Lab: Batch Scoring in DSX

November 30th, 2017

Author: Elena Lowery elowery@us.ibm.com

Table of contents

Contents

Overview

In this lab you will learn how to configure batch scoring in DSX.

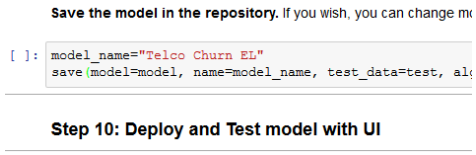
Required software, access, and files

- To complete this lab, you will need access to a DSX Local cluster.
- You will also need to download and unzip this GitHub repository: https://github.com/elenalowery/DSX_Local_Workshop

Part 1: Load the sample project and create model

1. If you haven't already created a project from *DSX_Local_Workshop.zip* file, follow instructions for *Use Case 1* in this repository *https://github.com/elenalowery/DSX_Local_Workshop*

2. If you haven't run through the *TelcoChurn* notebook, run through it so that you generate a model. The easiest way to do this is to open the notebook, scroll down to **Step 10**, click on it, then in the menu select **Cell -> Run all above**.



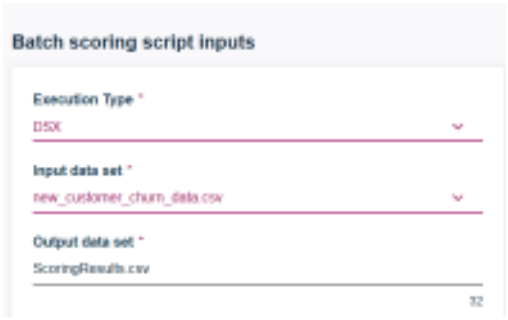
   Navigate to the **Assets** view and make sure that the model has been created. Your model may have a different name and version.

Part 2: Configure batch scoring of a model

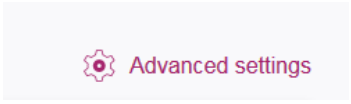
1. Click on the ellipses next to the model and select **Batch Score**.

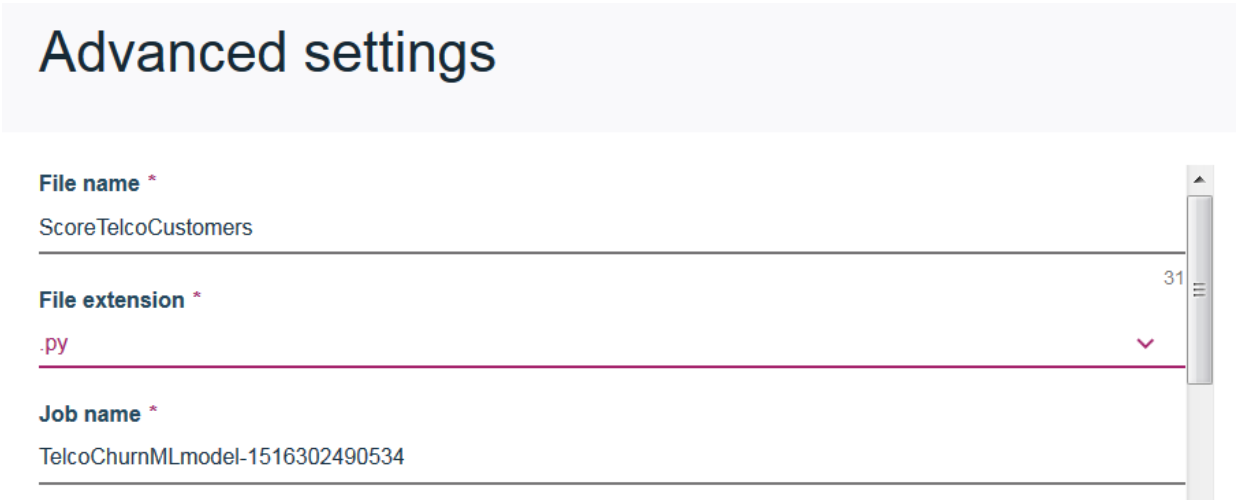
2. Fill out the required fields:

    - Input data set: *new_customer_churn_data.csv* (this data set contains data that should be scored)
    - Output data set: *ScoringResults.csv.* Make sure to provide **.csv** extension – otherwise you won't be able to preview and download the output.

