

AutoAI Lab

Introduction

This lab will demonstrate the award-winning AutoAI capability to build and deploy an optimized model based on the Titanic data set.

AutoAI in Cloud Pak for Data automatically analyzes your data and generates candidate model pipelines customized for your predictive modeling problem. AutoAI algorithms analyze your dataset to discover data transformations, estimator algorithms, and parameter settings that work best for your problem setting. Results are displayed on a leaderboard, showing the automatically generated model pipelines ranked according to your optimization objective.

Using AutoAI, you can build and deploy a machine learning model with sophisticated training features and no coding. The tool does most of the work for you.

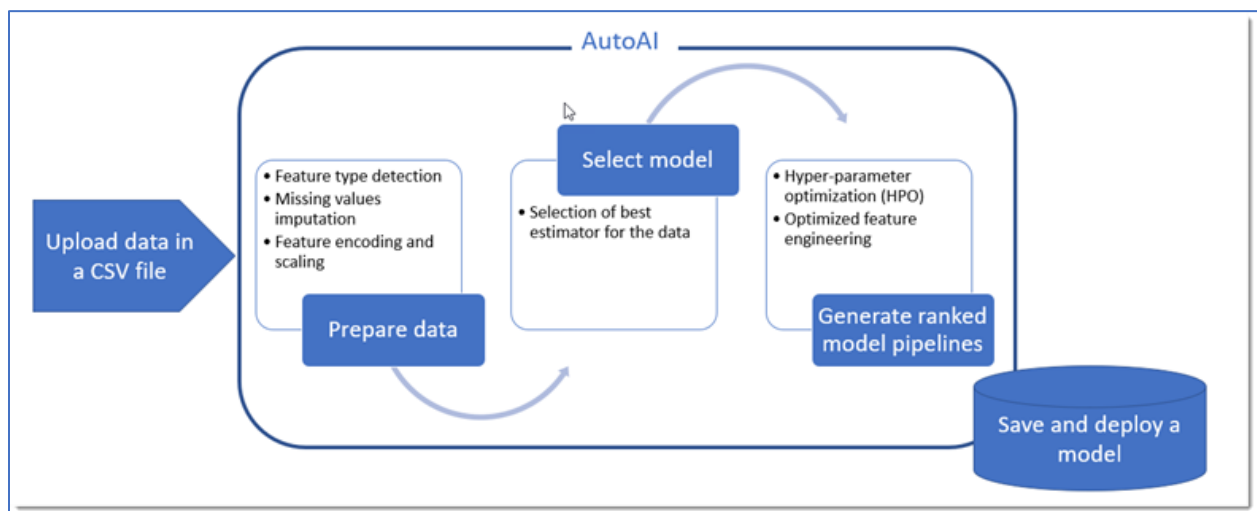


Figure 1- Auto AI Flow

The AutoAI process follows this sequence to build candidate pipelines:

- **Data pre-processing** - Most data sets contain different data formats and missing values, but standard machine learning algorithms work with numbers and no missing values. AutoAI applies various algorithms, or estimators, to analyze, clean, and prepare your raw data for machine learning. It automatically detects and categorizes features based on data type, such as categorical or numerical. Depending on the categorization, it uses hyper-parameter optimization to determine the best combination of strategies for missing value imputation, feature encoding, and feature scaling for your data.
- **Automated model selection** - The next step is automated model selection that matches your data. AutoAI uses a novel approach that enables testing and ranking candidate algorithms against small subsets of the data, gradually increasing the size of the subset

for the most promising algorithms to arrive at the best match. This approach saves time without sacrificing performance. It enables ranking a large number of candidate algorithms and selecting the best match for the data.

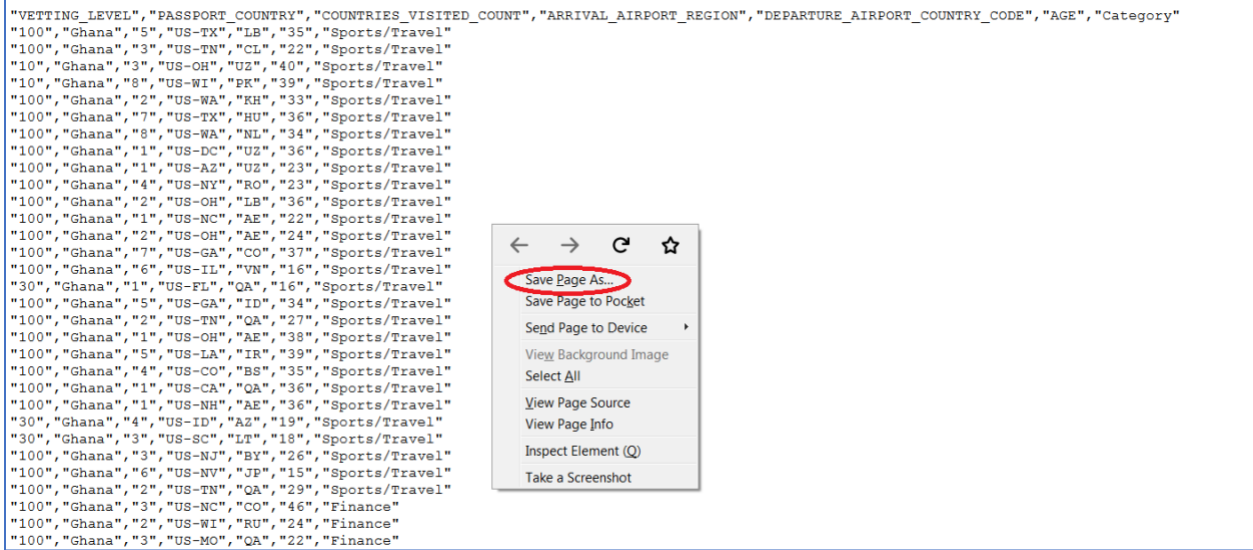
- **Hyperparameter optimization** - Hyper-parameter optimization refines the best performing model pipelines. AutoAI uses a novel hyper-parameter optimization algorithm optimized for costly function evaluations such as model training and scoring that are typical in machine learning. This approach enables fast convergence to a good solution despite long evaluation times of each iteration.
- **Automated feature engineering** - Feature engineering attempts to transform the raw data into the combination of features that best represents the problem to achieve the most accurate prediction. AutoAI uses a unique approach that explores various feature construction choices in a structured, non-exhaustive manner, while progressively maximizing model accuracy using reinforcement learning. This results in an optimized sequence of transformations for the data that best match the algorithms of the model selection step.
- **Repeat Hyperparameter optimization** – The Hyperparameter optimization step is repeated including the derived features from the feature engineering step.

