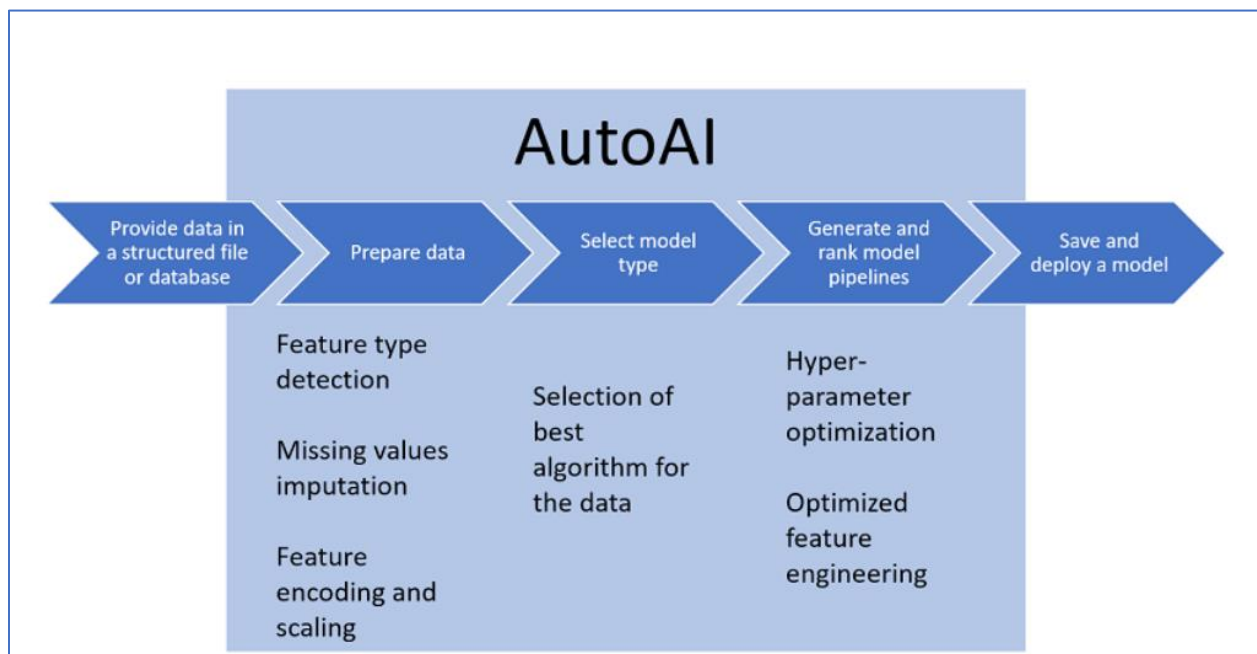


AutoAI Lab

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

AutoAI in Watson Studio 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.



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.

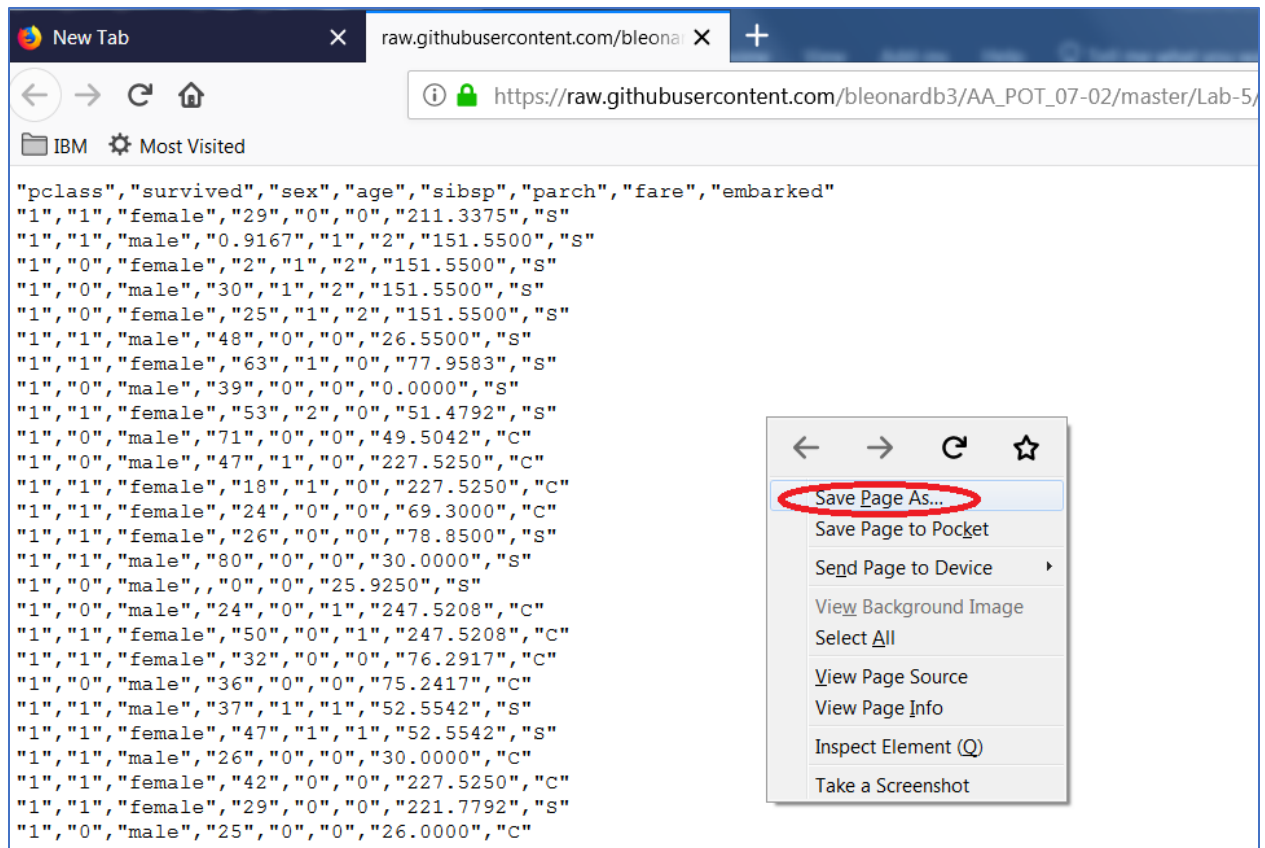
- **Automated model selection** - The next step is automated model selection that matches your data. AutoAI uses a novel approach that enables testing and ranking candidate algorithms against small subsets of the data, gradually increasing the size of the subset for the most promising algorithms to arrive 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:

