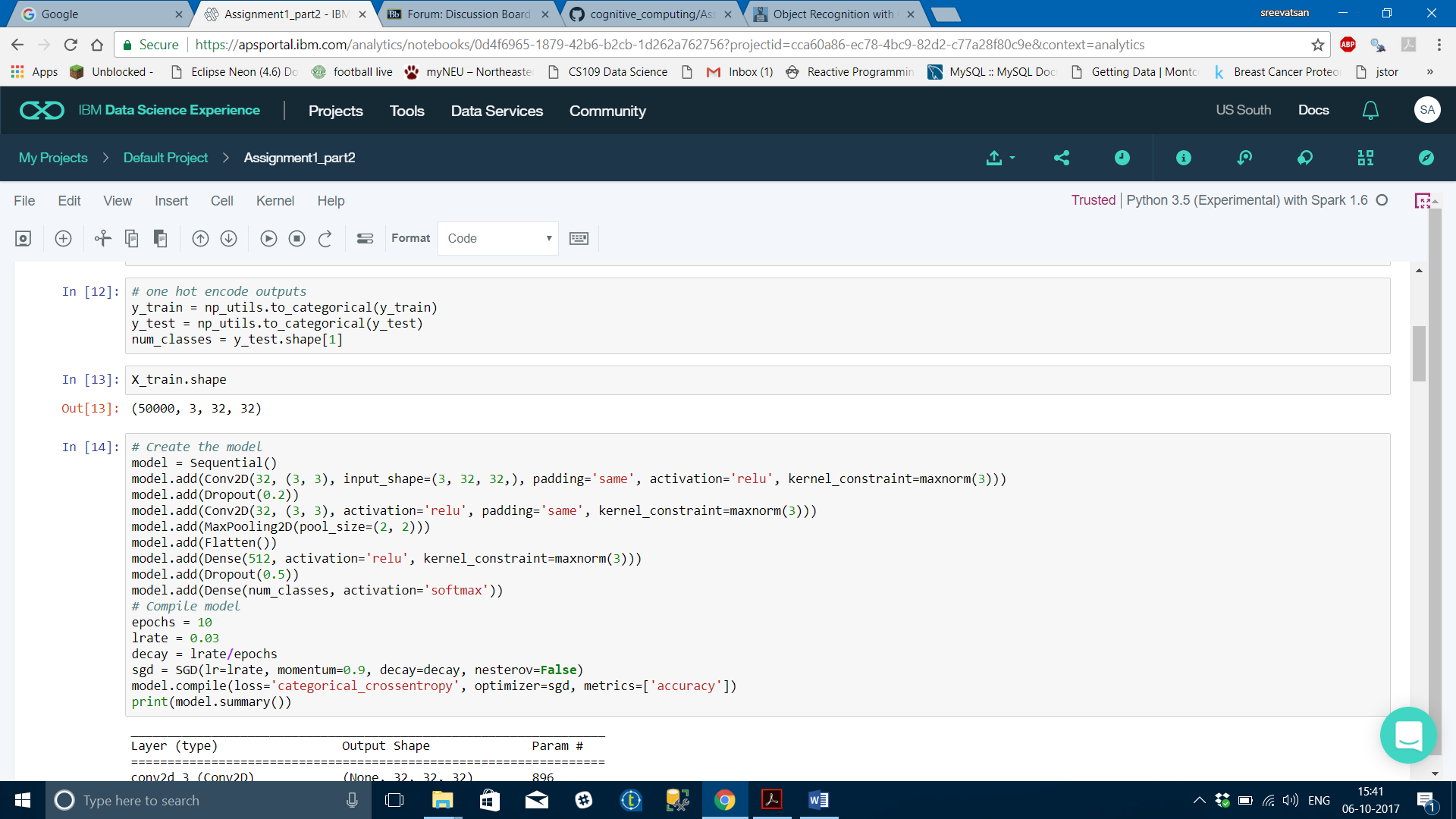
**Building Model:**

The one-hot encoding is done to change the output variable y into 10 categories as this is a classification problem.

The initial model structure is built with two convolutional layers followed by max pooling and a flattening out of the network to fully connected layers to make predictions.

