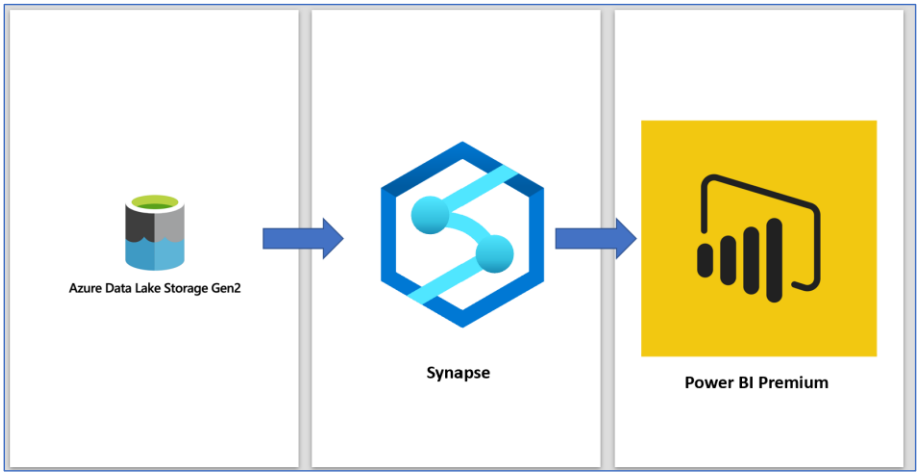
Tutorial: Create an AI/ML Model with Power BI Premium

In this tutorial, you learn how to create a powerful Classification Model without writing a single line of code using automated machine learning functionality provided within Power BI Premium. In our example, we will use an Accounts Receivable dataset and leverage Power BI premium to build a Classification Model that predicts if a customer will pay his/her account on time (i.e. before or on the invoice due date). Optionally – we could also run a Regression Model and predict when the customer will pay (i.e. the actual number of days it will take a customer will pay).

In this scenario, we have an Accounts Receivable dataset that resides in Azure Data Lake and can be queried with Azure Synapse Serverless. At a high level, the architecture of the lab looks as follows:



Architecture Components:

1. Azure Data Lake
   - Infinitely scalable azure storage
2. Synapse
   - Infinitely scalable compute that can be leveraged in a serverless or dedicated capacity
3. Power BI Premium
   - Analytics Platform providing users the ability to create AI/ML models

With automated machine learning functionality within Power BI Premium, you can automate away time intensive tasks of experimentation and testing ML models. Automated machine learning within Power BI Premium rapidly iterates over many combinations of algorithms and hyperparameters to help you find the best model based on a success metric of your choosing.

In this tutorial, you will go through the following high level tasks in order to complete the exercise end to end:

a) Login to powerbi.com and create a PowerBI Dataflow in your HackathonXX (XX denotes your hackathon # - i.e. Hackathon1, Hackathon2,and so on and so forth)
b) Go into the Power BI Dataflow created above and create a Machine Learning Model.
c) Review the Model Validation report.
d) Apply the model and see the predicted values on your dataflow dataset

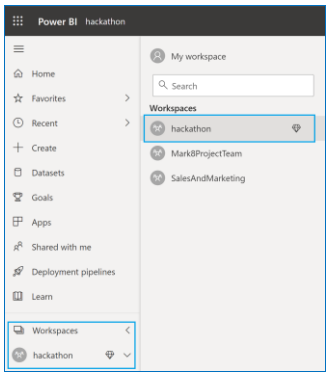
A) Login to Power BI and Create Dataflow

Login to *app.powerbi.com* using your credentials provided but should follow the format:

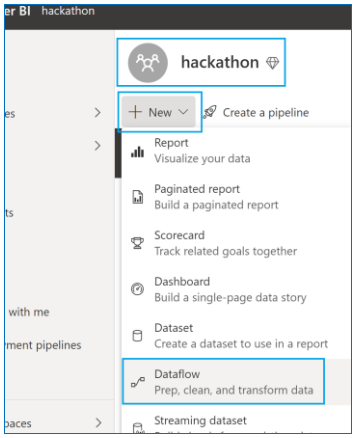
Username:

Password:

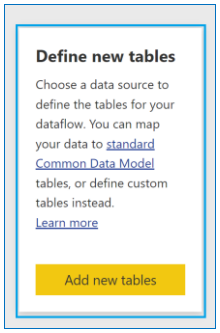
Go to your Hackathon workspace as shown below:



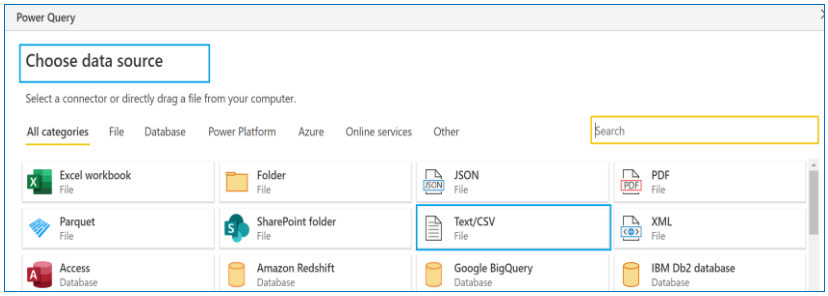
Create a Dataflow as by clicking on the + New button as shown:



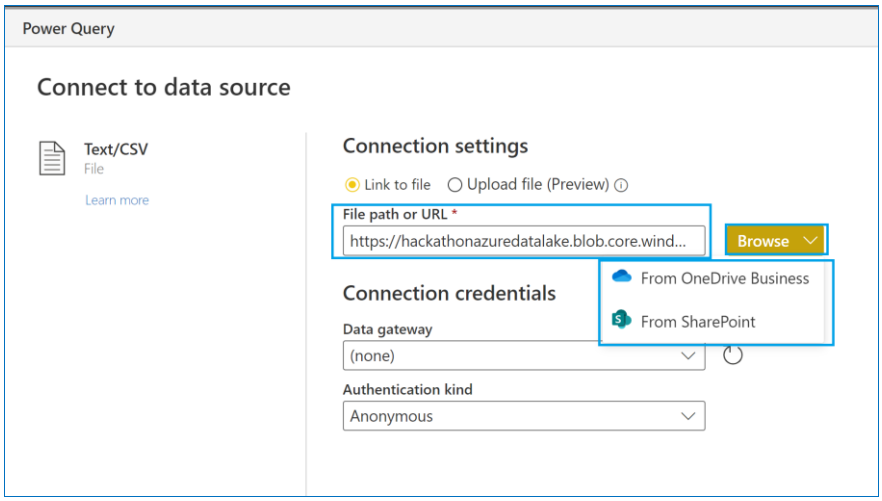
Click on Define New Tables:

