

Northeastern University

Data Management and Database Design

INFO6210

Spring 2019

Hyperparameter Database Team 11

Project Report

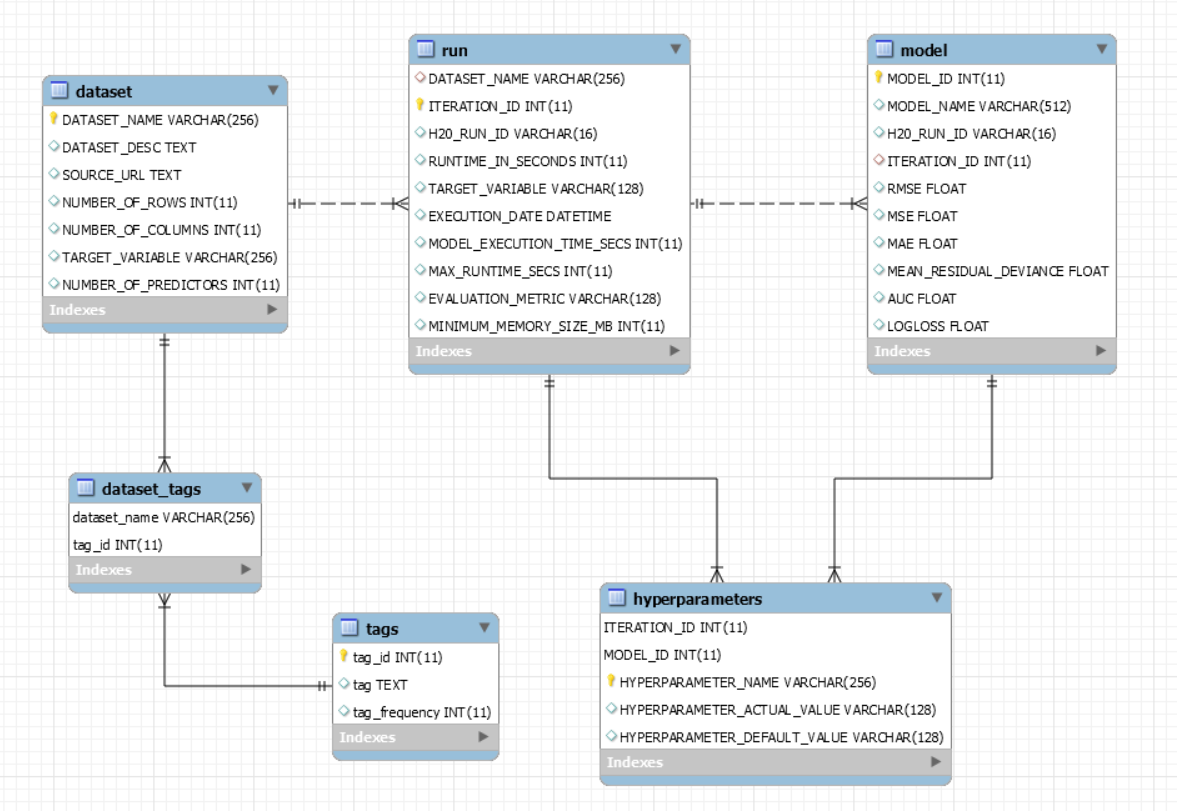
By Darshan Durve & Shriram Karthikeyan

# Abstract

The goal of this hyperparameter project is to create a database consisting of hyperparameters that were extracted by running various machine learning models (classification/regression) on multiple datasets. Using the database of hyperparameters that is generated we will suggest hyperparameters for the dataset that the user would like to run ML models on; this aspect of the project is not currently within our scope therefore we will limit ourselves to building a database of hyperparameters for one dataset.

# The Dataset We have got the CalCOFI dataset from Kaggle.com. The data set represents the longest (1949-present) and most complete (more than 50,000 sampling stations) time series of oceanographic and larval fish data in the world. It includes abundance data on the larvae of over 250 species of fish; larval length frequency data and egg abundance data on key commercial species; and oceanographic and plankton data.

