Building and Deploying a machine learning model-

1. Create and Deploy a machine learning workspace with the details.

2. Select Go to resource once the deployment is complete-

3. Under the machine learning studio, select AutomatedML on the left pane of the page.
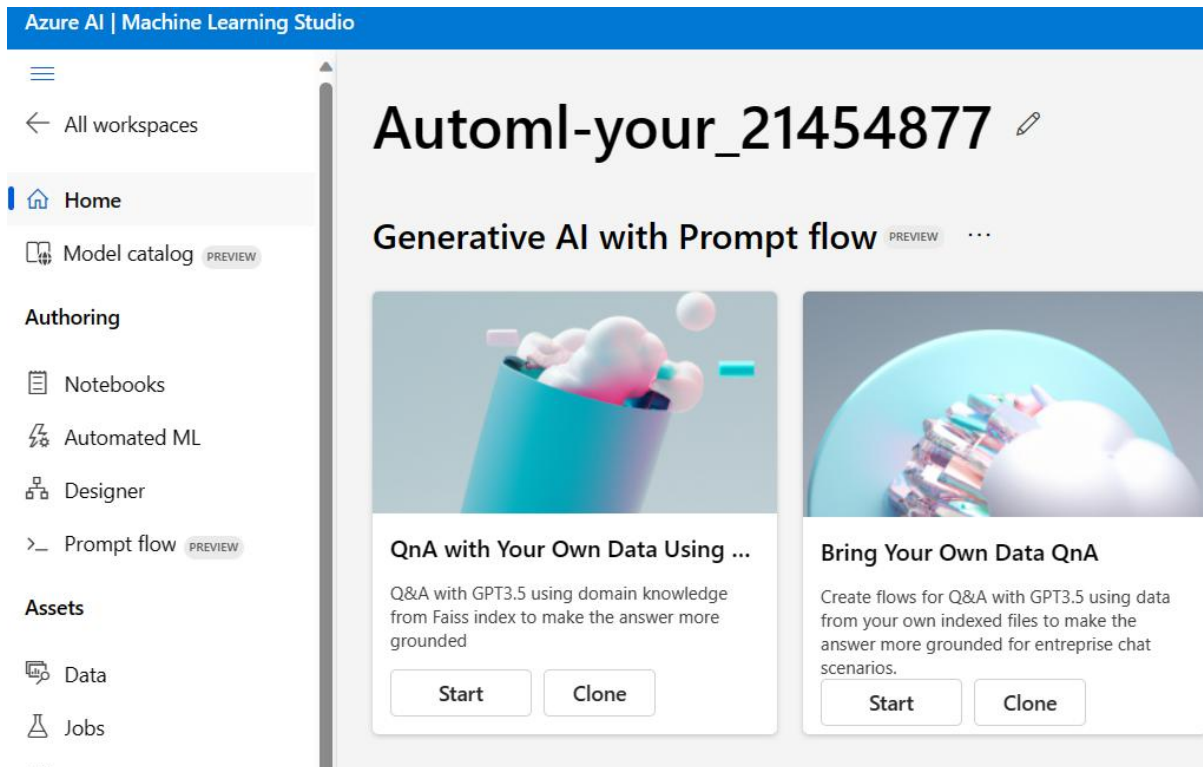
4. Create a new Automated job and select the options as seen below-
   - Training method- Train automatically, Click next
   - Basic Settings- Add the experiment name, the job name will be filled automatically
   - Task & Type data- Select as classification and add the bankmarketing_train csv file
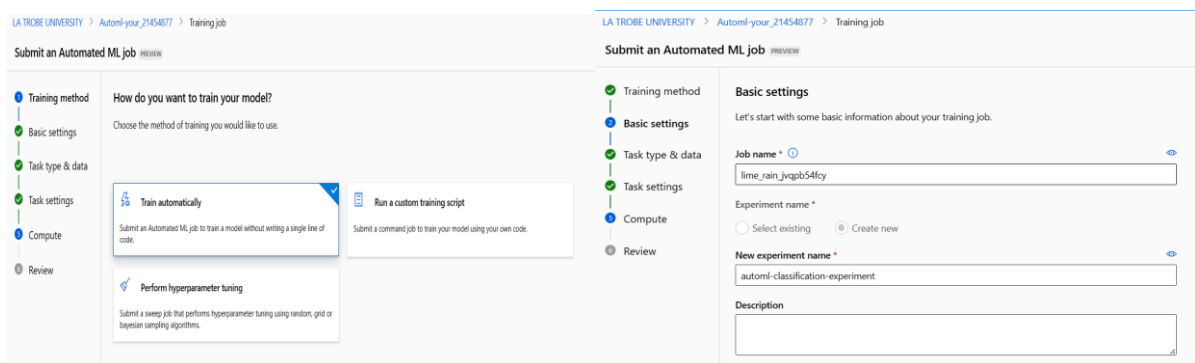   - Select the target as y and under Valid & test select k-fold cross validation and number of cross validations as 2.
   - Under Compute, select Compute instance and create a Compute Instance