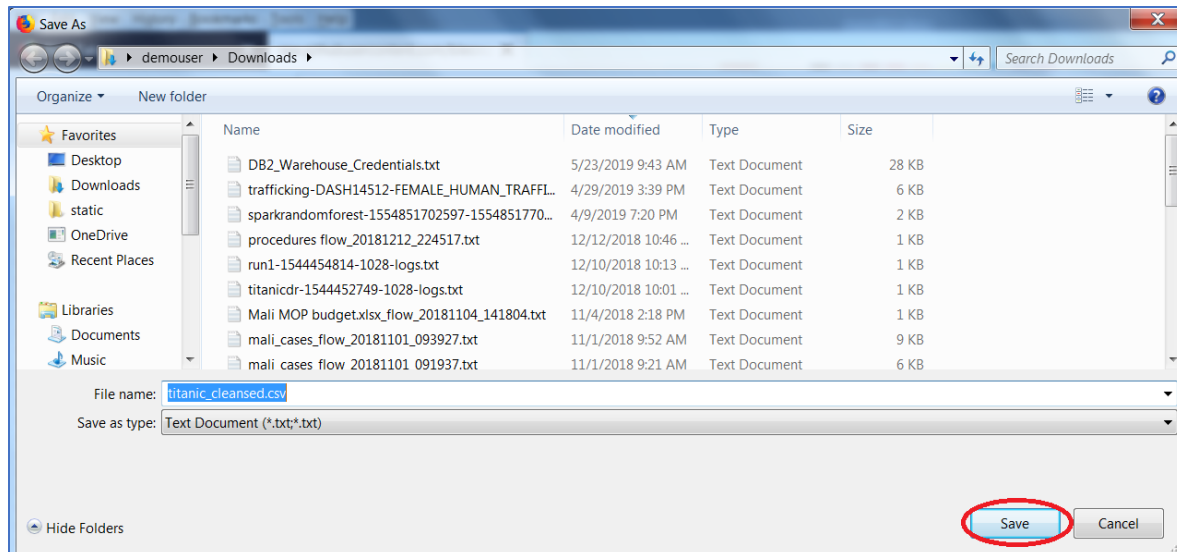
1. Download a Titanic cleansed data set
2. Add an Auto AI Experiment
3. Save and Deploy the selected model
4. Test the Deployment


Step 1: Download the `titanic_cleansed.csv` data set

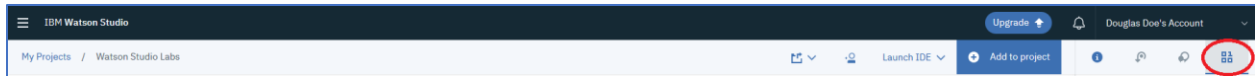
1. Download the **titanic_cleansed.csv** data file from the following location by clicking on the link [here](#). Note this is a different file than used in the previous labs.
2. Right-click on the window, and click **Save Page As...**



3. Click on **Save**. Note, if the file is named `titanic_cleansed.csv.txt`, change it to be `titanic_cleansed.csv`.

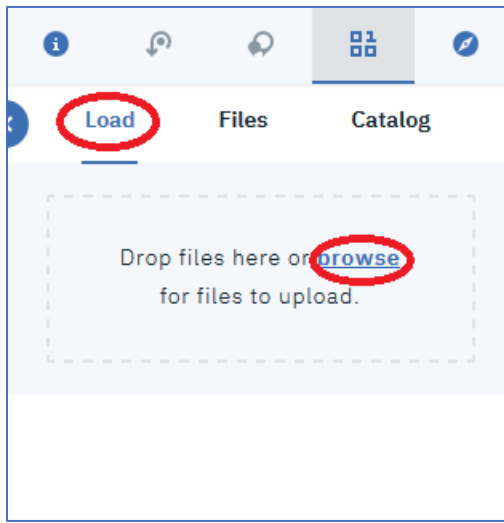


4. Go back to your Watson Studio Labs project. Click on the  icon.

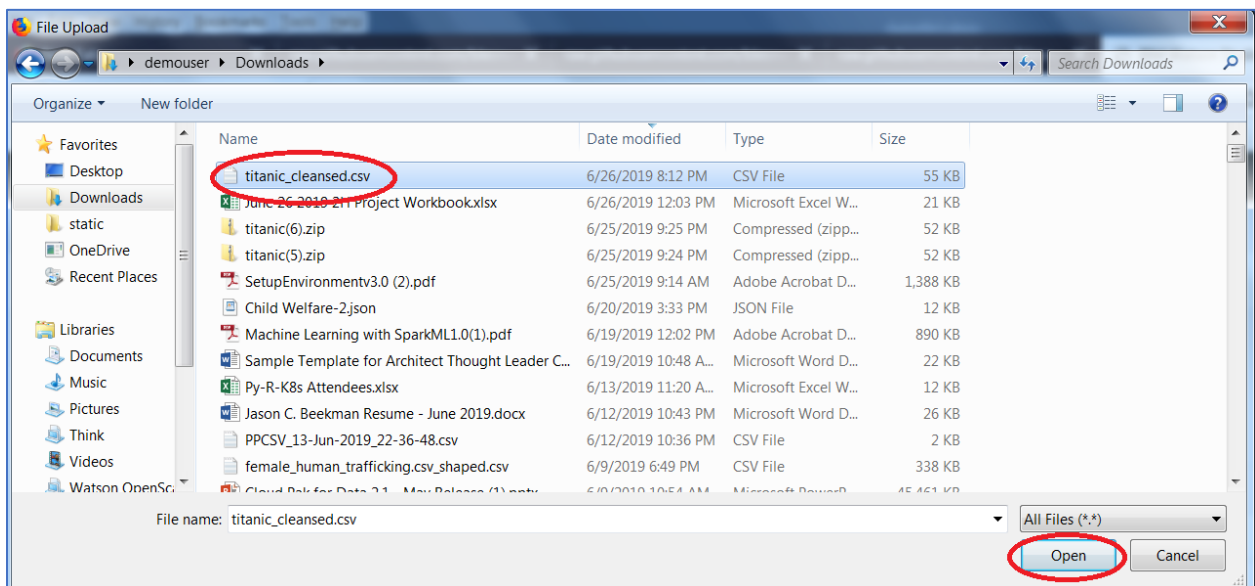


5. Click on the **Load** tab and then click on **browse**. If you don't see the **Load** tab, click on

the  icon again.