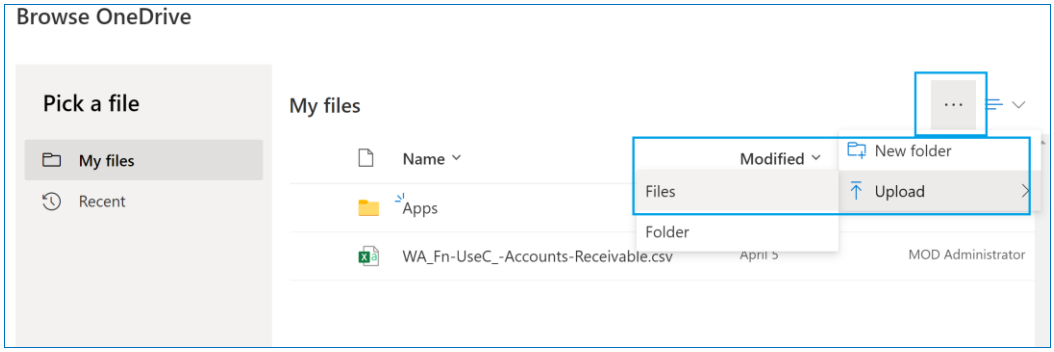


Choose a Data Source -> Text/CSV(didn't work)



Use Azure Data Lake:



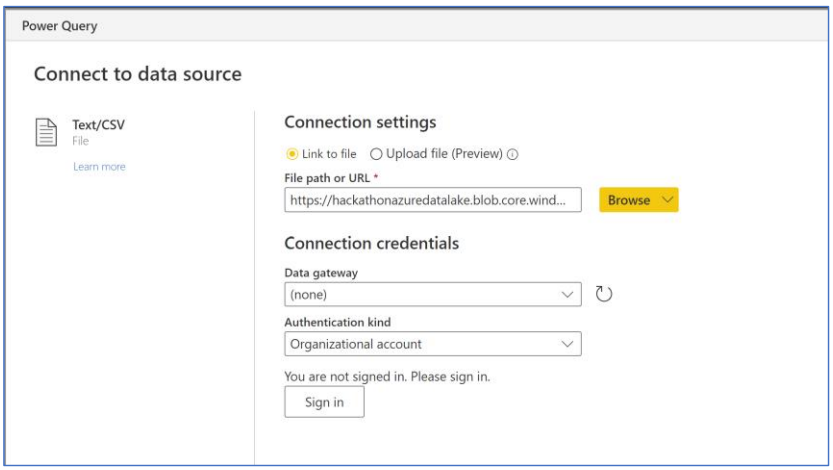


Click on Sign in – Organizational Account and enter your U/P (i.e. hacker@demotenant/password)

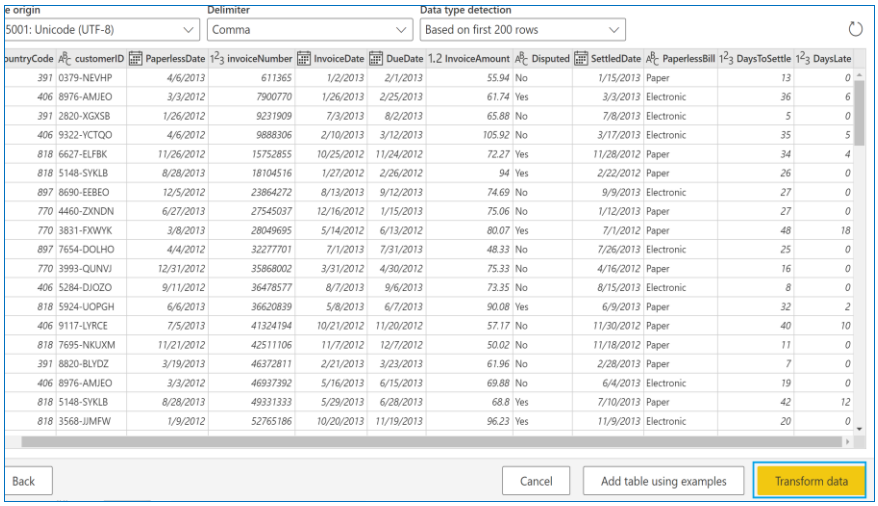
Data can be found on GitHub:

https://github.com/alpkayaMSFT/saponazureml/blob/main/WA_Fn-UseC_-Accounts-Receivable.csv

For File Path – Upload File (you can download the file locally and then upload)