# ER DIAGRAM



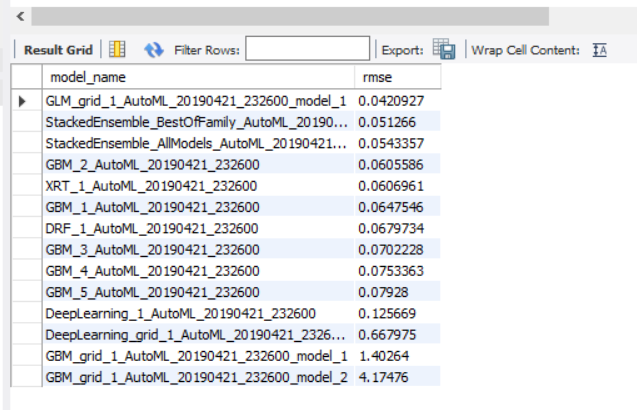
Above E-R diagram explains the relationship between the models for each run with their hyperparameters. Each run has different models generated by H2O and each model has different hyperparameters associated to it. We have also stored tags to associate the dataset with keywords.

# Use Cases

1. Retrieve the rmse value for the models of the lowest runtime.

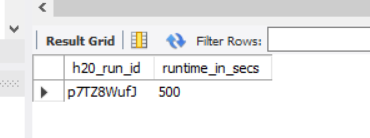
* select m.model\_name, m.rmse from model m join run r on m.iteration\_id=r.iteration\_id

where r.max\_runtime\_secs=(select min(max\_runtime\_secs) from run);



1. Retrieve the runid from the meta-data for the lowest runtime.

- Select h20\_run\_id, max\_runtime\_secs as runtime\_in\_secs from run where max\_runtime\_secs=(select min(max\_runtime\_secs) from run);

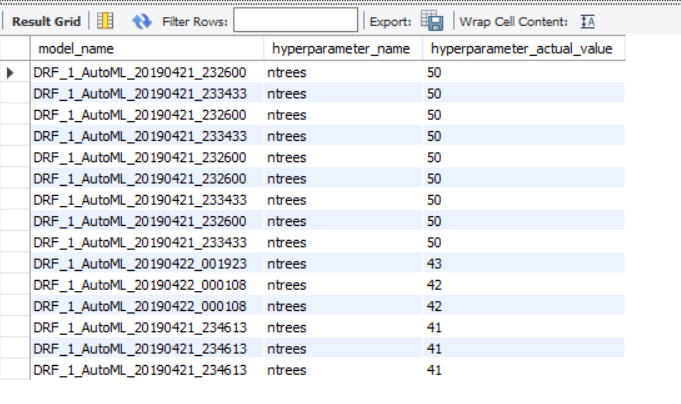


1. Retrieve the ntrees from the Random Forest algorithm.

- select m.model\_name, h.hyperparameter\_name, h.hyperparameter\_actual\_value

from model m join hyperparameters h on m.model\_id=h.model\_id

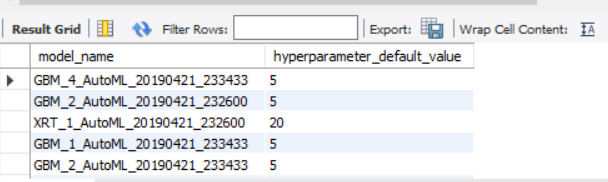
where m.model\_name like 'drf%' and h.hyperparameter\_name = 'ntrees' order by h.hyperparameter\_actual\_value desc;



1. The default max\_depth of the models

- select m.model\_name, h.hyperparameter\_default\_value from hyperparameters h join model m on h.model\_id=m.model\_id

where h.hyperparameter\_name = 'max\_depth';

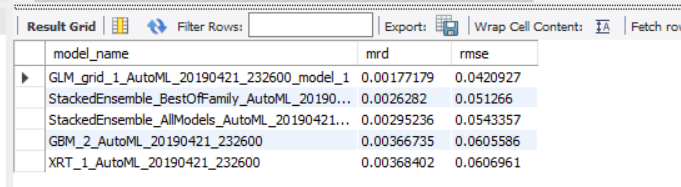


1. The mean residual deviance for the top 5 models of the lowest runtime.

#Based on rmse

select model\_name, mean\_residual\_deviance as mrd, rmse from model m join run r on m.iteration\_id=r.iteration\_id

where r.max\_runtime\_secs=(select min(max\_runtime\_secs) from run) order by rmse limit 5;

