Домашняя работа №2 Дедлайн 16 Марта 2021, 23:59

Задание

В этой домашке мы работаем с теми же самыми данными, что и в первом домашнем задании, и задача для обучения модели стоит точно такая же. Прочтите описание данных и задачи ниже.

Однако мы будем работать с данными используя только Hive.

За основу взят датасет и соревнование Criteo Display Advertising challenge

Задание

Предскажите вероятность клика на рекламный баннер. Признаки включают в себя некоторое количество целочисленных и категориальных переменных. Данные в обучающем датасете покрывают 7 последовательных дней, данные валидационного датасета покрывают 8-ой день.

Первая колонка датасета - идентификатор записи - целое число. Вторая - целевая переменная: 1 - клик, 0 - нет клика Следующие 13 - целочисленные признаки, имеются пропуски. Следующие 26 - категориальные признаки, так же имеются пропуски Последняя колонка - порядковый номер дня. Заметим что нет никакой информации о днях недели, которым соответствуют номера дней в датасетах.

Для простоты, возьмите следующее определение полей датасета:

```
numeric_features = ["if"+str(i) for i in range(1,14)]
categorical_features = ["cf"+str(i) for i in range(1,27)] + ["day_number"]

fields = ["id", "label"] + numeric_features + categorical_features
```

Метрика - Log loss (http://wiki.fast.ai/index.php/Log_Loss).

Более подробное описание задачи можно найти на сайте соревнования по ссылке выше. Заметьте, что в соревновании на Kaggle в датасете не было поля day_number, оно было

добавлено при сэмплировании из расширенного датасета Criteo, в котором файлы данных содержали один день. Вы можете использовать day_number на ваше усмотрение.

Вы можете пользоваться моделями, описанными на форуме соревнования, искать среди них идеи, но ваше решение все равно должно укладываться в заданные рамки оформления работы, см. ниже.

Что делать

Приготовьте и положите в репо в подпапку projects/2 следующие скрипты .hql:

- create_test.hql - создание внешней таблицы hw2_test с тестовым датасетом /datasets/criteo/criteo_test_large_features. Таблица должна быть temporary external.

- create_pred.hql - создание managed таблицы hw2_pred для предсказаний. В качестве location укажите <ваш логин>_hw2_pred. В этой таблице должны быть две колонки - для id записи, и для предсказания.

- filter_predict.hql - наполнение таблицы hw2_pred путем запроса с фильтрацией по заданному условию 20< if1 <40 и одновременным предсказанием модели на фильтрованных данных.

- select_out.hql - выгрузка данных из таблицы hw2_pred в текстовом виде в относительный (без начального /) путь <ваш логин>_hiveout в HDFS.

Кроме того, включите файлы model.py, train.py, train.sh, predict.py. После тренировки модели, она должна сохраняться в 2.joblib.

Чекер

- Последовательно выполнит ваши скрипты и проверит их результат.

- скачает из HDFS файл <ваш логин>_hiveout и рассчитает метрику

Чекер выполняется из корня вашего репо. Команды Hive будут подаваться вот таким образом, где NAME_ - ваш логин с тире (-) замененным на подчерк (_):

```
create database if not exists ${NAME_}_checker;
use ${NAME_}_checker;
drop table hw2_test;
```

```
source projects/2/create_test.hql;
describe hw2_test;
select count(id) from hw2_test;
drop table hw2_pred;
source projects/2/create_pred.hql;
describe hw2_pred;
source projects/2/filter_predict.hql;
select count(id) from hw2_pred;
source projects/2/select_out.hql;
```

Подсказки

Обратите внимание, как Hive обрабатывает пропущенные значения.

Зачет

Зачет при метрике лучше, чем 0.7, то есть <0.7. Если не добились этого скора (чекер показывает PASSED 0), то работу все равно зачтем.

После того как добились PASSED 1 (или 0) от чекера, сделайте скриншот и сдайте работу в классрум