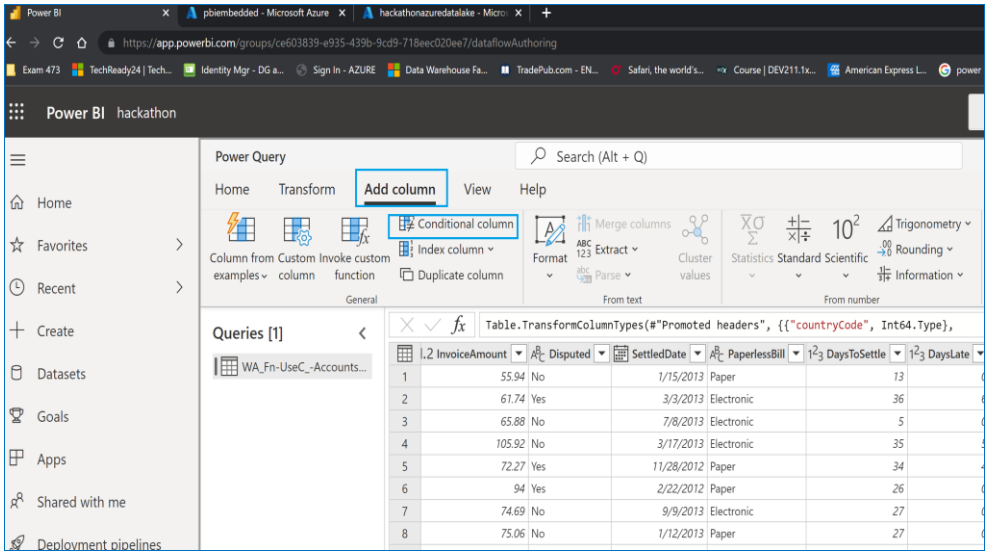


Click Next

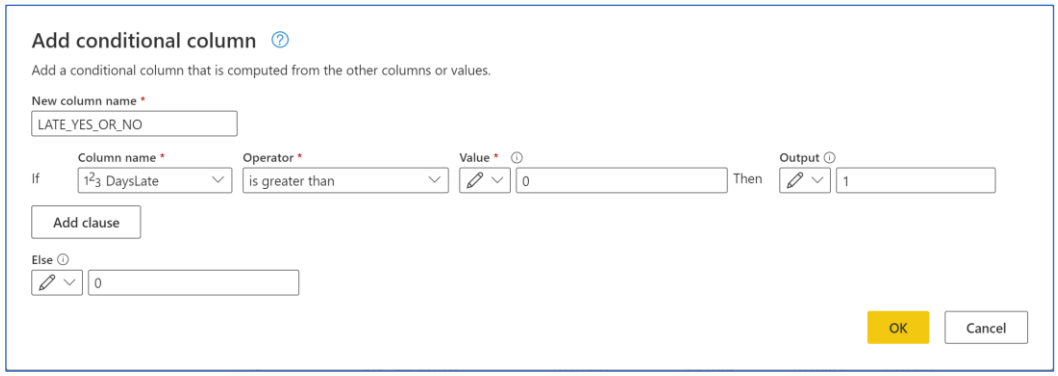


Click on Transform data

Click on Add Column > Conditional Column



In the conditional column select Days Late > 0 is late (i.e. 1) otherwise 0 (i.e. not late). The logic should look as follows:

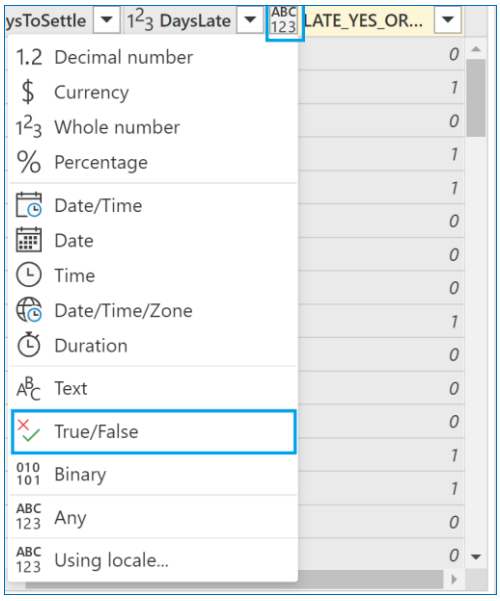


In PowerQuery it should look as follows:

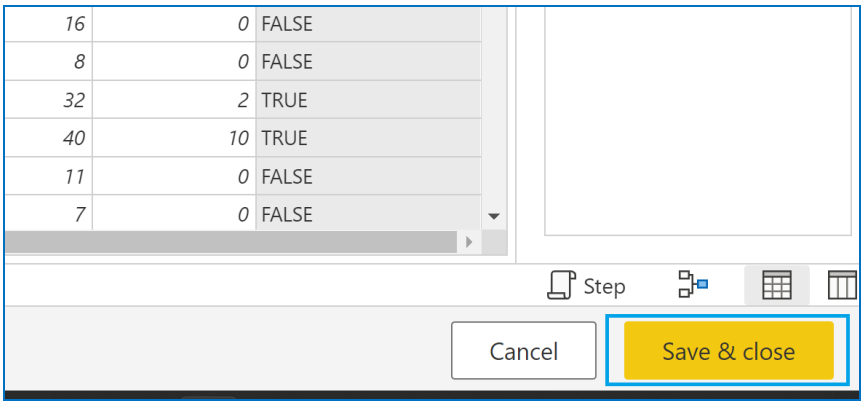
```
Table.AddColumn(#"Changed column type", "LATE_YES_OR_NO", each if [DaysLate] > 0 then 1 else 0)
```

Commented [ST1]: The screen shot has column name as 'Late', but he PowerQuery has different name

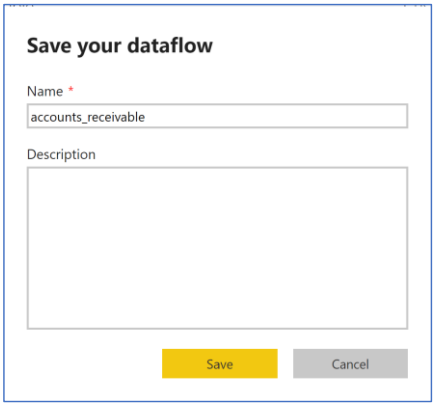
Click on the new conditional column you just created and change its datatype into Boolean (i.e. True/False)



