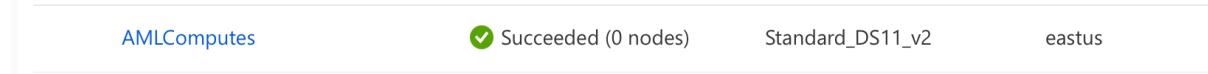


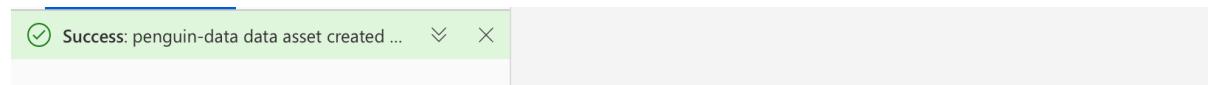
Friday, 20 December 2024

02c-create-clustering-model

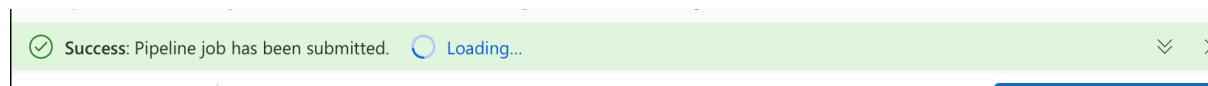
By MONISH NULE



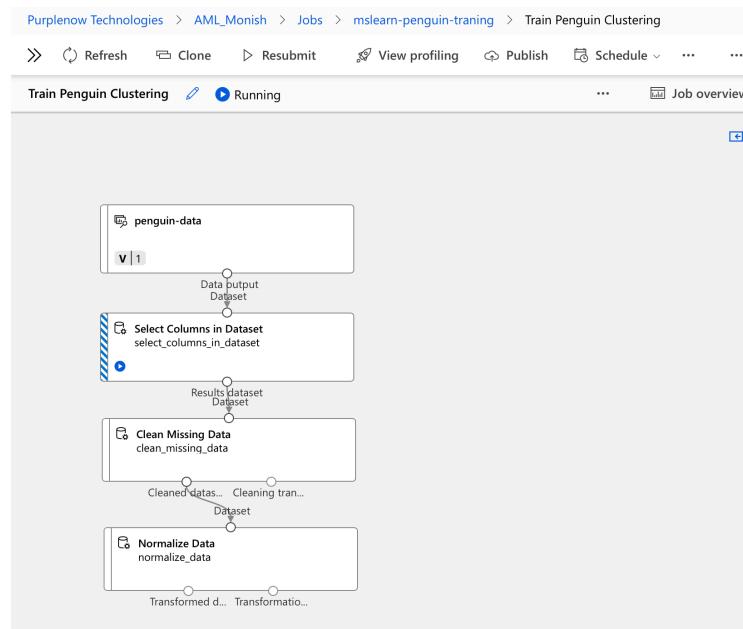
- created computing resource for AML.



- created dataset asset to drop from webURL.



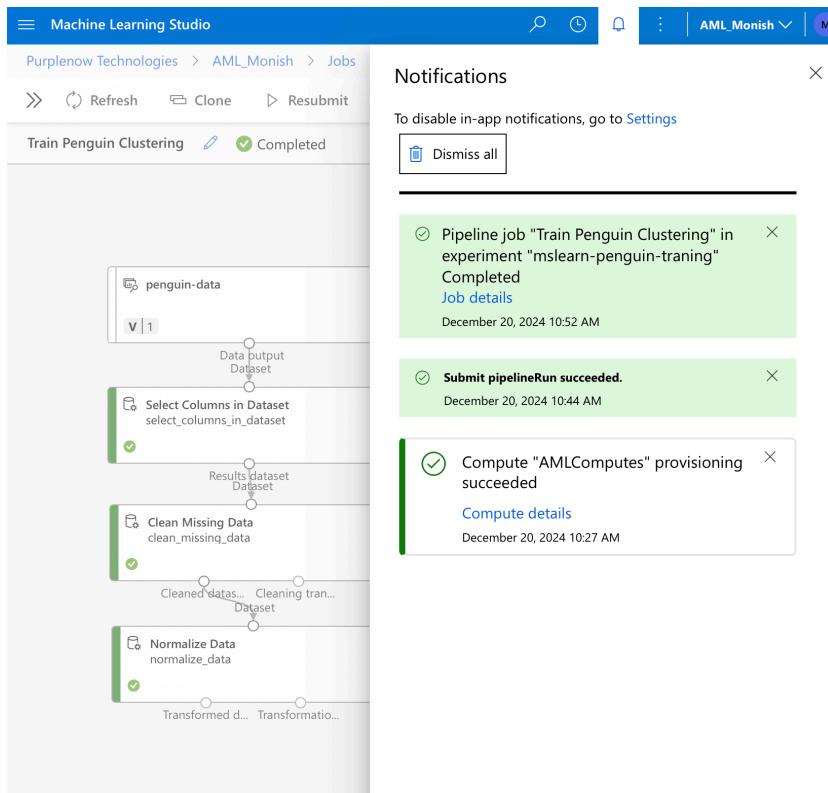
- Configured the pipeline with cleaning, transforming and loading dataset, job submitted.
- Pipeline running, started at 10.49am