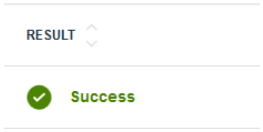


3. Click on **Advanced Settings** and change the file name to *ScoreTelcoCustomers*. Click **Save**.

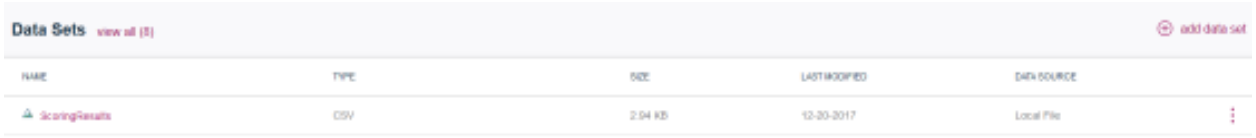




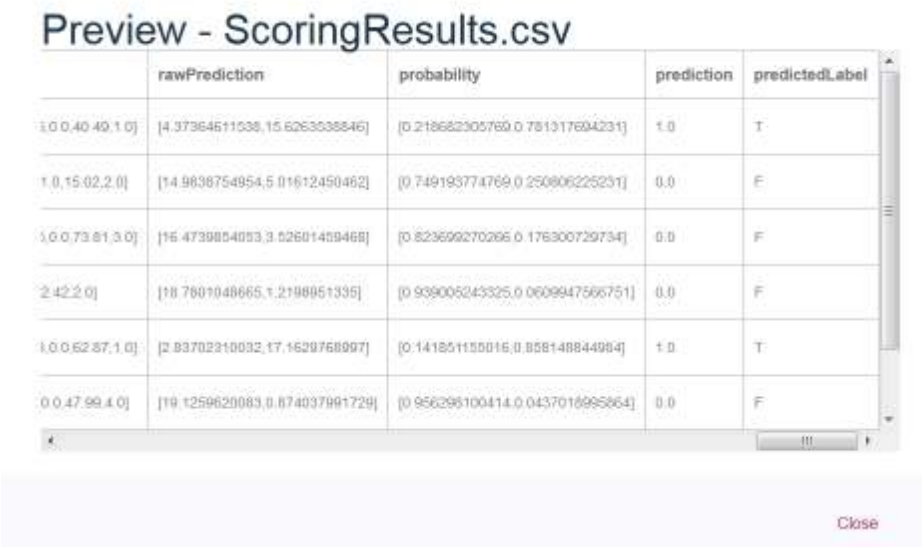
4. Click **Generate Batch Script**. Then click **Run now** and wait till the status changes to *Success*.



5. Navigate to the **Assets** view of the project. Scroll down to **Data Sets**. You should see the generated ScoringResults.csv file.



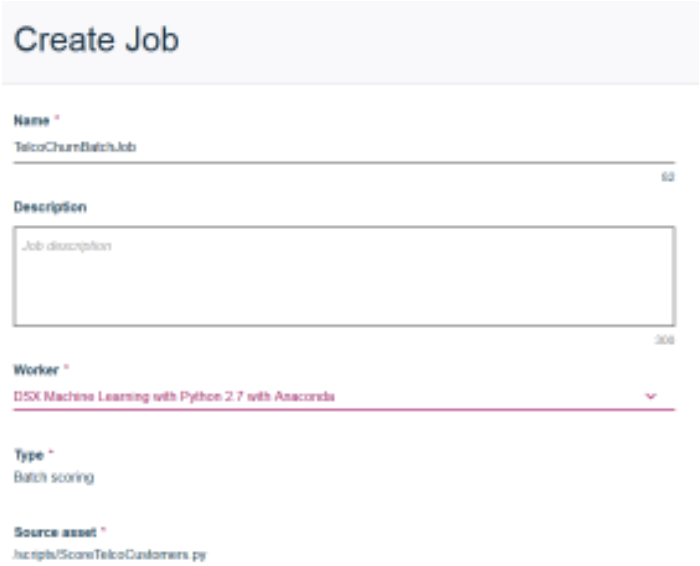
6. You can choose to preview or download the file (click on the ellipses). The scoring results are the last few columns: *rawPrediction, probability, prediction, predictedLabel.*



