Lab Guide

Hands-on-Lab:
Building and deploying model using AutoAI

Shivam R Solanki
Data Scientist
Shivam.raj.solanki@ibm.com

IBM

With the aim of creating AI for AI, IBM introduced a service on Watson™ Studio called AutoAI.

AutoAI is a capability that automates machine learning tasks to ease the tasks of data scientists. It automatically prepares your data for modeling, chooses the best algorithm for your problem, and creates pipelines for the trained models.

AutoAI can be run in public clouds and in private clouds, including IBM Cloud Pak® for Data.

Learning objectives

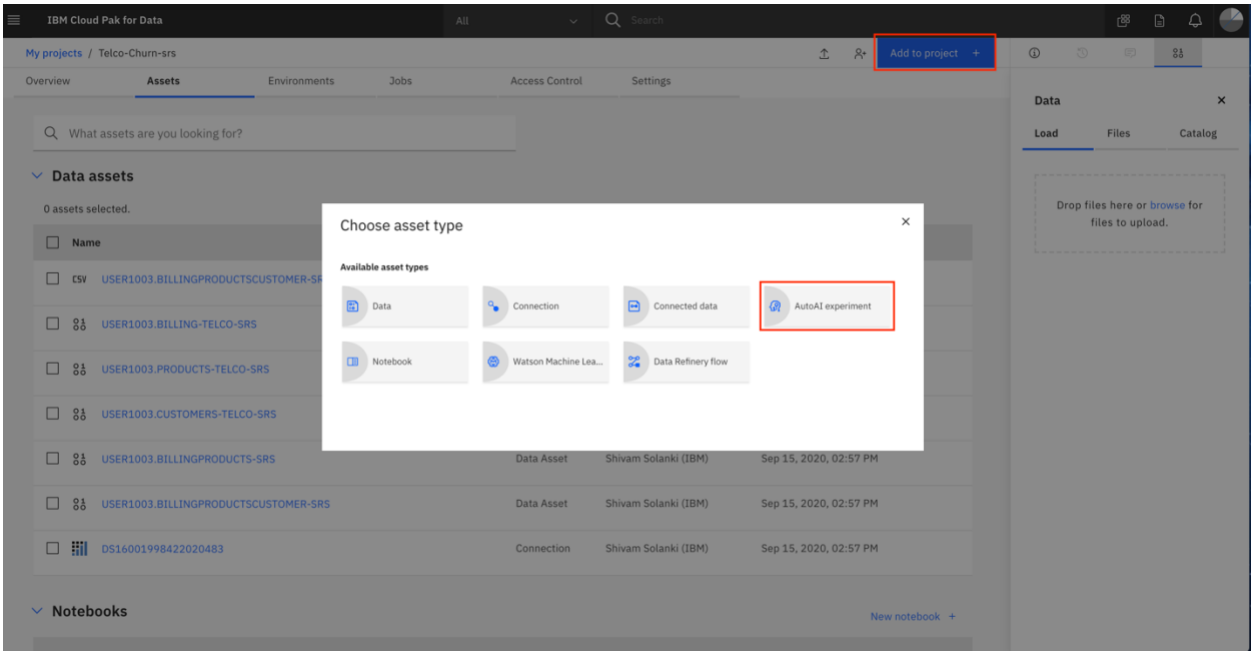
This tutorial explains the benefits of the AutoAI service on a use case. This will give you a better understanding of how regression and classification problems can be handled without any code — and how the tasks (feature engineering, model selection, hyperparameter tuning, etc.) are done with this service. This tutorial also includes details for choosing the best model among the pipelines and how to deploy and use these models via IBM Cloud Pak for Data platform.

- Setup your AutoAI environment and generate pipelines
- Save AutoAI model
- Deploy the model
- Test the model