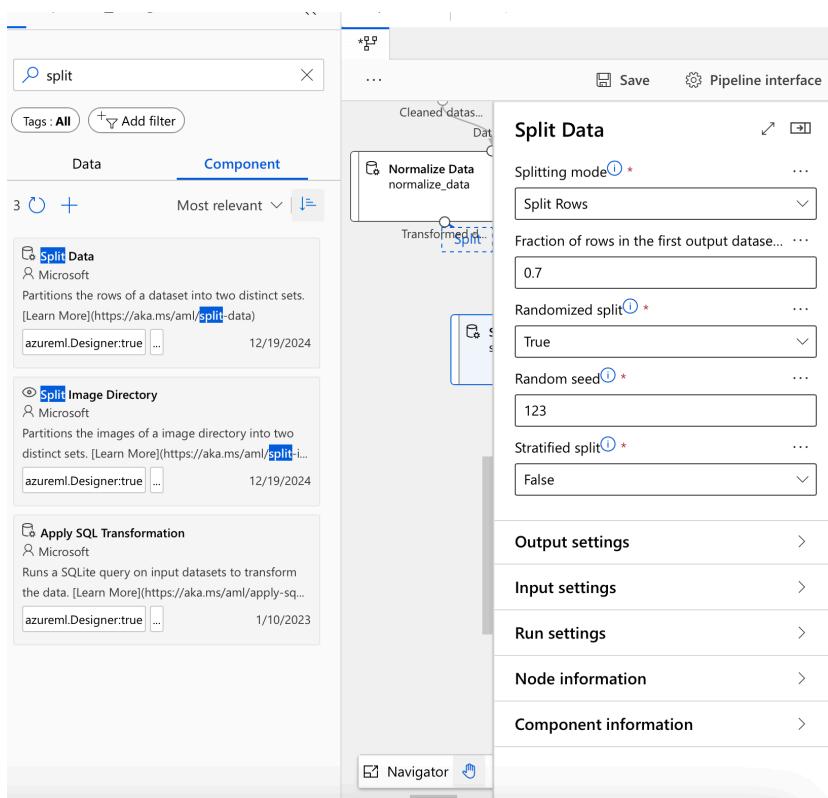
- Pipeline job completed at 10.52 took 3mins.

-

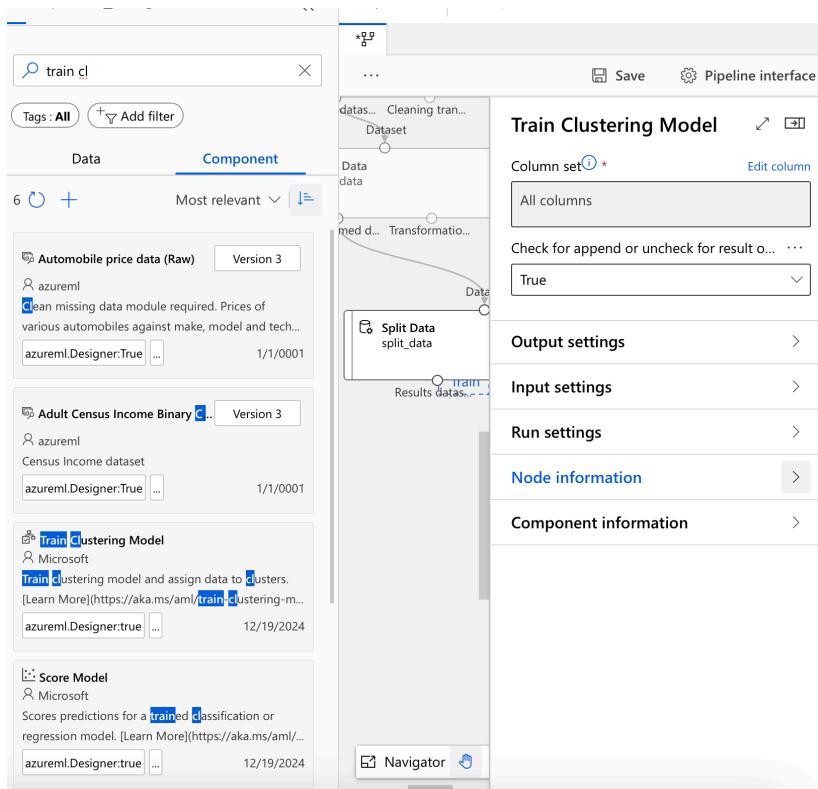
- Adding traning modules.



