**Deep Learning Categorical**

Originally, we created 2 hcc categories, one below 1.08 and one above.

Think of hcc as a cost/illness indicator. Higher number is higher cost and illness.

* Below 1.08 we called low risk
* Above 1.08 we called high risk

Non-LIPA- or ***non-*Low Utilization** Payment Adjustment. Our data was somewhat ambiguously labeled.

So maybe all counts and numbers were for higher utilization facilities and lower utilization facilities were excluded, hence the numbers for

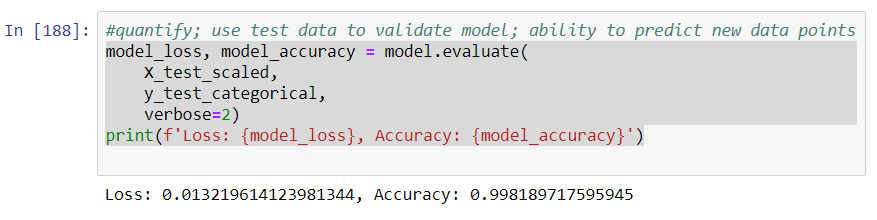
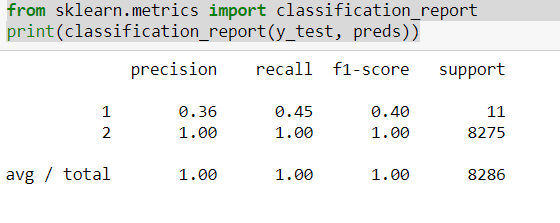
* 42k were above 1.08
* And just 50 were below 1.08

**Deep Learning Regression**

This might explain:

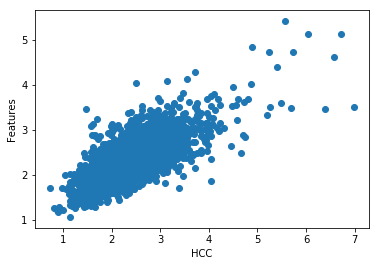
* An accuracy near 1
  + Epoch 68/100
  + - 2s - loss: 9.9153e-04 - acc: 0.9998
  + Epoch 69/100
  + - 2s - loss: 6.2780e-04 - acc: 0.9999
  + Epoch 70/100
  + - 2s - loss: 5.3688e-04 - acc: 0.9999
  + Epoch 71/100
  + - 2s - loss: 4.8836e-04 - acc: 1.0000
  + Epoch 72/100
  + - 2s - loss: 4.8769e-04 - acc: 1.0000
  + Epoch 73/100
  + - 2s - loss: 4.8750e-04 - acc: 1.0000
  + Epoch 74/100
  + - 2s - loss: 4.8738e-04 - acc: 1.0000
* And small losses

For both our training and test data sets

* Test
* 
* This classification report reports lower numbers for low risk group ‘1’. This may be related to the small number of data to test with on low hcc.  
  

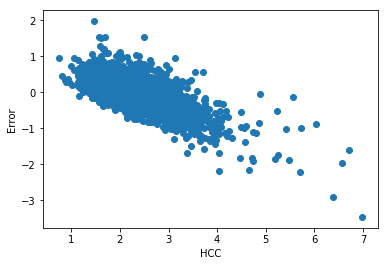
In addition to evaluating our data categorically with deep learning, the data was evaluated with regression and mean squared error(mse) via deep learning.

* Again our test train had a lower loss of around 0.08
* Since this was regression we did not have an accuracy number
* The prediction was plotted:
  + import matplotlib.pyplot as plt
  + %matplotlib inline
  + plt.scatter(y\_test,model.predict(X\_test\_scaled))
  + plt.xlabel('HCC')
  + plt.ylabel('Features')



Additionally. we checked the error in regards to HCC and found that the error increased with higher hcc scores:

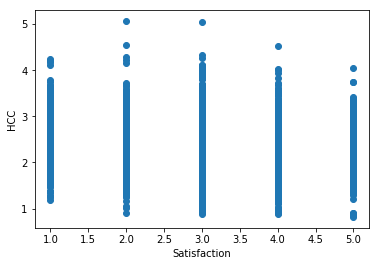
* err = model.predict(X\_test\_scaled)[:,0] - y\_test
* plt.scatter(y\_test, err)
* plt.ylabel('Error')
* plt.xlabel('HCC')



**HHC vs Satisfaction**

Hypothesis: with higher hcc you may see lower satisfaction

Raw data:



The raw data does seem to suggest this as you see less 4 and 5 satisfaction ratings with a 4 and 5 hcc.

Give a test y(hcc), what will X (satisfaction) be:



The y\_test shows a bigger spread with describe:

count 4302.000000

mean 2.185625

std 0.345743

min 0.810000

25% 1.960000

50% 2.160000

75% 2.380000

max 4.330000

Name: Average HCC Score, dtype: float64

So I still don’t really understand the graph above.