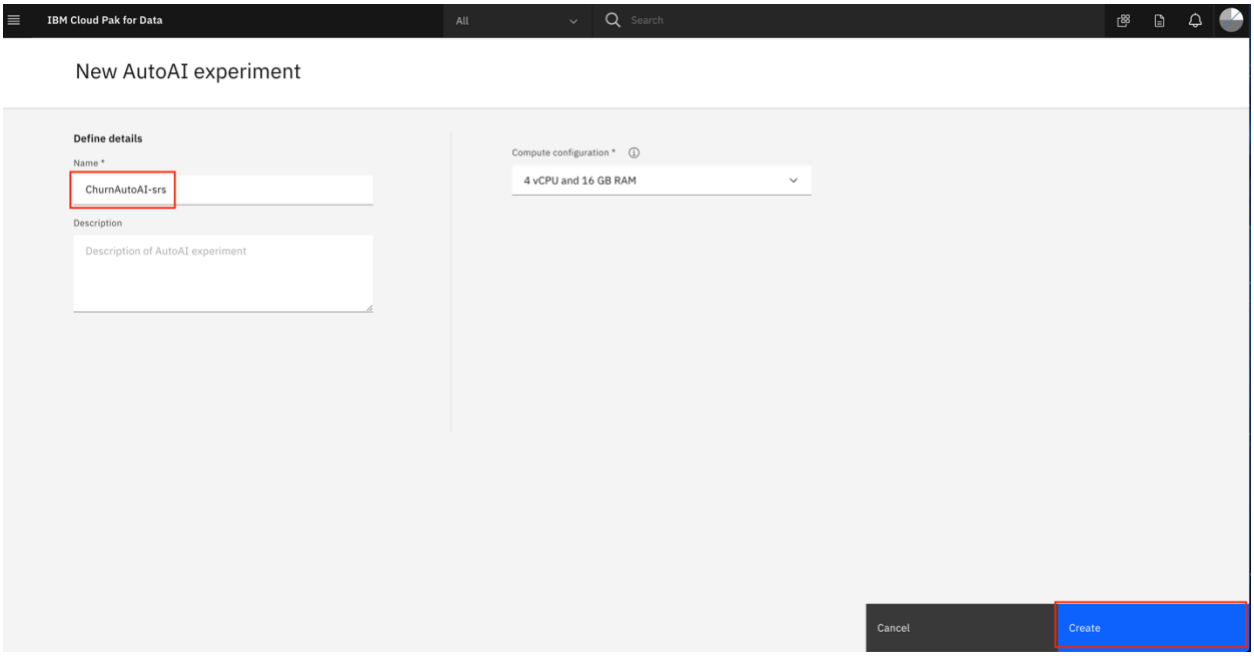
Steps

Step 1. Setup AutoAI environment and generate pipelines

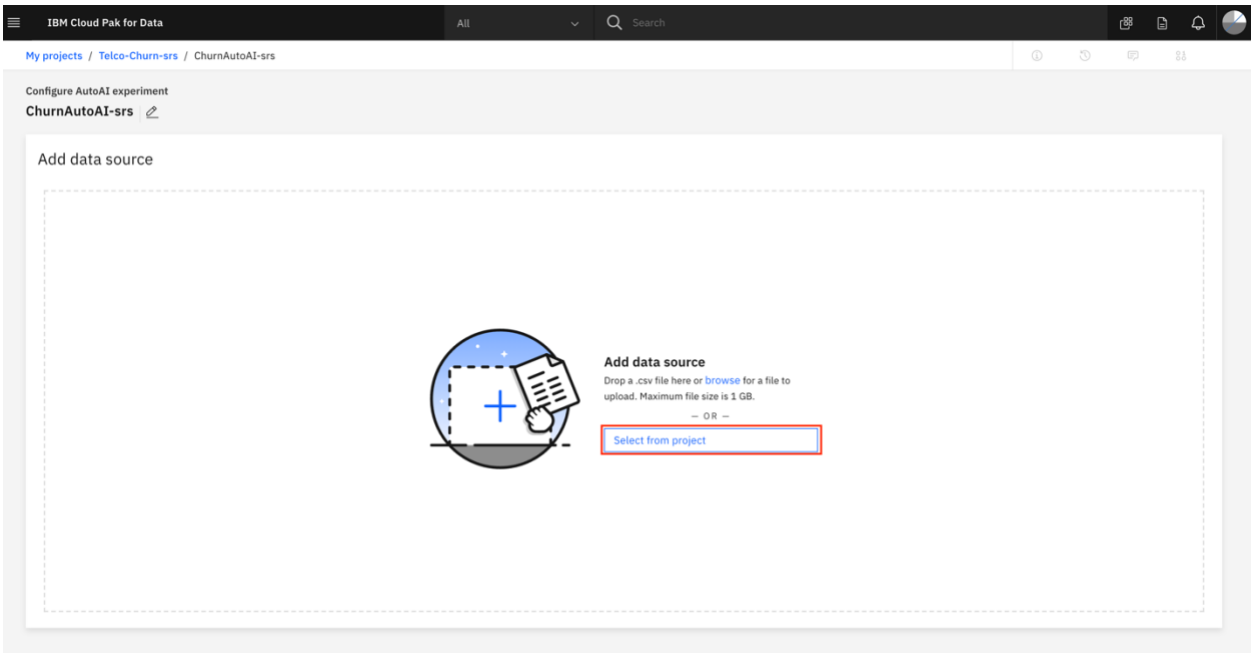
1. To start the AutoAI experience, click **Add to Project +** and select **AutoAI experiment**.



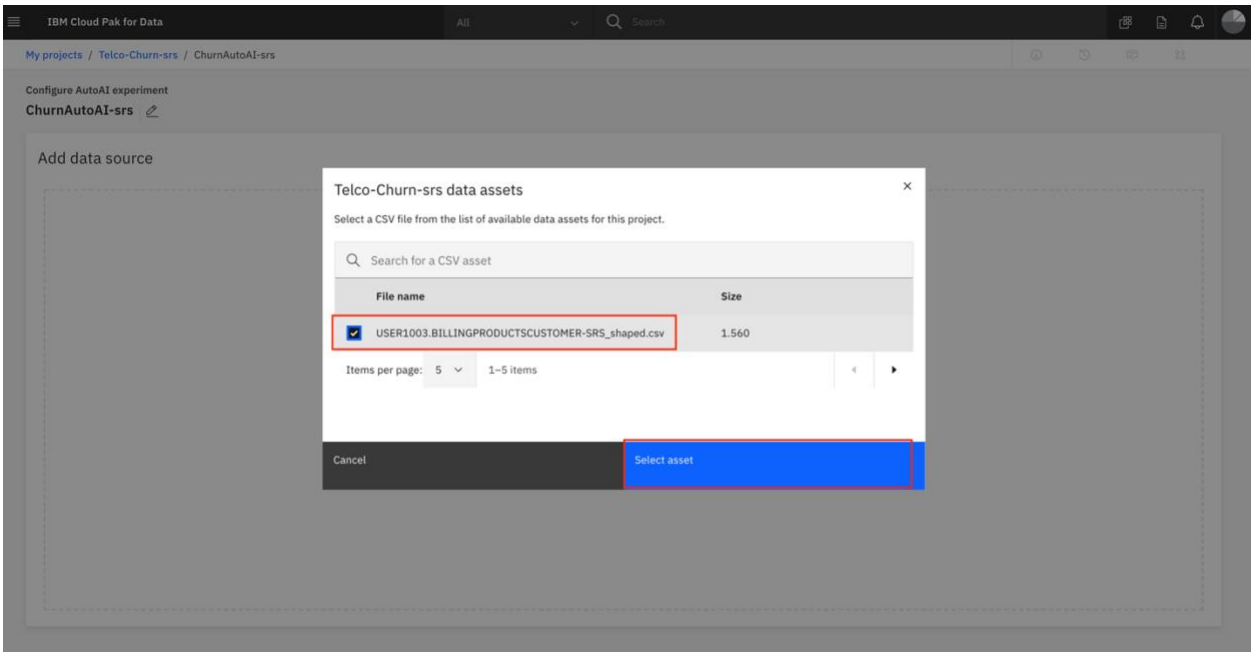
2. Name your AutoAI experiment asset and leave the default compute configuration option listed in the drop-down menu. Click **Create**. *(Note – For the purpose of this lab, please add your initials at the end of the* AutoAI experiment name *to avoid naming conflicts since all the participants are working in the same environment. For ex: ChurnAutoAI-srs)*



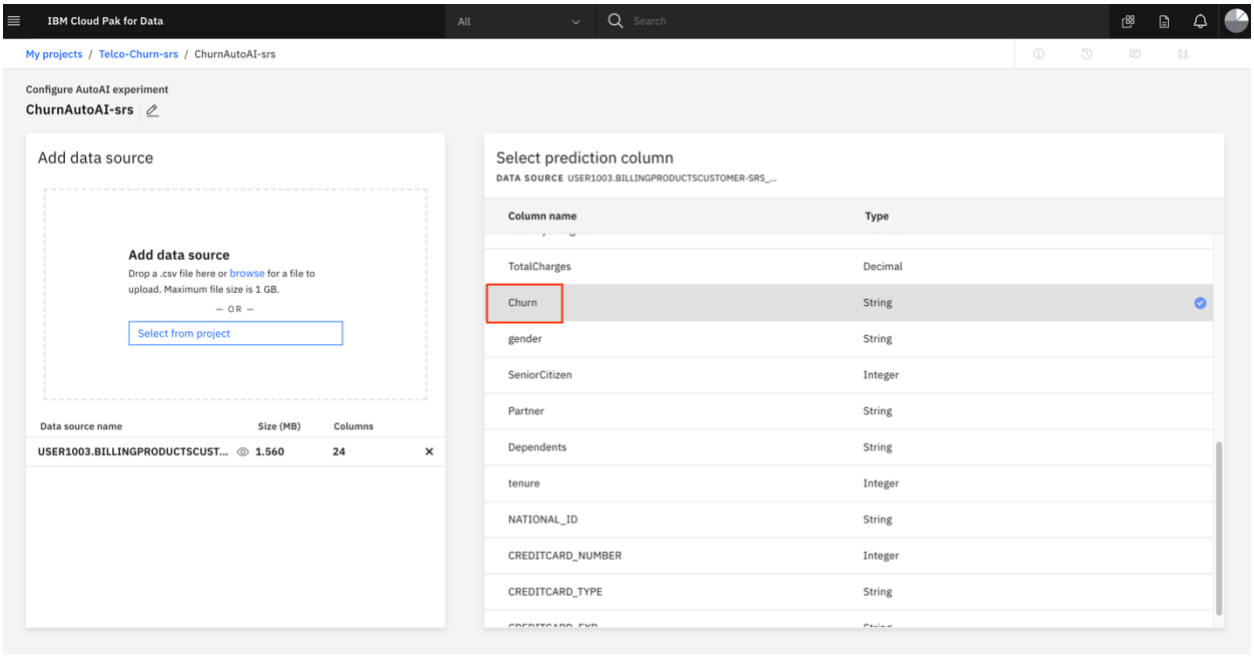
3. To configure the experiment, we must give it the dataset to use. Click on the **Select from project** option.



