Workshop: ICP for Data – Analyze

Lab 2 – Build , Save, Test, and Evaluate a Machine Learning Model

*May 29, 2018*

*Author: Rui Fan* [*rui.fan@ibm.com*](mailto:rui.fan@ibm.com)

*Reference assets from: Sidney Phoon* [*yfphoon@us.ibm.com*](mailto:yfphoon@us.ibm.com)

**Table of contents**

Contents

**5. Build and save Spark ML model with Juypter Notebook**

5.1 Review and execute the code in TelcoChurn\_SparkML notebook

5.2 Verify that the model is saved under the model section of the project

**6. Test, Batch Score and Evaluate saved model**

6.1 Test the model

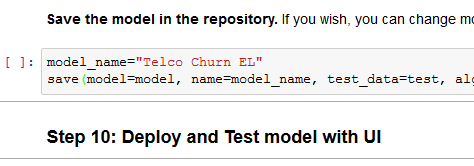
6.2 Create a batch score

6.3 Create an evaluation

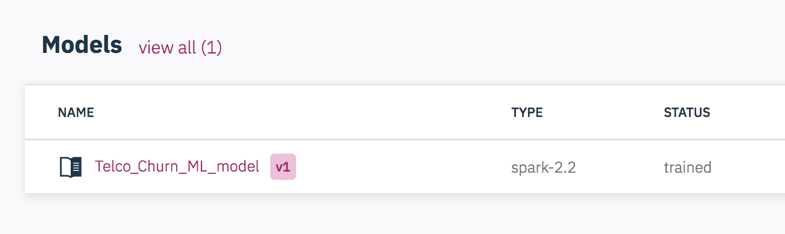
# **5: Build and save Spark ML model with Juypter Notebook**

In this section we will be loading the existing notebook TelcoChurn\_SparkML . The notebook contains comments which will guide you through its use. As you review each cell click in it with you mouse and use the run button on the toolbar to execute it.

1. Navigate to the ‘**Notebooks’** tab and open the prepared ‘**TelcoChurn\_SparkML**’ notebook.
2. In step 2, you will first load the merged remote customer data you created in the earlier sessions. You will need to automatically generate the python code just like you did in the previous exercise.
3. In Step 9 you will saved the model into the project repository for further use.



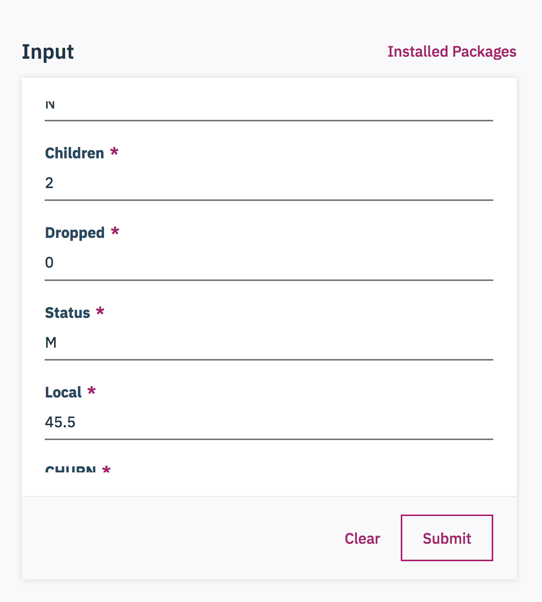
1. Verify that the model is saved under the model section of the project by navigating to the **Assets** view. Look under the ‘**Models**’ tab to make sure that the model is shown. Your model may have a different name and version.



# **6: Test, Batch Score and Evaluate saved model**

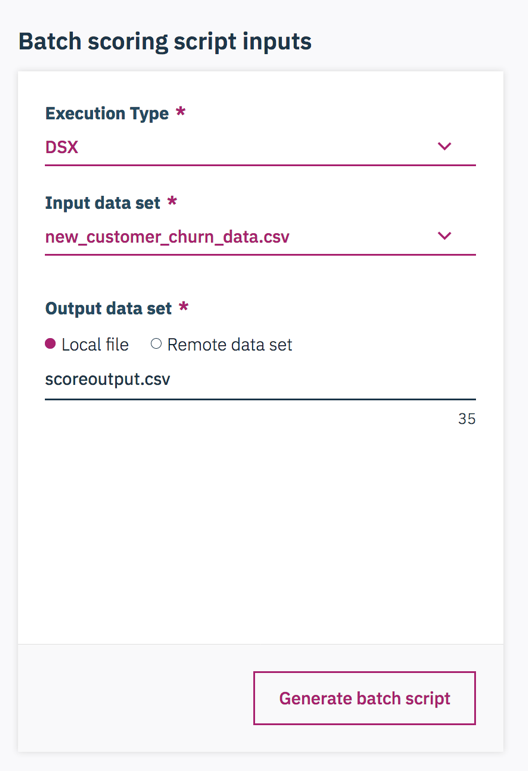
Let’s start by testing the model by creating an sample single case scoring scenario

