

ASEANZK Cloud Pak for Data – Practicum Scenario:

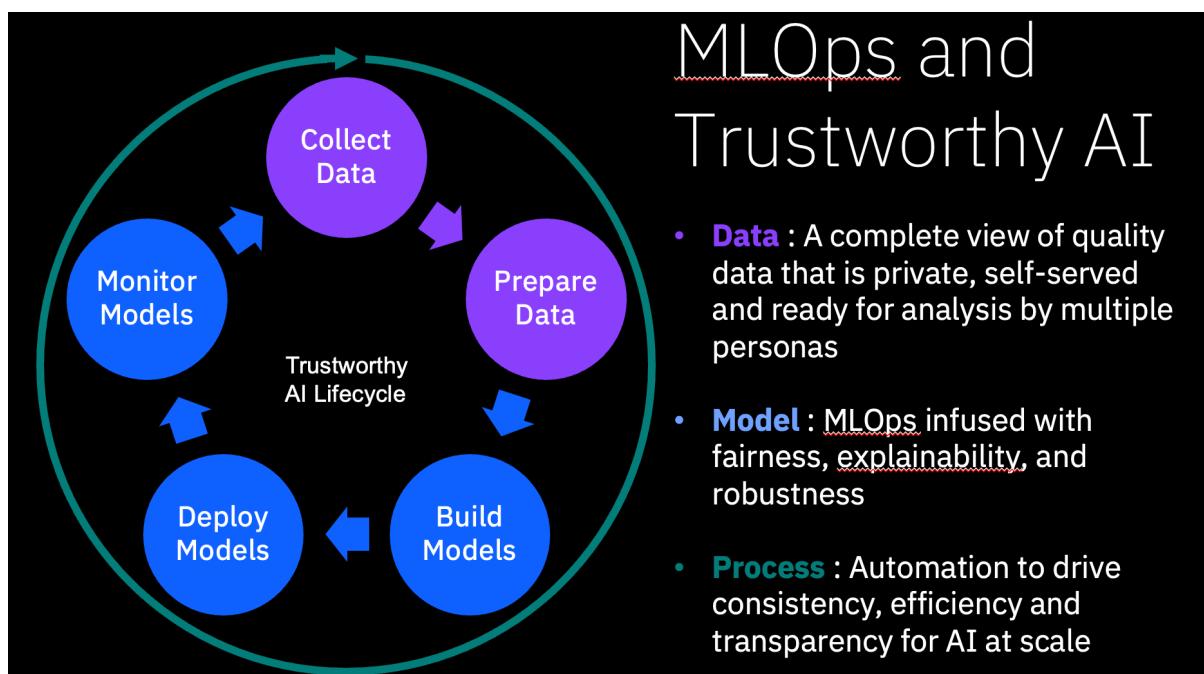
Author:

Teerarat Siwapathomchai (Toon), Customer Success Manager

Satit Pongbundit (Oat), Customer Success Manager

Use Case

MLOps and Trustworthy AI (focusing on Model Development / Model Deployment / Monitoring)



Scenario Description

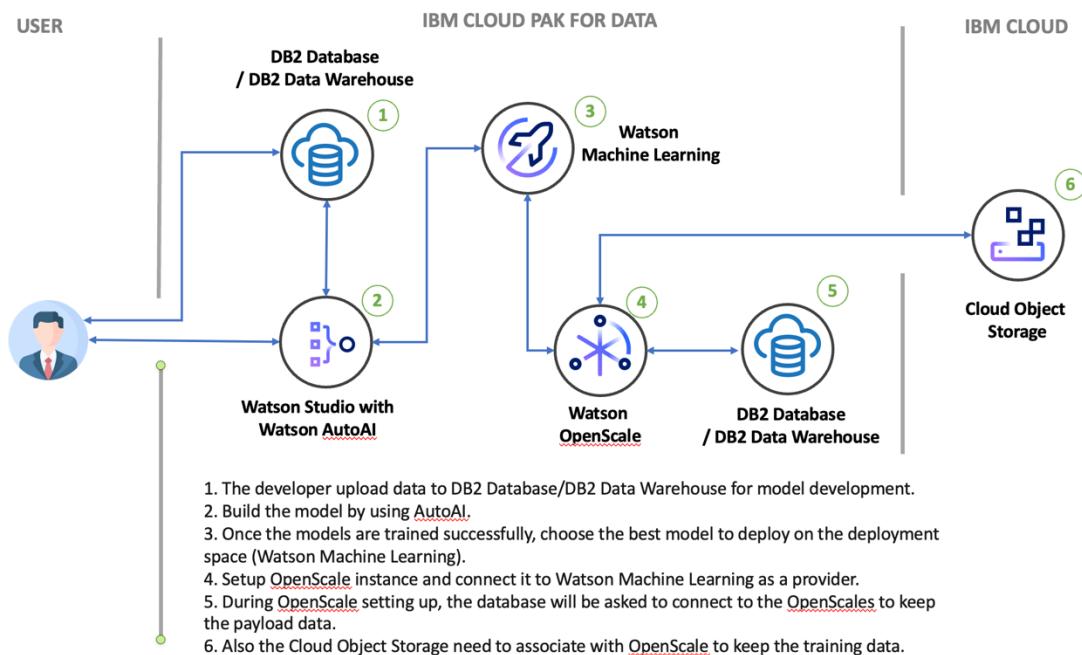
Credit risk is significant for the banking business. If the borrower fails to repay, it results in a loss. To minimize loan from the bank's perspective, ABC Bank needs a decision rule regarding to give approval of the loan and who not to. They decided to implement Application Score (A-Score) model to access loan application based on application's demographic and economic profiles. Loan analyst who has no programming skill has been asked to develop the model. The saved model need to be deployed machine learning engineer and generate the API for the front-end application. Due to economic uncertainty can cause the customer behaviour change which affects to the model. To avoid the opportunity cost in business from dropping in model accuracy. The team also has been asked to set up the instance to monitor the model after deploying.

The company has recently acquired Cloud Pak for Data v4 and are planning to utilize its key features to achieve the following goals :

- i. Accelerate model development without programming skill.
- ii. Save and deploy model seamlessly to reduce friction between model developer and machine learning engineer.
- iii. The model need to be an "on-line" deployment with API for the front end application.
- iv. After deployment successfully, you also have been asked to set up instance to monitor the model to see there are any changes affect to the model (Drift and Fairness for example).
- v. Use the Python Client to test the model and check the payload is logging in the OpenScale.