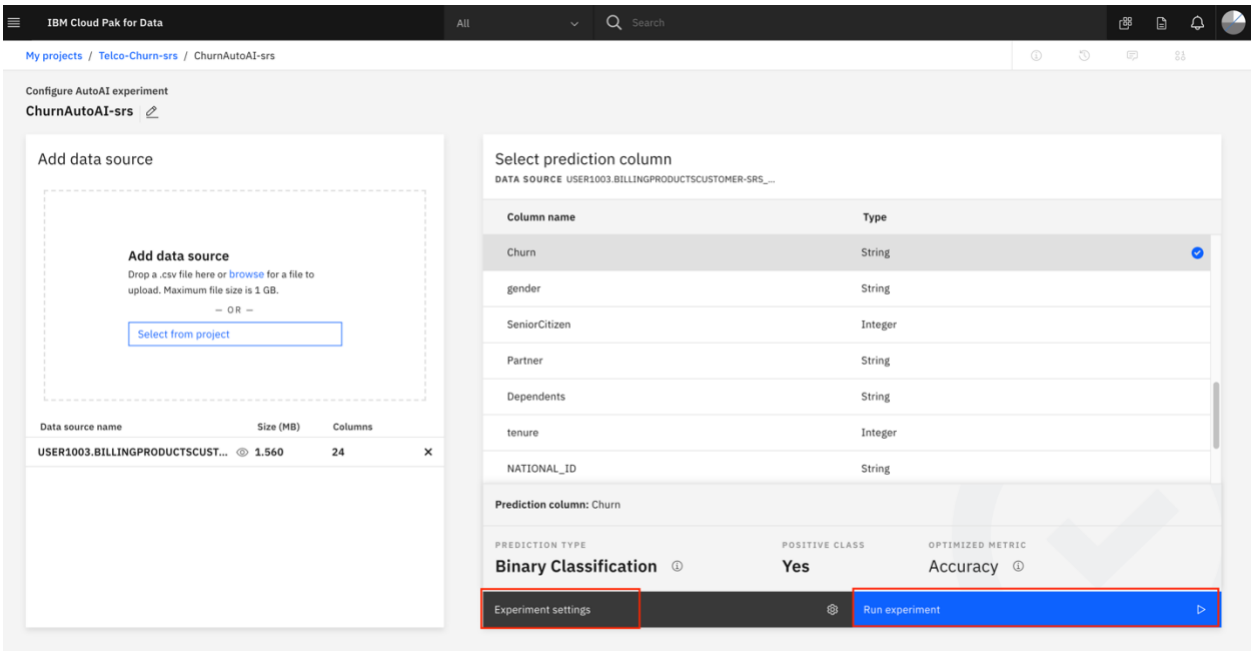
4. In the dialog, select the name of the shaped dataset that was the output of the data refinery step. Click **Select asset**.



5. Once the dataset is read in, we need to indicate what we want the model to predict. Under the Select prediction column, find and select the **churn** row.

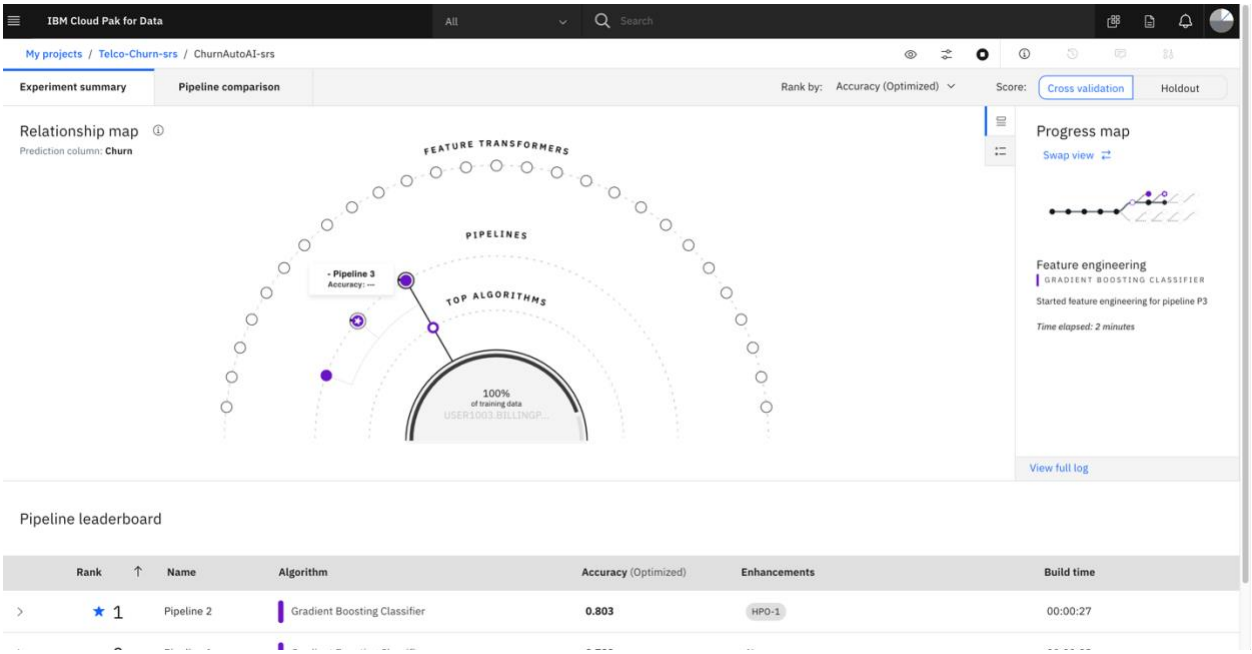


6. Click on **Experiment Settings** and explore some of the customizable features of an AutoAI experiment.

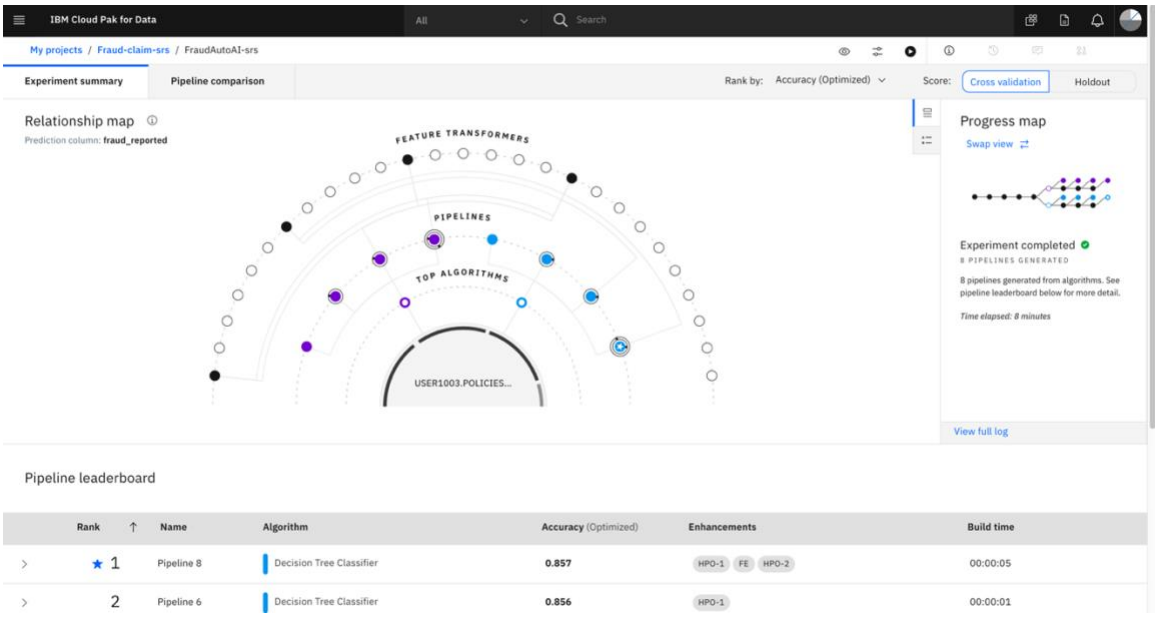


7. Click **Run experiment**.

8. The AutoAI experiment will run and the UI will show progress as it happens.



9. The experiment can take several minutes to run. Upon completion, you will see a message that the pipelines have been created.