1. Navigate to the Models tab in the Assets view of your project.
2. Click the the model name ‘**Telco\_Churn\_ML\_model**’ ; In the next view select the ‘**Test’** tab.
3. There will be some data automatically generated for each of the fields in the model. They are displayed in the ‘**Input’** section.
4. Click ‘**submit**’ and the online score will show on the right side ‘**Result**’ section.
5. You can change some of the input values to see how it changes the prediction.

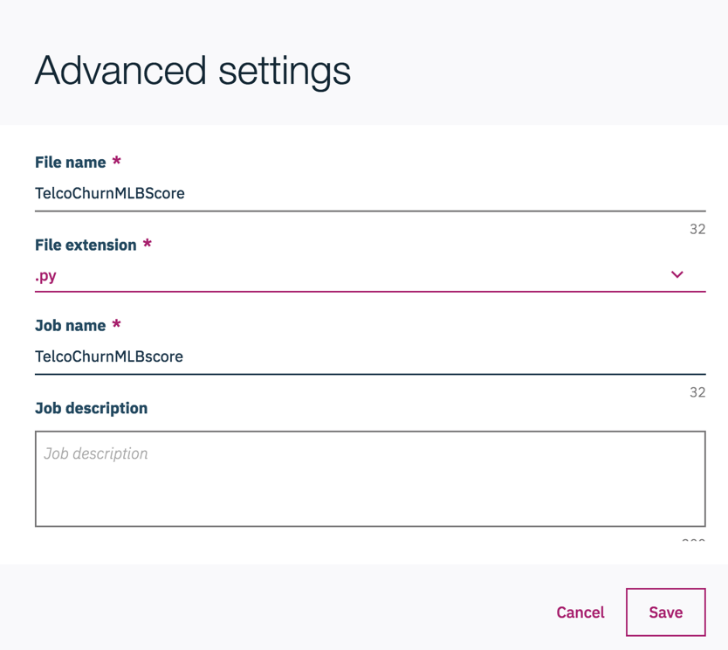


Now let’s create a reate a batch scoreing process to score more than one case at a time.

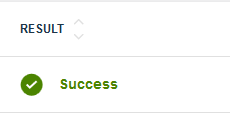
1. Select the ‘**Batch Score**’ tab for our model ‘Telco\_Churn\_ML\_Model’
2. Under ‘Execution Type’ select ‘DSX’
3. Under ‘**Input data set’** select ‘**new\_customer\_churn\_data.csv’** . This is a preloaded data set that contains cases that haven’t yet been scored by this model.
4. Under ‘**Output data set’** select ‘**Local file’** and type in the filename ‘**scoreoutput.csv’**. **Note**: You can call the output data anything you want but make sure to provide ‘**.csv’** extension otherwise you won’t be able to preview and download the output.



1. Click on ‘**Advanced Settings’** and change the file name to ‘**TelcoChurnMLBscore’**. You could also change the file into ‘.ipynb’, but ‘.py’ also works because your subsequent code changes take place in a python script. Click ‘**Save’.**



1. Click ‘**Generate batch script**’ and ‘**Run now’**. In the new window scroll down to the Runs table and wait utill the status changes to **Success**. This means your imported data set was successfully scored by the saved Model.