We will perform the following steps in this lab:

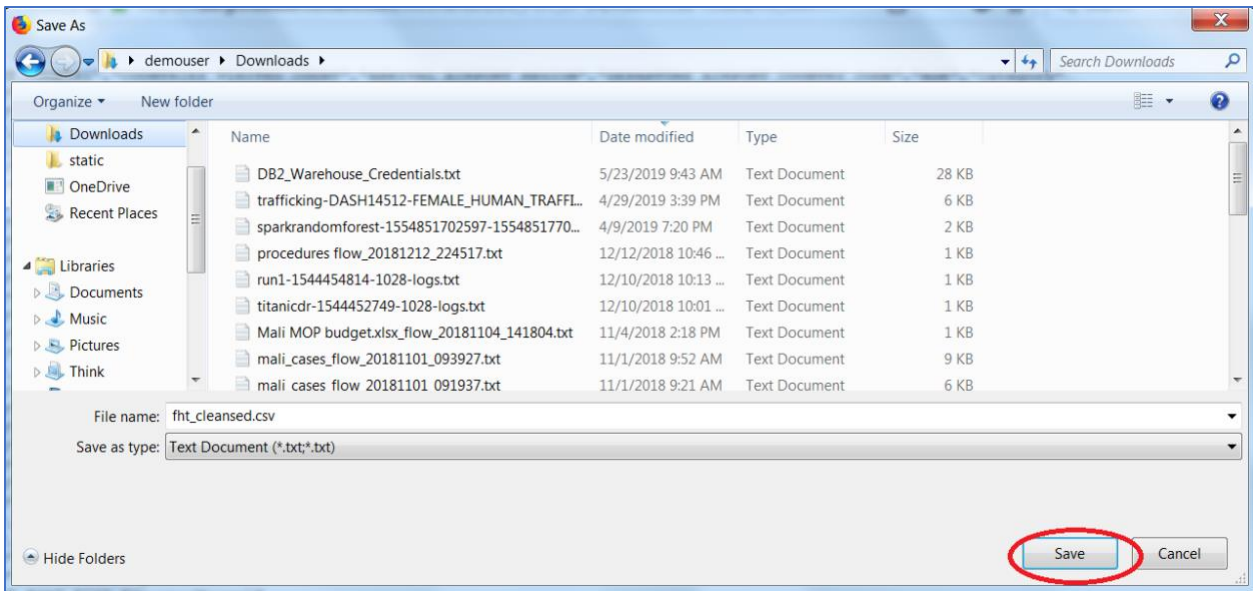
1. Download a female human trafficking cleansed dataset from the github repo
2. Add an Auto AI Experiment to create a model to predict the trafficking risk
3. Save and Deploy the model
4. Test the model

Step 1: Download a female human trafficking cleansed dataset

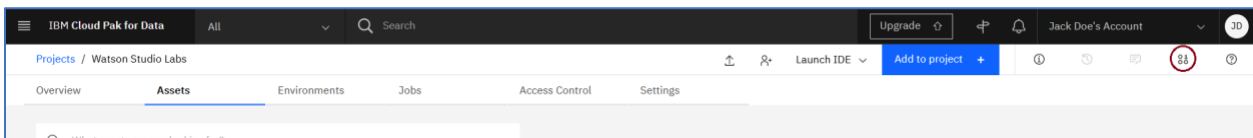
1. Download the fht_cleansed.csv data file from the following location by clicking [here](#)
2. Right-click on the window, and click **Save Page As...**



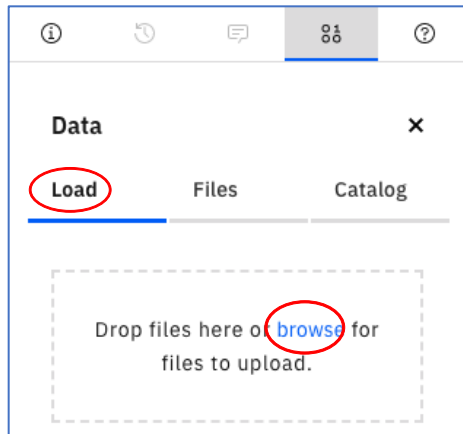
3. Click on **Save**.



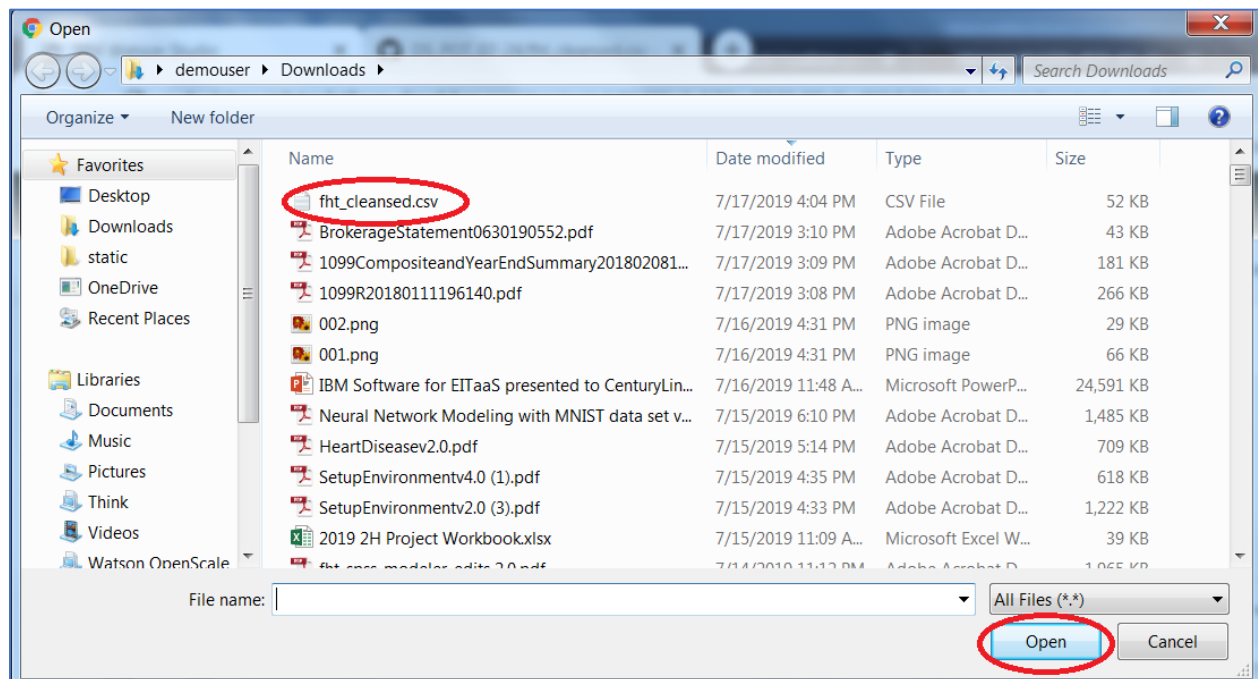
4. Go back to your project. Click on the icon.