High Level Architecture

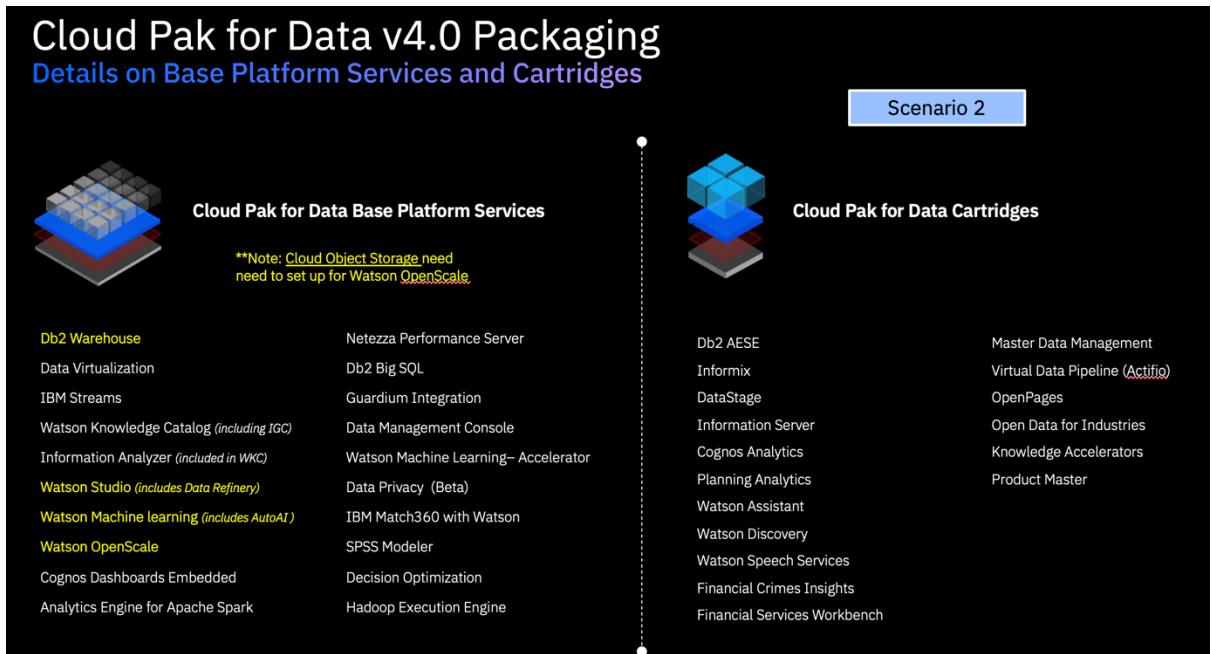
Below is the high level architecture that we plan to implement as part of this scenario.



Product Used

Your objective is to use IBM Cloud Pak for Data features to develop, deploy and monitor model under the MLOps and trustworthy AI concept. The key products you will be used for this project are here;

- 1x Watson Studio (WSL) with AutoAI
- 1x Watson Machine Learning (WML)
- 1x Watson OpenScale instance.
- 2x DB2 Database or DataWarehouse (1x for Training data storing, 1x for Watson OpenScale set up)
- 1x Cloud Object Storage (for Watson OpenScale to keep training data)



Implementation Step

Task 0: Preparing Steps

Check the products and number of instances will be use in this project.

- 1) From the navigation menu, go to service > instances.
- 2) Check the products and number of instances are correct per information here;
 - 1x Watson Studio (WSL) with AutoAI
 - 1x Watson Machine Learning (WML)
 - 1x Watson OpenScale instance.
 - 2x DB2 Database or Data Warehouse (1x for Training data storing, 1x for Watson OpenScale set up)
- 3) If there are missing on products or number of instances, please do create it by clicking new instance and keep the configurations as default (if any).
- 4) Create a Cloud Object Storage for Watson OpenScale to keep training data.
 1. Going to <https://www.ibm.com/cloudibm>
 2. Login to IBM console, if you don't have account, please create one.
 3. After login, go to the "catalog" at the top bar. Click "catalog", in the search menu type "Object".
 4. Select "Object Storage by IBM"
 5. Choose the lite plan and create it.

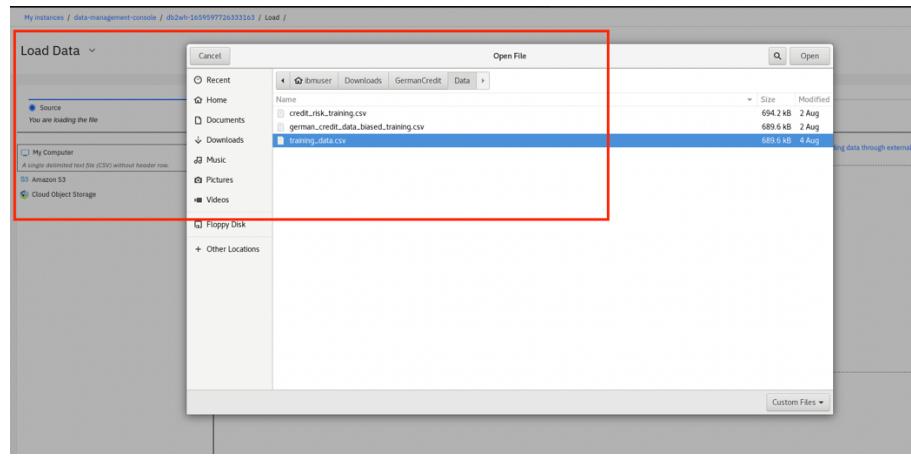
Loading the training data to the data warehouse for further model development.

- 1) Go to https://github.com/sandeepved/ASEANZK-CP4D-Practicum/blob/main/scenario2/data/training_data.csv and loading training_data to your local repository.
- 2) Go back to IBM Cloud Pak for Data.
- 3) From navigation menu, go to Data > Database.

Type	Created by	vCPU requests	Memory requests (GiB)	Users	Status	Created on
db2wh	admin	6.20	18.25 Gi	1	Green	Aug 4, 2022
db2wh	admin	6.20	18.25 Gi	1	Green	Aug 4, 2022
dv	admin	11.60	38.70 Gi	1	Green	Apr 5, 2022
dmc	admin	3.70	9.50 Gi	1	Green	Apr 5, 2022
azos	admin	-	-	1	Green	Apr 5, 2022

- 4) Open the first data warehouse you can see the “Summary” in the left hand-side. Click drop down and go to “Load”. Under “Load”, click load data.

- 5) From the Load Data GUI, browse file and upload training_data to DB2 data warehouse.



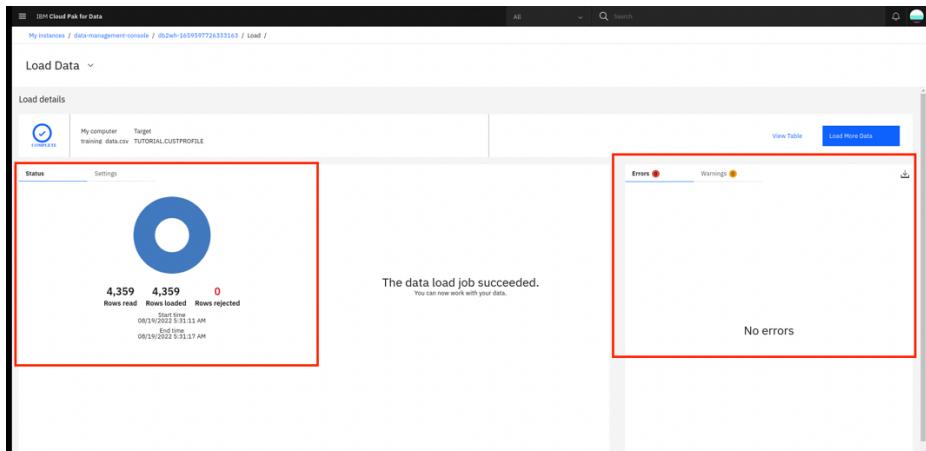
- 6) Create new schema name “TUTORIAL” and Create new table name “CUSTPROFILE”

- 7) Click next and review columns name. Click next and begin load.

Load Data ▾

	LOANAMOUNT	SEX	AGE	HOUSE	CREDITCARD	JOB	DEPENDENT	TELEPHONE	FOREIGNER	RISK
1	1889	female	32	own	1	skilled	1	none	yes	No Risk
2	462	female	37	own	2	skilled	1	none	yes	No Risk
3	250	male	28	own	2	skilled	1	yes	no	No Risk
4	3693	male	32	own	1	skilled	1	none	yes	No Risk
5	6235	male	57	own	2	skilled	1	none	yes	Risk
6	9604	male	57	free	2	skilled	2	yes	yes	Risk
7	3109	female	36	own	2	skilled	1	none	yes	No Risk
8	7138	male	49	free	2	skilled	2	yes	yes	Risk
9	4302	male	34	free	1	skilled	1	none	yes	No Risk
10	3310	male	40	free	1	skilled	1	yes	yes	No Risk

- 8) In the final page, check the data is download completely.