5. Now click on Review and submit the job. Now wait for approximately 2 hours for the job to run and complete. Once the job is completed, we need to explore the models and describe how to select the best models.

6. To evaluate the models, Azure calculates the weighted AUC and selects the best model. In this case, we see that the VotingEnsemble is the best algorithm with a weighted AUC of 0.94829 and a 100% sampling.

Algorithm name	Explained	Responsible AI	AUC weighted ↓
VotingEnsemble	View explanation	View responsible AI dasl	0.94829
SparseNormalizer, XGBoostClassifier			0.94632
MaxAbsScaler, LightGBM			0.94606
StandardScalerWrapper, XGBoostClassifier			0.94593
SparseNormalizer, LightGBM			0.94593

7. On selecting the model, we can analyse the most important variables to be the duration, employed, variance rate and confidence index that's provided in the data.

Top 4 features by their importance
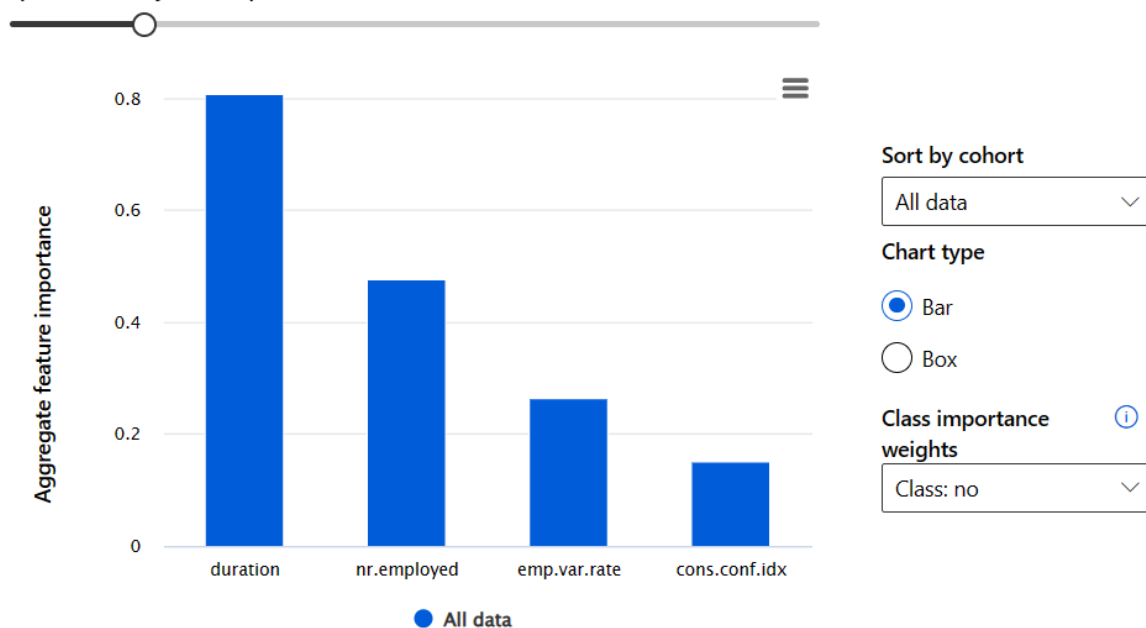
8. Now this model needs to be deployed, in order to deploy, open the selected model and click on Deploy> Real time endpoint and wait for 15 minutes for the model to deploy.
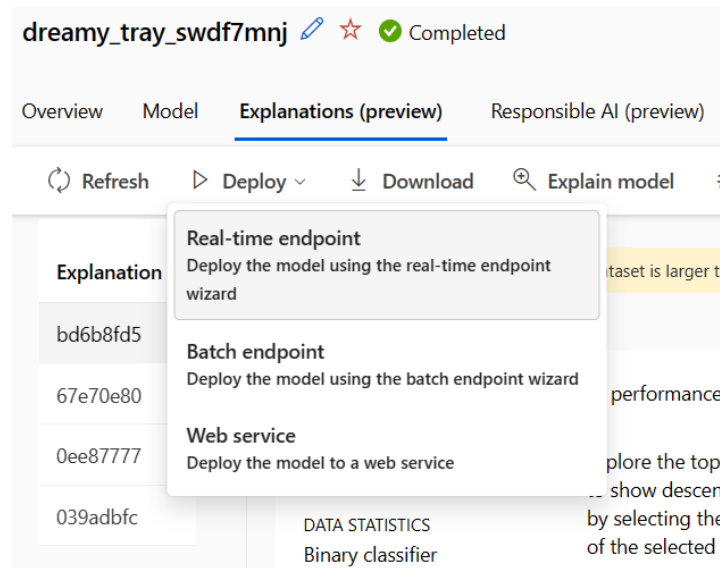
9. Now use this deployed model to classify the unknown data. To predict unknown data, click on the Endpoints option on the left pane. Under real time enpoints, we can see our selected model deployed. Open this model and select the Consume menu.