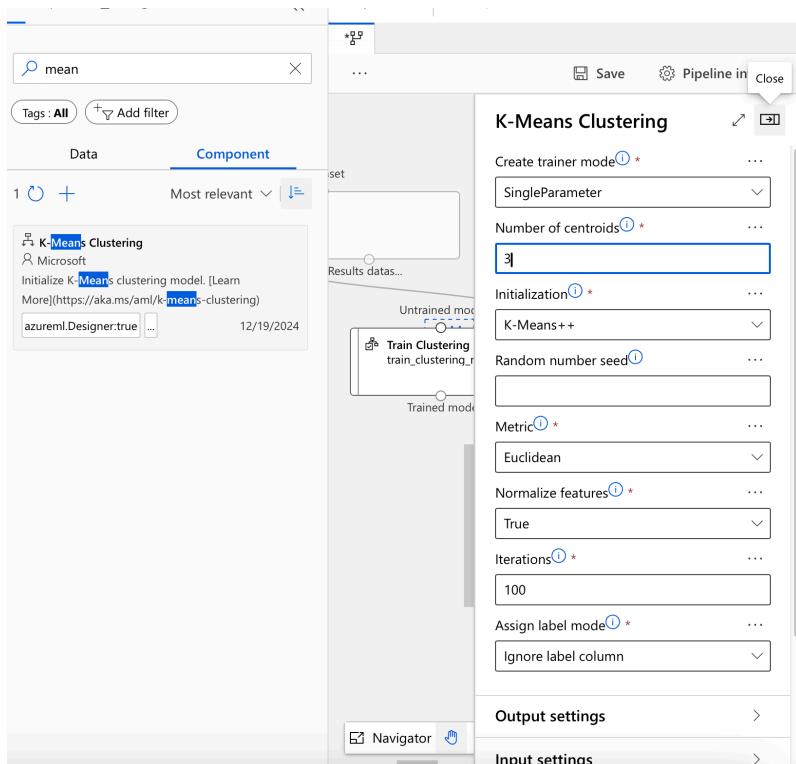
- Configuring split data.



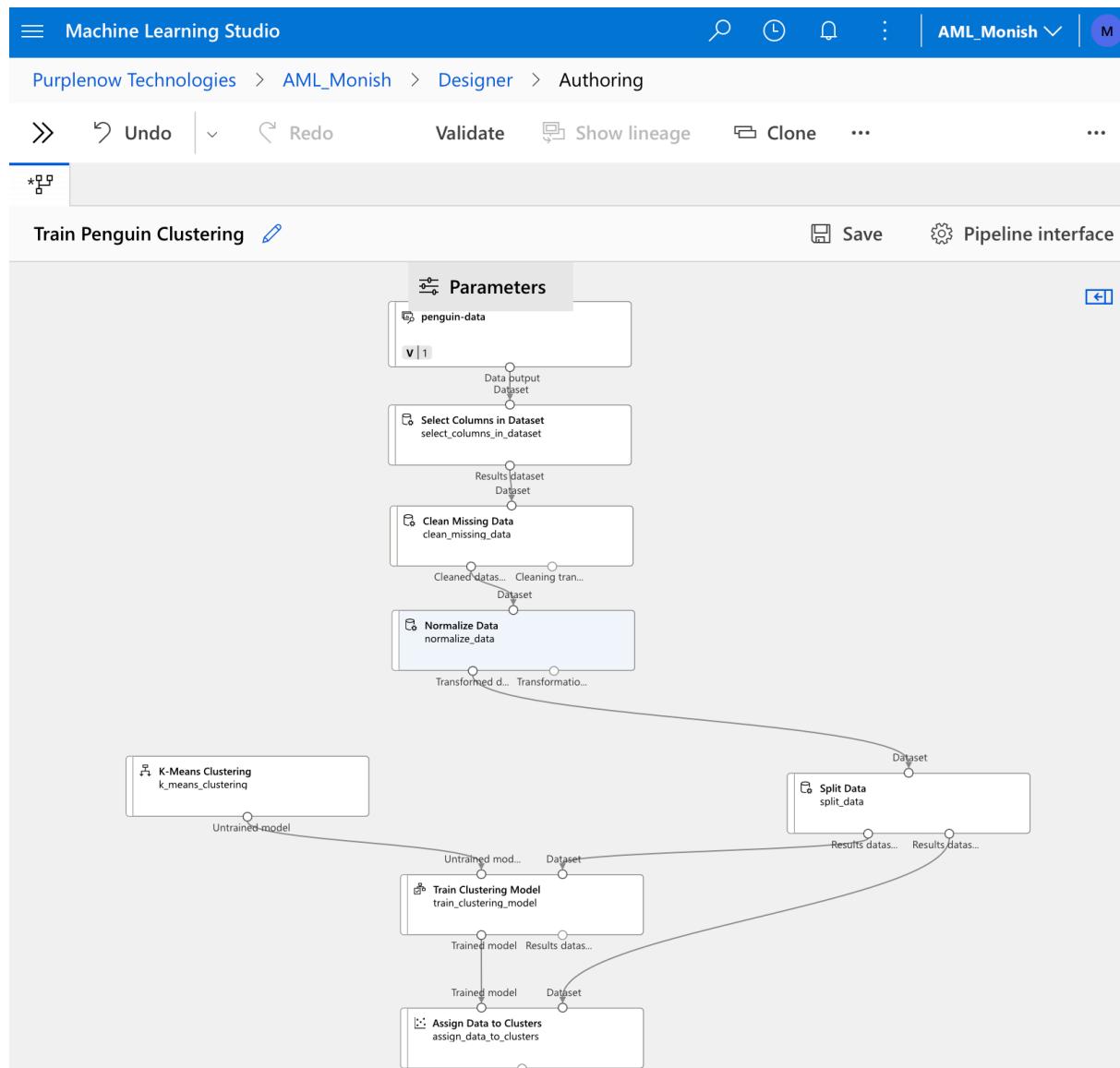
- Configuring train clustering model.



