QBS Competition 3 - Report

(A) what I do for data pre-processing (remove NA or feature engineering……)

(B) explorative data analysis (EDA)

For (A) and (B), This time I didn't do pre-processing and EDA, Because it seems nothing to do with the data this time. But I still do something easy.

The Total num of the Train pictures. The num of the pictures having defects:

	ImageId	allMissing
0	1.JPG	0
1	2.JPG	1
2	3.JPG	1
3	5.JPG	0
4	6.JPG	0
...	...	...
8793	12561.JPG	0
8794	12563.JPG	1
8795	12564.JPG	1
8796	12567.JPG	0
8797	12568.JPG	0

8798 rows × 2 columns

	ImageId	allMissing
0	1	0
1	5	0
2	6	0
3	7	0
4	8	0
...	...	...
4677	12558	0
4678	12559	0
4679	12561	0
4680	12567	0
4681	12568	0

4682 rows × 2 columns

We can see that we have 4682 pictures having defects and 4116(8978–4682) pictures having no defect. Because the amount of them are close, we needn't to set weight_class to the first model, which is used to predict whether or not picture having defects, in my code.

(C) model development process:

I. DL model draft, parameter initialization, parameter tuning

I used the model pre-train by others directly, and use this as a standard to decide how to tune hyper-parameters and choose the optimizer and other choice.

II. DL finalized model, parameter initialization, parameter tuning

I start to freeze some part of the model, and train it by myself to see whether it would perform better. I have two model one for predicting whether the pictures

have defects, and another of predicting the genres of defects if there are. For the first one, It doesn't perform better when I re-train the model using our data, so I chose to use its original version. As for the second one, I find that if I train almost all the layers but 2~3 layers in the bottom, it would perform better than just using it without any re-train.

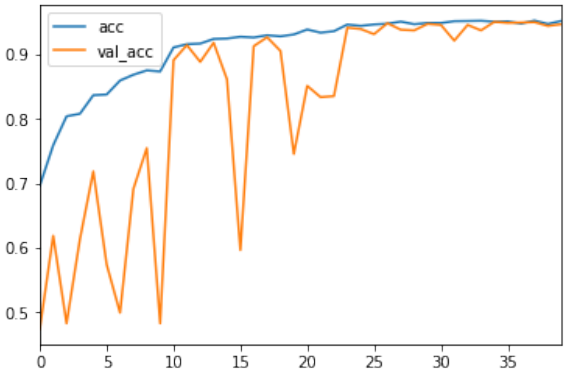
(d) The prediction result

I got about 0.667 dice-coefficient at the end. And I think that the reason I can't get higher is because I don't have enough training data. For instance, the original Kaggle competition have about 12,000 pictures, but I only have about half of their data, but I'd tried my best!

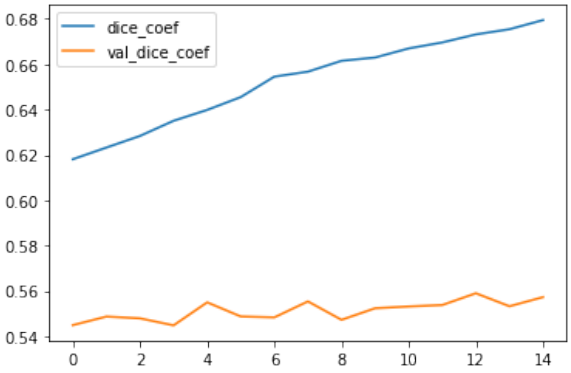
(e) explaining the model prediction to make people trust and understand whether your predictive machine is reasonable, understandable, and trustable.

I think my method is reasonable. Because I check if the picture has defects first, which is much easier and therefore would have higher accuracy, and then, I check the genres of them.

This is the accuracy of predicting whether pictures have defects:



This is the accuracy of predicting the genre of defects:



(f) Your modeling and data analysis based on the lectures, tutorials, and assigned readings.

The basic concept is based on the textbook (FC) and tutorials.

The hyper-parameters part is based on the textbook (FC)

The class weight setting is based on the information given by TA and google and a classmate.

Thinks a lot to Kaggle's everyone.

(e) learning progress, reflection, and feedback for the teaching team's reference

In my learning process, I consider discussing with is important, and it's really important to take Kaggle as reference. For feedback, I hope that TA and Teacher could taught us how to deal with some particular situation, because when it's my time to put developing the prediction in practice, I find much more problem that I've never seen in the course or textbook, even self-reading. Maybe it's because of lacking of experience, but, if possible, I still want to know more technique in class.