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



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

My Projects / Watson Studio Labs

Overview Assets Environments Bookmarks Deployments Access Control Settings

What assets are you looking for?

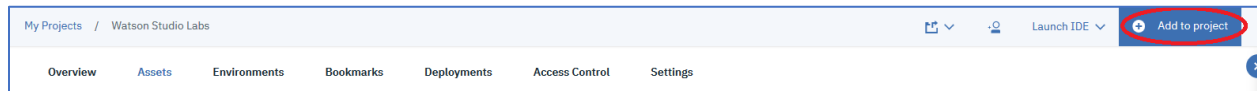
▼ Data assets

0 asset selected.

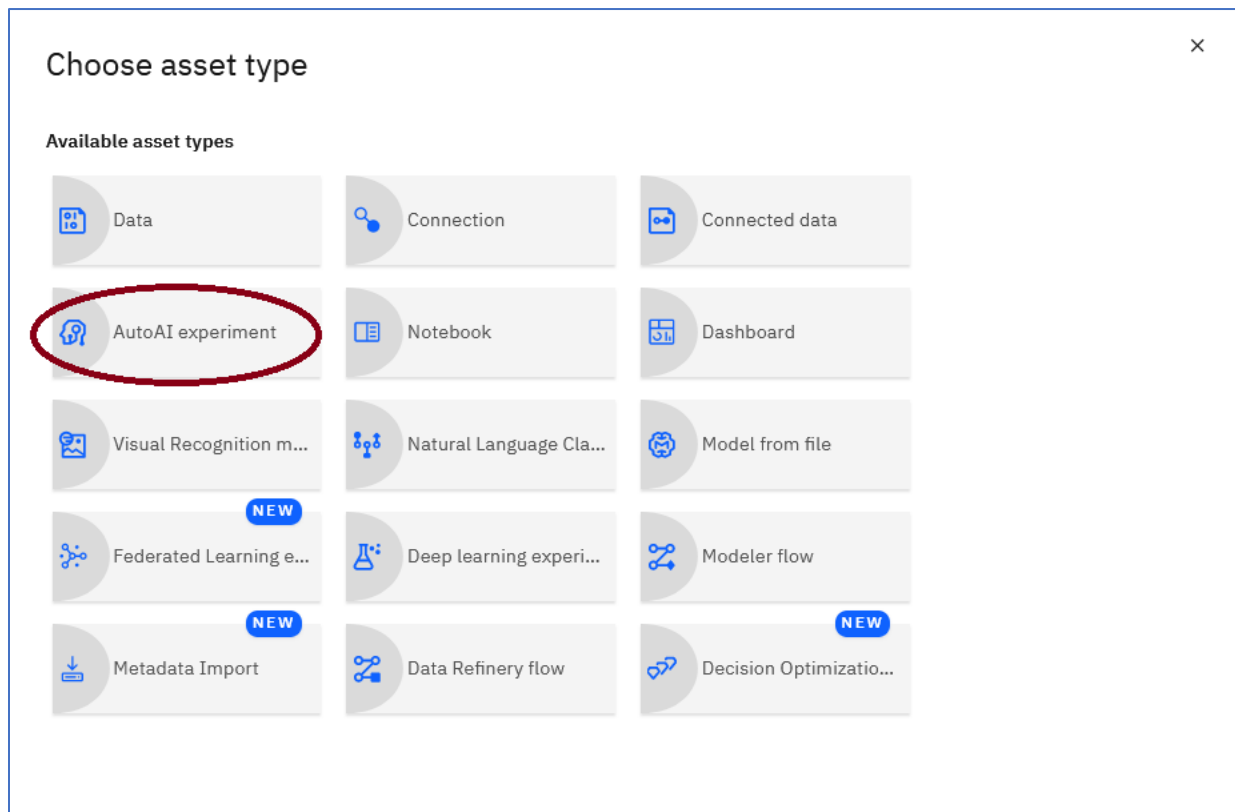
<input type="checkbox"/>	NAME	TYPE	CREATED BY	LAST MODIFIED	ACTIONS
<input type="checkbox"/>	CSV titanic_cleansed.csv	Data Asset	Douglas Doe	26 Jun 2019, 12:35:27 pm	
<input type="checkbox"/>	CSV titanic.csv	Data Asset	Douglas Doe	25 Jun 2019, 6:39:35 pm	

Step 2: Add an AutoAI Experiment

1. Click on **Add to project**.



2. Click on **AutoAI experiment**



3. Enter an **Asset name**, leave the defaults for the **Watson Machine Learning** and **Compute configuration**.

Define details

Name

Titanic AutoAI

Description (optional)

What's the purpose of this AutoAI experiment?

Tags (optional)

Add tags to make assets easier to find.

Start typing to add tags

Define configuration

Watson Machine Learning Service Instance

WatsonMachineLearning

Environment definition ⓘ

8 vCPU and 32 GB RAM


This environment definition consumes **20 capacity units per hour** for training. The consumption rate differs for joining multiple data sources, depending on your configuration. For details, see [Watson Machine Learning plans](#).

4. Click **Create**

Cancel

Create

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



Add data source

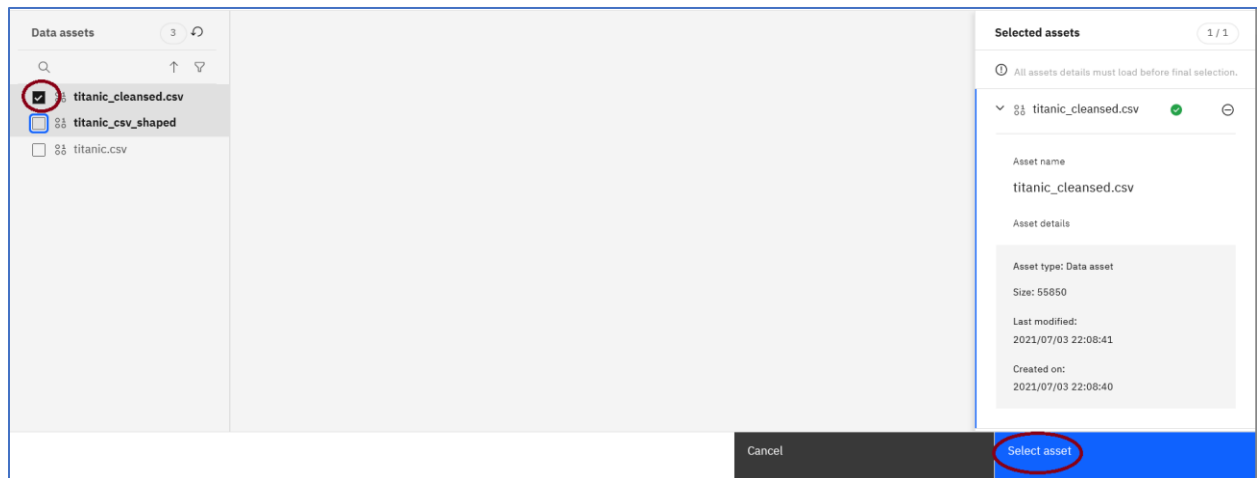
Drop or browse for a csv file.