Click Save & Close

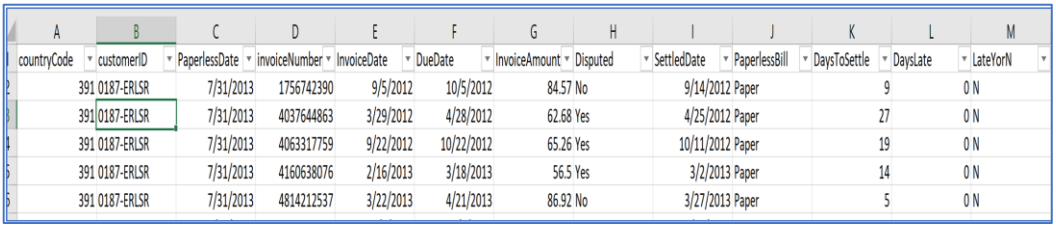


Name your dataflow – in our example we used accounts_receivable and click Save



B) Create a Machine Learning Model

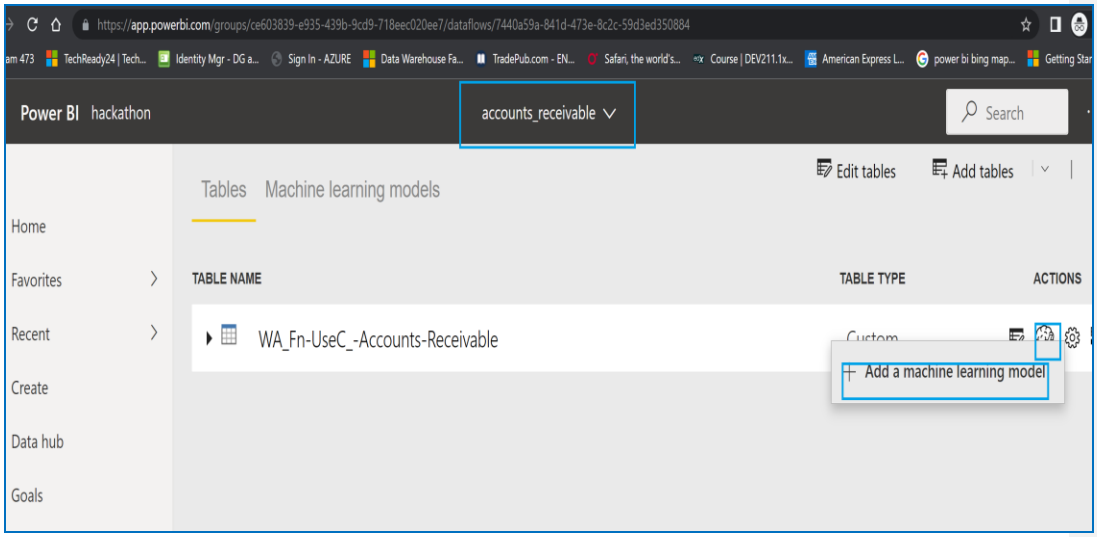
What does the Accounts Receivable Dataset Look like?



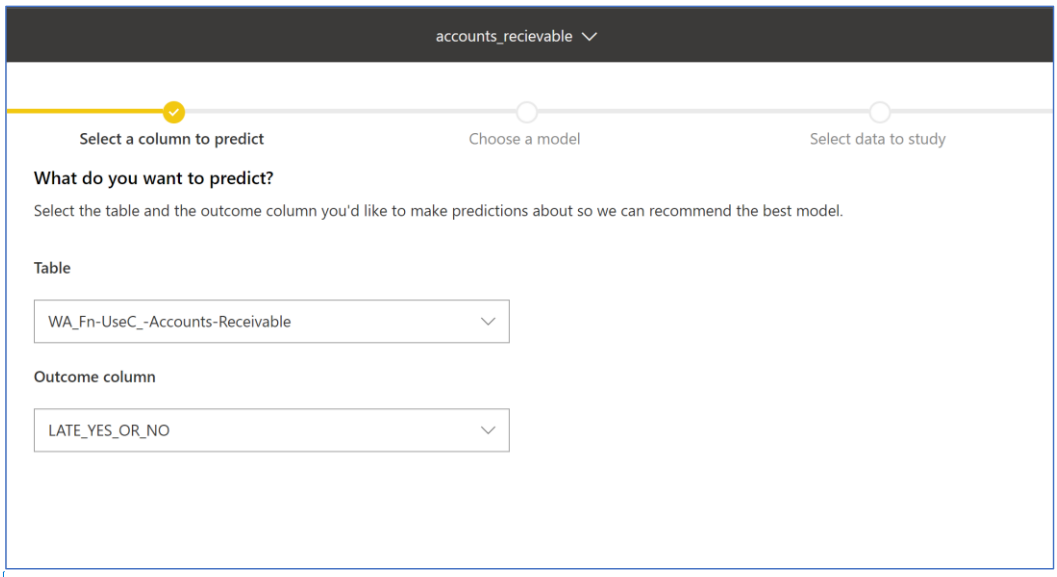
countryCode	customerID	PaperlessDate	invoiceNumber	InvoiceDate	DueDate	InvoiceAmount	Disputed	SettledDate	PaperlessBill	DaysToSettle	DaysLate	LateYorN
391	0187-ERLSR	7/31/2013	1756742390	9/5/2012	10/5/2012	84.57	No	9/14/2012	Paper	9	0	N
391	0187-ERLSR	7/31/2013	4037644863	3/29/2012	4/28/2012	62.68	Yes	4/25/2012	Paper	27	0	N
391	0187-ERLSR	7/31/2013	4063317759	9/22/2012	10/22/2012	65.26	Yes	10/11/2012	Paper	19	0	N
391	0187-ERLSR	7/31/2013	4160638076	2/16/2013	3/18/2013	56.5	Yes	3/2/2013	Paper	14	0	N
391	0187-ERLSR	7/31/2013	4814212537	3/22/2013	4/21/2013	86.92	No	3/27/2013	Paper	5	0	N

- CustomerID represents an ID for a customer
- InvoiceNumber is the number of the invoice for a customer
- InvoiceDate is the date that the customer was invoiced
- DueDate is the data that the bill is due (we are trying to predict if the customer will pay before/after this date)
- InvoiceAmount is the amount of the invoice

