# Lab 2: AutoAI Model Development & Deployment

## Introduction:

In this lab you will develop a model for deployment and governance.

The first model, in lab 2, we will use **AutoAI** which is part of **watsonx.ai.** AutoAI in watsonx.ai 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, perform evaluations of fairness, 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 problem optimization objective

The second model, in lab 3, will use a traditional code base ML development process with scikit-learn and Python.

### AutoAI:

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.

## Objectives:

The AutoAI process follows this sequence to build candidate pipelines:

* Data pre-processing
* Automated model selection
* Automated feature engineering
* Hyperparameter optimization

## German Credit Risk Synthetic Dataset

This dataset contains synthetic representations of credit applications. The data itself is varied and has demographic, financial, and other elements. It also contains a label column that determines if they are a risk, as it is a supervised learning model from prior data. We will use this dataset to generate a model that predicts if someone should be accepted or denied credit based on this data.

The data is available here as well as in the workshop box folder; <https://dataplatform.cloud.ibm.com/exchange/public/entry/view/ded91b00566440f4712f61e7a805b922?context=wx>

## Add the data to the Project

Use the project you created in lab 1, and add this dataset to it. You can either click add to project from the data page above to clone it into your project, or you can download the data and upload it in the UI. There are also lots of other connection methods available to data stores that is outside the scope of this lab.

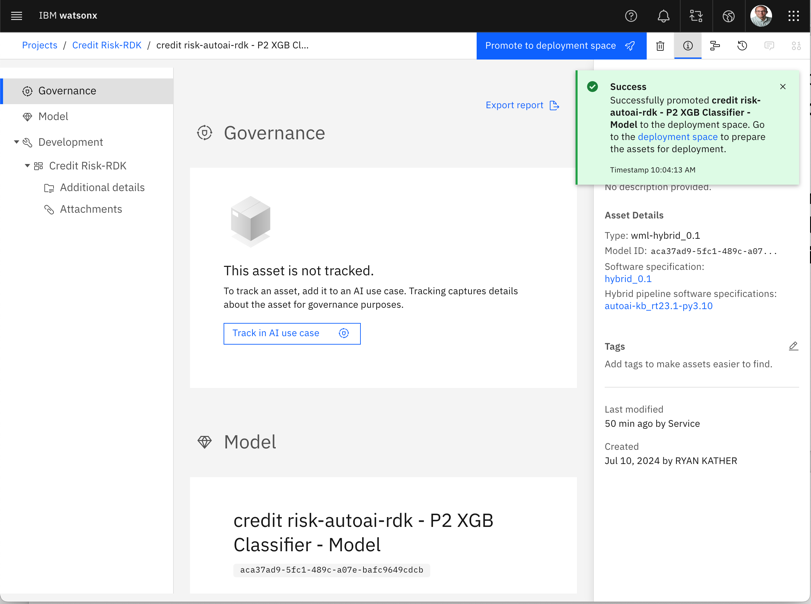
## Create the new AutoAI Experiment

1. From within your project click the **Assets** tab.
2. Click on **New Asset** and then click **Build Machine Learning Models Automatically** from the card selection options.
3. Name the model **Credit Risk – AutoAI {INITIALS}**  or whatever else you would like that is unique. (Replace initials with your own or some other unique identifier) and leave everything else as is.
4. Click **Create**

## Add file and Run Experiment

1. Under **Add Data source**, click on **Select from project**, choose the **german\_credit\_data\_biased\_training.csv** file under **Data Asset** by checking the check box. Click “Select asset”.
2. Answer no the time series prediction question.
3. Under **Select prediction column** click the **Risk** attribute

AutoAI assesses the characteristics of the attribute to be predicted and identifies the type of prediction method to be used. In the case of the Risk feature AutoAI chose a binary classification since Risk only has two distinct values.

1. Click on **Experiment settings**
2. On the **Data source settings** change the **Training data split** to **85%**
3. Click on **Prediction** under the **Experiment settings** and explore the settings. You can see that AutoAI defaulted to **Binary classification** and it selected the value **1** for the **Positive class** value. Scrolling down you can choose which metric to optimize, **Optimized metric**. There is a list of **Algorithms to include** when running the experiment. All these options will vary based upon which Prediction type that is chosen, Binary Classification, Multiclass classification, or Regression.
4. Explore the fairness tab, and feel free to add a fairness metric if desired for evaluation.
5. Click **Save settings**
6. Click **Run experiment** (This will take a few minutes), as it runs feel free to browse around the various items presented, such as logs, pipeline stages, etc…
7. When the experiment completes you will see the results of each of the pipelines under the **Experiment summary**. Explore areas of interest to you.
8. Click on the **Pipeline comparison** tab. Review the results for each of the **Optimized metrics** across the pipelines that were run. Below that you will see the ranked pipelines, **Pipeline leaderboard**.
9. Return to the **Experiment summary** tab.
10. Under the **Pipeline leaderboard** click on the **Pipeline with a rank of 1**
11. You can further explore the experiment.
12. To the right of the pipeline, Click on **Save as** and select **Model**. Keep the default Model name and click **Save.** This will save the model as a project asset in your project’s Watson Machine Learning instance and will be included Models asset list.
13. Click on **Save as** and select **Notebook**. This will create a Notebook for the flow which you can use as an example or reference going forward.
14. Keep the default Runtime **Runtime 23.1 Python …**
15. Click **Create notebook** (This might take a few minutes)
16. Explore the generated notebook
17. Click on the Project list (your project name in the breadcrumb menu)
18. Scroll down to the model you just generated in the Models section and Click on it to open it
19. Click on the **Promote to deployment space (top right)**
20. **Create a new deployment space** with a unique name of your choosing.
21. Indicate a production level (Development, Testing, Production) of your choosing.
22. Select the machine learning service you used for the project creation and click **Create**.
23. Click **Promote** to publish the model to the deployment space.
24. View the deployed model in the space with the notification link in the green box.
25. In the assets tab of the deployment space select the **elipses** of the deployed model and click **deploy** on the popup.A screenshot of a computer

    Description automatically generated
26. Select **Online** and give the deployment a name **CreditRisk{Initials}** where your initials are your own or a unique identifier.

1. Click **Create**

After a few moments (30 seconds), the status should change from “In Progress” to "Deployed".

1. Click on the newly **Deployed model**
2. Note the **Code Snippets** tab. You can see information on how to access the model. Note the code snippets for multiple languages at the bottom of the screen
3. Click on the **Test** tab
4. Enter relevant values into the **Enter input data**
5. Click **Predict**

The model is online, and predictions can be made by sending it data through REST.