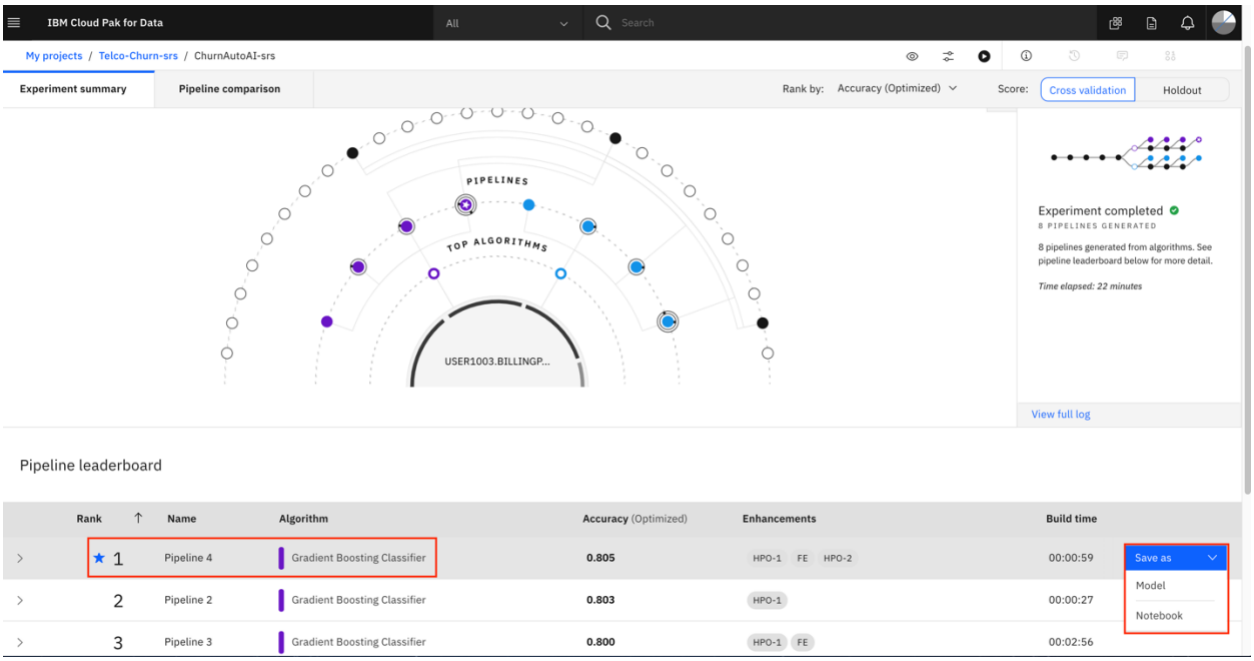
Step 2. Save AutoAI model

The AutoAI process by default selects the top-two performing algorithms for a given dataset. After executing the appropriate data pre-processing steps, it follows this sequence for each of the algorithms to build candidate pipelines:

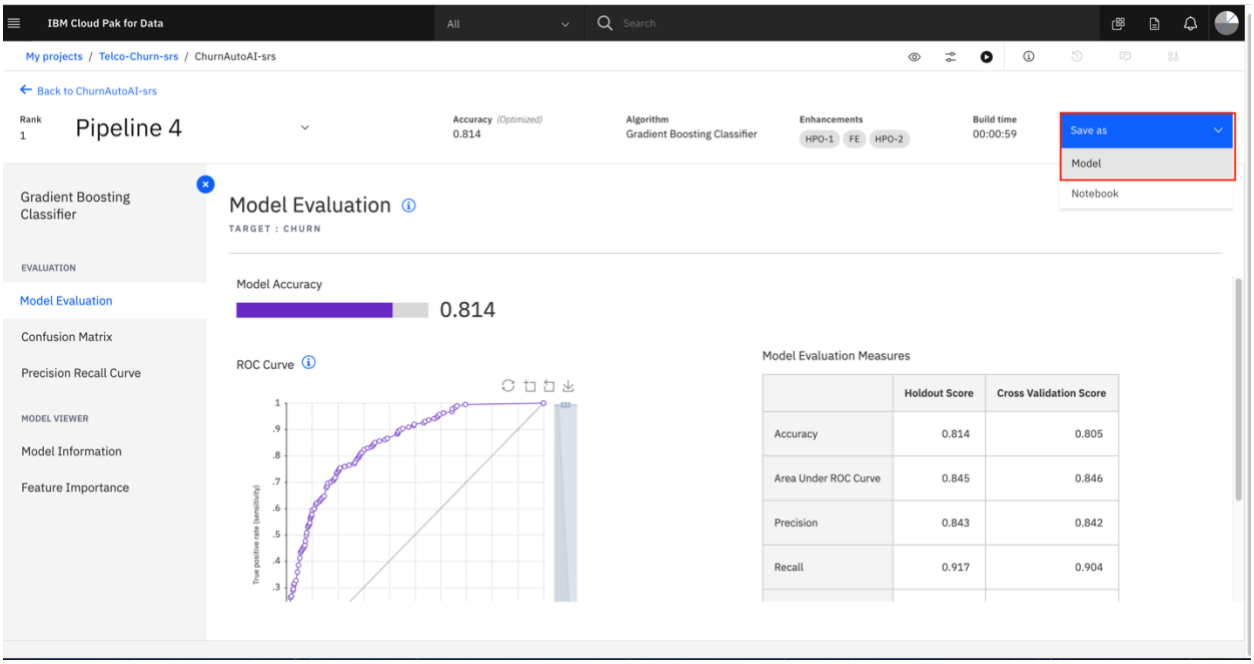
- Automated model selection
- Hyperparameter optimization
- Automated feature engineering
- Hyperparameter optimization

You can review each pipeline and select to deploy the top-performing pipeline from this experiment.