5. Click on the **Load** tab and then click on **browse**. If you don't see the **Load** tab, click on the  icon again.



6. Go to the folder where the fht_cleansed.csv file is stored. Select the fht_cleansed.csv file and then click **Open**.

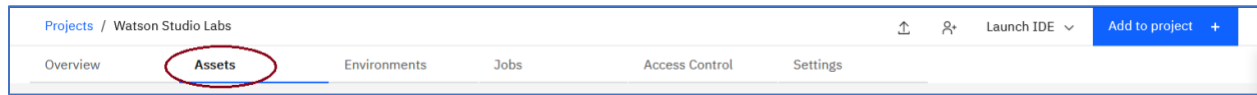


7. The fht_cleansed.csv file is now added as a Data Asset.

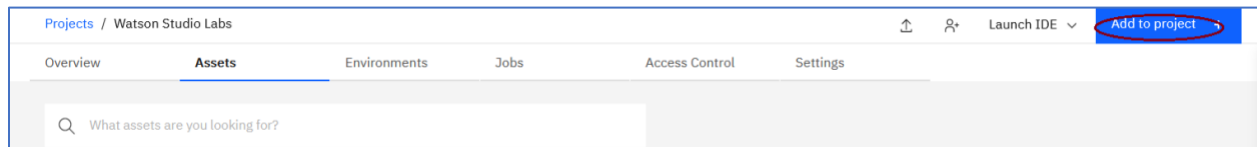
Data assets				
0 assets selected.				
<input type="checkbox"/>	Name	Type	Created by	Last modified
<input type="checkbox"/>	CSV fht_cleansed.csv	Data Asset	FCOT Labs	Aug 03, 2020, 10:18 AM

Step 2: Add an AutoAI Experiment.

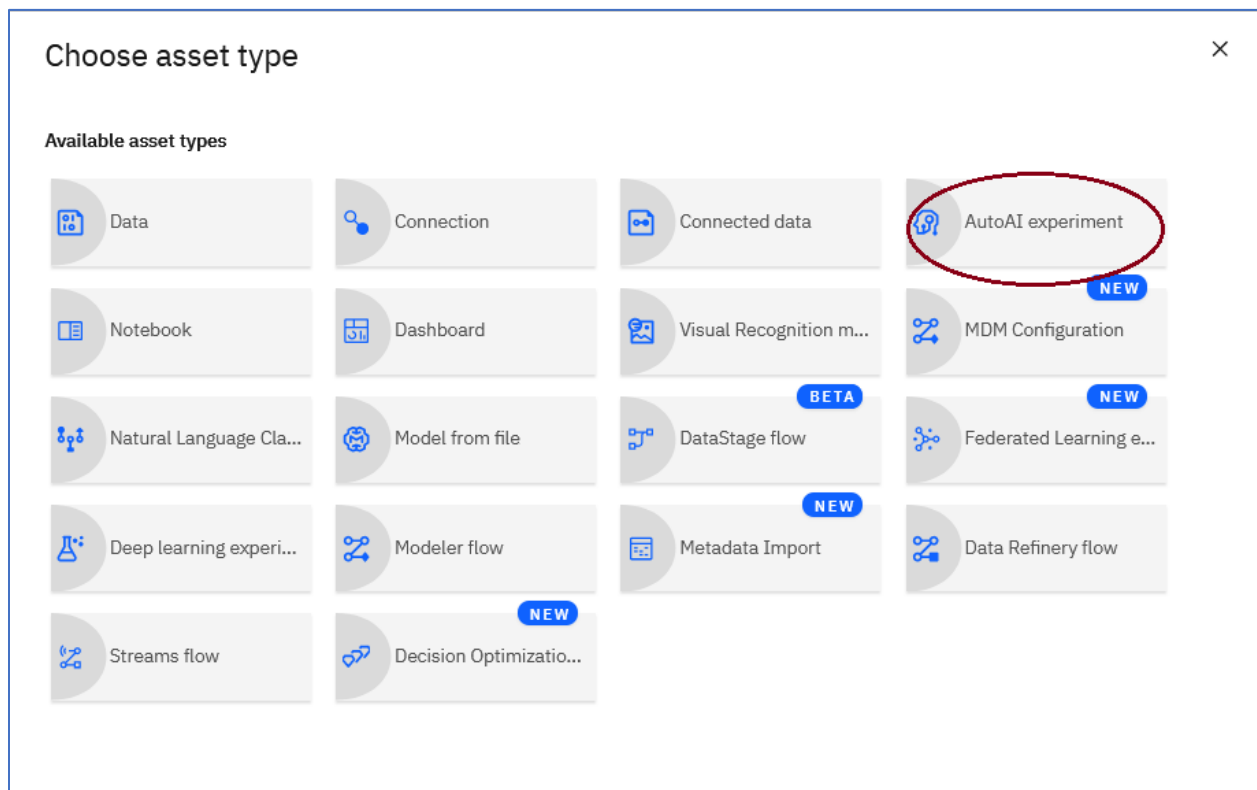
1. If not on the **Assets** page, click on the **Assets** Tab



2. Click on **Add to project**.



3. Click on **AutoAI Experiment**.



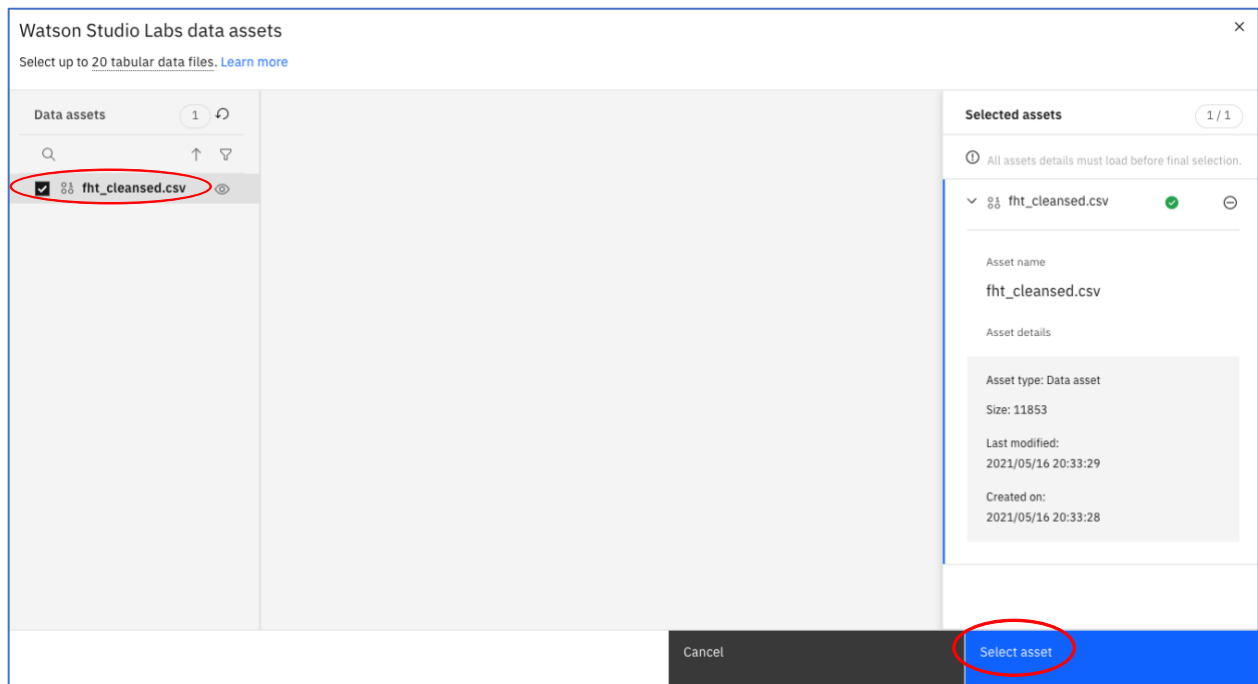
4. Enter an **Asset name**, leave the defaults for the **Watson Machine Learning** and **Compute configuration** and click on **Create**.