Browse

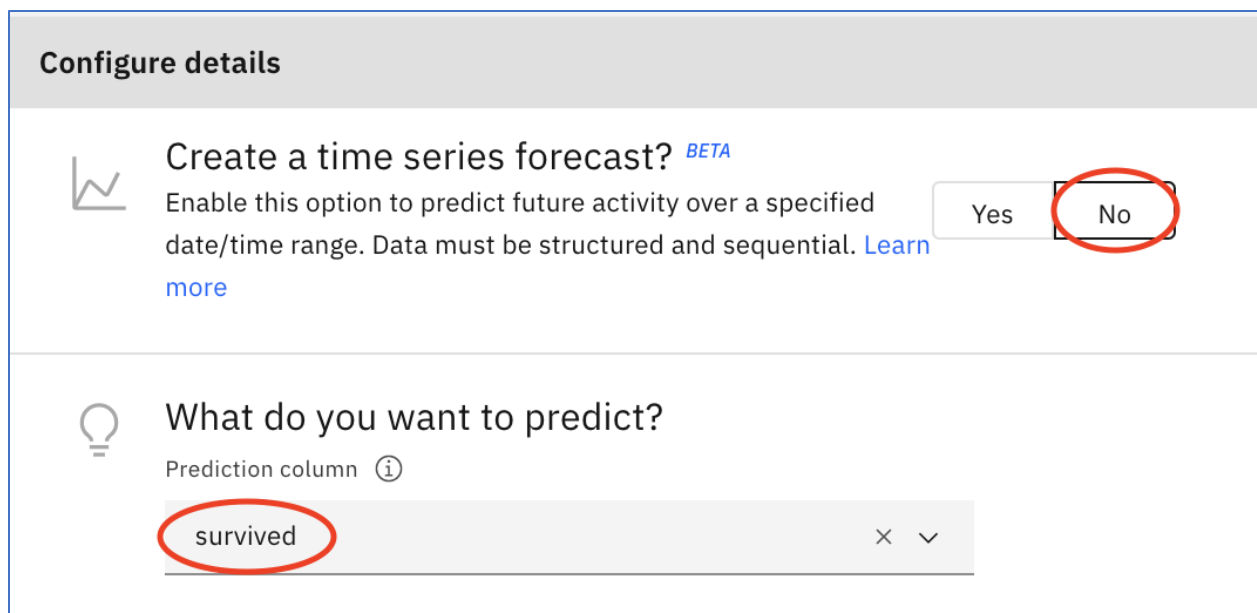
 or

Select from project

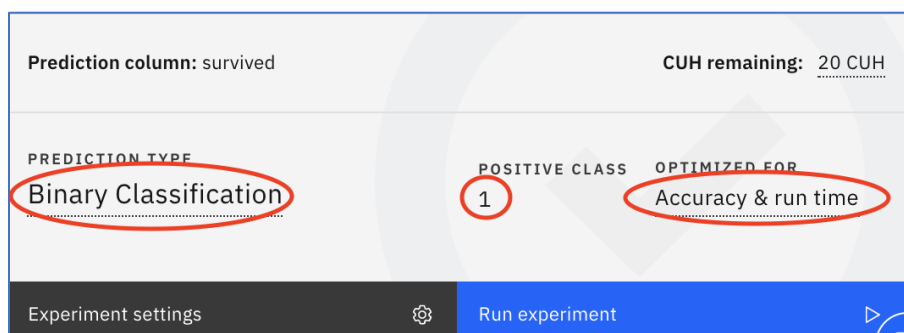
6. Click on **titanic_cleansed.csv** and then click on **Select asset**.



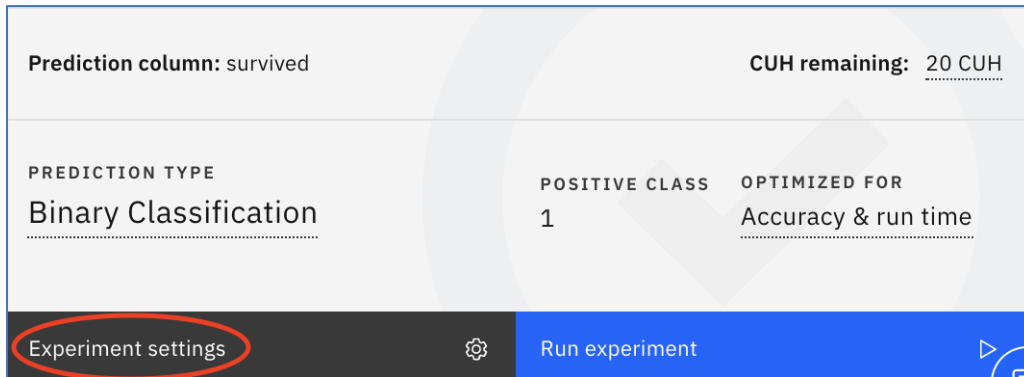
7. Select **No** when prompted on whether to **create a time series forecast** and click on **survived** as **the column to predict**.