Create a project

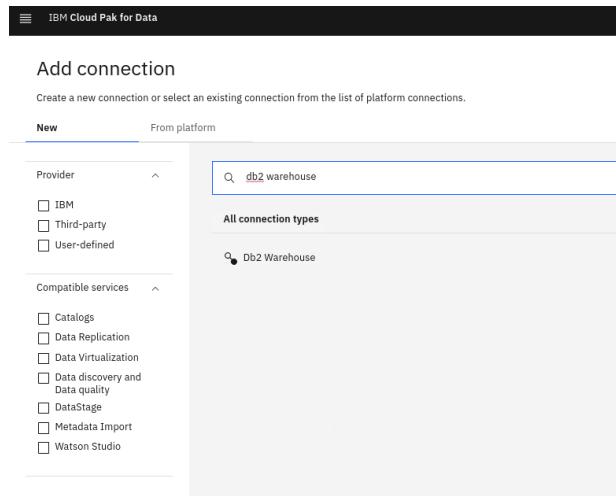
- 1) If you have an existing project, open it. If you don't have an existing project, click **Create a project** on the home page or click New project on your **Projects** page.
- 2) Select **Analytics Project** and Next.
- 3) **Create an empty** project.
- 4) On the Create a project screen, add a name and optional description for the project.
- 5) Click Create.

Task 1: Build Model

Task 1.1: Connect to the Data Warehouse.

Before you build the model using Auto AI, you need to connect the data to the project first. So, the data we will use is already located in the DB2 Data Warehouse which we have already loaded data in the previous step. To connect the Data Warehouse, follow the steps below;

- 1) Go to the project,
- 2) Set up the project connection first for CP4D communicate with DB2 Data Warehouse to get data for model development.
 - o Add to project.
 - o Go to connection and add connection.
 - o In the connection page find “DB2 Warehouse”, click and select.



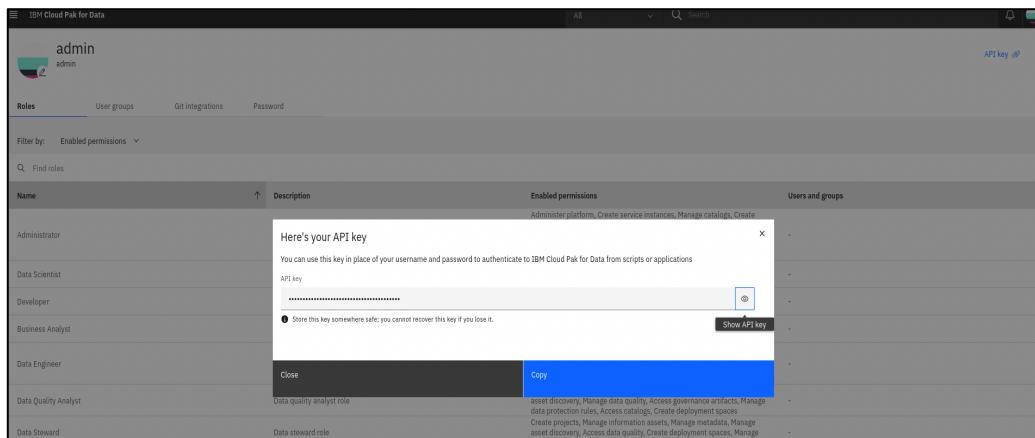
- Open another window and navigate to DB2 data warehouse instance. Click on the instance name that used for storing training_data and click to see the details.

About this database		Storage	
Database name	BLUDB	Storage class (System storage)	managed-nfs
Database type	db2wh	Size (System storage)	100 GiB
Database software version	11.5.7.0-cn3-x86_64	Storage class (User storage)	managed-nfs
Processor	x86-64	Size (User storage)	100 GiB
Deployment id	db2wh-1659597726333163	Storage class (Backup storage)	managed-nfs
Created on	Aug 4, 2022 3:22 AM	Size (Backup storage)	100 GiB
Status	Available	Storage class (Transaction logs storage)	managed-nfs
Nodes		Size (Transaction logs storage)	100 GiB
HOSTNAME	CPU	Storage class (Temporary table spaces storage)	managed-nfs
worker2	6.1 cores	Size (Temporary table spaces storage)	100 GiB

- Use the information fill in to the “Create connection: DB2 Warehouse”. After filling all information, then click create.
- If the the data warehouse is connected to data based properly, you will see the screen like this.

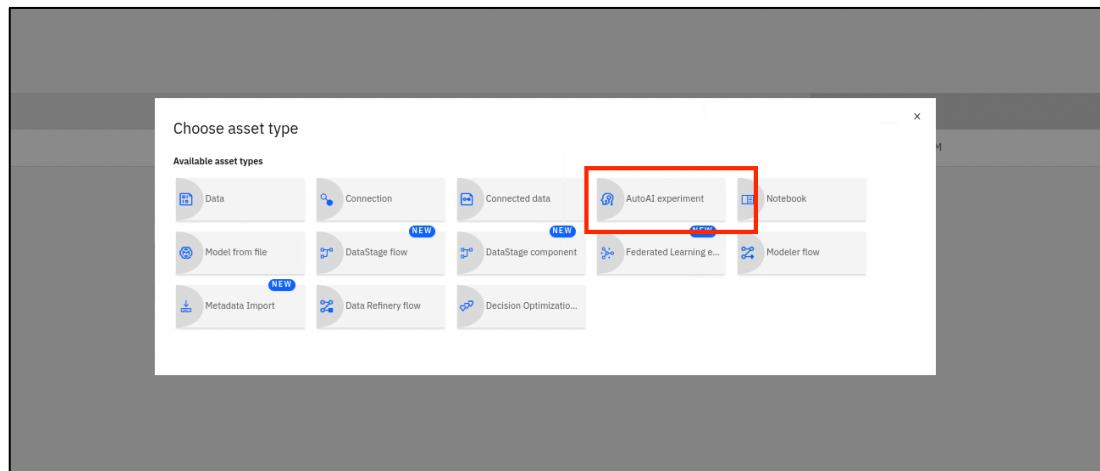
Name	Type	Created by	Last modified
Data Warehouse	Connection	admin	Aug 21, 2022, 10:42 PM

Note: If you need to connect via using API key, please generate the API key by going to your “profile and setting” which located in the top right corner. Then generate API key.