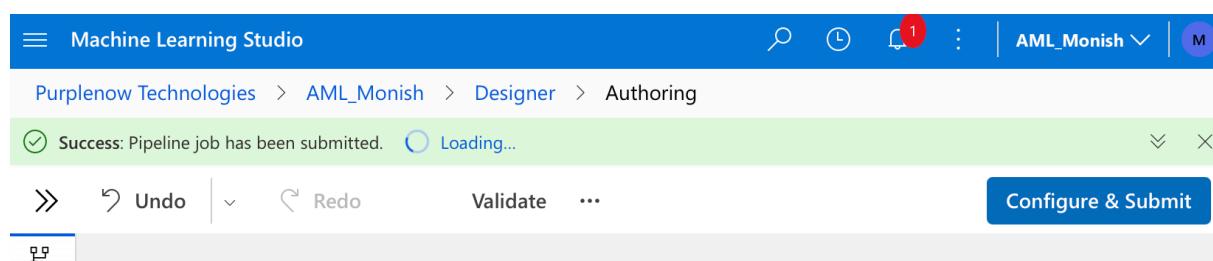
- K-Means algorithm to train the clustering model, configuring...



- Assign to data module connected to train clustering model as to train with remaining 30% data.

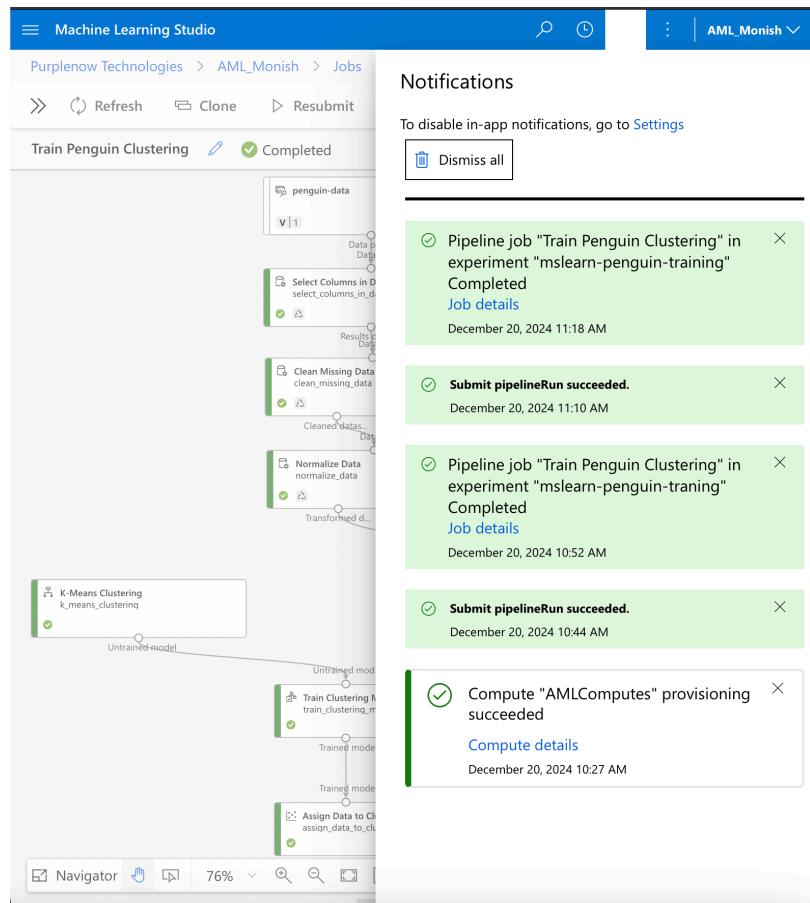


- Pipeline started running at 11.10am



- Dd
- Dd

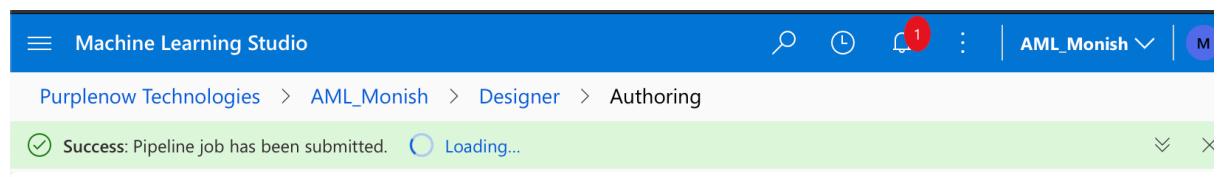
- Pipeline completed at 11.18am



The model is predicting clusters for the penguin observations, but how reliable are its predictions? To assess that, you need to evaluate the model.