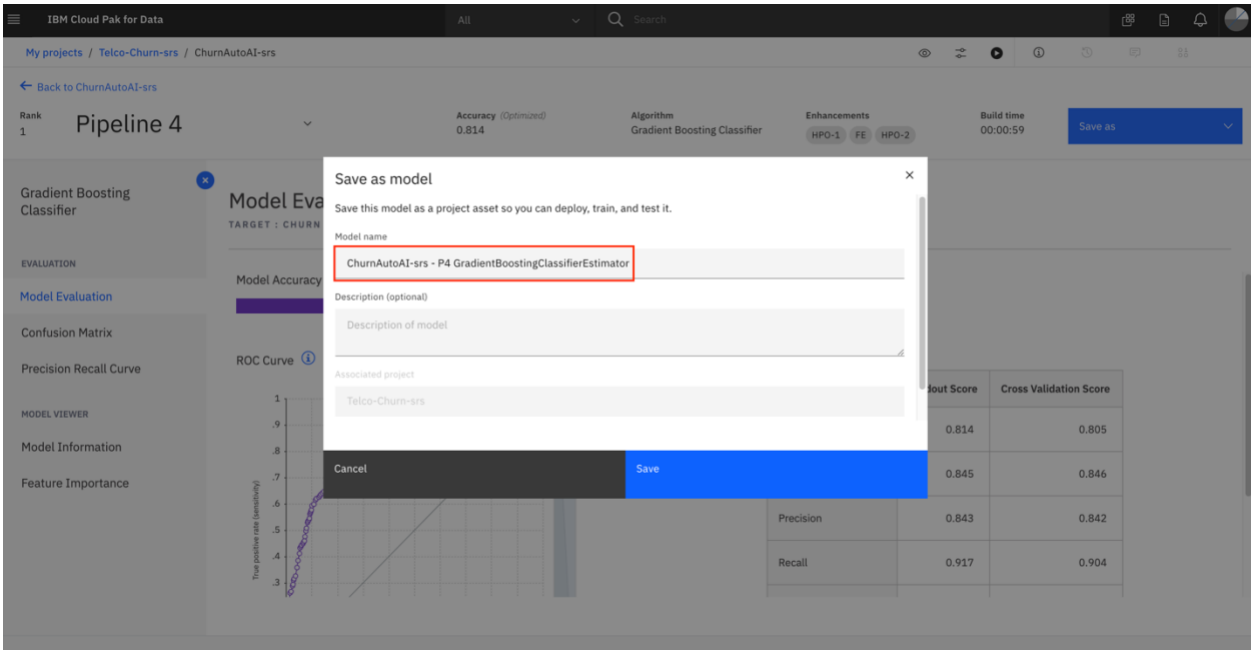
1. Scroll down to see the Pipeline leaderboard. The top-performing pipeline is in the first rank.
2. The next step is to select the model that gives the best result by looking at the metrics. In this case, Pipeline 8 gave the best result with the metric “Accuracy (optimized)”. You can view the detailed results by selecting the corresponding pipeline from the leaderboard.



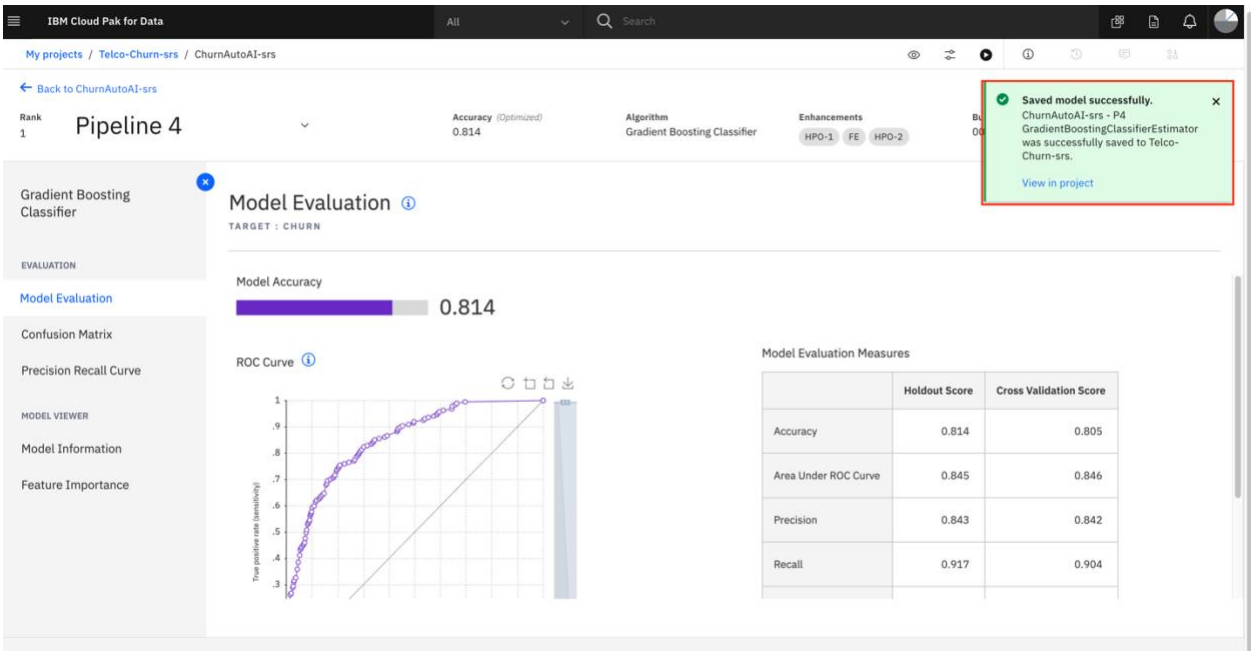
3. The model evaluation page displays metrics for the experiment, feature transformations performed (if any), which features contribute to the model, and more details about the pipeline.



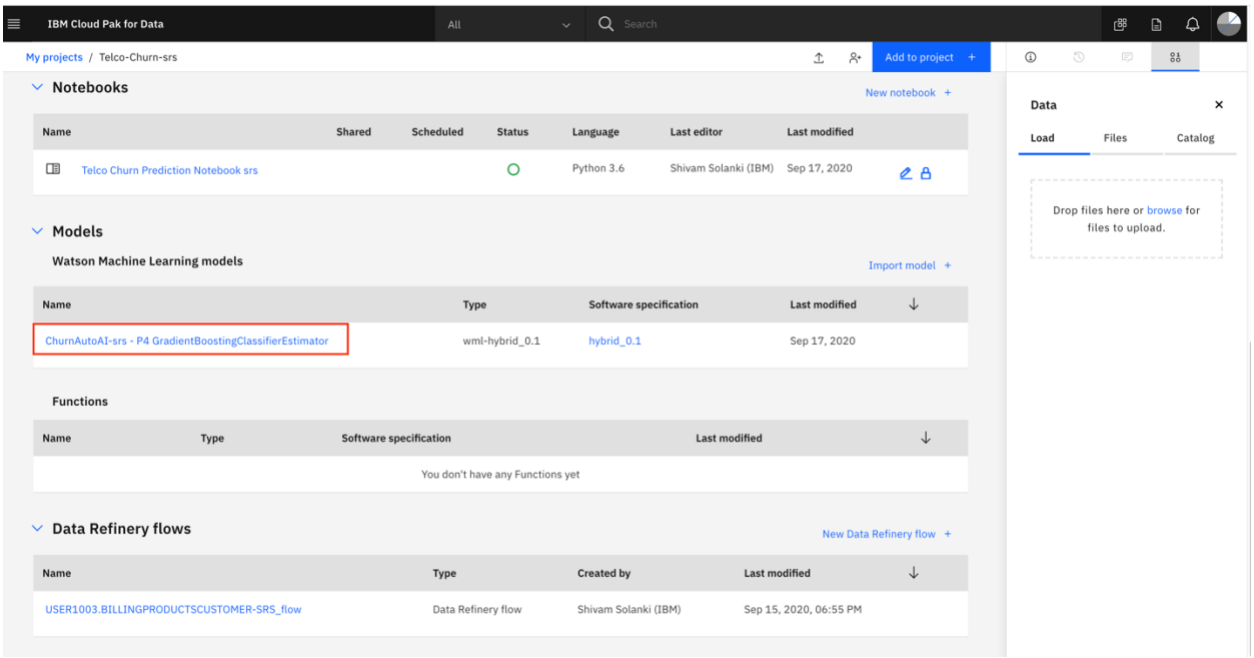
4. To deploy this model, click **Save as** → **Model** to save it.
5. A window opens that asks for the model name, description (optional), etc. You can accept the defaults or give the model a meaningful name/description. Click **Save**.



6. A notification indicates that the model is saved to the project. Go back to the project's main page by clicking on the project name on the navigator on the top left.