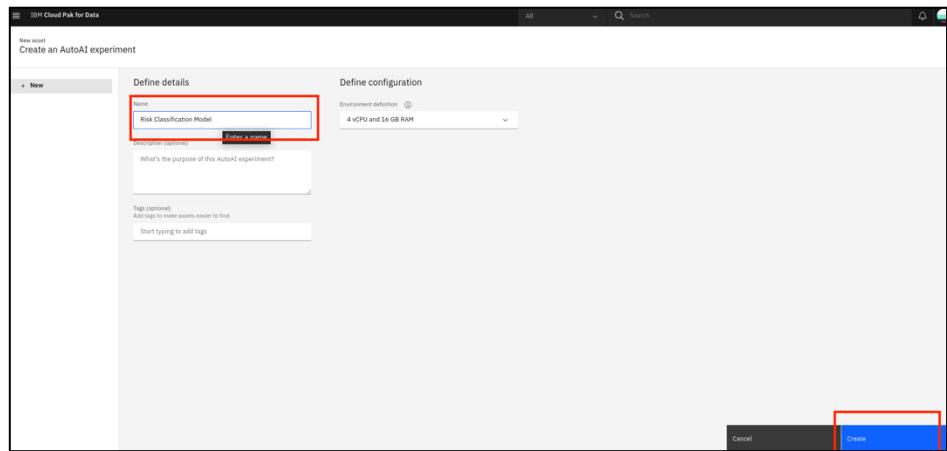


Task 1.2: Build the model using Auto AI

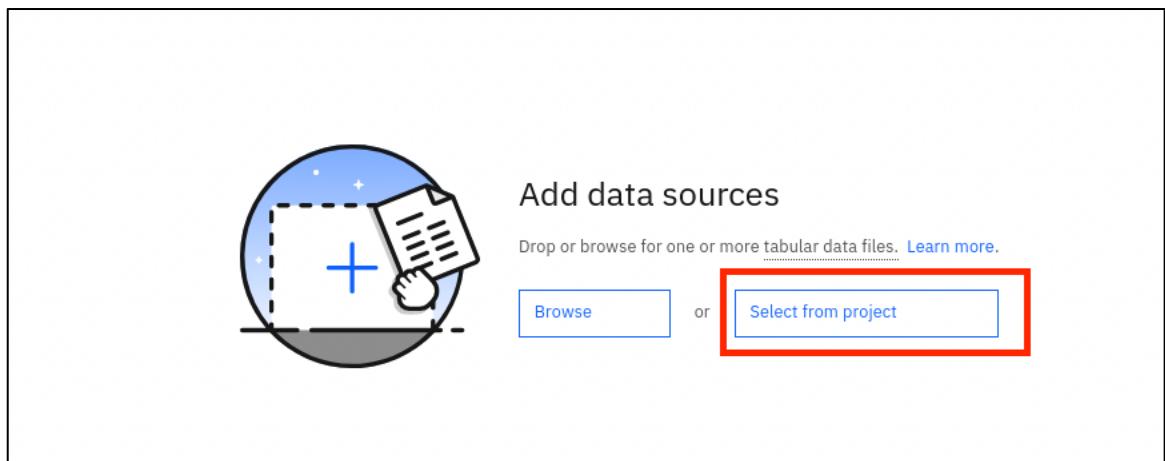
- 1) In the project page under asset tab, go to the right hand side click “ADD to Project”
- 2) Add AutoAI Experiment to the project.



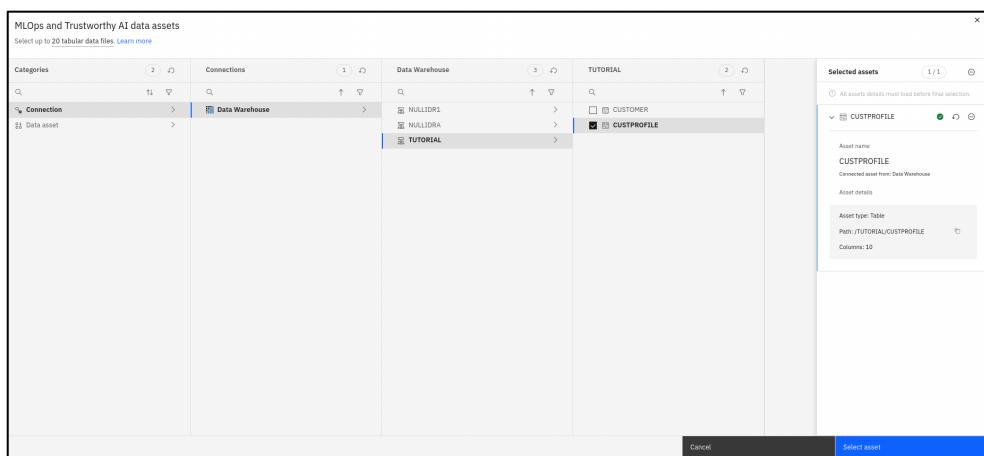
- 3) Give the name of the AutoAI Experiment and then create.



- 4) Add the data source, in our case we will use the connection from our project. So, choose select from project.

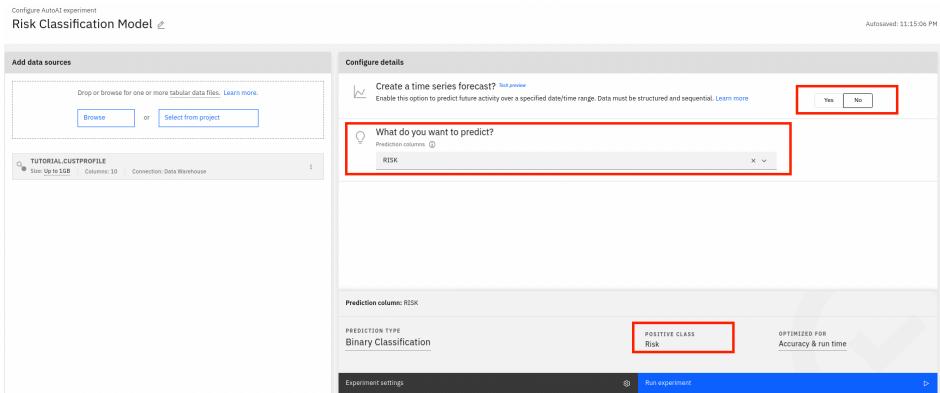


- 5) Select “Connection” > “Data Warehouse” > “TUTORIAL” > “CUSTOMERPROFILE” and select asset.



- 6) In the configuration page
- o Create a time series forecasting? Select “No”, because we will do the classification model to classify the customer “Risk”

- What do you want to predict? Choose the target variable in this case is “Risk”
- Select the Positive Class to be “RISK”. This is what we are interested in.
- Then run the experiment.



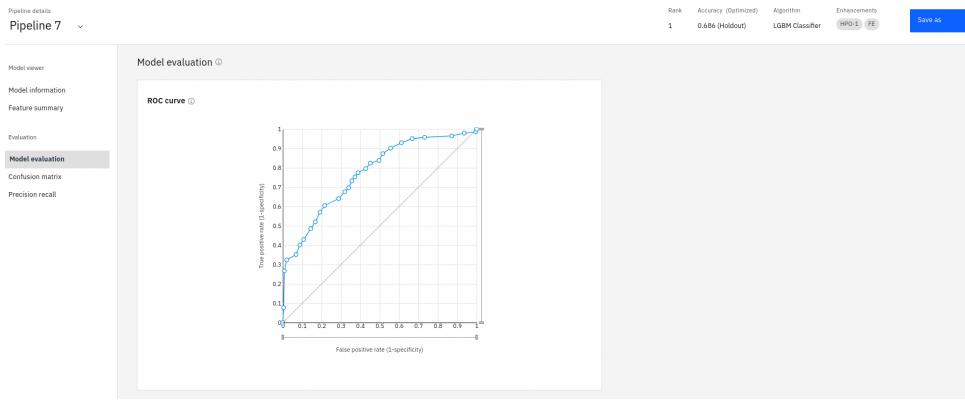
- 7) During the AutoAI Experiment has been run, you can see the Experiment Summary. Please explore the AutoAI Page and list down any questions you have.
- 8) In the pipeline leader board, you can see the eight model pipeline are generated and they're already ranked by performance (Accuracy).

	Rank	↑ Name	Algorithm	Accuracy (Optimized) Cross Validation	Enhancements	Build time
★	1	Pipeline 7	○ LGBM Classifier	0.733	HPO-1 FE	00:00:17
	2	Pipeline 8	○ LGBM Classifier	0.733	HPO-1 FE HPO-2	00:00:28
	3	Pipeline 3	○ Snap SVM Classifier	0.731	HPO-1 FE	00:00:14
	4	Pipeline 4	○ Snap SVM Classifier	0.731	HPO-1 FE HPO-2	00:00:01
	5	Pipeline 6	○ LGBM Classifier	0.727	HPO-1	00:00:16
	6	Pipeline 1	○ Snap SVM Classifier	0.720	None	00:00:01
	7	Pipeline 2	○ Snap SVM Classifier	0.720	HPO-1	00:00:01
	8	Pipeline 5	○ LGBM Classifier	0.705	None	00:00:01

- 9) Under name click the top pipeline in your leader board to see the information of the model.