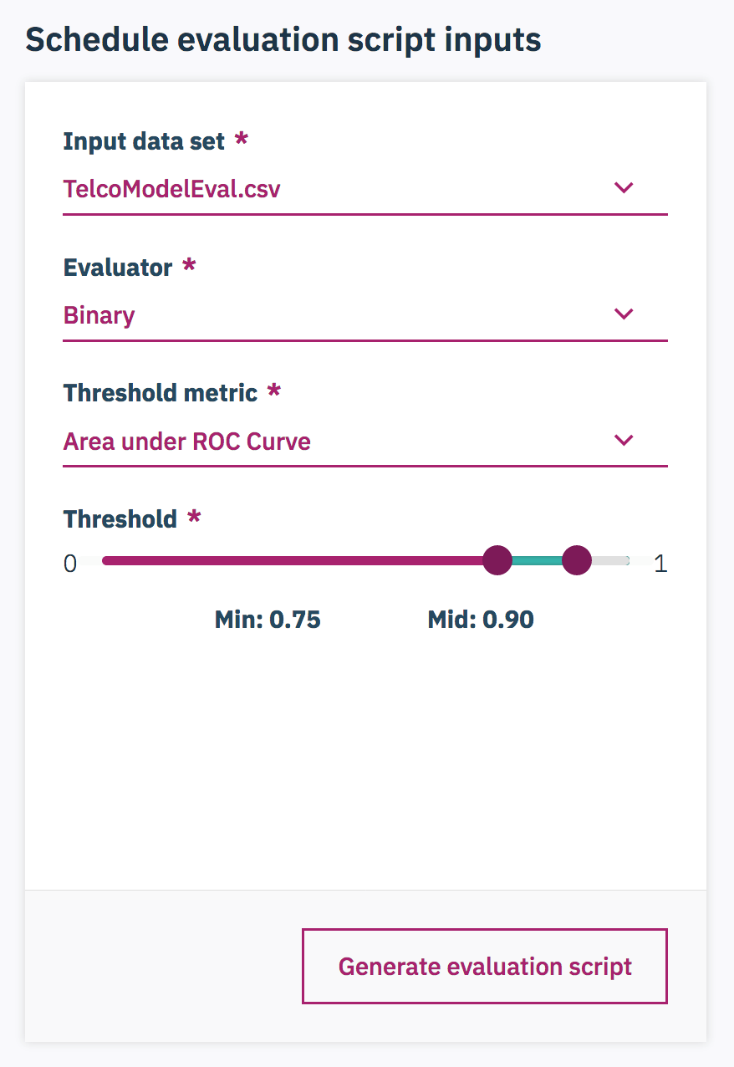
1. Navigate to the **Assets** view of the project. Scroll down to **Data Sets**. You should now see the generated **ScoreOutput.csv** file.



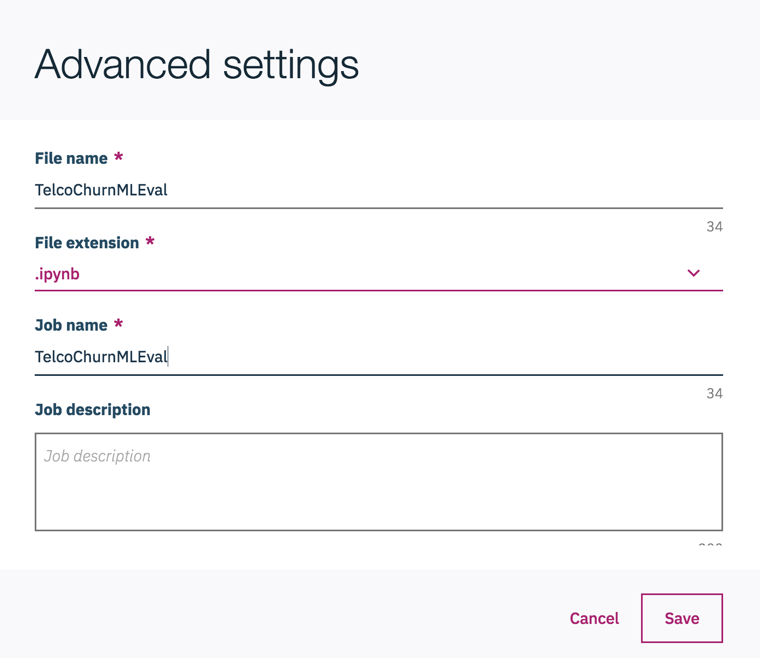
1. You can choose to download the file by clicking on the ellipses. The scoring results are the last few columns: rawPrediction, probability, prediction, predictedLabel.

Next let’s create an Evaluation scenario for the model so we can monitor its performance over time

1. Go back to the Models area under assets and click on the model we created.
2. Slect the ‘**Evaluate’ Tab**.
3. Under Input data set select ‘**TelcoModelEval.csv’** which contains the predictions and the actual values of Churn.
4. Under Evaluator select ‘Binary’ since we are predicting a binary output.
5. Under Threshold Metric select ‘**Area under ROC Curve**’ . This is a common option when we are predicting a binary output.
6. Define your own thresholds for what accuracy would consitite a good model or stick with the defaults.



1. Click ‘**Advanced Settings**’ and change the file name and job name to **‘TelcoChurnMLEval’** so that you can keep track of this job. Also you could change the file into **‘.ipynb’**. This ensures you can check the python script for evaluation. Click ‘**Save’**.



1. Click ‘**Generate Batch Script**’. In the ‘Result’ window, you could see the python code that will do the evaluation.Select ‘**Run’ .** Scroll down andwait till the status changes to success.