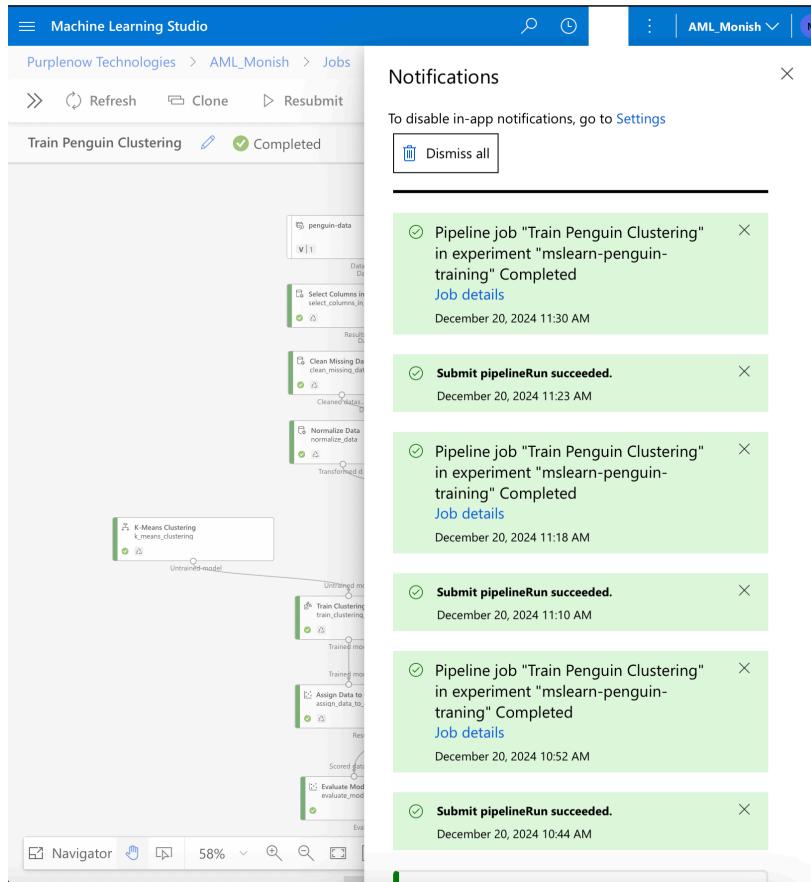
Evaluating a clustering model is made difficult by the fact that there are no previously known true values for the cluster assignments. A successful clustering model is one that achieves a good level of separation between the items in each cluster, so we need metrics to help us measure that separation.

- Adding Evaluating model now..

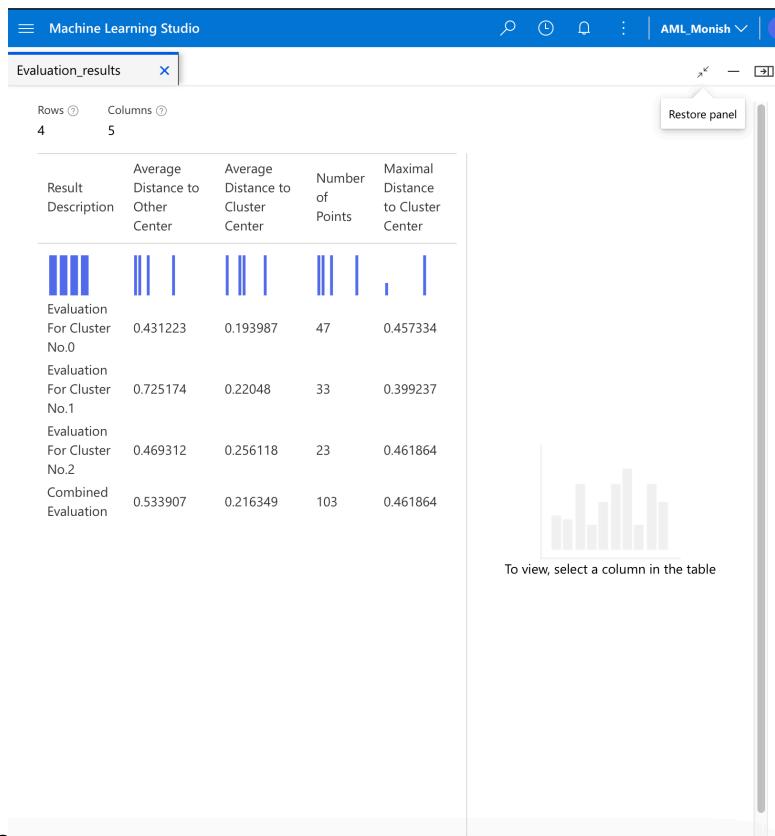


- After adding evaluating model, running the pipeline again at { 11.23am } to get the evaluation of training cluster model.
- Pipeline completed at 11.30am