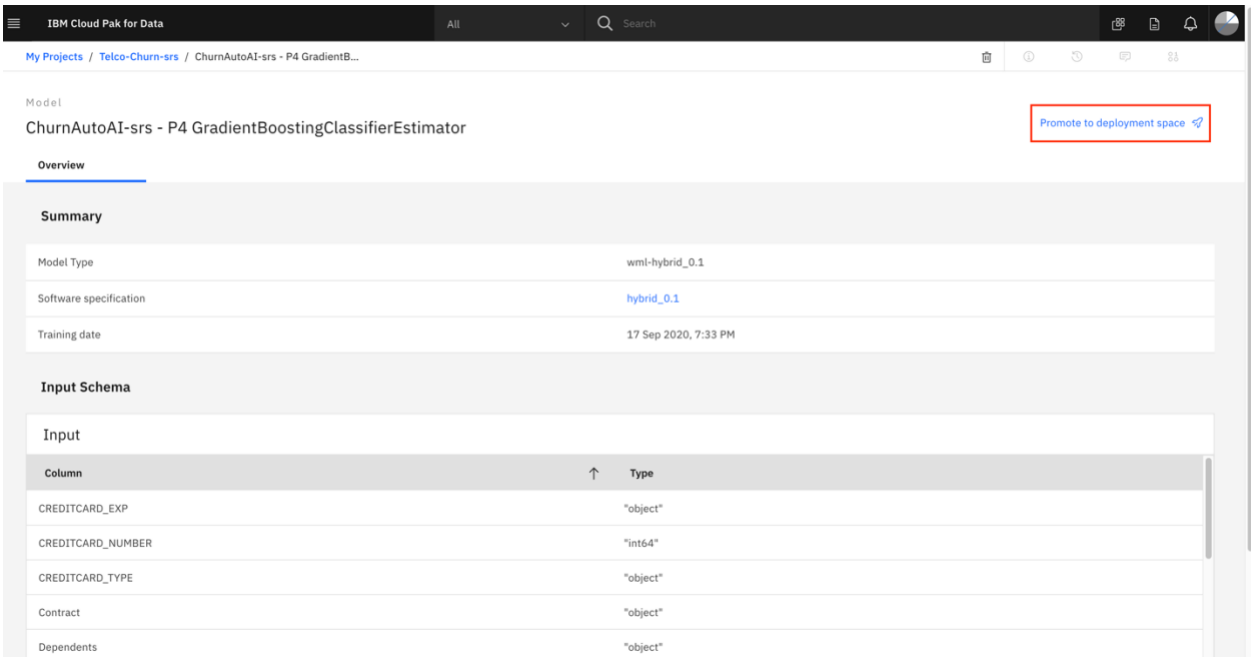


The new model is listed under the Models section of the Assets page.

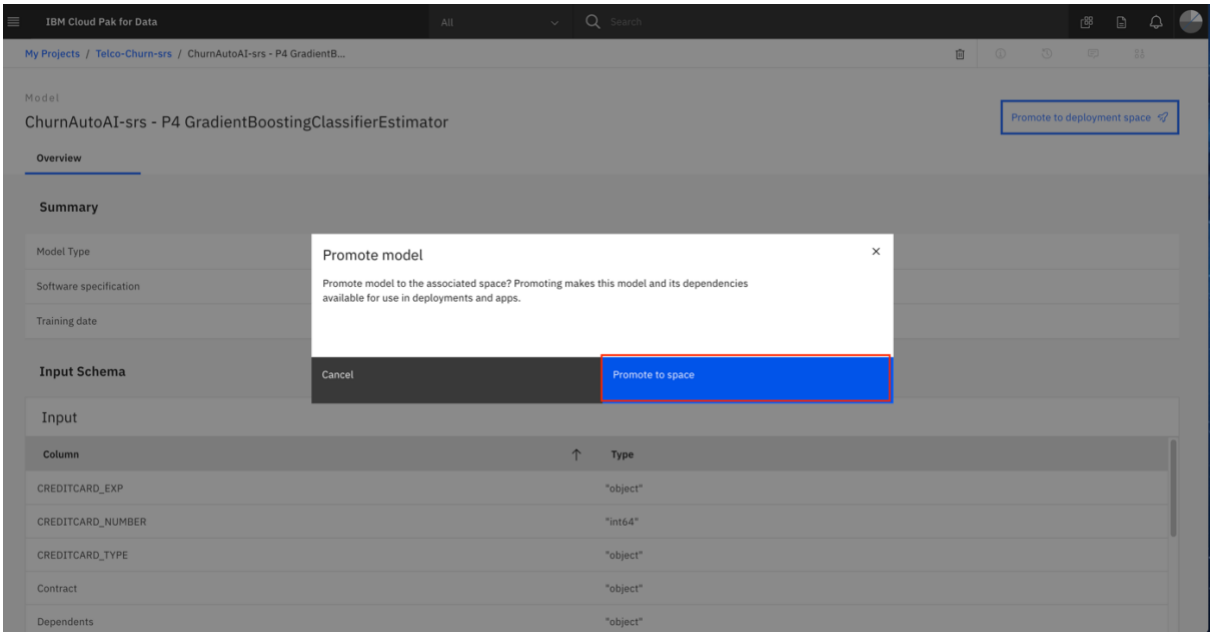


Step 3. Deploy the model

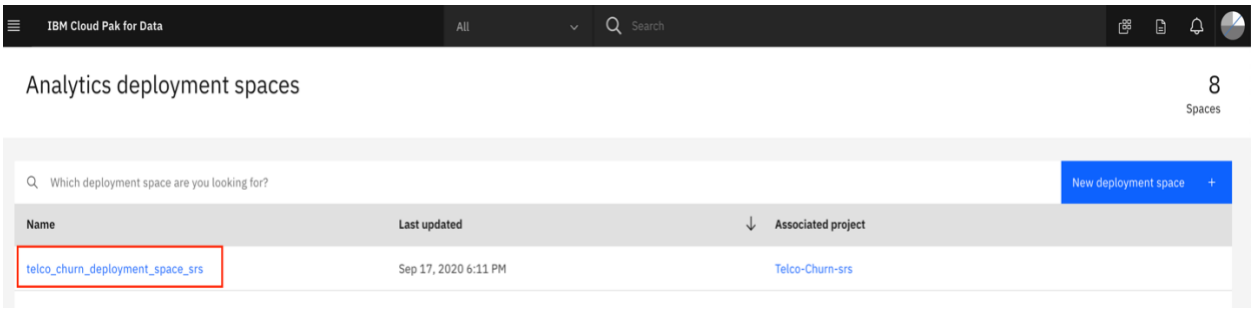
1. Under the Models section of the Assets page, click the name of the saved model.
2. To make the model available to be deployed, we first need to make it available in the deployment space. Click on **Promote to deployment space**.