Summary:

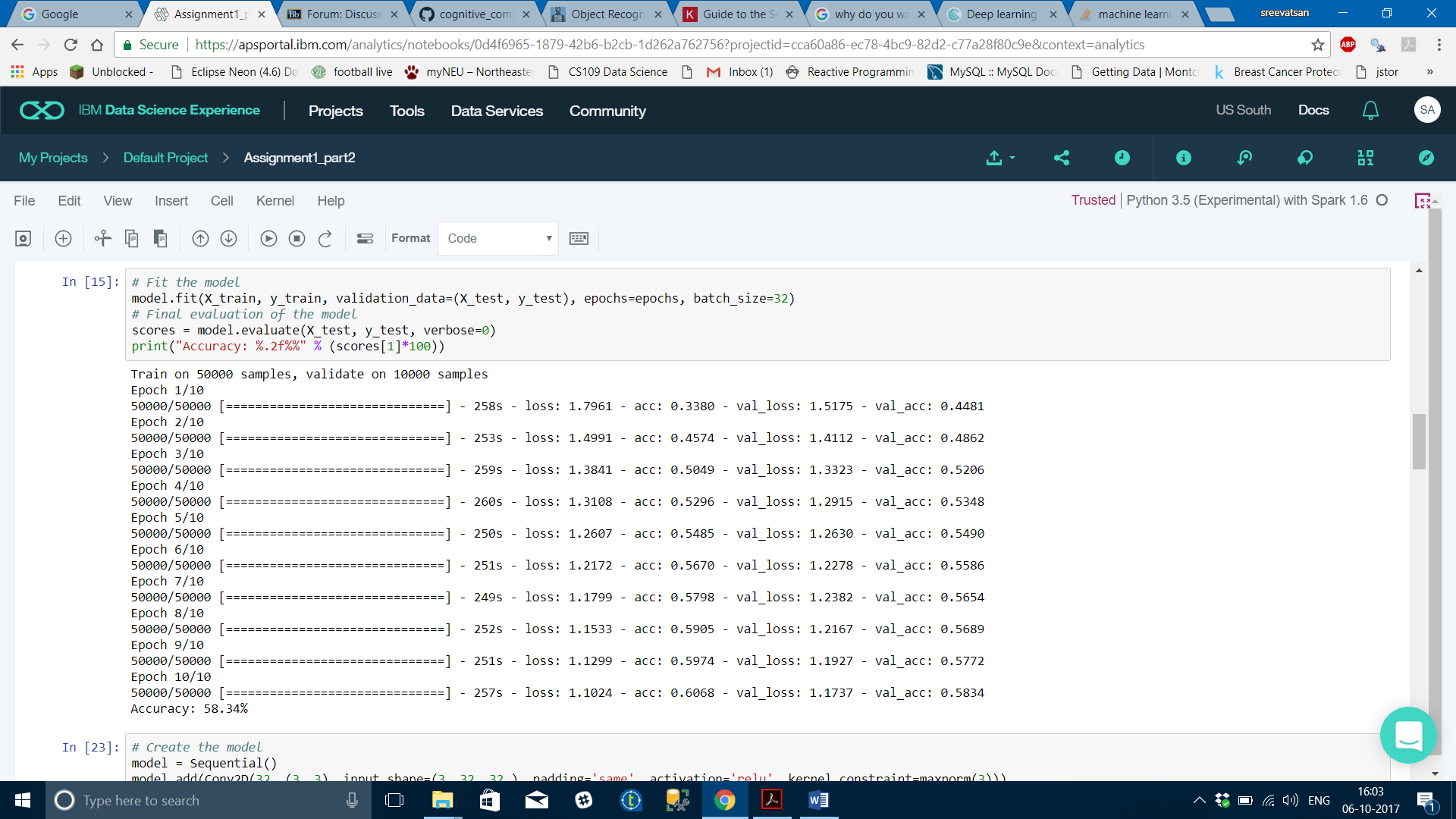
* Sequential model is used as we want to add the layers and execute it in a sequential way.
* Next the convolution layer with 32 maps and 3\*3 resolution is added . RELU activation function is used. Dropout is set to 20%
* Now another convolutional layer with same configuration as added .
* Now a max polling layer of 2\*2 dimension is added for down sampling.
* Next another layer is added to flatten the vectors.
* In the next step a fully connected layer with 512 units is added with RELU activation function with a dropout of 50%.
* Finally a dense layer is added with softmax function to get output of 10 classes.
* Epochs is set to 10 and learning rate as 0.01. The SGD optimizer is used to minimize the loss function and the model is compiled together.



**Model Validation:**

Now the built model is validated against the 10000 test data by fitting it with batch size of 32 and run for 10 epochs.