The screenshot shows the 'Define details' form in the Watson Machine Learning console. On the left, a sidebar contains a '+ New' button and a 'Gallery sample' link. The main form is divided into two sections: 'Define details' and 'Associate services'. In the 'Define details' section, the 'Name' field is populated with 'FHT AutoAI' and is circled in red. The 'Description' field contains the text 'Description of AutoAI experiment'. In the 'Associate services' section, the 'Watson Machine Learning Service Instance' is set to 'WatsonMachineLearning' and the 'Compute configuration' is set to '8 vCPU and 32 GB RAM'. At the bottom right, there are 'Cancel' and 'Create' buttons, with the 'Create' button circled in red.

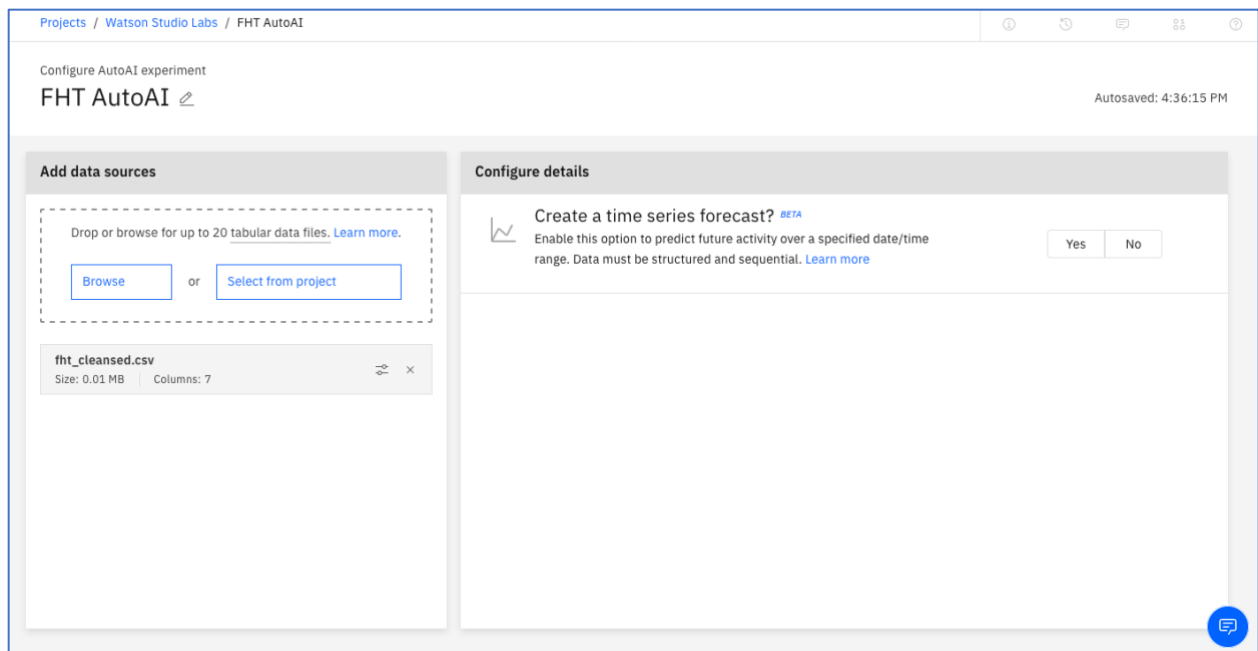
5. Click on **Select from project**.

The screenshot shows the 'Add data source' dialog. At the top, the title 'FHT AutoAI' is followed by an edit icon. Below the title is a section labeled 'Add data source'. The main area of the dialog is a large dashed box. At the bottom, there is an icon of a document with a plus sign. To the right of the icon, the text 'Add data source' is followed by the instruction 'Drop or browse for a csv file.' Below this, there are two buttons: 'Browse' and 'Select from project'. The 'Select from project' button is circled in red.

6. Select the **fht_cleansed.csv** dataset and click on **Select asset**.



7. **Click No** next to Create a time series forecast?



8. Select **VETTING_LEVEL** as the column to predict, leave the default for **OPTIMIZED METRIC** and click on **Run Experiment**. Note the system scanned the **VETTING_LEVEL** values to determine that a **Multiclass Classification** was the **PREDICTION TYPE**.

Configure details

Create a time series forecast? [BETA](#)
Enable this option to predict future activity over a specified date/time range. Data must be structured and sequential. [Learn more](#)

What do you want to predict?
Prediction column ⓘ
VETTING_LEVEL x v

Prediction column: VETTING_LEVEL CUH remaining: 20 CUH

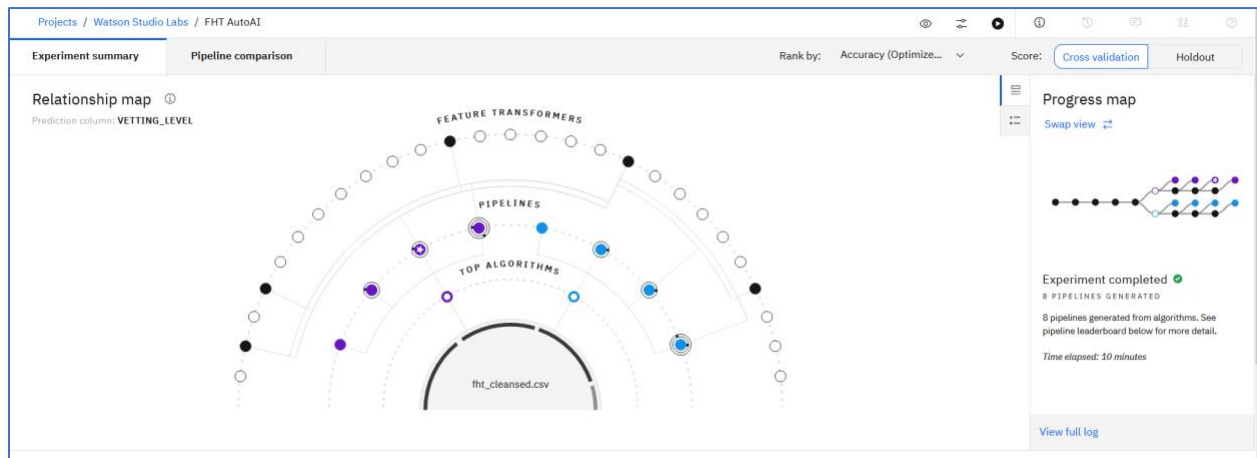
PREDICTION TYPE
Multiclass Classification