Click on the brain icon and “Add a machine learning model” as shown below

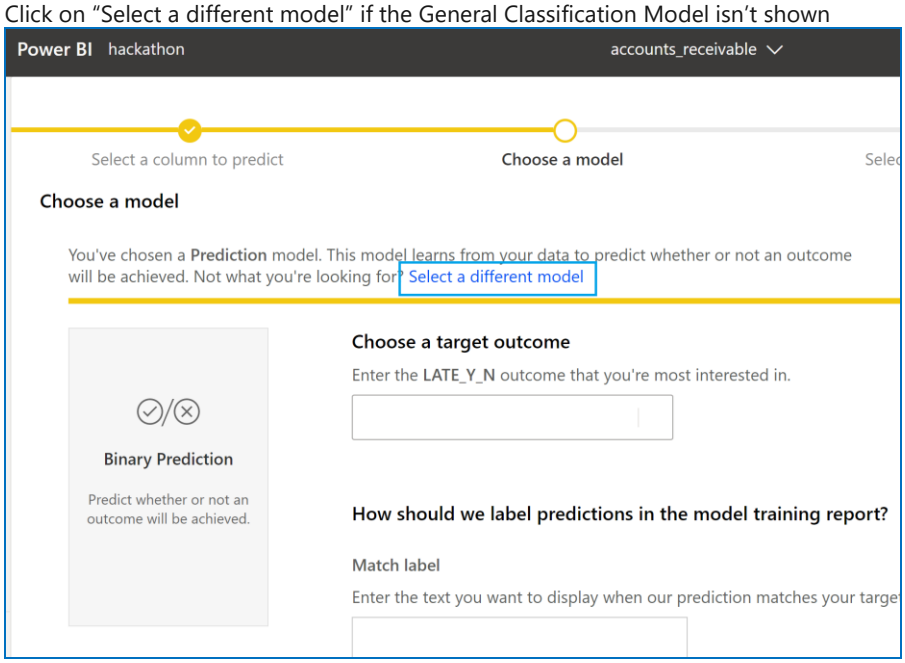


Select the table you created from the dataflow earlier – in our case it was "WA_Fn_UseC_Accounts-Receivable"

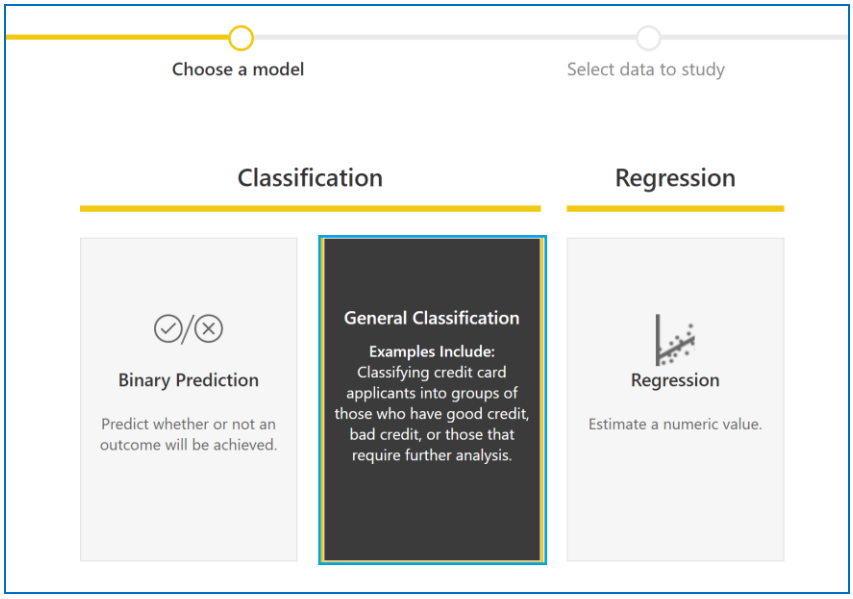


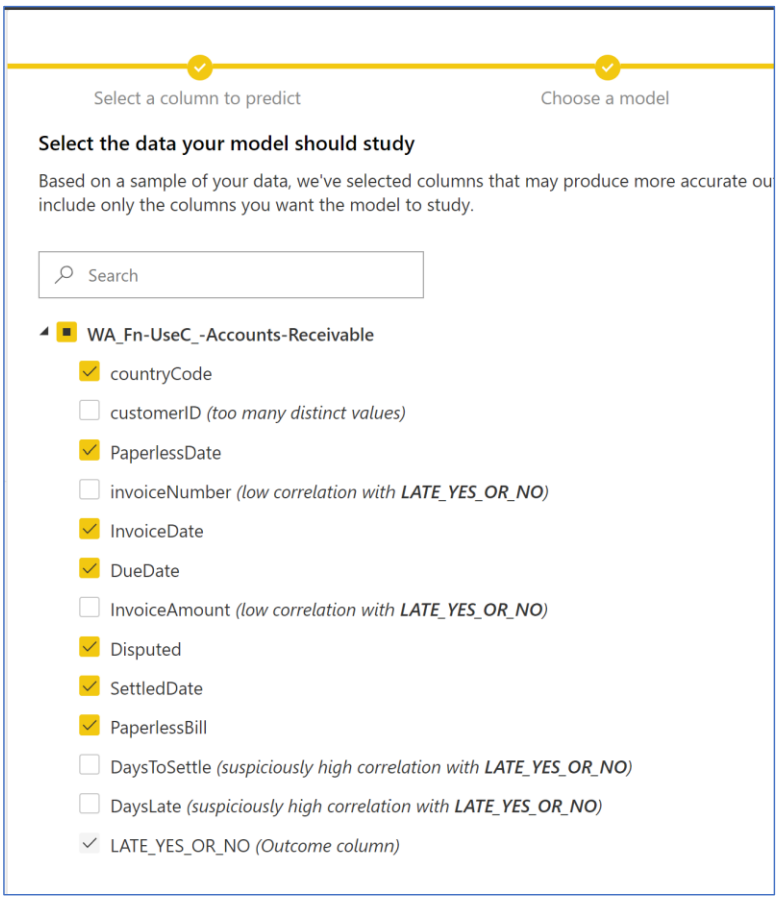
Commented [ST2]: The outcome column is name different - LateYorN

Click on “Select a different model” if the General Classification Model isn’t shown