Pipeline leaderboard ▾						
	Rank	↑ Name	Algorithm	Accuracy (Optimized) Cross Validation	Enhancements	Build time
★	1	Pipeline 7	○ LGBM Classifier	0.733	HPO-1 FE	00:00:17
	2	Pipeline 8	○ LGBM Classifier	0.733	HPO-1 FE HPO-2	00:00:28
	3	Pipeline 3	○ Snap SVM Classifier	0.731	HPO-1 FE	00:00:14

- 10) You can see the details of model performance. The model information, feature summary, model evaluation and metric are generated for data scientist to examine.



Congratulation!!! You have done for the model building part.

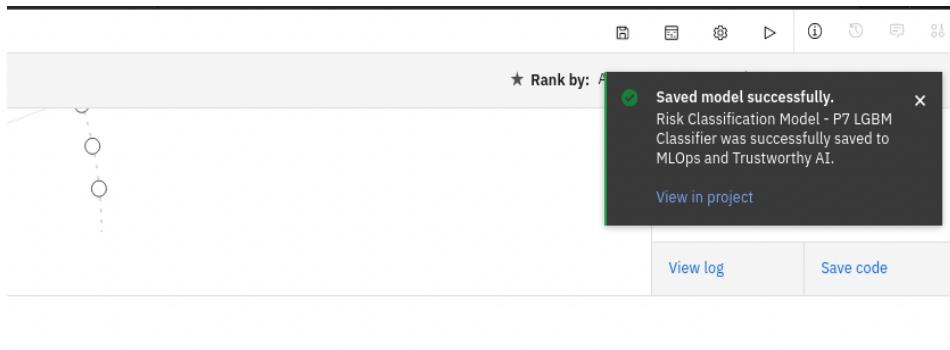
Task 2: Save and Deploy Model

- 1) Go back to the model leader board.
- 2) Click “Save as” on the top rank model.

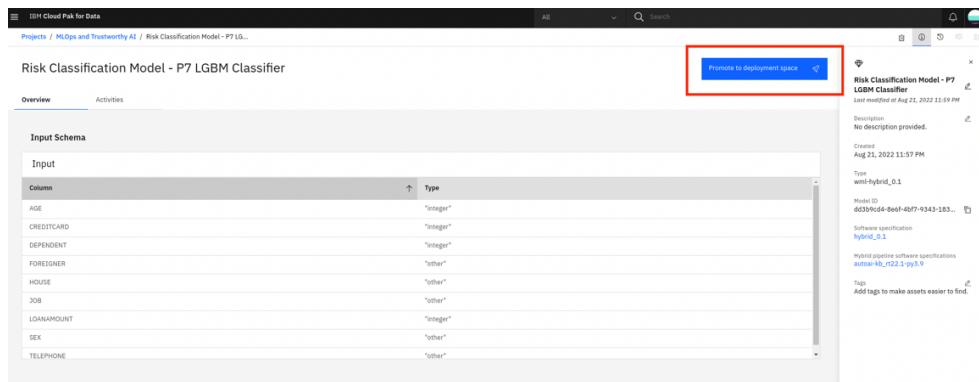


- 3) Go back to the model leader board.
- 4) Click “Save as” on the top rank model.
- 5) We keep everything default. We save the model as Model and Use the default name which generated by system. Click Create.

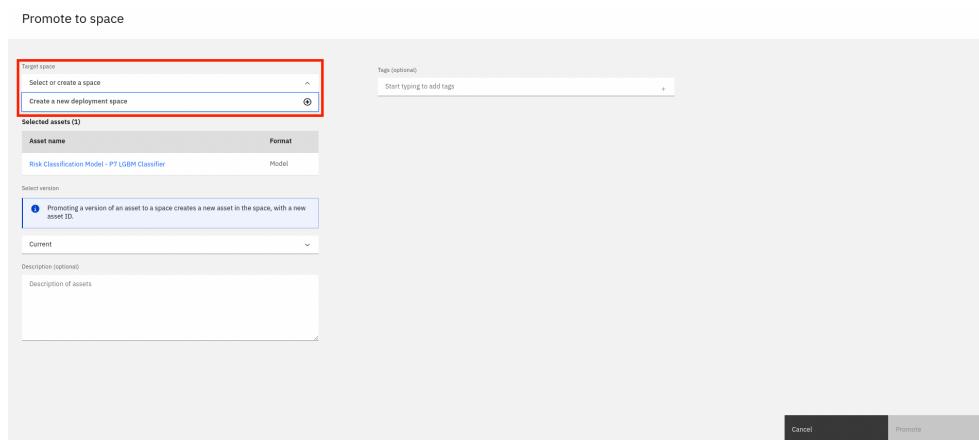
- 6) Once you save, it will back to AutoAI page, click view in project.



- 7) Review input schema and model information. On the top right corner, click "Promote to deployment space" to promote our model to deployment space (Watson Machine Learning).



- 8) Under target space, please click to create a new deployment space.



- 9) Give the deployment space name and then create.

Create a deployment space

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

Define space details

Name	Credit Risk
Description (optional)	Deployment space description

Deployment space tags (optional) (optional)

Add a tag

Create

- 10) Back to promote to space page, select the target space that we just already created and keep everything default. Then click promote to promote our model.

Promote to space

Target space
Credit Risk

Go to the model in the space after promoting it

Selected assets (2)

Asset name	Format
Risk Classification Model - P7 LGBM Classifier	Model

Select version

Promoting a version of an asset to a space creates a new asset in the space, with a new asset ID.

Current

Description (optional)

Promote

- 11) Once you promote it, it will lead you to the project page. You can see your model is saved and on the top right corner click "go to the deployment space"

IBM Cloud Pak for Data

Projects / ML Ops and Trustworthy AI

Overview Assets Environments Jobs Access control Settings

What assets are you looking for?

Data assets

0 assets selected.

Name	Type	Created by	Last modified
Data Warehouse	Connection	admin	Aug 21, 2022, 10:42 PM

AutoAI experiments

Name	Status	Model type	Last modified
Risk Classification Model	Completed	Binary Classification	Aug 21, 2022, 11:22 PM

Models

Watson Machine Learning models

Name	Type	Software specification	Last modified
Risk Classification Model - P7 LGBM Classifier	wml-hybrid_0.1	hybrid_0.1	Aug 21, 2022

Success message

Successfully promoted Risk Classification Model - P7 LGBM Classifier to the deployment space. Go to the deployment space to explore the assets for deployment.

- 12) On the deployment space under asset tab, you can see your model is registered to the repository. Click at your model name.

13) Click new deployment to deploy your model.

14) Under create a deployment page.

- Choose deployment type to be “Online” because we need the model to return result on-line (real-time) upon inference.
- Give the name of this deployment
- Create.

15) Wait until the model is deployed successfully and then please click at the name of the model to see the details of deployment.

The screenshot shows the 'Deployments' section of the IBM Cloud Pak for Data interface. It displays a table of deployments under the 'Online' category, with one entry: 'Credit Risk Model LGBM Deployment' which is 'Deployed'. To the right of the table, there is a detailed view of the deployment, including its name, status, creation date, type (wmn-hybrid_0.1), model ID, software specification (hybrid_0.1), and source asset details.

16) Under deployment page, you can see the detail of the deployment.

Explore API Reference and Test

- API Reference – The endpoint and code snippets are created for further use in the other front-end system.
- Test – The testing area to test the API and model inference.
- Next step in the test tab, we will test the endpoint and inference. Please enter input data as follow;
 - LOANAMOUNT : 2000
 - SEX : male
 - AGE : 32
 - HOUSE : own
 - CREDITCARD : 1
 - JOB : skilled
 - DEPENDENT : 1
 - TELEPHONE : yes
 - FOREIGNER :no
- Add to list > Predict.
- Check the model can give you the result properly or not.

Congratulation!!! You have done for the model deployment and testing part.