OPTIMIZED FOR
Accuracy & run time

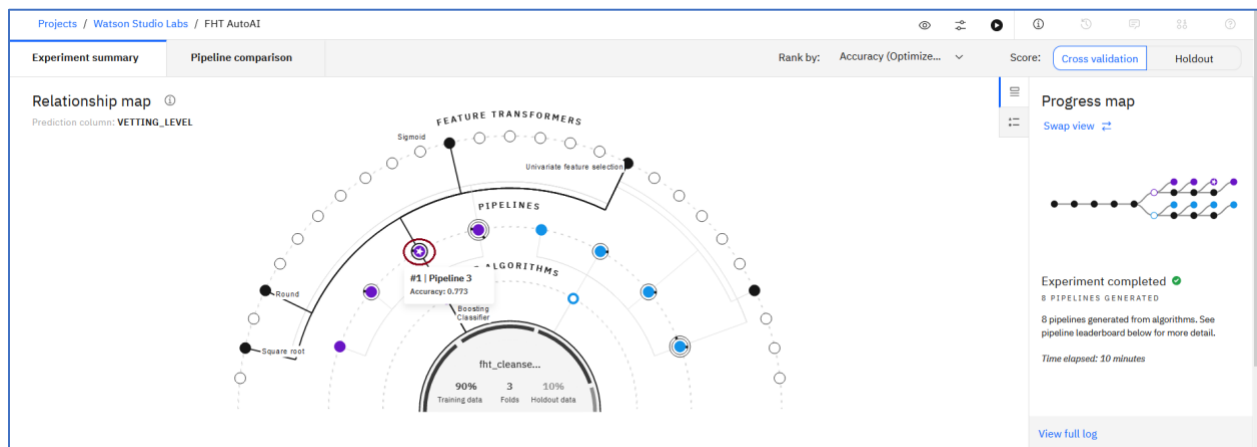
Experiment settings **Run experiment**

9. It will take several minutes for the eight alternative pipelines to be analyzed. The first pipeline picks the best algorithm. In this case, it is a **Gradient Boosting** classifier. The second pipeline performs a hyperparameter optimization to see if tuning the algorithm parameters will improve the performance metric. The third pipeline will derive new features (i.e. feature engineering) to try to improve the performance metric. The fourth pipeline will do another hyperparameter optimization including the newly derived features. The next 4 pipelines do the same thing for the second-best algorithm. Note, you can move ahead at any point after the first pipeline has been created. We are going to proceed as if Pipeline 3 is the best ranked pipeline. Yours could be different.

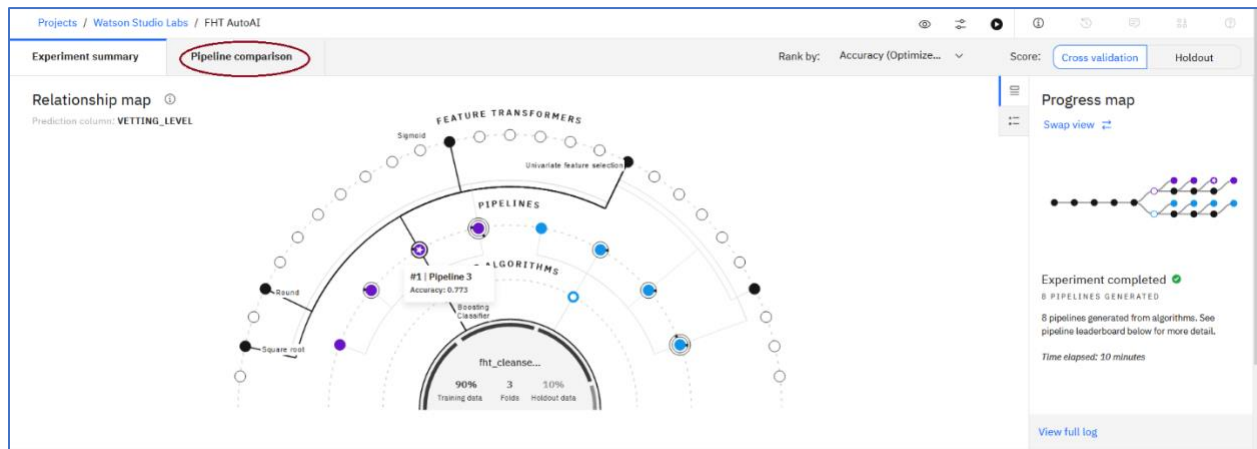
10. When the AutoAI run is completed you will see a display similar to below.



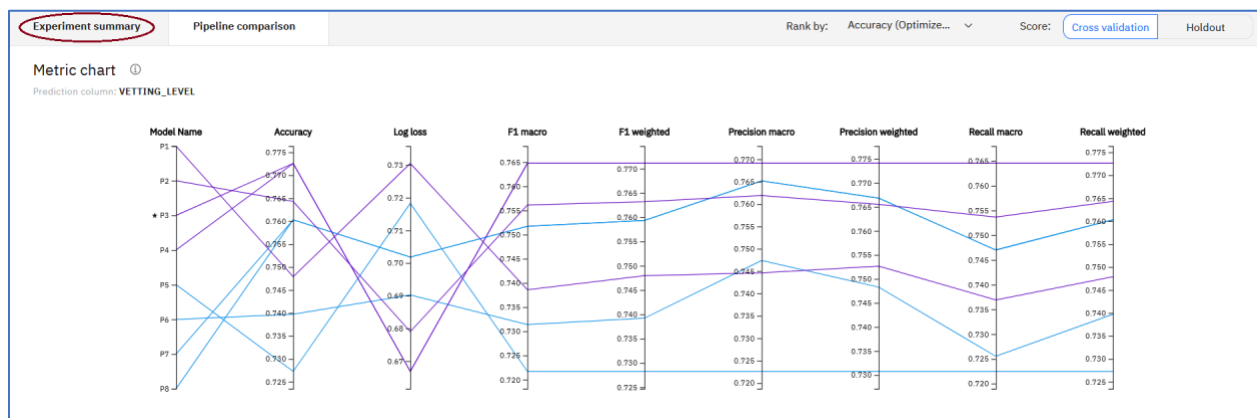
11. Hover over the  icon in the Pipelines semi-circle. This is the top ranked pipeline based on the accuracy metric. It is shown to be Pipeline 3 with a Gradient Boosted classifier and an accuracy of .77. Note, if you click on the  icon, you will be taken to the metrics for Pipeline 3 (don't do this yet).