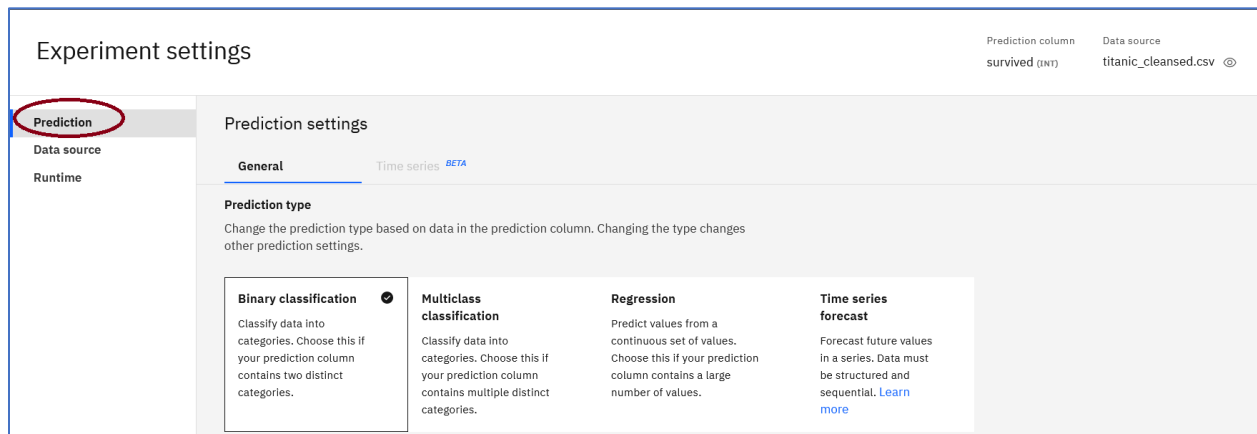
8. Note, based on this selection, the **Prediction Type** is **Binary Classification**, and the **Optimized Metric** is **Accuracy**. Further note, the **Positive Class** is correctly defaulted as "1" – survived.



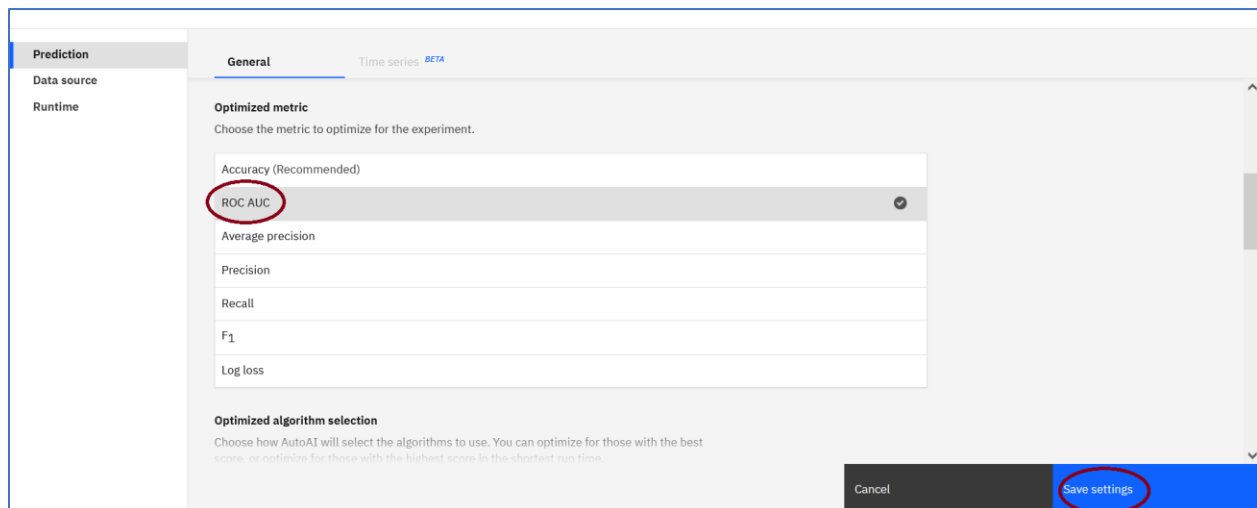
9. Click on **Experiment settings** to change the default optimized metric.



10. Click on **Prediction**.



11. Ensure that the **Prediction type** is set to **Binary classification**, the **Positive class** is set to **1**. Then, scroll down and click on **ROC AUC** (Receiver Operating Characteristic Area Under the Curve) and then click on **Save Settings**



12. Click on **Run experiment**.

Configure details

☐ 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 ⓘ

survived

×

▼

Prediction column: survived

CUH remaining: 20 CUH

PREDICTION TYPE	POSITIVE CLASS	OPTIMIZED FOR
Binary Classification	1	ROC AUC & run time