Task 3: Set up Model Monitoring with Watson OpenScales

- 1) Go back to the IBM Cloud Pak for Data Navigation bar. Service > Instance
- 2) Check the OpenScale instance is already created. If not, please contact your facilitator.

The screenshot shows the 'Instances' section of the IBM Cloud Pak for Data interface. It lists several instances, each with details like Name, Type, Created by, vCPU requests, Memory requests (GiB), Users, Status, and Created on. One instance, 'openscale-defaultinstance', is highlighted with a red box.

Name	Type	Created by	vCPU requests	Memory requests (GiB)	Users	Status	Created on
Db2 Warehouse-1 Db2 Warehouse 11.5.7.0-cn3-x86_64	db2wh	admin	6.20	18.25 Gi	1	Green	Aug 4, 2022
Db2 Warehouse-2 Db2 Warehouse 11.5.7.0-cn3-x86_64	db2wh	admin	6.20	18.25 Gi	1	Green	Aug 4, 2022
data-virtualization	dv	admin	11.60	38.70 Gi	1	Green	Apr 5, 2022
data-management-console Data Management Console	dmc	admin	3.70	9.50 Gi	1	Green	Apr 5, 2022
openscale-defaultinstance IBM Watson OpenScale	aos	admin	-	-	1	Green	Apr 5, 2022

- 3) On the right hand-side of OpenScale instance row, you will see triple dot. Click “Open” to the OpenScale Application.
- 4) On the OpenScale landing page, click “Manual setup”
- 5) In the system setup, first set up the database.
 - o Choose the database type to be “DB2”. We will use the second DB2 Data warehouse which is created for OpenScale (not the same as we use to keep training data)
 - o Go back to service > instance to get the data warehouse credential.

The screenshot shows the 'Details' page for 'Db2 Warehouse-2'. It includes sections for Deployment id, Created on, Status, Nodes, and Access information.

Deployment id		db2wh-1659596976865976																	
Created on	Aug 4, 2022 3:09 AM																		
Status	Available																		
Nodes	<table border="1"> <thead> <tr> <th>HOSTNAME</th> <th>CPU</th> <th>MEMORY</th> <th>Storage class (Temporary table spaces storage)</th> <th>Size (Temporary table spaces storage)</th> <th>Storage class (Transaction logs storage)</th> <th>Size (Transaction logs storage)</th> <th>managed-nfs</th> </tr> </thead> <tbody> <tr> <td>master1</td> <td>6.1 cores</td> <td>18.0 GiB</td> <td>managed-nfs</td> <td>100 GiB</td> <td>managed-nfs</td> <td>100 GiB</td> <td>managed-nfs</td> </tr> </tbody> </table>			HOSTNAME	CPU	MEMORY	Storage class (Temporary table spaces storage)	Size (Temporary table spaces storage)	Storage class (Transaction logs storage)	Size (Transaction logs storage)	managed-nfs	master1	6.1 cores	18.0 GiB	managed-nfs	100 GiB	managed-nfs	100 GiB	managed-nfs
HOSTNAME	CPU	MEMORY	Storage class (Temporary table spaces storage)	Size (Temporary table spaces storage)	Storage class (Transaction logs storage)	Size (Transaction logs storage)	managed-nfs												
master1	6.1 cores	18.0 GiB	managed-nfs	100 GiB	managed-nfs	100 GiB	managed-nfs												

Access information

JDBC Connection URL	jdbc:db2://<CLUSTER_ACCESSIBLE_IP>:32236/BLUDB:user=-;password=<password>;securityMechanism=9;encryptionAlgorithm=2
JDBC Connection URL (SSL)	jdbc:db2://<CLUSTER_ACCESSIBLE_IP>:30057/BLUDB:user=-;password=<password>;securityMechanism=9;sslConnection=true;encryptionAlgorithm=2

- o Go back to the openpage data base configuration page and fill in the credential information. Then get connect.

System setup

Connect to a database, machine learning providers, and integrated services. Optionally enable batch support.

Database

Required

Machine learning providers

Batch support (optional)

Integrations (optional)

Users & roles

Database

Required

Database

Select a database to store your model transactions and model evaluation results. For Db2 options that are part of your cluster, see [Services](#), [Data Sources](#) where you find options, such as Db2 Warehouse and Db2 Advanced Enterprise Server Edition. For an external database, you can use [IBM DB2 Database](#).

Database type

Db2

Database location

Other

Hostname or IP address

master1

Port

32236

Use SSL

Database

BLUDB

Username

admin

Password

Connect Success

- Check the database is connected to OpenScale successfully.
- Then scroll down a bit, under schema choose Auto-create a new schema, then save.

Database

BLUDB

Username

admin

Password

Connect Success

Schema

No schema found

Auto-create a new schema

Cancel Save

- Finish OpenScale database setup.

6) Setup machine learning provider.

- On the machine learning providers page, click “add machine learning provider”

System setup

Connect to a database, machine learning providers, and integrated services. Optionally enable batch support.

Machine learning providers

Description

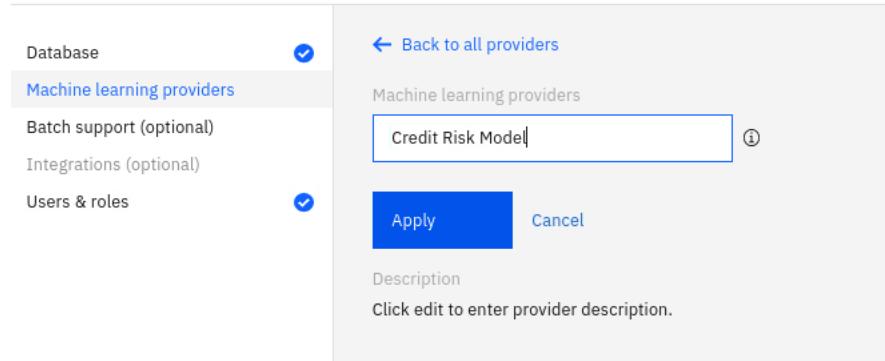
Watson OpenScale connects to deployed models stored in a machine learning environment.

Add machine learning provider

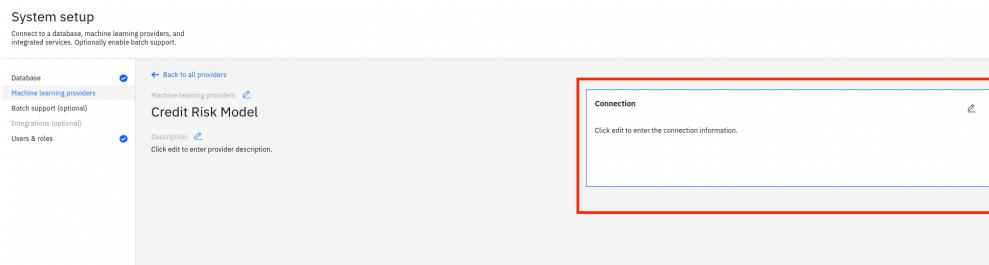
- Give the machine learning provider name.

System setup

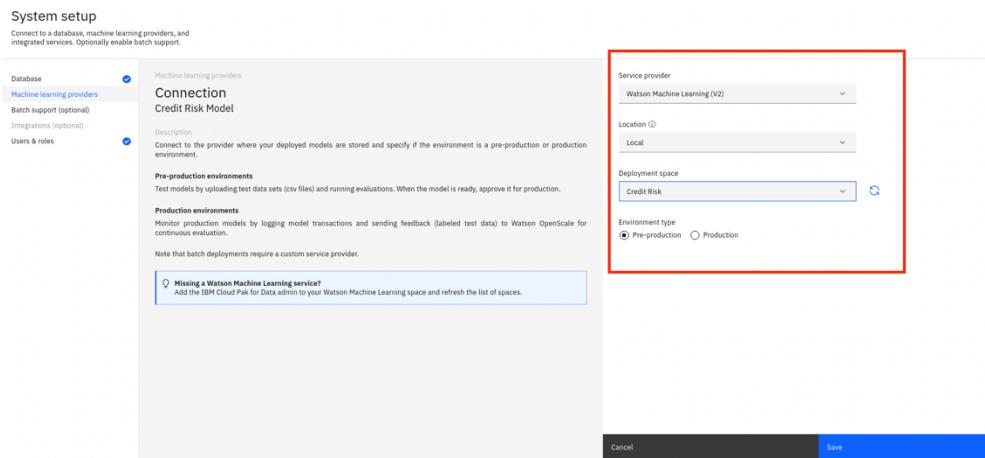
Connect to a database, machine learning providers, and integrated services. Optionally enable batch support.