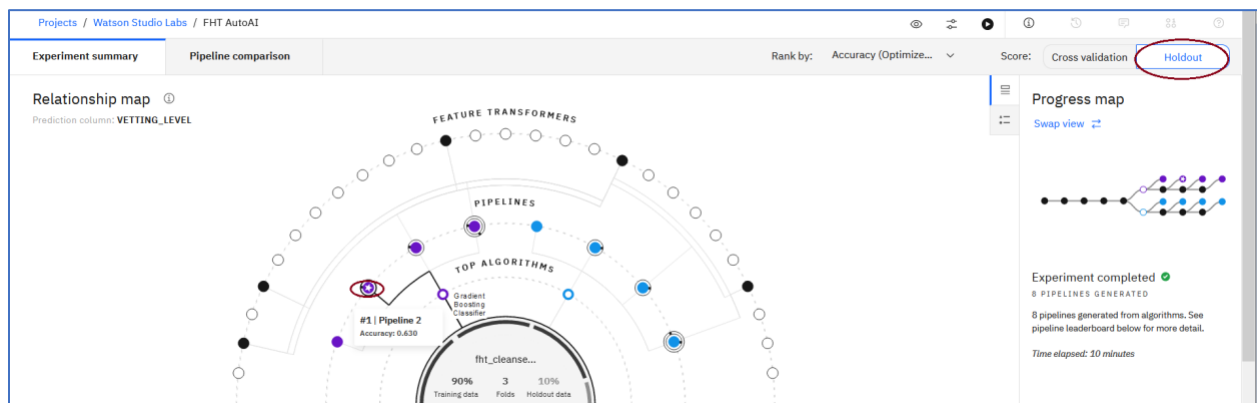
12. Click on the **Pipeline comparison**.



13. This will display a diagram showing the metrics associated with each pipeline on a single chart. Click on **Experiment summary** to return to the prior panel.



14. Click on **Holdout**. Note that the  icon may have shifted.



15. Click on the **Cross validation** tab and then scroll down. The Pipeline leaderboard is displayed with the pipelines ranked by accuracy.

Experiment summary

Pipeline comparison


Rank by: Accuracy (Optimize... ▾)

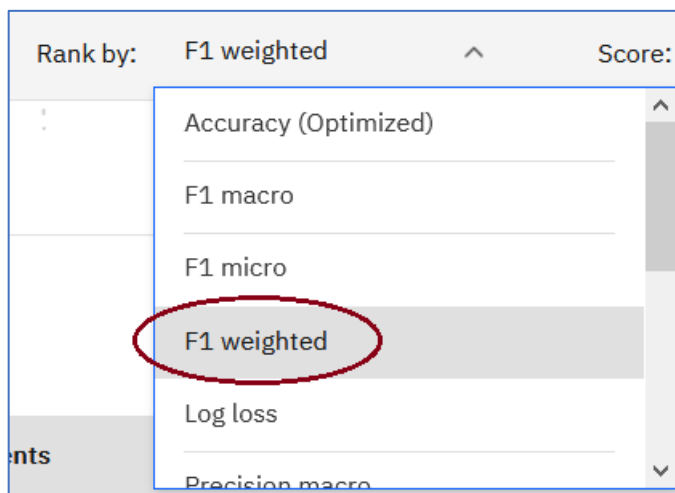
Score:

Cross validation

Holdout

</

16. Click on the  icon adjacent to Accuracy and click F1 Weighted



17. The list is now ranked by the F1 Weighted metric.

Experiment summary


Pipeline comparison

Rank by: F1 weighted

Score:

Cross validation

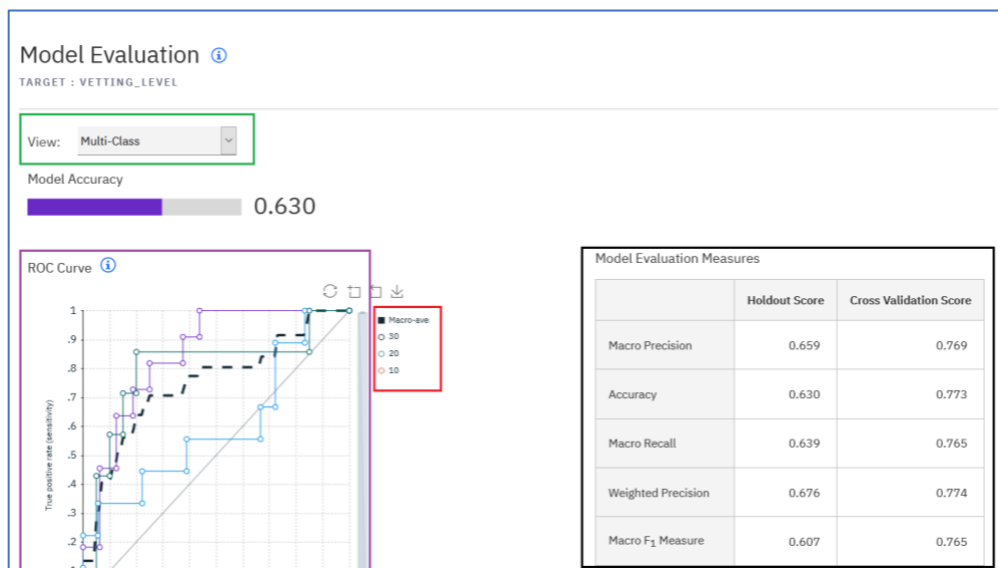
<

18. Return to the Accuracy ranked list by clicking on the  and selecting **Accuracy (Optimized)**. Click on the highest ranked pipeline (**pipeline 3 in the picture**).

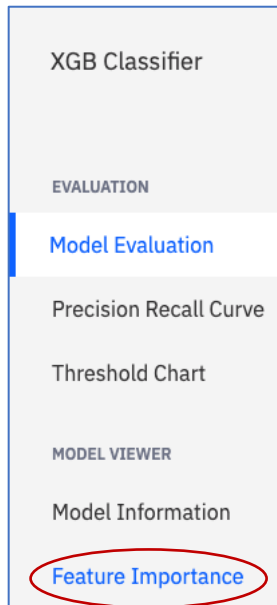
Experiment summary

Pipeline comparison

19. The **Model Evaluation** panel is displayed. The model evaluation metric for the holdout sample is displayed. Highlighted in blue rectangle below are links to additional model information and metrics. Highlighted in the green rectangle is a drop down containing the options multi-class, and then options for 3 binary metrics (30 versus rest, 20 versus rest, and 10 versus rest). Highlighted in the purple rectangle is the Receiver Operating Characteristic (ROC) curve. This plots the true positive rate versus the false positive rate at varying thresholds. Highlighted in the red rectangle is a clickable legend. Clicking on the circles will toggle the corresponding display on/off. Finally, highlighted in the black rectangle are model metrics for both the holdout sample, and the cross validation sample.