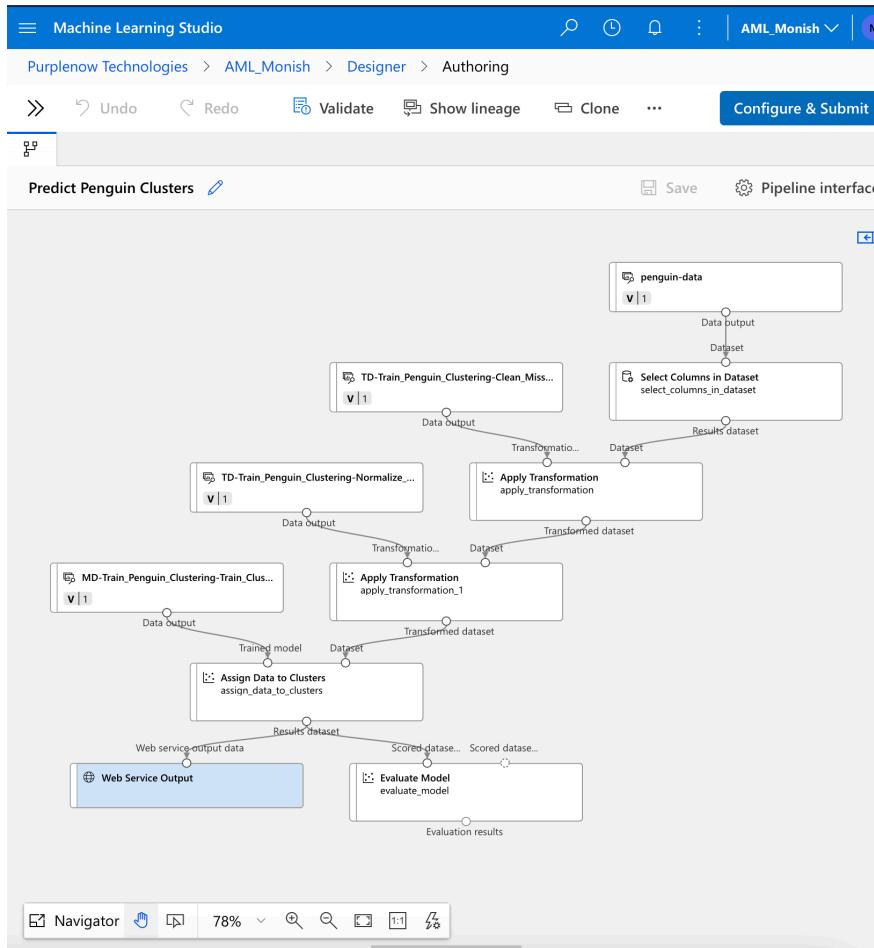
- Screenshot of evaluation the model prediction.



- Evaluations final results...



- Pipeline Inference, The transformations and clustering model in your training pipeline are a part of this pipeline. The trained model will be used to score the new data. The pipeline also contains a web service output to return results.



- Removed Select Colunm, and Penguin Data, and Evaluate Model, The inference pipeline assumes that new data will match the schema of the original training data, so the **penguin-data** dataset from the training pipeline is included. However, this input data includes a column for the penguin species, which the model does not use. Delete both the **penguin-data** dataset and the **Select Columns in Dataset** modules, and replace them with an **Enter Data Manually** module from the **Asset library**.

- Now entering data manually...