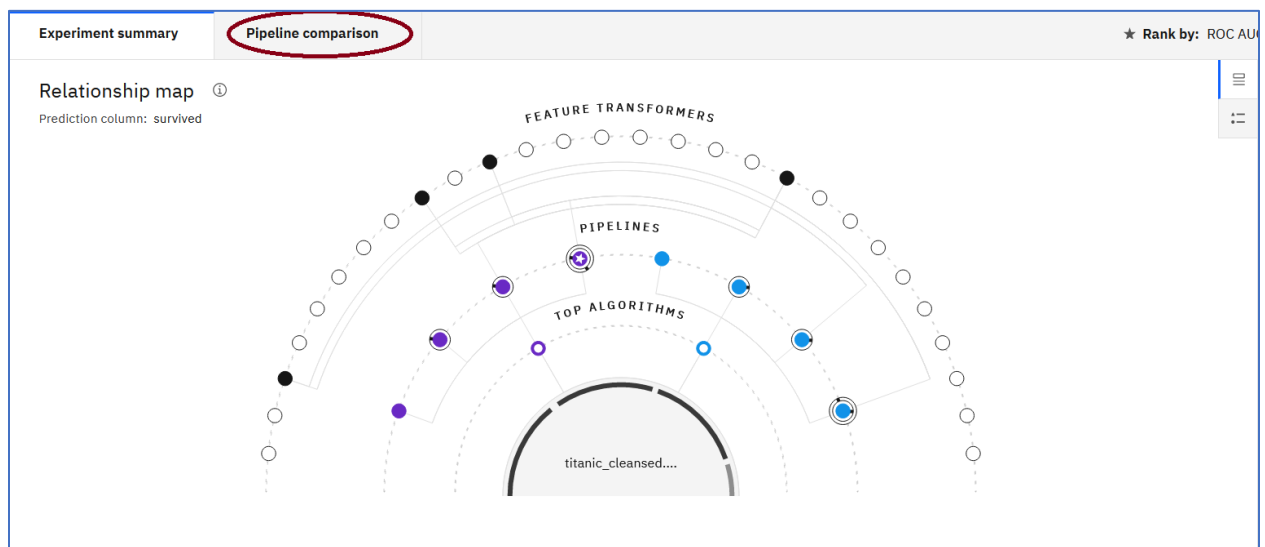
Experiment settings

⚙️

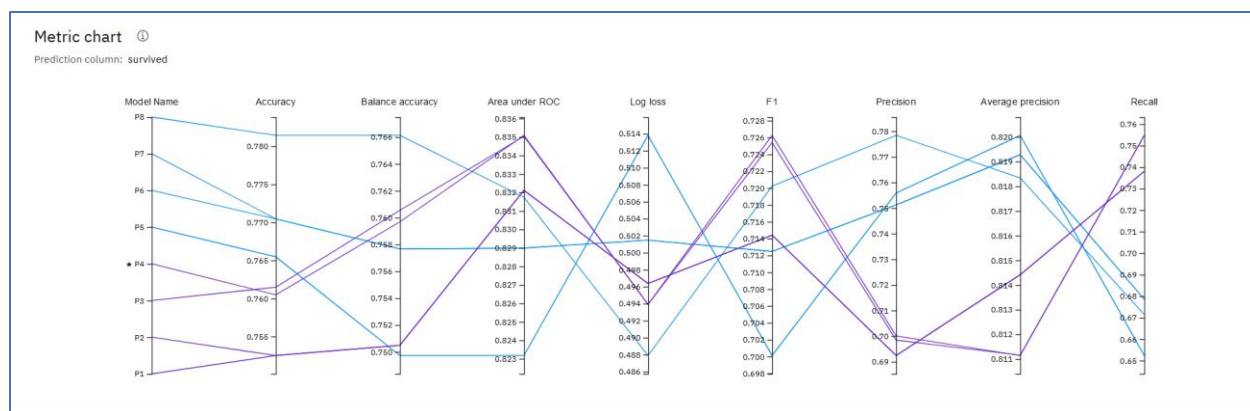
Run experiment

▶

13. It will take several minutes for the eight alternative pipelines to be analyzed. The first pipeline picks the best algorithm. The second pipeline optimizes the hyper-parameters for the selected algorithm. The third pipeline does a feature transformation to try to improve the performance of the algorithm. The fourth pipeline repeats the hyper-parameter tuning with the new set of features. The next 4 pipelines do the same thing for the second best algorithm. Click on **Pipeline comparison**.



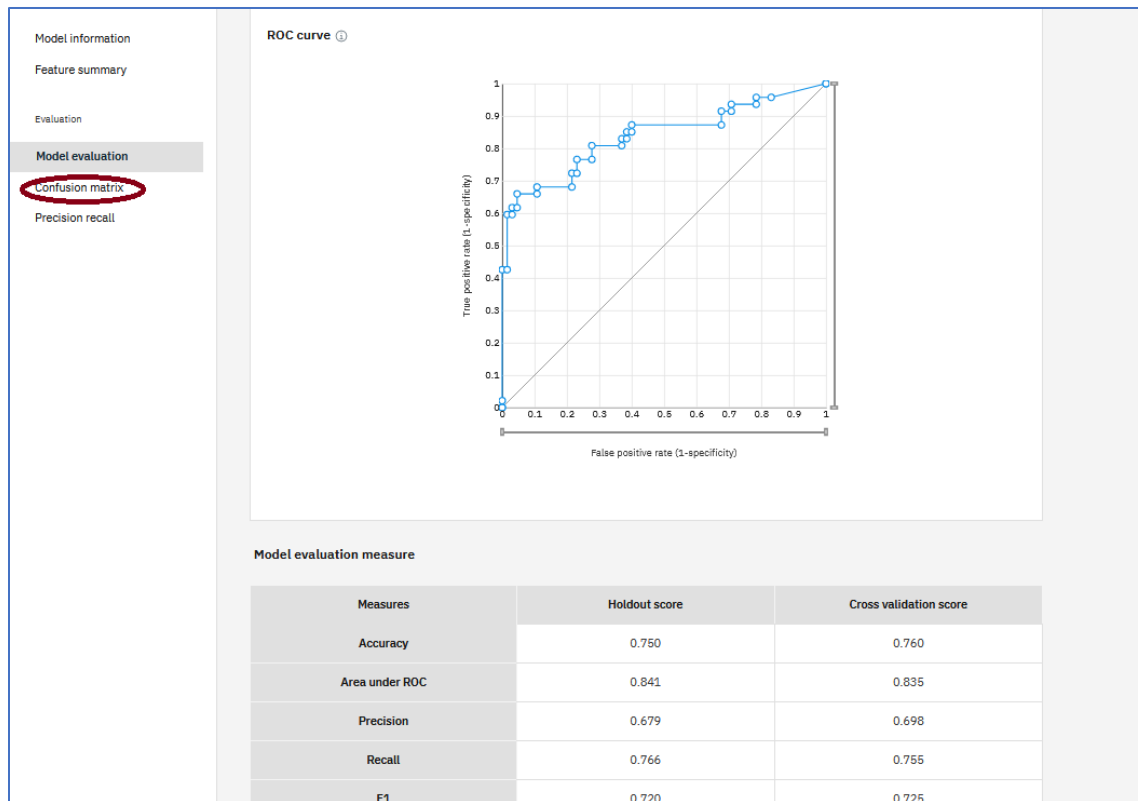
14. A comparison of all the pipelines against several performance metrics is displayed.



15. Scroll down to view the **Pipeline leaderboard**. The pipeline summary is then displayed. Click on the right arrow **Pipeline 4**.

	Rank	↑	Name	Algorithm	ROC AUC (Optimized) Cross Validation	Enhancements	Build time
★	1		Pipeline 4	Logistic Regression	0.835	HPO-1 FE HPO-2	00:00:12
	2		Pipeline 3	Logistic Regression	0.835	HPO-1 FE	00:00:20
	3		Pipeline 1	Logistic Regression	0.832	None	00:00:01
	4		Pipeline 2	Logistic Regression	0.832	HPO-1	00:00:04
	5		Pipeline 8	XGB Classifier	0.832	HPO-1 FE HPO-2	00:00:19
	6		Pipeline 6	XGB Classifier	0.829	HPO-1	00:00:06
	7		Pipeline 7	XGB Classifier	0.829	HPO-1 FE	00:00:23
	8		Pipeline 5	XGB Classifier	0.823	None	00:00:01