20. Click on **Feature Importance**.



21. According to the **Feature Importance**, the **COUNTRIES_VISITED_COUNT** feature is the most important.

Step 3: Save and Deploy the Model

In this section, we will save and then deploy the model. You create a deployment for a machine learning model so you can submit new data to a model and get a score, or prediction back. To deploy the model, you must have a *deployment space* where you can organize the assets you need to create and monitor deployments. A space contains an overview of deployment status, the deployable assets, deployments, associated input and output data, and the associated environments.

To deploy a model, we first **promote** the model to a deployment space. When you promote a model to a space, components required for a successful deployment, such as a training library, model definition, other dependent assets, or environment definition are automatically promoted as well. After promoting a model and the data assets needed to a deployment space, you can create a deployment in the space.

You can create a deployment space, promote the model, and create a deployment through the user interface or programmatically.

1. **Click on Save as.**



- AutoAI can generate both a machine learning model as well as a Jupyter notebook. The Jupyter notebook can contain the source code to build the model, or the source code plus the deployment code. Feel free to experiment with saving the notebooks. For this lab, we will save the model. Make sure the Model is checked, optionally change the **Name**, and click **Create**.

The screenshot shows the 'Define details' form for creating a model asset. On the left, under 'Select asset type', the 'Model' option is selected and circled in red. The 'Model' description states: 'Create a Watson Machine Learning model asset that you can test with new data, deploy to generate predictions, and trace lineage activity.' Below it, the 'Notebook Beta' option is also visible. On the right, the 'Name' field is filled with 'FHT AutoAI - P3 GradientBoostingClassifierEstimator'. The 'Description (optional)' field is empty. The 'Tags' section has a dropdown menu labeled 'Add a tag'. At the bottom right, there are 'Cancel' and 'Create' buttons, with the 'Create' button circled in red.

- A message is displayed showing the model was successfully saved. Click on **View in project**.

The screenshot shows the model overview page for 'Pipeline 3'. The top bar displays 'Rank 1', 'Holdout Accuracy (Optimized) 0.630', 'Algorithm Gradient Boosting Classifier', 'Enhancements HPO-1 FE', and 'Build time 00:01:19'. A 'Save as' button is visible. Below the top bar, the 'Feature Transformations' section is expanded, showing 'TARGET : VETTING_LEVEL'. A green success message box is displayed on the right, stating: 'Saved model successfully. FHT AutoAI - P3 GradientBoostingClassifierEstimator was successfully saved to Watson Studio Labs.' The 'View in project' link in the message box is circled in red.

- As mentioned above, we will need to promote the model to a deployment space. Click on **Promote to deployment space**.

The screenshot shows the model overview page for 'FHT AutoAI - P3 GradientBoostingClassifierEstimator'. The 'Promote to deployment space' button is circled in red. Below the button, the 'Input Schema' section is visible, showing a table with columns 'Column' and 'Type'. The table lists the following input features and their types:

Column	Type
AGE	"integer"
ARRIVAL_STATE	"other"
COUNTRIES_VISITED_COUNT	"integer"
Category	"other"
DEPARTURE_AIRPORT_COUNTRY_CODE	"other"

5. A deployment space already exists due to the Watson OpenScale set up in Lab-1. We will create a new space for our model deployment. Click on **New space**.

Promote to space

Target space

Select or create a space ^

Create a new deployment space ⊕

openscale-express-path-15c8dbf8-a3e6-4da4-ac8d-d7b4180b6a12

6. Enter **AutoAI Deployment Space** for the Name, scroll down and select **WatsonMachineLearning** for the **Select machine learning service**, and click **Create**.

Create a deployment space

Use a space to collect assets in one place to create, run, and manage deployments

Name
AutoAI Deployment Space

Description (Optional)
Deployment space description

Deployment space tags (optional) ⓘ
Add a tag

Select storage service ⓘ
Cloud Object Storage-zj

Select machine learning service (optional) ⓘ
WatsonMachineLearning

Cancel Create

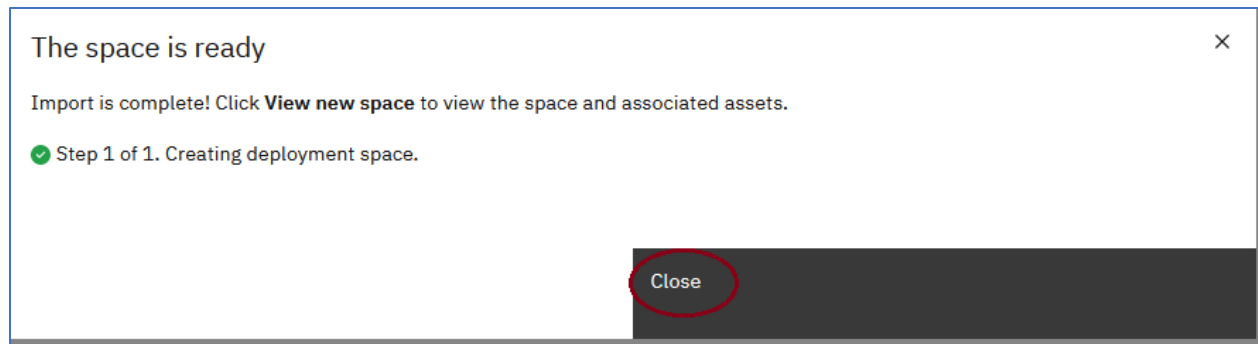
7. Two message boxes are displayed. Click **Close** in the second message box.

The space is being prepared...

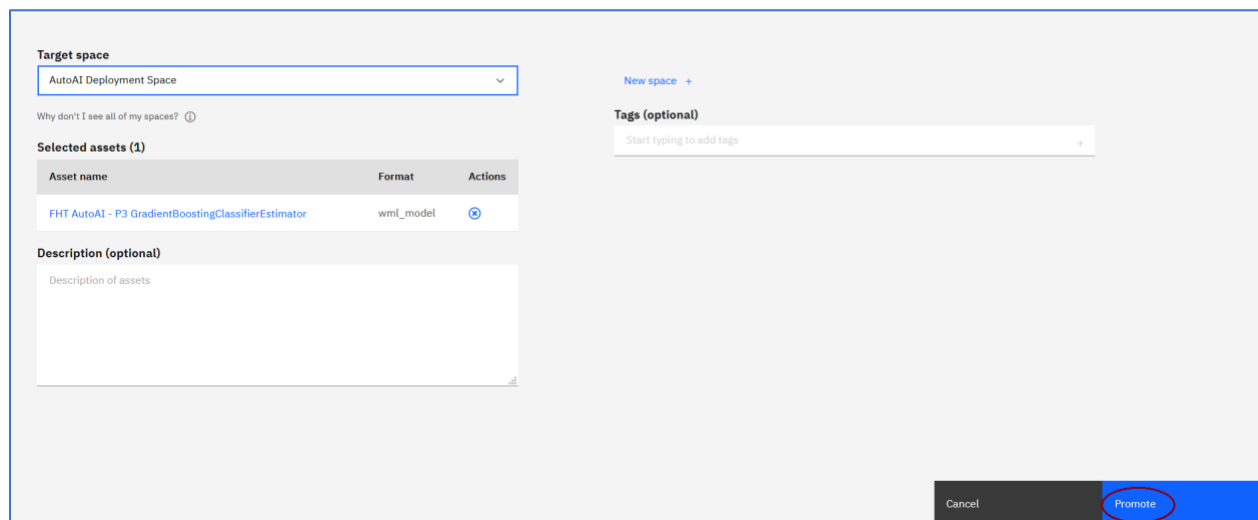
The space "AutoAI Deployment Space" is being created.