10. The python code to use for classification can be seen below, copy this python code for further use-
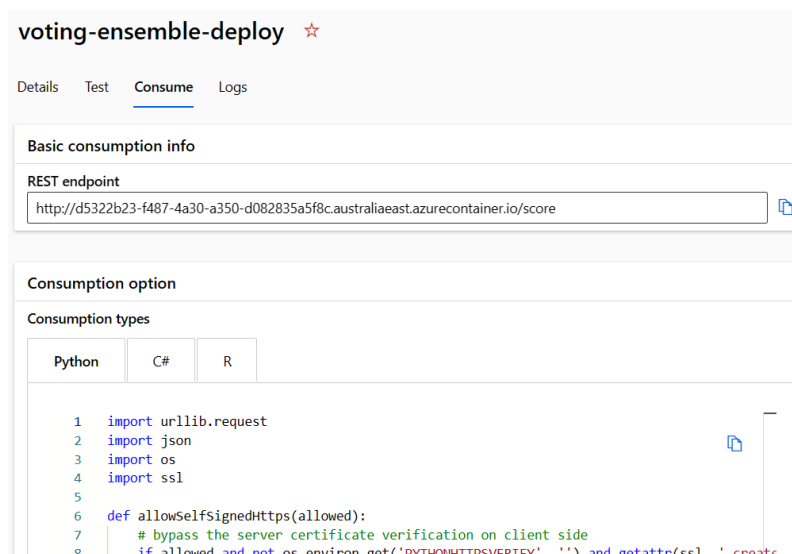
11. Select notebooks on the left pane and create a new notebook. Now create a new ipynb notebook and paste the code.
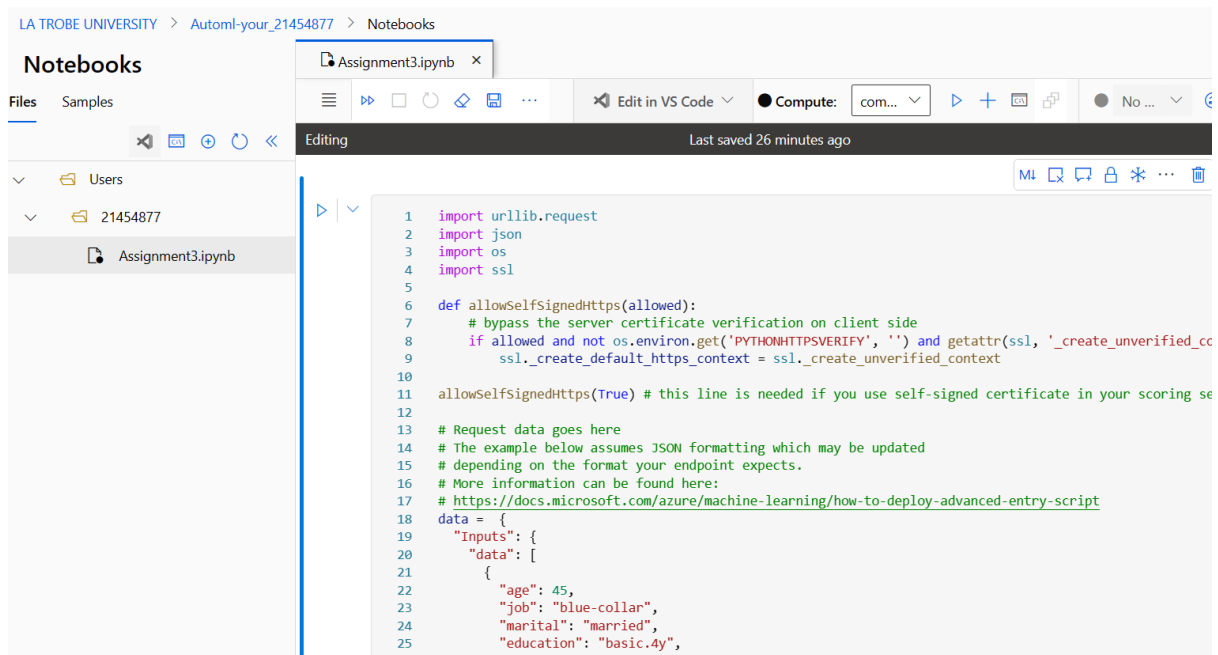
12. Replace the values in the data with the data provided in the classify file. Upon running each value, we get the predictions as below-

campaign.id	age	y
1501	41	Yes
1502	60	No
1502	35	No
1502	50	No
1502	50	No
1502	45	No

Benefits of no-code AutoML-

- **Saving resources**- AutoML accelerates the workflow as it allows the data scientists to spend more time on solving more complex problems rather than focusing on repetitive tasks. It increases their efficiency and thus reduces the time and capital needed to build the models.

- **Easy to implement**- AutoML makes machine learning accessible to broader audience. It can be used by business analysts and non technical users to analyze and predict the data they have. This democratizes the process and people with minimal coding experience can collaborate and innovate across teams. (*The complete list of pros and cons of Automl 2023*)

Drawbacks of no-code AutoML-

- **Lack of Control**- The models are created by data scientists in Azure and thus the data scientists of an organization won't have enough information about the running of the algorithm or fine tuning it for better results.

No domain specific knowledge- Prediction of outcomes greatly depends on the domain specific knowledge. These models make predictions just based on the data that is provided to them which is a smaller sample compared to the entire data. Still there are multiple factors that require a domain knowledge, which a no-code Automl lacks.