We were able to get an accuracy of 58% with simple two convolutional layers.



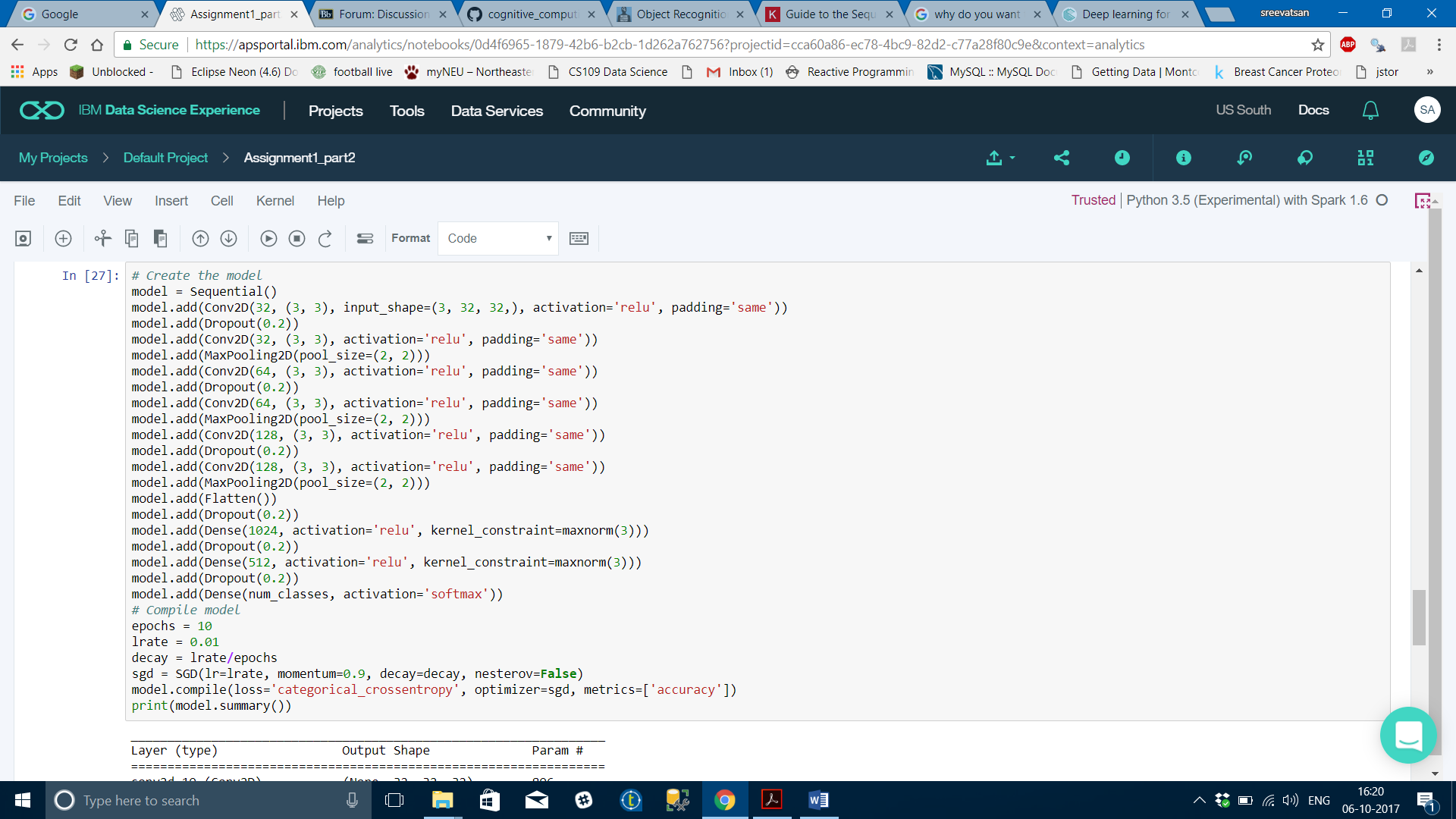
Now as a next step we are going to build a much deep layer network which will help in resulting a model with much greater accuracy.

**Rebuilding Deep layer model:**

The same pattern of Convolutional, Dropout, Convolutional and Max Pooling layers is followed here.