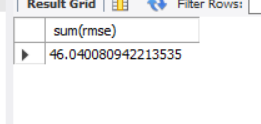


1. Sum of rmse values of any least or highest runtime.

#For highest

select sum(rmse) from model m join run r on m.iteration\_id=r.iteration\_id

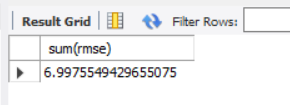
where r.max\_runtime\_secs in (select max(max\_runtime\_secs) from run);



#For lowest

select sum(rmse) from model m join run r on m.iteration\_id=r.iteration\_id

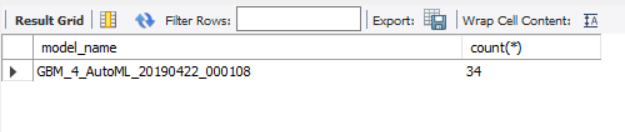
where r.max\_runtime\_secs in (select min(max\_runtime\_secs) from run);



1. The number of models of the same algorithm for a particular runtime(1100).

select model\_name, count(\*) from model m join run r on m.iteration\_id=r.iteration\_id

where r.max\_runtime\_secs = '1100' and model\_name like 'GBM%';

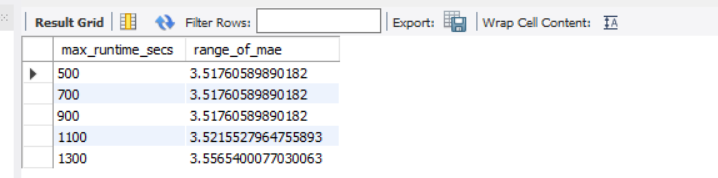


1. The difference between the highest and lowest mae values of the models for a runtime.

select r.max\_runtime\_secs, max(m.mae)-min(m.mae) as range\_of\_mae

from run r join model m on m.iteration\_id=r.iteration\_id group by r.max\_runtime\_secs

order by max\_runtime\_secs;



# Functions

1. Function to suggest a dataset based on the given tag –

CREATE DEFINER=`root`@`localhost` FUNCTION `fetch\_dataset`(tag\_name varchar(128)) RETURNS varchar(128) CHARSET utf8mb4

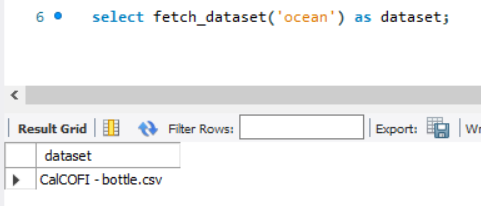
BEGIN

declare dataset varchar(128);

select distinct d.dataset\_name into dataset from tags t join dataset\_tags d on t.tag\_id=d.tag\_id where tag like concat('%',tag\_name,'%');

RETURN dataset;

END



1. Function to suggest best value of specified hyperparameter of specified model at specified runtime

CREATE DEFINER=`root`@`localhost` FUNCTION `best\_hyperparameter`(runtime varchar(128), model varchar(128), hyper varchar(128)) RETURNS varchar(128) CHARSET utf8mb4

BEGIN

declare best\_value varchar(128);

select distinct HYPERPARAMETER\_ACTUAL\_VALUE into best\_value from model m join run r on m.iteration\_id=r.iteration\_id

join hyperparameters h on m.model\_id=h.model\_id

where r.max\_runtime\_secs = runtime and

m.model\_name like concat('%',model,'%')

and h.hyperparameter\_name like concat('%',hyper,'%')

and m.rmse = (select min(m.rmse) from model m join run r on m.iteration\_id=r.iteration\_id

where r.max\_runtime\_secs = runtime and m.model\_name like concat('%',model,'%'));

RETURN best\_value;

END