Machine Learning Studio

Purplenow Technologies > AML_Monish > Designer > Authoring

Configure & Submit

Predict Penguin Clusters Edit

Save Pipeline interface

Enter Data Manually

Data format * CSV

Has header * True

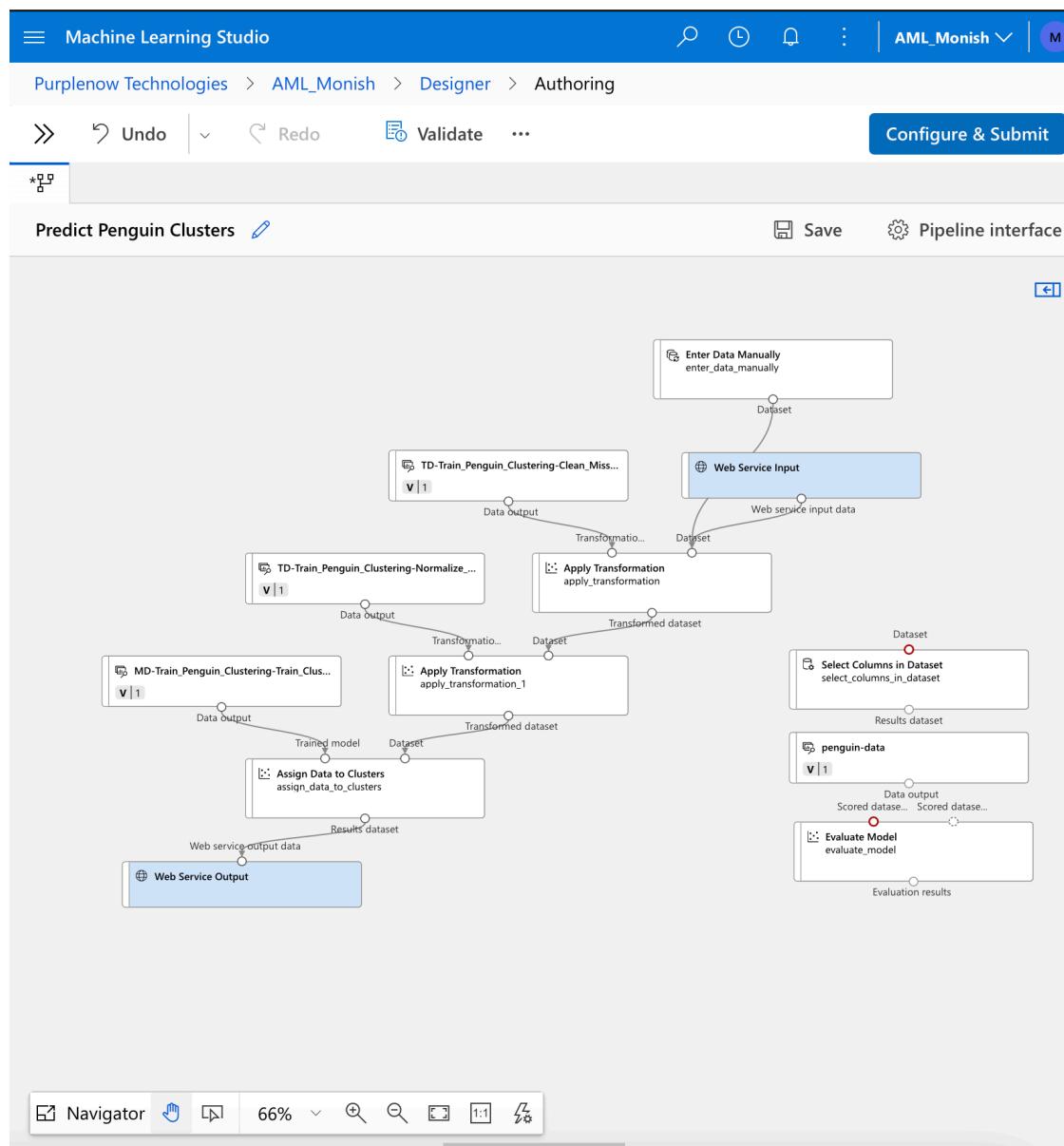
Data *

```

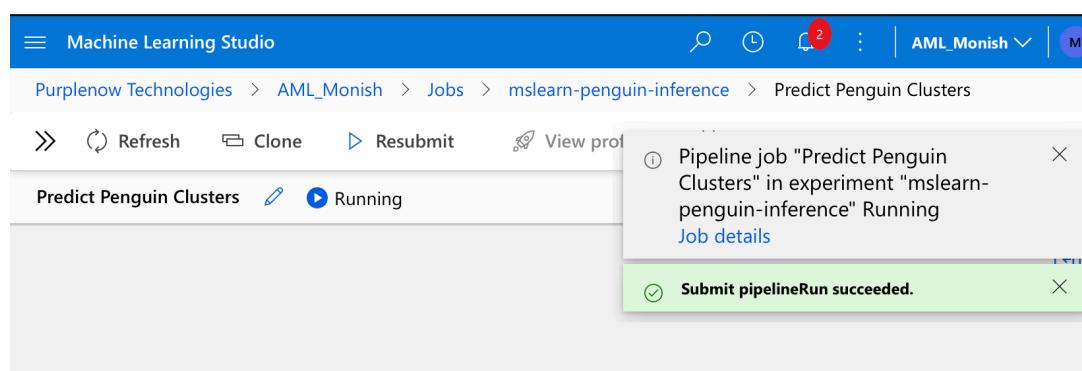
1 CulmenLength,CulmenDepth,FlipperLength,BodyMass
2 39.1,18.7,181,3750
3 49.1,14.8,220,5150
4 46.6,17.8,193,3800

```

- After modification to Inference Pipeline.