○ Step 1 of 1. Creating deployment space.

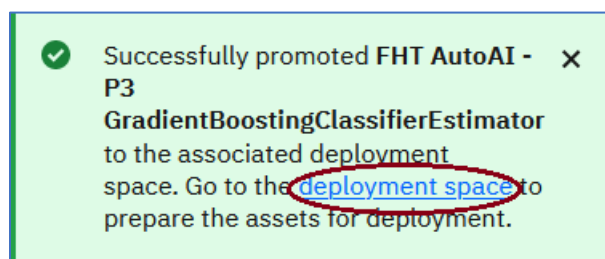
Close





8. Click **Promote**.



9. A message is displayed. Click **Deployment Space** in the message.






10. Hover over the promoted model until you see the deploy icon  appear on the right. Click on the  icon.

AutoAI Deployment Space

Assets Deployments Jobs Access control Settings

What assets are you looking for?

Models (1) [Import model +](#)

Name	Type	Software specification	Asset tags	Last modified	
 FHT AutoAI - P3 GradientBoostingClassifierEstimato...	wml-hybrid_0.1	hybrid_0.1		Jan 12, 2021 9:22 PM	 

Deploy

11. Click **Online** for an **Deployment type**, enter **FHT AutoAI Deployment** for the **Name**, and click **Create**.

Associated asset
FHT AutoAI - P3 GradientBoostingClassifierEstimator

Deployment type

Online
Run the model on data in real-time, as data is received by a web service.

Batch
Run the model against data as a batch process.

Name
FHT AutoAI Deployment

Description
Deployment description

Software specification
hybrid_0.1
The software specification is predefined for the asset type. You can update or customize the software specification programmatically. [Learn more](#)

Cancel Create

12. Click the Deployments tab. Once the asset has “Deployed” under Status, click on the deployment name.



Deployments / AutoAI Deployment Space [Add to space +](#)

AutoAI Deployment Space

Assets **Deployments** Jobs Manage

What deployments are you looking for?

Deployments (1)

Name	Type	Status	Asset	Tags	Last modified
 FHT AutoAI Deployment	Online	 Deployed	FHT AutoAI - P2 XGBClassifier		May 16, 2021 6:42 PM

13. The **API Reference** panel provides information for the application developers to invoke the deployed model. It includes sample code in various programming languages and the scoring endpoint to be used when invoking the web service.

The screenshot shows the 'FHT AutoAI Deployment' interface. At the top right, there are status indicators: a green checkmark for 'Deployed' and a grey circle for 'Online'. Below the title, there are two tabs: 'API reference' and 'Test'. The 'Test' tab is selected and circled in red. The 'Direct link' section shows an 'Endpoint' text area with a URL and a 'Bearer <token>' dropdown menu set to 'IAM'. The 'Code snippets' section has tabs for 'cURL', 'Java', 'JavaScript', 'Python', and 'Scala'. The 'cURL' tab is active, displaying a series of curl commands for authentication and scoring. A note at the top of the cURL section states: '# NOTE: you must set \$API_KEY below using information retrieved from your IBM Cloud account.' The commands include an insecure POST request for authentication and a subsequent POST request for scoring using the token.

14. Enter the following values:

PASSPORT_COUNTRY – **Ghana**

COUNTRIES_VISITED_COUNT – **3**

ARRIVAL_STATE – **OH**

DEPARTURE_AIRPORT_COUNTRY_CODE – **RU**

AGE – **20**

Category – **Sports/Travel**

Click **Add to list**.

Deployments / AutoAI Deployment Space / FHT AutoAI - P2 XGBClassifier / FHT AutoAI Deployment

FHT AutoAI Deployment Deployed Online

API reference **Test**

Enter input data

PASSPORT_COUNTRY

Ghana

COUNTRIES_VISITED_COUNT

3

ARRIVAL_STATE

OH

DEPARTURE_AIRPORT_COUNTRY_CODE

RU

AGE

20

Category

Sports/Travel

Add to list

+

Input list (0)

The prediction list is empty.
Add one or more items to the list to request a prediction.

Predict

Result

Submit a prediction to see scoring results.

15. Click Predict (1)

Deployments / AutoAI Deployment Space / FHT AutoAI - P2 XGBClassifier / FHT AutoAI Deployment

FHT AutoAI Deployment Deployed Online

API reference **Test**

Enter input data

PASSPORT_COUNTRY

other

COUNTRIES_VISITED_COUNT

Integer

ARRIVAL_STATE

other

DEPARTURE_AIRPORT_COUNTRY_CODE

other

AGE

Integer

Category

other

Add to list

+

Input list (1)

[Ghana, 3, OH, RU, 20, Sports/Travel]

Predict (1)

Result

Submit a prediction to see scoring results.

16. The prediction for VETTING_LEVEL is **30** with 44.9 % confidence.

The screenshot displays the 'FHT AutoAI Deployment' interface. The top navigation bar shows the path: Deployments / AutoAI Deployment Space / FHT AutoAI - P2 XGBClassifier / FHT AutoAI Deployment. The main header indicates the model is 'Deployed' and 'Online'. Below this, there are tabs for 'API reference' and 'Test', with 'Test' being the active tab.

The 'Test' section is divided into three main areas:

- Enter input data:** This section contains several input fields for testing the model:
 - PASSPORT_COUNTRY: other
 - COUNTRIES_VISITED_COUNT: Integer
 - ARRIVAL_STATE: other
 - DEPARTURE_AIRPORT_COUNTRY_CODE: other
 - AGE: Integer
 - Category: other
- Input list (1):** This section shows a single input example: [Ghana, 3, OH, RU, 20, Sports/Travel].
- Result:** This section displays the model's output in JSON format. The prediction is 30, which is circled in red. The probability is 0.4497275650501251, which is also circled in red. The full JSON output is:

```
{
  "predictions": [
    {
      "fields": [
        "prediction",
        "probability"
      ],
      "values": [
        30,
        0.4497275650501251
      ]
    }
  ]
}
```

You have successfully completed the Lab!!!

- ✓ Downloaded a female human trafficking cleansed dataset from the github repo
- ✓ Added an Auto AI Experiment to create a model to predict the trafficking risk
- ✓ Saved and Deployed the model
- ✓ Tested the model