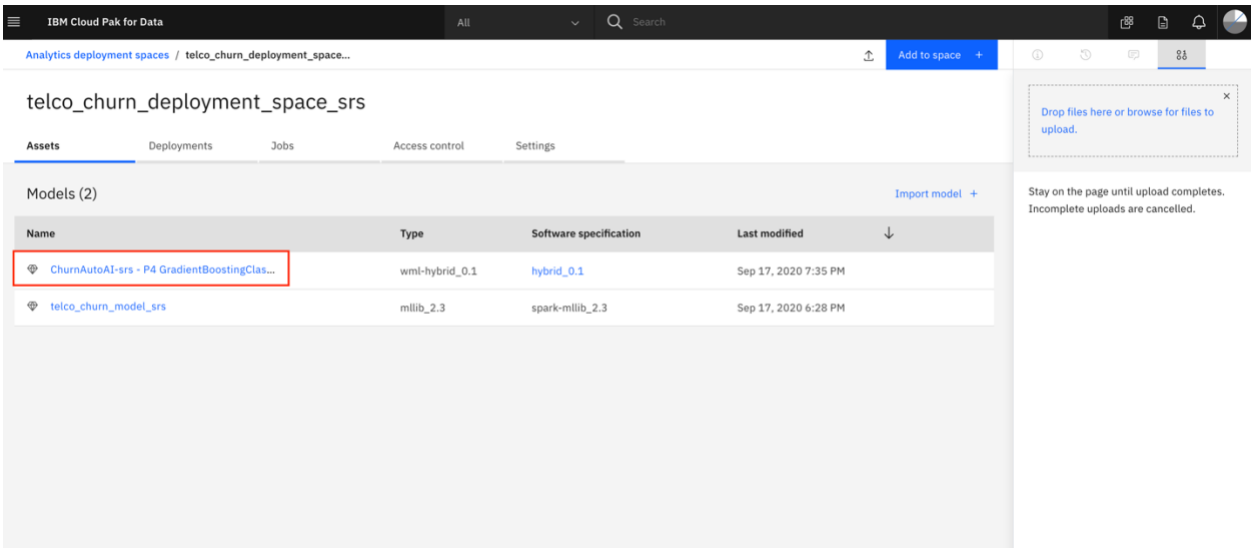
3. Since we have already associated a deployment space in the previous lab, we can promote this AutoAI model to that deployment space.



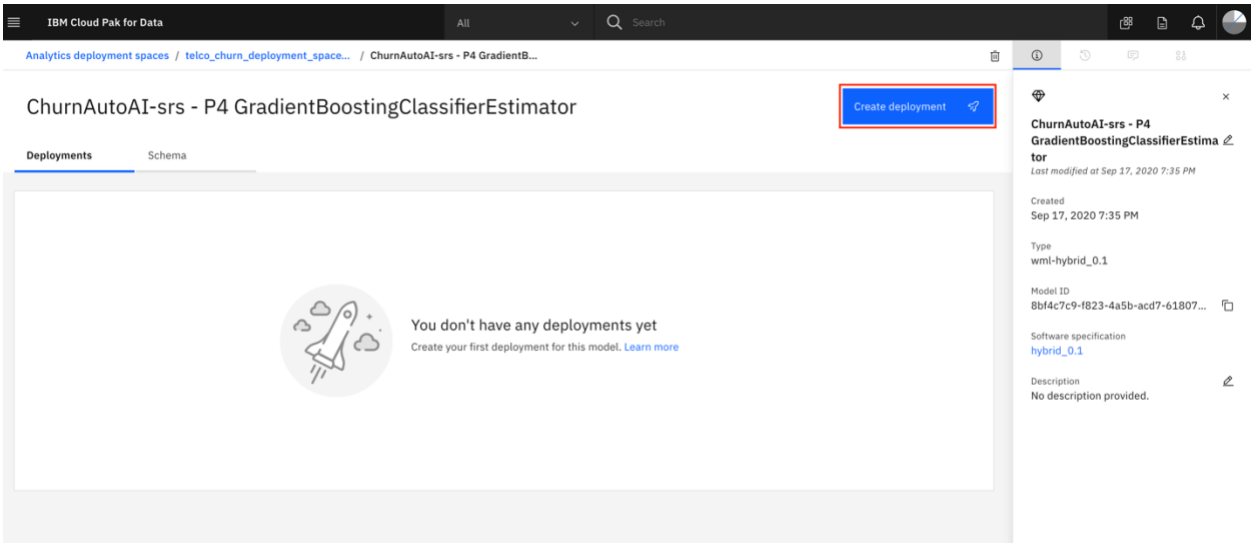
4. You will see a notification that the model was promoted to the deployment space successfully. Click **Deployment space** from this notification. You can also reach this page by using the hamburger (☰) menu and selecting **Analyze** → **Analytics deployments**. Select the deployment space.



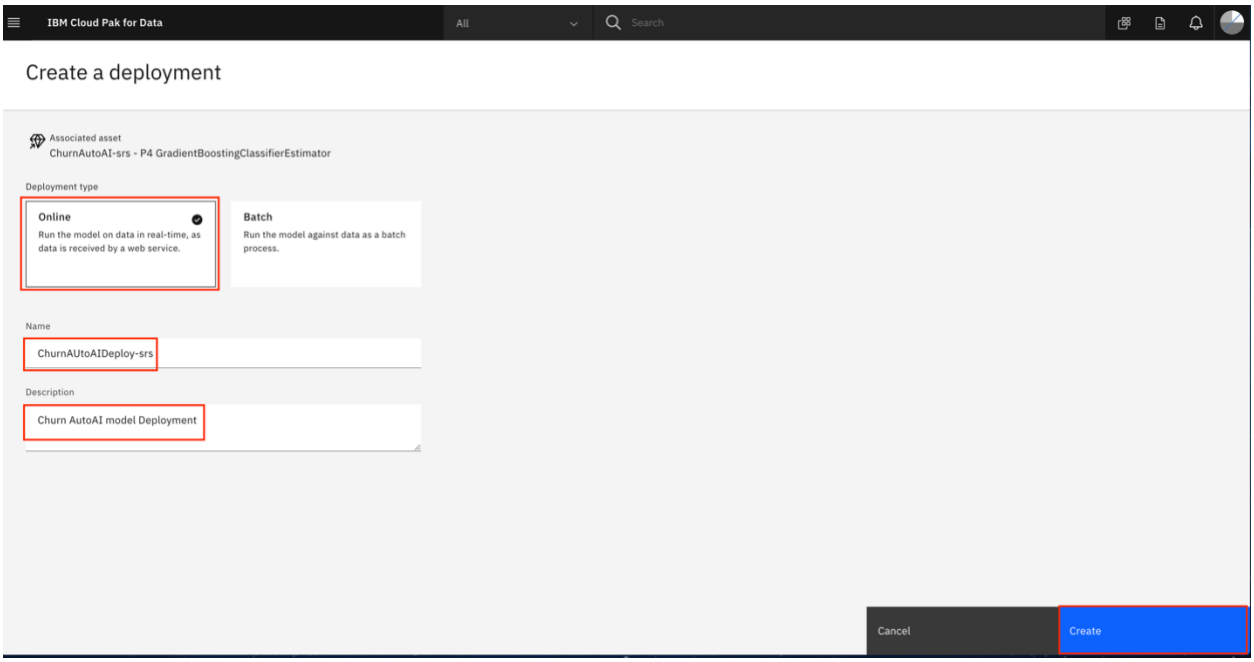
5. Under the Assets tab, click on the AutoAI model you just promoted.