- Click connection to set up the connection with machine learning provider.

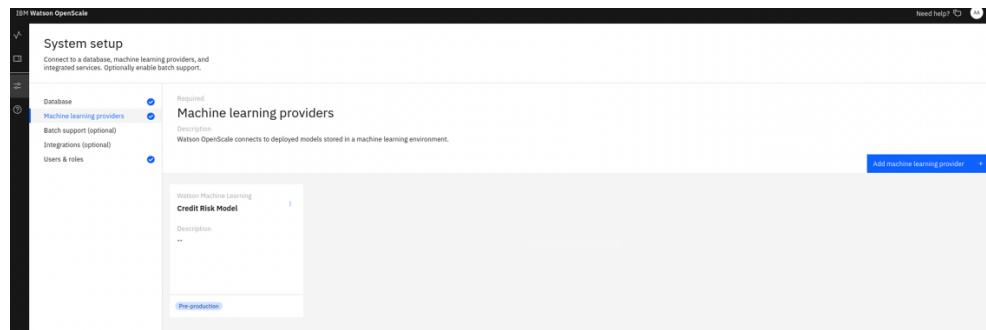


- Under connection set up page, please configure it as follows;
 - Service provider : Watson Machine Learning (V2)
 - Location : Local
 - Deployment space : <your deployment space name>
 - Save.



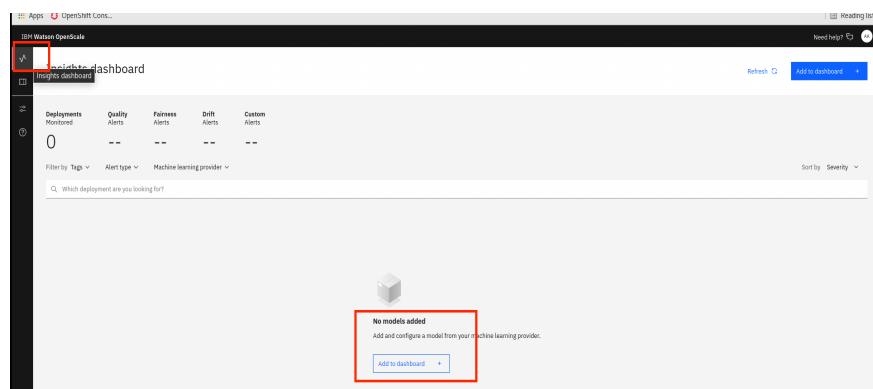
- Machine Learning Provider is setup successfully.

-

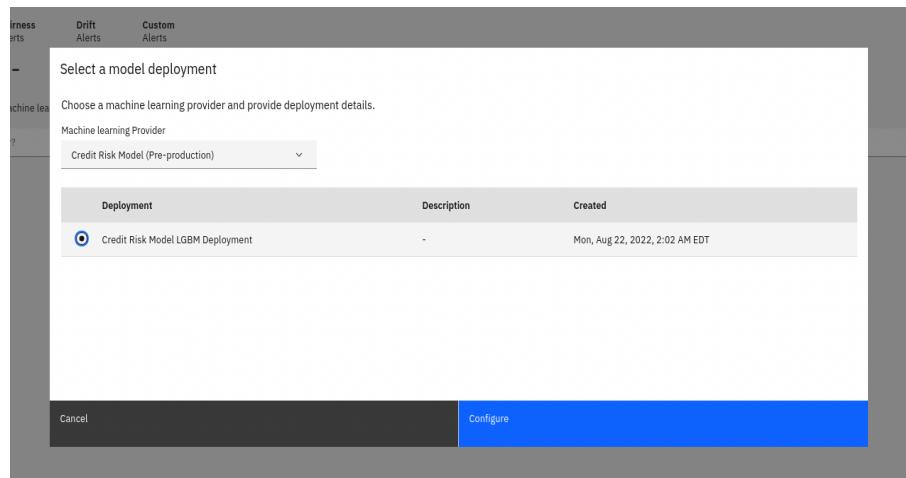


7) Setup OpenScale dashboard.

- Click insight dashboard on the left hand-side of the menu bar.
- From the insight dashboard page, click add to dash board.



- Add our machine learning provider to the dashboard.



- Our machine learning provider is added to the dashboard successfully.

The screenshot shows the 'Insights dashboard' for IBM Watson OpenScale. At the top, there are tabs for 'Deployments', 'Quality Alerts', 'Fairness Alerts', 'Drift Alerts', and 'Custom Alerts'. Below these, a summary table shows 1 deployment monitored, with all alert counts at zero. A search bar at the bottom allows filtering by 'Tags', 'Alert type', and 'Machine learning provider'. A detailed view for 'Credit Risk Model' is expanded, showing 'Issues' (0) and a 'Configure monitors' button highlighted with a red box.

8) Configure monitors.

- From the machine learning provider, click triple dots and configure monitor.

This screenshot shows a detailed view of a deployment. At the top, it lists 'Deployments Monitored' (1), 'Quality Alerts' (0), 'Fairness Alerts' (0), 'Drift Alerts' (0), and 'Custom Alerts' (0). A search bar below allows filtering by 'Tags', 'Alert type', and 'Machine learning provider'. The main section shows a deployment named 'Credit Risk Model' with a status of 'Credit Risk'. Under 'Issues', the 'Configure monitors' button is highlighted with a red box. Other options shown include 'View details', 'Remove deployment', and buttons for 'Pre-production' and 'Evaluation pending'.

- Model details configuration.

- Model input:
 - Data type : Numeric/categorical
 - Algorithm type : Binary Classification
 - Save and continue.
- Training data:
 - Going to <https://www.ibm.com/cloudibm> and login to IBM console.
 - From the navigation bar left hand-side, select the resource list.
 - Find Storage and select your object storage.
 - Create bucket
 - Choose quickly get start
 - Give a bucket name for example “trainingdataforwos”
 - Next and the bucket will be created.
 - Upload the training data to the “trainingdataforwos” bucket

- Back to service credential to see the credential information.
- Back to model detail > Training data setup in OpenScale. Then configure and fill in Object Storage credential
 - Storage type : Database or cloud storage
 - Location : Cloud object Storage
 - Resource instance id : <your object storage resource instance id>
 - API Key : <your object storage API key>
 - Click connect
 - Choose the bucket
 - Choose the data set.
 - Next

Storage type
Database or cloud storage

Location
Cloud Object Storage

Resource instance ID
crn:v1:bluemix:public:cloud-object-storage:global:a/432ff2d283604e4f8a4e4c

API key
.....