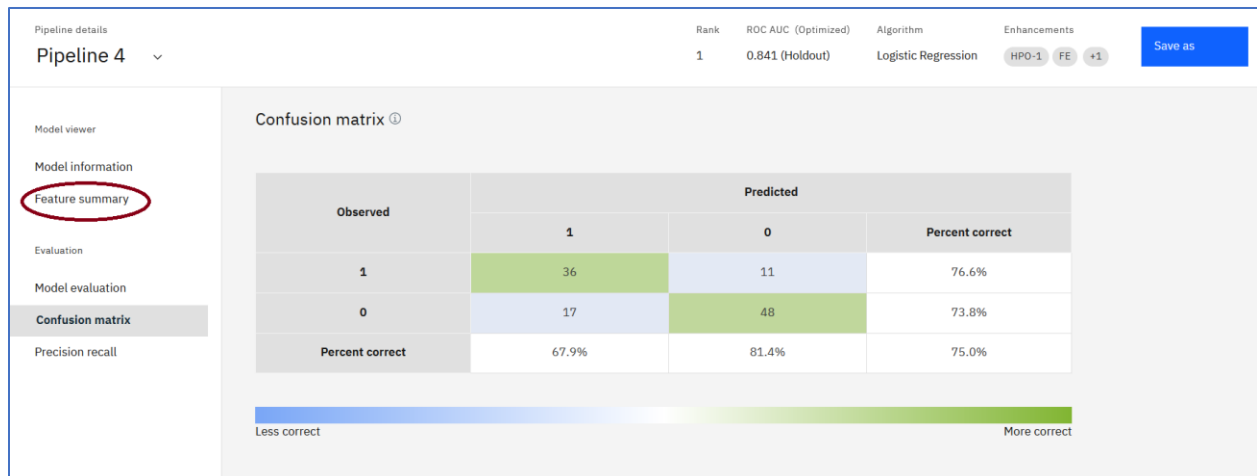
16. Metrics are displayed for both the holdout sample and the training sample (cross-validation). Click on **Confusion Matrix**.



17. The Confusion Matrix is displayed for the holdout sample. The different metrics are computed based on the numbers in the Confusion Matrix. For example, Precision is defined by the percentage of predicted positives that are actually positive (i.e. the percentage of predicted survivors that survived). Recall is defined as the percentage of observed positives that the model predicts are positive (i.e. the percentage of actual survivors that the model predicted would survive). Note the higher the Precision the lower the Recall.

Precision = True Positive/ (True Positive + False Positive) – shown inside blue rectangle on diagram below.

Recall = True Positive/(True Positive + False Negative) - shown inside green rectangle on diagram below. This is also called the True Positive Rate. After viewing the Confusion Matrix, click on the **Feature Summary** option.



18. Eight new features are derived. Most of these are shown to be more impactful on the prediction of survivability than the original features.

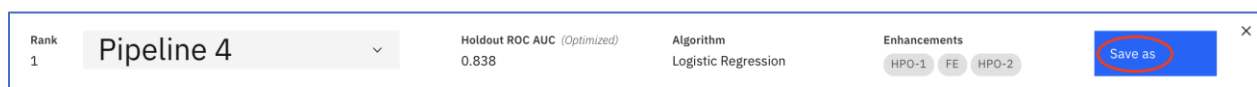
Feature summary ①

All features Search feature or transformer names

Feature name	Transformation	Feature importance
NewFeature_6	pca(ALL)	20.80%
NewFeature_5	pca(ALL)	11.50%
NewFeature_2	pca(ALL)	9.40%
sex	None	8.50%
NewFeature_7	pca(ALL)	7.70%
NewFeature_3	pca(ALL)	7.50%
NewFeature_4	pca(ALL)	7.50%
fare	None	6.70%

Step 3 – Save and Deploy the Selected Model

1. Click on **Save as**.



2. You have the option to save a Model or a Notebook. The notebook contains the code used to generate the pipeline. In this way a data scientist could use this as a starting point

to tune the model even further. We will save the model. Optionally change the default name and click on **Save**.

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 highlighted with a red box. The 'Model' description reads: 'Create a Watson Machine Learning model asset that you can test with new data, deploy to generate predictions, and trace lineage activity.' Below it is the 'Notebook' option. On the right, the 'Name' field contains 'Titanic AutoAI - P4 Logistic Regression'. The 'Description (optional)' field is empty. The 'Tags' section has an 'Add a tag' input field. At the bottom right, the 'Create' button is highlighted with a red circle.

3. The model is successfully saved. Click on **View in Project**.

The screenshot shows the model project page for 'Titanic AutoAI - P4 Logistic Regression'. The top bar displays 'Rank 1', 'Pipeline 4', 'Holdout ROC AUC (Optimized) 0.838', and 'Algorithm Logistic Regression'. A 'Feature transformations' section is visible. A success message overlay is present: 'Saved model successfully. Titanic AutoAI - P4 Logistic Regression was successfully saved to Watson Studio Labs.' The 'View in project' link in the message is highlighted with a red circle.

4. In order to deploy the model, we need to promote it to the deployment space that was created in Lab-1. Click on **Promote to deployment space**.

The screenshot shows the model project page for 'Titanic AutoAI - P4 Logistic Regression'. The 'Promote to deployment space' button is highlighted with a red circle. The page also shows 'Overview' and 'Activities' tabs.

5. Set the **Target space** to **Watson Studio Labs** and click on **Promote**.

Promote to space

Target space
Watson Studio Labs

Tags (optional)
Start typing to add tags +

☐ Go to the model in the space after promoting it

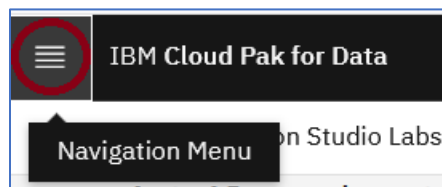
Selected assets (1)

Asset name	Format
Titanic AutoAI - P4 Logistic Regression	Model

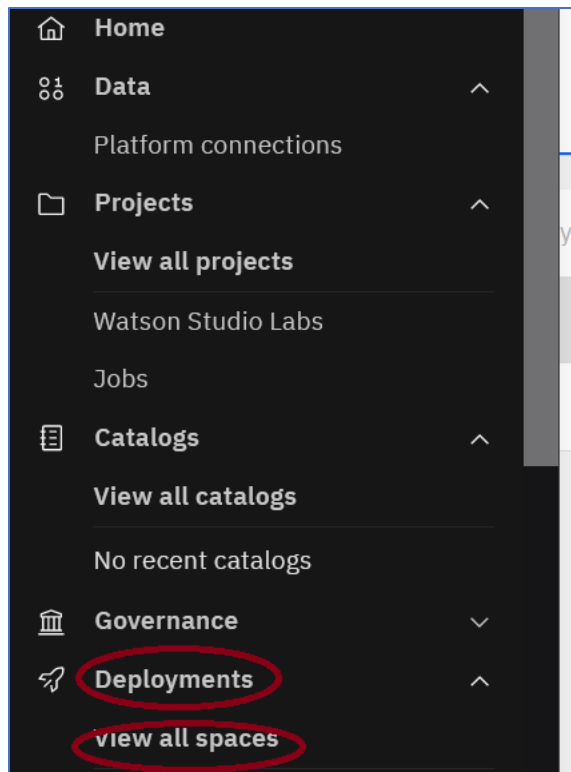
Description (optional)
Description of assets