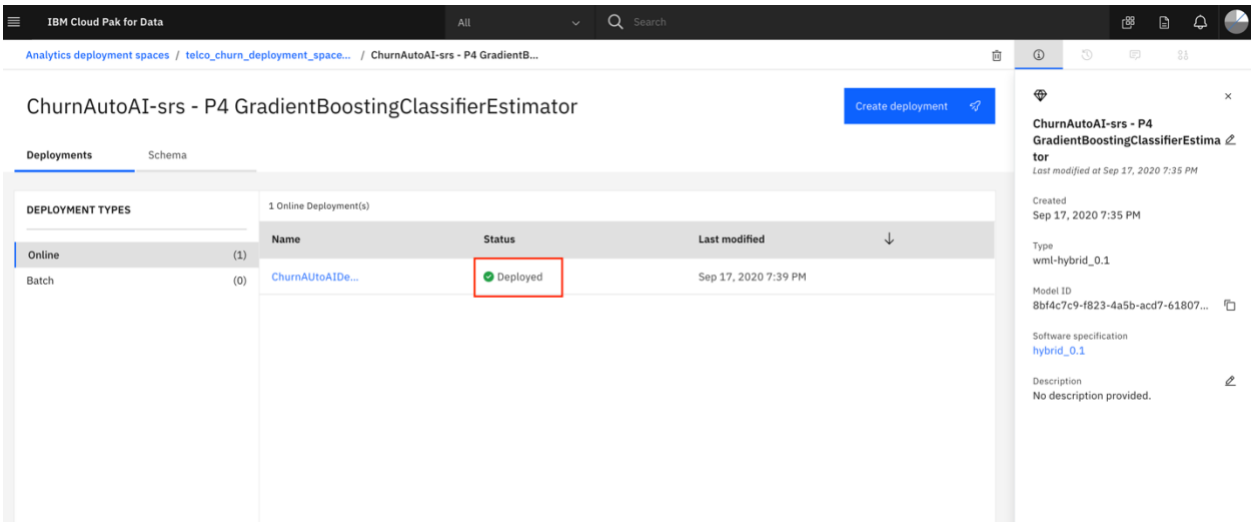
6. Click **Create deployment** in the top-right corner.



7. On the Create a deployment screen, choose **Online** for the deployment type, give the deployment a name and an optional description. Click **Create**.



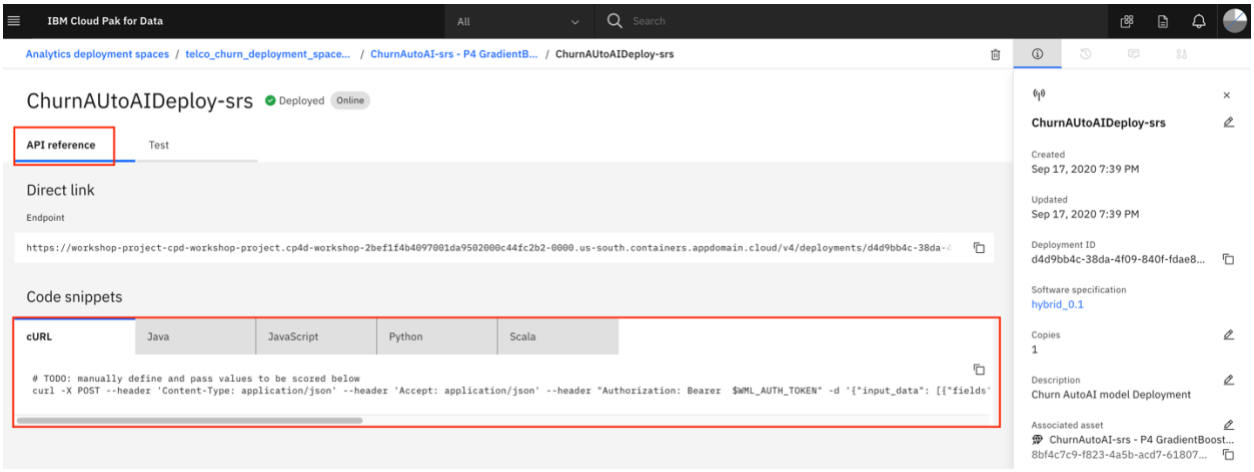
8. The deployment status will show as “In progress” and change to “Deployed” when finished.



Step 4. Test the model

IBM Cloud Pak for Data offers tools to quickly test out Watson machine learning models. We begin with the built-in tooling.

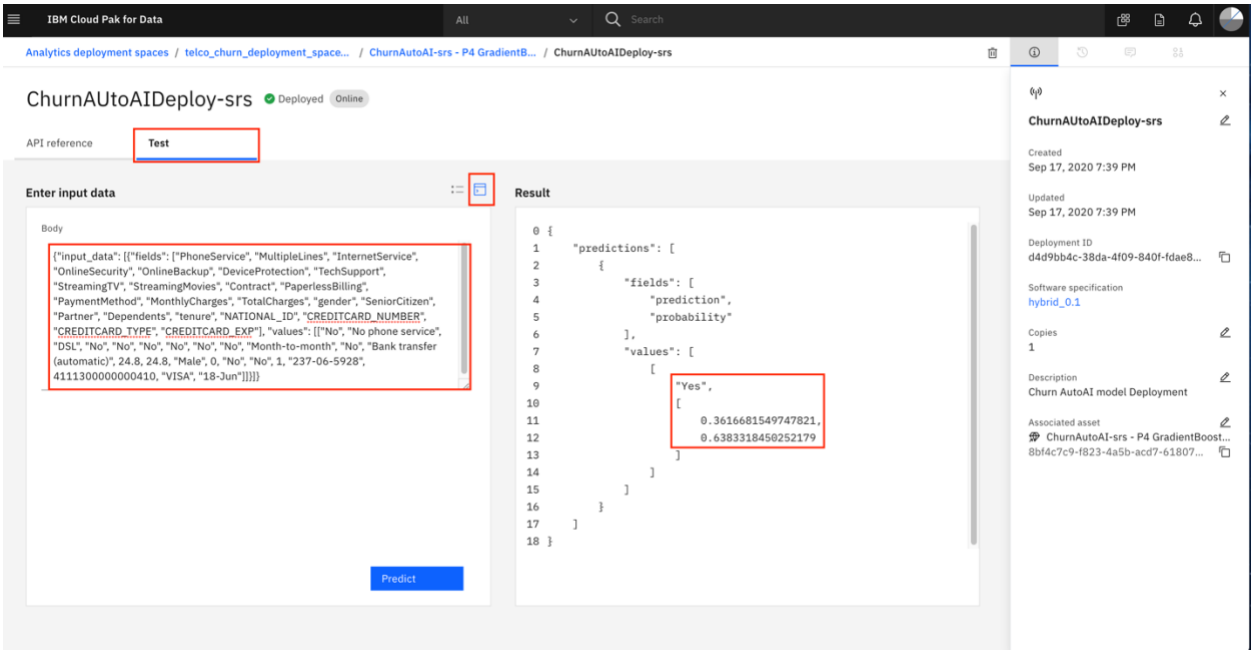
1. Click on the deployment. The deployment API reference tab shows how to use the model using cURL, Java, JavaScript, Python, and Scala. Click on the corresponding tabs to get the code snippet in the language you want to use.



2. To get to the built-in test tool, click the **Test** tab, then click on the **Provide input data as JSON** icon and paste the following data under Body:

{"input_data": [{"fields": ["PhoneService", "MultipleLines", "InternetService", "OnlineSecurity", "OnlineBackup", "DeviceProtection", "TechSupport", "StreamingTV", "StreamingMovies", "Contract", "PaperlessBilling", "PaymentMethod", "MonthlyCharges", "TotalCharges", "gender", "SeniorCitizen", "Partner", "Dependents", "tenure", "NATIONAL_ID", "CREDITCARD_NUMBER", "CREDITCARD_TYPE", "CREDITCARD_EXP"], "values": [["No", "No phone service", "DSL", "No", "No", "No", "No", "No", "No", "Month-to-month", "No", "Bank transfer (automatic)", 24.8, 24.8, "Male", 0, "No", "No", 1, "237-06-5928", 4111300000000410, "VISA", "18-Jun"]]}]}

3. Click the **Predict** button and the model will be called with the input data. The results will display in the Result window. You can see the prediction result ("Yes" or a "No" for Customer Churn).



For this example, it has been predicted as Chrun (Yes) with a probability of around 63%.

Summary

This lab tutorial showed you how to setup your AutoAI environment and generate pipeline. You have also learnt how to save the AutoAI model, Deploy and test the model and everything can be done without a single line of code. That's awesome!