Batch Learning w/Random Forest Sklearn [closed]

Asked 1 year, 11 months ago Active 1 year, 11 months ago Viewed 5k times

2

2

Closed. This question is off-topic. It is not currently accepting answers.

Want to improve this question? Update the question so it's on-topic for Cross Validated.

Closed 2 years ago.

I have a data set of approximately 5 million rows and wanted to run a RandomForestClassifier. I ran my RandomForestClassifier with only 50 trees, I tried to use the fit function but I receive a memory error. I tried running it on the AWS box with 64GB worth of memory but still I run into this issue.

I was wondering if it was possible to use some sort of Batch Learning to overcome this issue using sklearn? I am open to other suggestions if anyone has any.

machine-learning random-forest scikit-learn large-data

asked Feb 7 '18 at 15:31

ml_enthusiast

41 1 4

1 Answer

5

Yes, Batch Learning is certainly possible in scikit-learn. When you first initialize your RandomForestClassifier object you'll want to set the warm_start parameter to True. This means that successive calls to `model.fit` will not fit entirely new models, but add successive trees.

Here's some pseudo-code to get you started. This will build one tree for each sub chunk of your data.

```
# split your data into an iterable of (X,y) pairs
# size each one so that it can fit into memory
data_splits = ...

clf = RandomForestClassifier(warm_start = True, n_estimators = 1)
for _ in range(10): # 10 passes through the data
    for X, y in data_splits:
        clf.fit(X,y)
        clf.n estimators += 1 # increment by one so next  will add 1 tree
```

I'm surprised to see that there's not a `subsample` parameter in RandomForestClassifier similar to the one in GradientBoostingClassifier that controls the number of observations visible to each tree. If you switched to GradientBoostingClassifier you might be able to simply set `subsample` to be a very small number to achieve the same results.

answered Feb 7 '18 at 16:14

user1993951
371 1 3

How does this resolve the memory error that OP has? – Sycorax says Reinstate Monica Feb 7 '18 at 16:25

1 If the memory error is related to computing the random forest on large data then this would resolve it by limiting the amount of data the model trains on at each iteration. I believed this to be the case based on the question's description of the error occurring during the fitting stage. If the memory error is actually the result of not being able to hold all the training data in memory at all, then the solution will have to be adapted to save chunks of data to disk then to only read in the data needed for each training iteration at training time. – user1993951 Feb 7 '18 at 16:29