Select General Classification and click Next



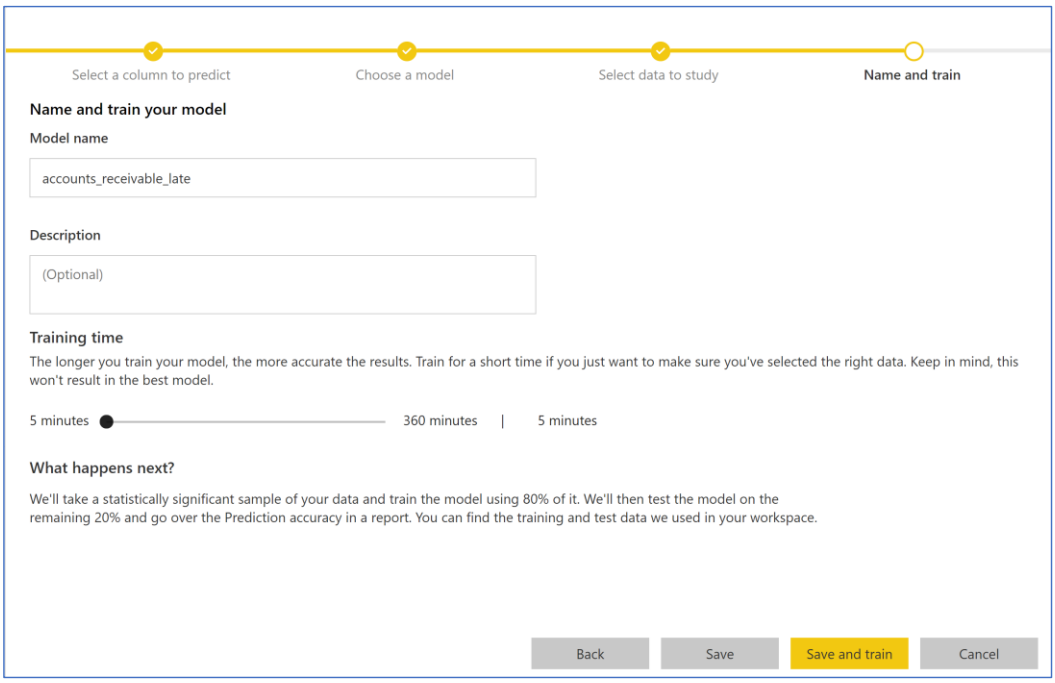


Select all attributes **except** customerID, invoiceNumber, daystosettle, and dayslate – (these last 2 are directly related to being late or not)

Provide the Model Name – accounts_receivable_late

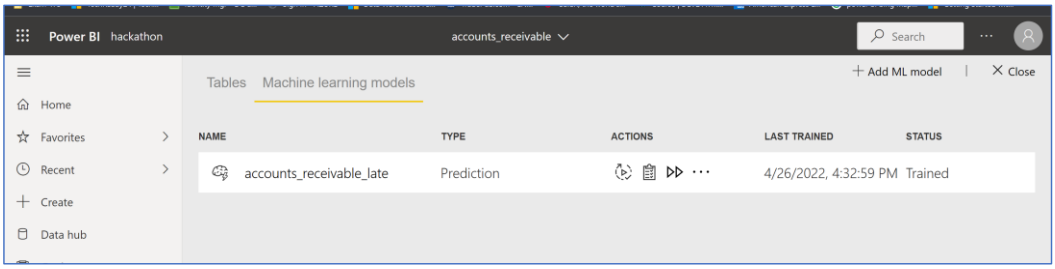
Commented [ST3]: Added the step

Click Save & Train – just go with 5 mins (Pl. use the slider to bring down the value to 5)



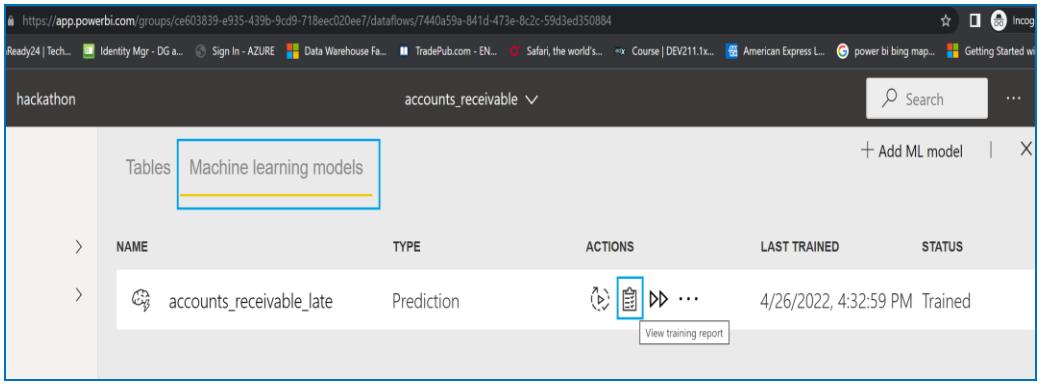
C) Review the Power BI Model Validation Report