Bucket
trainingdataforwos

Data set
training_data.csv

Cancel Next

- Select label columns
 - Choose “Risk”

- **Next**

Select the label column		X
Features (10)	Type	
AGE	81	
CREDITCARD	81	
DEPENDENT	81	
FOREIGNER	A	
HOUSE	A	
JOB	A	
LOANAMOUNT	81	
RISK	A	
SEX	A	
TELEPHONE	A	

Back **Next**

- **Select the training features**

- Choose all feature.
- **Next**

Select the training features Selected features 9

Features (9) Type Categorical

<input checked="" type="checkbox"/>	AGE	81	<input type="checkbox"/>
<input checked="" type="checkbox"/>	CREDITCARD	81	<input type="checkbox"/>
<input checked="" type="checkbox"/>	DEPENDENT	81	<input type="checkbox"/>
<input checked="" type="checkbox"/>	FOREIGNER	A	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	HOUSE	A	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	JOB	A	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	LOANAMOUNT	81	<input type="checkbox"/>
<input checked="" type="checkbox"/>	SEX	A	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	TELEPHONE	A	<input checked="" type="checkbox"/>

Items per page: 25 1 - 9 of 9 items 1 of 1 page ◀ ▶

Back Next

- Configure Scoring
 - Scoring Method : JSON payload
 - Down Load JSON_Payload_Example from Github
https://github.com/sandeepved/ASEANZK-CP4D-Practicum/blob/main/scenario2/data/JSON_Payload_Example.
 - Open the file and Copy the content.
 - Paste the content to the blank under enter a sample request and repones.
 - Click send now
 - Next.

Scoring method

JSON payload

Enter a sample request and response. Replace the fields and values to match your model input and output. [Learn more.](#)

```
prediction :  
  "predictedLabel"  
],  
"values": [  
  [  
    2000,  
    "male",  
    32,  
    "own",  
    1,  
    "skilled",  
    1,  
    "yes",  
    "no",  
    1,  
    "RISK"  
  ]  
]  
}  
]
```

Scoring request successful Click Next

Send now

Back Next

The screenshot shows a user interface for a machine learning model's scoring endpoint. At the top, it says 'Scoring method' and 'JSON payload'. Below that is a text area containing a JSON object representing input features. A message box at the bottom says 'Scoring request successful Click Next'. There is a 'Send now' button and navigation buttons 'Back' and 'Next' at the bottom.

- Specify the model output.
 - Ensure you select prediction and probability.
 - Save

Use the checkboxes to make one or more selections

Features (3)	Type	Prediction	Probability
predictedLabel	A	<input type="checkbox"/>	0
prediction	B	<input checked="" type="checkbox"/>	0
probability	B	<input checked="" type="checkbox"/>	0



- Explainability
 - Explainability will be configured automatically, once you completed the previous steps correctly.

Important Note:

- You may face the error below after completing the configurations above. This is a known issue for this specific environment, we are investigating the root-cause. Please skip and ignore expandability for this lab.

Explainability

Controllable features

Description

Watson OpenScale can reveal changes that will result in a different model outcome for a selected transaction. You can choose to include or exclude features that you cannot control when running an analysis.

Note: All features (controllable and non-controllable) will be analyzed to determine which features were most important in determining the model outcome.

Error

Action 'Enable Monitor' has failed with status code 400
Binary classification model can not have more than two class labels

OK

Controllable features

Q

Feature

LOANAMOUNT

DEPEN

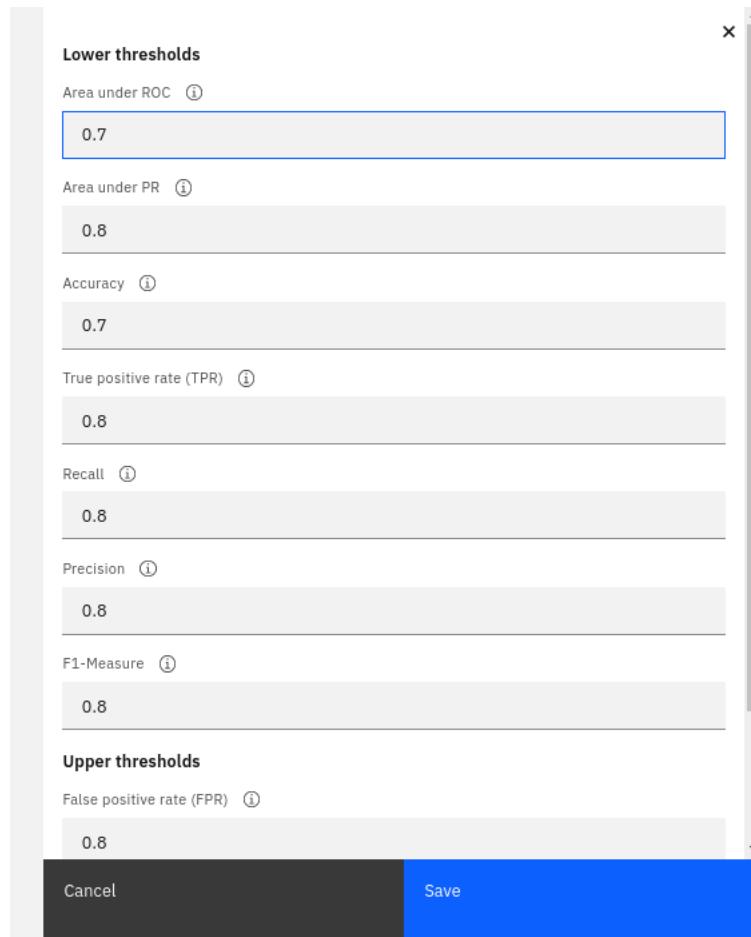
RD

DEPENDENT

- Quality
 - Click Quality
 - Quality threshold > edit

- The lower threshold and upper threshold is the configuration if the monitor metric get lower up higher than the threshold we set, it will be alert. Try to change some lower threshold (the

- Area under ROC : 0.7
- Accuracy : 0.7
- Save



- Sample size is the size of sample to get evaluate. We will keep 1,000 as default.

- Drift

- Click Drift
- Drift Model > edit
- Training option
 - Choose Train in Watson OpenScale (Suitable when you connected your training data to Watson OpenScale and it is less than 500 MB)
- Drift thresholds
 - There are two drift metrics, drop in accuracy and drop in data consistency. You can set the thresholds to get alert. We will

set to 5 percent., so it will give alert if drop in accuracy or data consistency greater than 5 percent.

- Drop in accuracy : 5%
- Drop in data consistency 5%

- Sample size is the size of sample to get evaluate. We will keep 1,000 as default.

- Save

- Fairness

- Click Fairness

- Favourable Outcome > edit

- Add 0 and 1 value
- Set 0 to favourable outcome
- Set 1 to unfavourable outcome
- Save

Select the favorable outcomes

x

1

- + Add value

Favorable values 1 Unfavorable values 1

Values	Favorable	Unfavorable
0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
1	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Cancel Save

- Sample size is the size of sample to get evaluate. We will keep 1,000 as default.
- Select the fields to monitoring
 - Select SEX (We will monitor bias based on SEX attributes)

Select one or more fields		
	Selected fields	1
Fields	Recommended	Type
<input type="checkbox"/> LOANAMOUNT		
<input checked="" type="checkbox"/> SEX		
<input type="checkbox"/> AGE		
<input type="checkbox"/> HOUSE		
<input type="checkbox"/> CREDITCARD		
<input type="checkbox"/> JOB		
<input type="checkbox"/> DEPENDENT		
<input type="checkbox"/> TELEPHONE		
<input type="checkbox"/> FOREIGNER		

- Specify the monitored groups for [SEX]
 - Add custom value
 - Type “male”, then add value. The check reference for male.
 - Type “female”, then add value. The check monitored for female.
 - Keep fairness alert threshold as default.
 - Save

Select the groups to monitor [SEX]

Add custom value

female Add value

Monitored group 1 Reference group 1

Values	Monitored	Reference	Recommended
female	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
male	<input type="checkbox"/>	<input checked="" type="checkbox"/>	

Set fairness alert threshold [SEX]

80

Congratulation!!! You have done for the model monitoring set up part.

Please note that we will not see any alert in the dashboard yet because there are no any new payload load to the model. Once, the payload reached out the threshold sample that we set (1,000 sample), it will evaluate and show the results.