DOCTORAL PROGRAM IN ENGINEERING SCIENCES AT ITESO

ADAPTATIVE DISCOVERING ALGORITHM BASED ON NEURAL NETWORKS

Edgar-D. Ramirez-de-L.
Maria-del-P. Pozos-P.
Ivan Villalon-Turrubiates

Algorithm 5 - trainInputFunction.doc

May 15, 2018
Tlaquepaque, Mexico 45604

Doctoral Program in Engineering Sciences
ITESO (*Instituto Tecnológico y de Estudios Superiores de Occidente*)

ADAPTATIVE DISCOVERING ALGORITHM BASED ON NEURAL NETWORKS

Edgar-D. Ramirez-de-L.
Maria-del-P. Pozos-P.
Ivan Villalon-Turrubiates

May 15, 2018

Doctoral Program in Engineering Sciences
ITESO (*Instituto Tecnológico y de Estudios Superiores de Occidente*)

Tlaquepaque, Mexico 45604
Tel +52 33 3669 3598
E-mail: dci@iteso.mx

Keywords *neural networks, deep learning, machine learning, classifiers, python*

Abstract

We present the Algorithm 5 (`trainInputFunction`) which is part of the Adaptative Discovering Algorithm based on Neural networks (ADAN algorithm).

```
Algorithm 5 trainInputFunction
Require: $features \neq \varnothing \wedge labels \neq \varnothing \wedge batchSize \neq \varnothing$
1: $dataSet \leftarrow toSlices(features, labels)$
2: $dataSet \leftarrow shuffle(dataSet, 1000)$
3: $dataSet \leftarrow repeat(dataSet, \varnothing)$
4: $dataSet \leftarrow batch(dataSet, batchSize)$
5: return $dataSet$
```

Some important considerations about the Algorithm 5:

1. The method and implementation is basically the same as you can see in the Premade Estimators[1] guide. More precisely, as in `iris_data.py`[2]. So, just to make it a little more clear, we will describe the functions in lines 1 to 4, and you can find more details in[20,21].

The `toSlices(tensors, labels)` function creates a data set whose elements are slices of the given tensors[3,4]. Note that `labels` could be empty/null (`None` in Python).

`shuffle(dataSet, bufferSize)`, in line 2, is a function that randomly shuffles

[1] Premade Estimators, TensorFlow™. May 7, 2018, https://www.tensorflow.org/get_started/premade_estimators
[2] models/iris_data.py, TensorFlow™. May 7, 2018, https://github.com/tensorflow/models/blob/master/samples/core/get_started/iris_data.py
[3] Tensors, TensorFlow™. May 7, 2018, https://www.tensorflow.org/programmers_guide/tensors
[4] tf.data.Dataset.from_tensor_slices, TensorFlow™. May 7, 2018, https://www.tensorflow.org/api_docs/python/tf/data/Dataset#from_tensor_slices

the elements of the specified dataSet. In the implementation this is achieved thanks to tf.data.Dataset.shuffle(buffer_size,seed=None,reshuffle_each_iteration=None)[5]

The function repeat(dataSet, count) repeats the dataSet count times. For more details please consult the documentation for this function[6].

Finally, in line 4, the batch(dataSet, batchSize) function, combines consecutive elements of the dataSet into batches from the size specified by batchSize. This is achieved thanks to tf.data.Dataset.batch(batchSize)[7].

[5] tf.data.Dataset.shuffle, TensorFlow™. May 7, 2018, https://www.tensorflow.org/api_docs/python/tf/data/Dataset#shuffle

[6] tf.data.Dataset.repeat, TensorFlow™. May 7, 2018, https://www.tensorflow.org/api_docs/python/tf/data/Dataset#repeat

[7] tf.data.Dataset.batch, TensorFlow™. May 7, 2018, https://www.tensorflow.org/api_docs/python/tf/data/Dataset#batch