After a few minutes, the model should complete



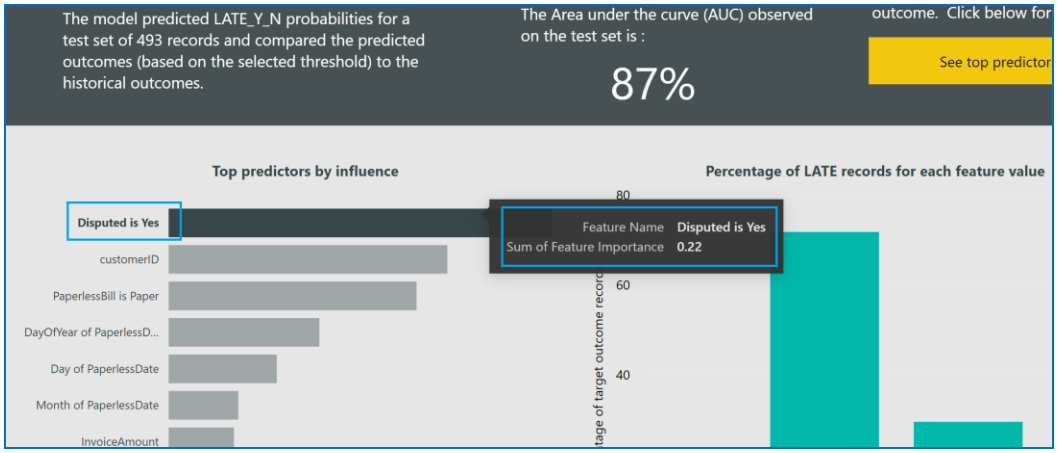
Go into Machine Learning models

View training report

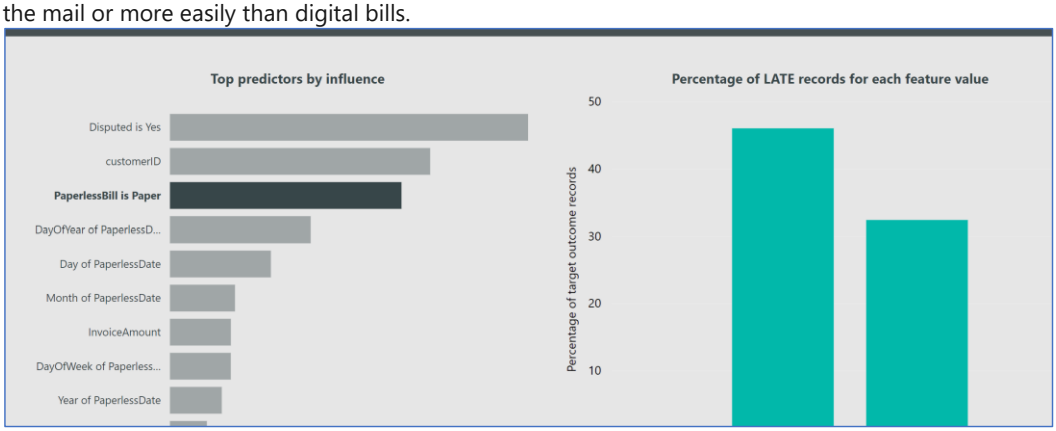


Click on top predictors:

Click on disputed – you can see it explain 22% of why a bill might be late. If a bill is under dispute, it could explain why the payment would be late.

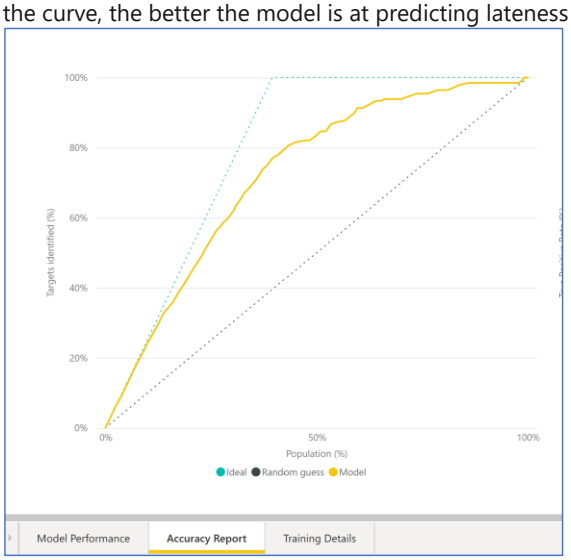


Click on Paperless Bill – which explains 14% of why a bill might be late. Paper bills could get loss in the mail or more easily than digital bills.



Notice how there is an inverse relationship between precision and threshold settings. As you increase the precision the recall decreases and vice versa. So, depending on what is more important – i.e. predicting late and actually being late or predicting late but wasn't late at all you can modify your objective

Click on the Accuracy Tab to see the area under the curve. The area under the curve represents the gain of not just randomly guessing late or not for each customer invoice. Hence, the more area under the curve, the better the model is at predicting lateness or not.