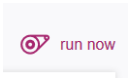
So far we have generated the batch job script and made sure that it works by running it interactively. Now we can schedule a batch job.

7. In the Project view scroll down to **Scripts** and click **Create Job** next to the script we created in previous steps.

8. On the **Create Job** screen provide *Job name* and make sure to select the right **Worker** (*Python 2.x or 3.x*) environment (check comments in the notebook or check with the instructor). Scroll down and select either "on demand" or a specific time. Click **Create**.



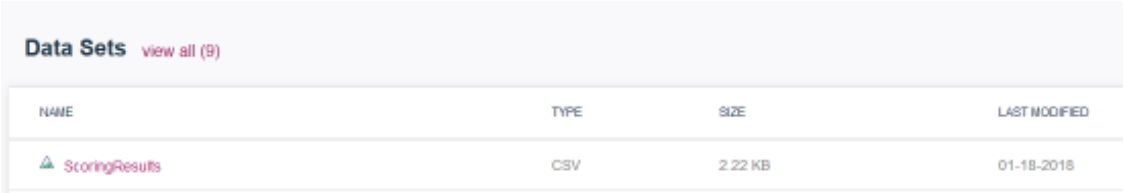
9. In the **Batch Job Details** view scroll down and select **Run Now**.



10. Provide Run name and click **Run**.

11. Job status is displayed. Wait till status is "Success".

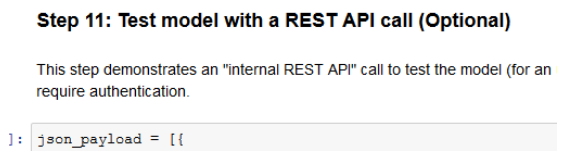
12. Navigate back to the Project view**,** and scroll down to **data sets.** Each batch job run overwrites the previous output file.



13. If you want to make sure that the file was updated, open the *RunSystemCommands* notebook and run the commands (check the timestamp on *ScoringResults.csv*).

Part 3: Configure batch scoring of a notebook

1. Open *TelcoChurn_SparkML* notebook and delete everything after **Step 11** (REST API call). These cells won't run in batch mode.



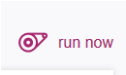
   *Note: If you want to make a copy of this notebook, don't use the **Make a copy** function in Jupyter (it's currently not working). Download the notebook, rename it, and add it to a project again.*

2. Navigate to the Project view then click on **Jobs**.

3. Click **create job**.

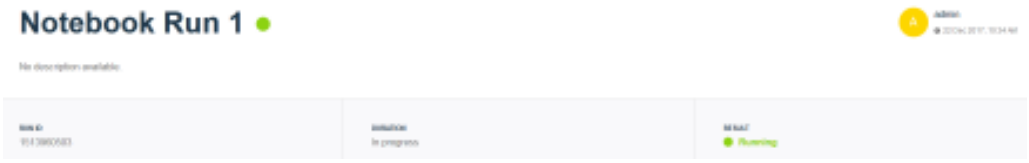
4. Enter Job information. Make sure to select **Worker** that matches version of Python in the notebook. Click **Create**.

5. In the **Batch Job Details** view scroll down and select **Run Now**.



6. Provide Run name and click **Run**.

7. Job status is displayed.



8. Navigate back to the **Batch Job Details** view to verify that the job ran successfully. You can also verify that the notebook ran by checking the version of the model in the **Project** view – it should be incremented.

You have finished working on **Batch Scoring in DSX lab**.