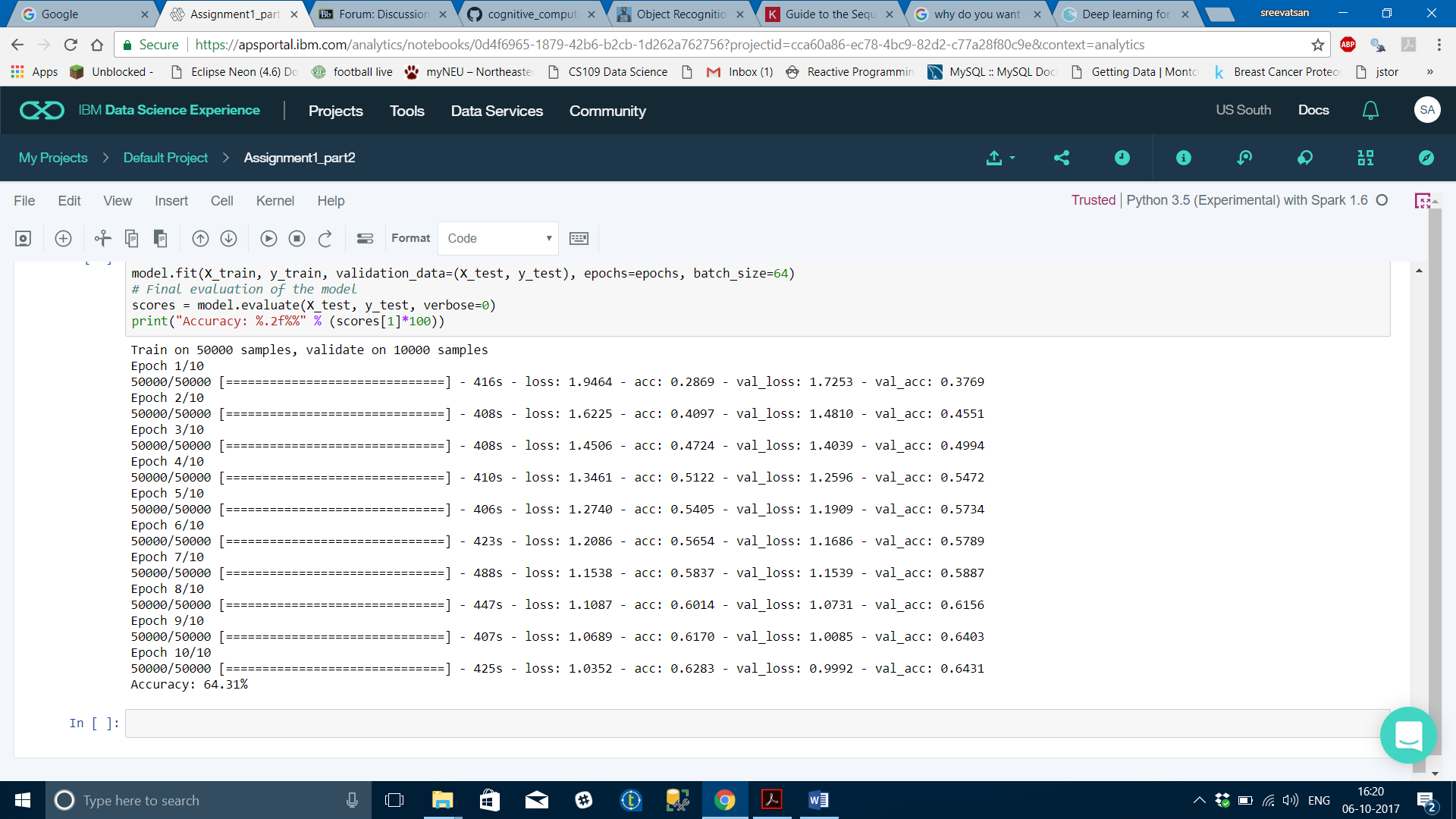
* Six convolutional layers are added twice in the increasing 2^n pattern such as 32,32,64,64,128,128 with dropout of 20%.
* Then two dense layer of 1024 units and 512 units are added respectively with ‘RELU’ activation function and same dropout of 20%.

By keeping the epochs, learning rate, optimizers and batch size as constant now the model is built and validated against the test data set.



**Result:**

 The estimate of classification accuracy for the final model is 64.31% which is better than our simpler model for 10 epochs.



**Conclusion:**

The accuracy can be further increased by

* Increasing the number of epochs
* Increasing the batch size
* Increasing the number of layers.
* Implementing Data augmentation Methods such as standardization and random shifts and horizontal image flips