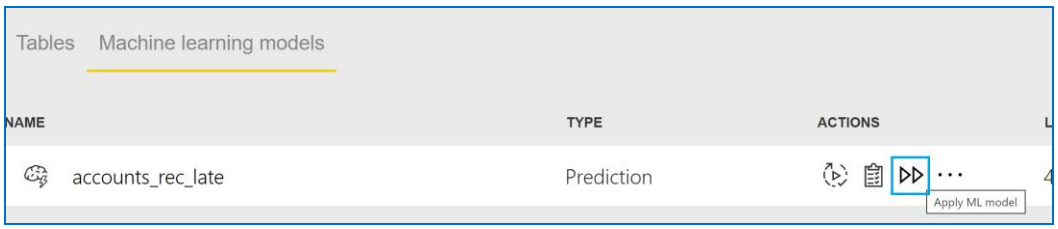


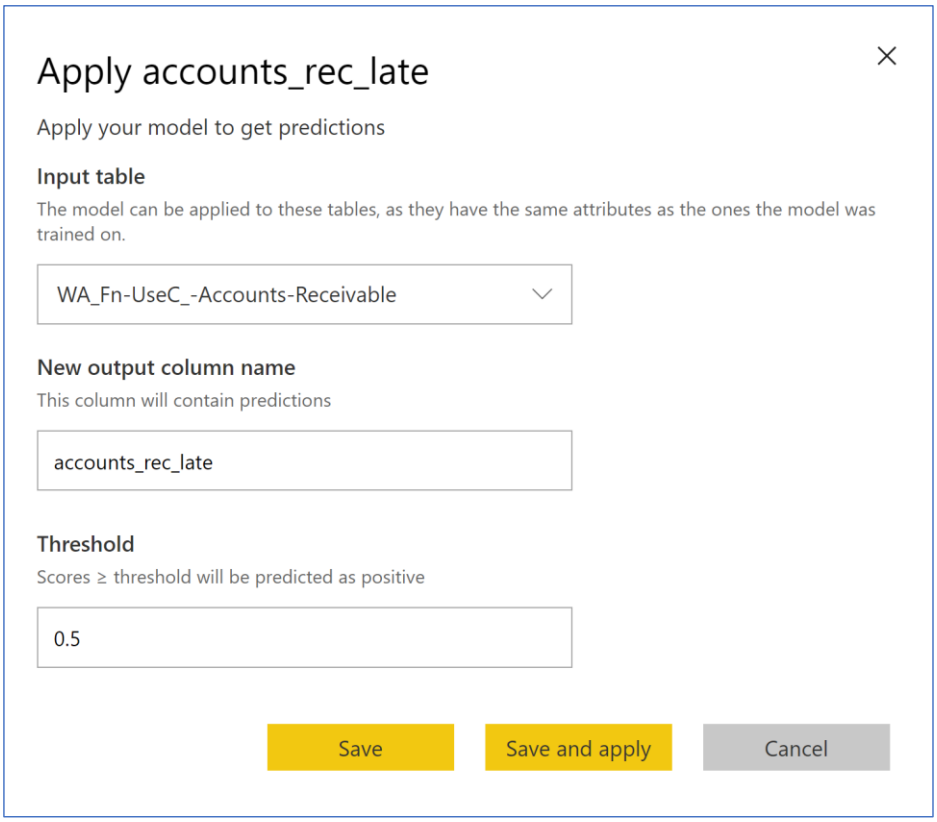
D) Apply the Model

Important

Preparation takes **10-15 minutes** to prepare the experiment run. Once running, it takes **2-3 minutes more for each iteration**.

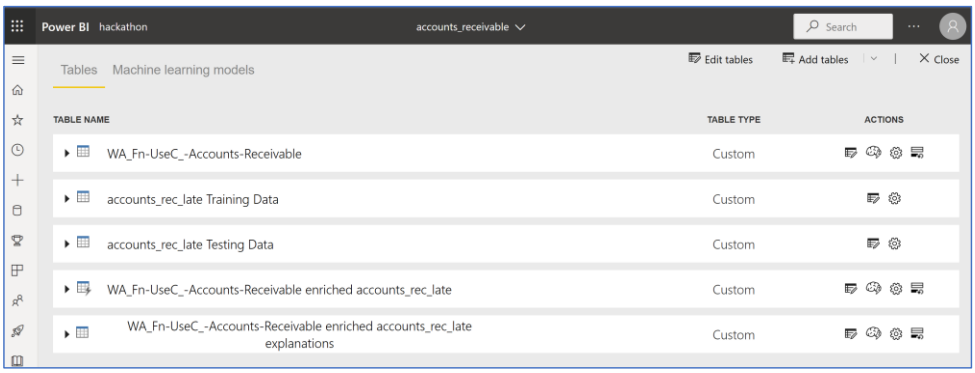
In production, you'd likely walk away for a bit. But for this tutorial, we suggest you start exploring the tested algorithms on the **Models** tab as they complete while the others are still running.



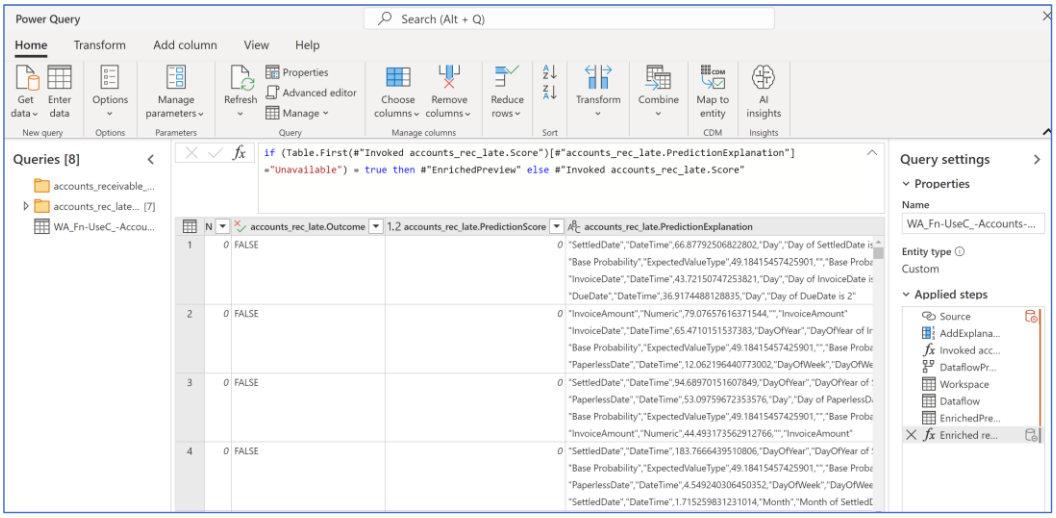


Click Save & Apply

Click on enriched late to see predictions and explanations on a record by record basis


Power BI hackathon
accounts_receivable
Search
Edit tables
Add tables
Close
Tables
Machine learning models
TABLE NAME
TABLE TYPE
ACTIONS
WA_Fn-UseC_-Accounts-Receivable
Custom
accounts_rec_late Training Data
Custom
accounts_rec_late Testing Data
Custom
WA_Fn-UseC_-Accounts-Receivable enriched accounts_rec_late
Custom
WA_Fn-UseC_-Accounts-Receivable enriched accounts_rec_late explanations
Custom

Example:


Power Query
Search (Alt + Q)
Home
Transform
Add column
View
Help
Queries [8]
accounts_receivable_...
accounts_rec_late... [7]
WA_Fn-UseC_-Accou...
if (Table.First(#"Invoked accounts_rec_late.Score")[#"accounts_rec_late.PredictionExplanation"] ="Unavailable") = true then #"EnrichedPreview" else #"Invoked accounts_rec_late.Score"
accounts_rec_late.Outcome
accounts_rec_late.PredictionScore
accounts_rec_late.PredictionExplanation
Query settings
Properties
Name
WA_Fn-UseC_-Accounts-...
Entity type
Custom
Applied steps
Source
AddExplana...
Invoked acc...
DataflowPr...
Workspace
Dataflow
EnrichedPre...
Enriched re...