Cancel Promote

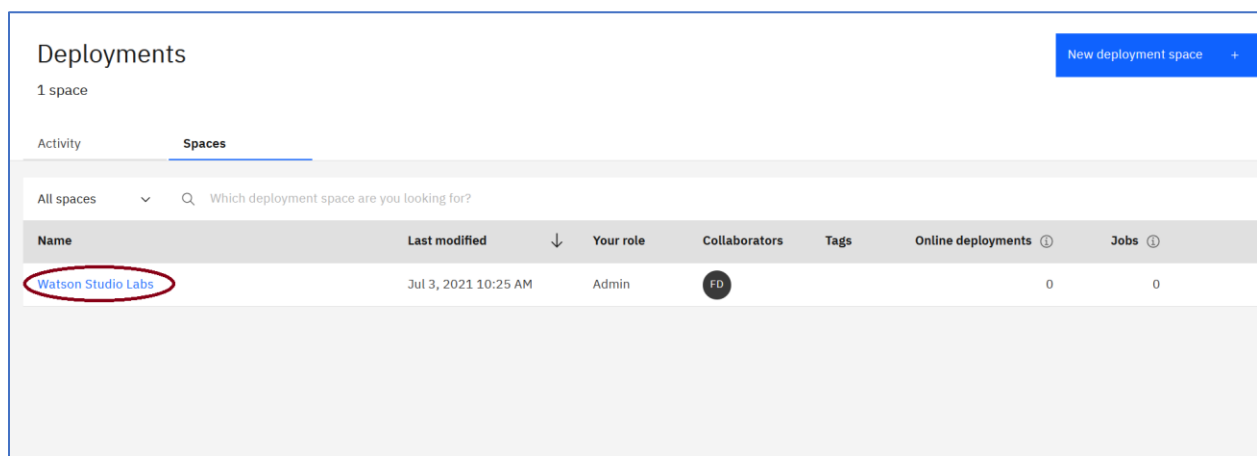
6. Click on the hamburger icon ☰.



7. Click on **Deployments** and **View All Spaces**.



8. Click on **Watson Studio Labs**.



9. Click on **Assets**.

Watson Studio Labs

Overview **Assets** Deployments Jobs Manage

Deployments All ▾

0	0
✓ Deployed	✗ Failed

[View deployments](#)

Job runs


0	0
● Active	✗ Failed last 24 hours ▾







[View jobs](#)

Assets

Space activity

🔔 No notifications
You will see your most recent notifications here.

10. Hover over the model and on the right the Deploy icon  will appear. Click on the icon.




Deployments / Watson Studio Labs  [Add to space](#) +     

Watson Studio Labs

Assets **Deployments** Jobs Manage

🔍 What assets are you looking for?

▾ Models (1) [Import model](#) +

Name	Type	Software specification	Tags	Last modified	↓
 Titanic AutoAI - P4 Logistic Regression	wml-hybrid_0.1	hybrid_0.1		Jul 1, 2021 1:15 PM	 

11. Click on **Online**, enter **Titanic AutoAI Deploy** for the **Name** of the deployment, optionally enter a **Description**, and click **Create**.

Create a deployment

Associated asset
Titanic AutoAI - P4 Logistic Regression

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
Titanic AutoAI Deploy

Description

Cancel Create

12. Click on the **Deployment** Tab to see the deployment status.

Deployments / Watson Studio Labs

Watson Studio Labs

Assets **Deployments** Jobs Manage

What deployments are you looking for?

Deployments (1)

Name	Type	Status	Asset	Tags	Last modified
Titanic AutoAI Deploy	Online	Deployed	Titanic AutoAI - P4 Logistic Regression		Jul 1, 2021 1:20 PM

13. When the status shows **Deployed**, click on **Titanic AutoAI Deploy**

Deployments / Watson Studio Labs

Watson Studio Labs

Assets **Deployments** Jobs Manage

What deployments are you looking for?

Deployments (1)

Name	Type	Status	Asset	Tags	Last modified
Titanic AutoAI Deploy	Online	Deployed	Titanic AutoAI - P4 Logistic Regression		Jul 1, 2021 1:20 PM

14. 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. Click on **Test**.

Titanic AutoAI Deploy

DeployedOnline

API reference

Test

Direct link

Endpoint

https://us-south.ml.cloud.ibm.com/ml/v4/deployments/61b61f08-873a-4c09-bc2a-5e

Bearer <token>

IAM

Code snippets

cURL	Java	JavaScript	Python	Scala
------	------	------------	--------	-------

```
# NOTE: you must set $API_KEY below using information retrieved from your IBM Cloud
curl --insecure -X POST --header "Content-Type: application/x-www-form-urlencoded" --he

# the above CURL request will return an auth token that you will use as $IAM_TOKEN in t
# TODO: manually define and pass values to be scored below
curl -X POST --header "Content-Type: application/json" --header "Accept: application/js
```

15. Enter values for a passenger. For example,

- pclass – 1
- sex – female
- age – 5
- sibsp – 1

Scroll down to add

- parch – 2
- fare – 23
- embarked – S

and click **Add to list.**

Enter input data

pclass
1

sex
female

age
5

sibsp
1

Add to list +

16. Click **Predict(1)**

Titanic AutoAI Deploy ✓ Deployed Online

API reference **Test**

pclass
Integer

sex
other

age
Double

sibsp
Integer

Add to list +

[1, female, 5, 1, 2, 23, S]

Predict (1)

17. The model predicts this passenger would survive with 98% confidence. Your number may be slightly different.

Deployments / Watson Studio Labs / Titanic AutoAI - P4 Logistic Regr... / Titanic AutoAI Deploy

Titanic AutoAI Deploy Deployed Online

API reference **Test**

pclass
Integer

sex
other

age
Double

sibsp
Integer

Add to list +

[1, female, 5, 1, 2, 23, S]

Predict (1)

```
0 {  
1   "predictions": [  
2     {  
3       "fields": [  
4         "prediction",  
5         "probability"  
6       ],  
7       "values": [  
8         [  
9           1,  
10          [ 0.01243982638529783,  
11             0.9865601736947822 ]  
12        ]  
13      ]  
14    }  
15  ]  
16 }
```

You have successfully completed the lab!!!

- ✓ Downloaded a Titanic cleansed data set
- ✓ Added an Auto AI Experiment
- ✓ Saved and Deployed the selected model.
- ✓ Tested the Deployment.