- Penguin Predict Pipeline submitted running started at 11.49am



- Sss

- Pipeline completed around 11.57am

The screenshot shows the Azure Machine Learning Studio interface. The top navigation bar includes 'Machine Learning Studio', a search icon, a clock icon, a notifications bell icon, and a dropdown for 'AML_Monish'. Below the navigation is a breadcrumb trail: 'Purplenow Technologies > AML_Monish > Jobs'. There are buttons for 'Refresh', 'Clone', and 'Resubmit'. A pipeline named 'Predict Penguin Clusters' is listed as 'Completed'. The pipeline diagram shows two main stages: 'TD-Train_Penguin_Clustering-Clean_Miss...' and 'MD-Train_Penguin_Clustering-Train_Clus...'. The first stage has a 'Data output' node. The second stage includes a 'Transform' node with an 'Apply Transform' step, followed by a 'Trained model' node, an 'Assign Data to Clusters' node, and a 'Results dataset' node. A 'Web service output data' node is also present. Below the pipeline is a 'Web Service Output' section. At the bottom left are 'Navigator', 'Download', 'Print', and zoom controls (59%). On the right, a 'Notifications' panel displays several green success messages from December 20, 2024, including 'Pipeline job "Predict Penguin Clusters" Completed', 'Submit pipelineRun succeeded.', 'Pipeline job "Train Penguin Clustering" Completed', 'Submit pipelineRun succeeded.', 'Pipeline job "Train Penguin Clustering" Completed', and 'Submit pipelineRun succeeded.'.

- Preview of resultant data after running web inference

The screenshot shows the 'Results_dataset' view in Azure Machine Learning Studio. The top navigation bar and breadcrumb trail are identical to the previous screenshot. The dataset summary shows 'Rows 3' and 'Columns 8'. The data table lists columns: CulmenLength, CulmenDepth, FlipperLength, BodyMass, and Assig. The data rows are:

CulmenLength	CulmenDepth	FlipperLength	BodyMass	Assig.
-0.884499	0.785449	-1.418347	-0.564142	0
0.949826	-1.192335	1.359205	1.184151	1
0.491245	0.329038	-0.563715	-0.501703	2

To the right of the table is a histogram visualization. At the bottom left is a 'TOC' icon.

- Now finally model is ready to take web inputs, deploying now... currently running the deployment started around 12.04pm
- Due to Endpoint not assigned to local system
- Created a pipeline with Model Evaluation results and modified the same pipeline for inference all sysyem ok, **Deployment Succeeded**
- **REST endpoints :** `{ http://bdd13e15-922d-472c-8235-62f2983db7a0.eastus.azurecontainer.io/score }`

The screenshot shows the Azure Machine Learning Studio interface. In the top navigation bar, it says "Machine Learning Studio" and "AML_Monish". On the left, there's a breadcrumb trail: "Purplenow Technologies > AML_Monish > Endpoints". Below that, the project name "penguin-predict-clusters" is displayed with a star icon. Underneath, there are tabs for "Details", "Test", "Consume", and "Logs", with "Details" being the active tab. To the right, a "Notifications" section is open, showing a message: "Deployment 'penguin-predict-clusters' creation completed" with a timestamp of "December 20, 2024 12:25 PM". There's also a "Dismiss all" button.

Tried running score.py with API failed with status code 400.

The screenshot shows a code editor with a Python file named "score.py" open. The code is as follows:

```

response = urllib.request.urlopen(req)
result = response.read()
print(result)
except urllib.error.HTTPError as error:
    print("The request failed with status code: " + str(error.code))
# Print the headers - they include the request ID and the timestamp, which are useful for debugging the failure
print(error.info())
print(error.read().decode("utf8", 'ignore'))

```

Below the code, the terminal output shows the execution of the script and the resulting error message:

```

/usr/local/bin/python3.12 /Users/amonish8/Desktop/Big-Data-Analytics/PML/Rest_Endpoints/score.py
The request failed with status code: 400
Access-Control-Allow-Origin: *
Content-Length: 885
Content-Type: application/json
Date: Fri, 28 Dec 2024 07:04:09 GMT
Server: nginx
X-Ms-Client-Request-Id: 619bf92e-7acc-4ba1-846f-089ef33c479f
X-Ms-Request-Id: 619bf92e-7acc-4ba1-846f-089ef33c479f
X-Ms-Run-Fn-Exec-Ms: 0.795
X-Ms-Run-Function-Failed: False
X-Ms-Server-Version: azmlinfsrv/0.8.4.2
X-Request-Id: 80088684-1e8d-4414-8ad0-fccf54b34628
Connection: close

{"error": {"code": 400, "message": "Input Data Error. Input data are inconsistent with schema.\nSchema: {'input1': {'columnAttributes': [{"name': 'CulmenLength', 'type': 'Numeric', 'isFeature': True}], "type": "Table"}}, "type": "InputDataValidationException"}}

Process finished with exit code 0

```

The status bar at the bottom indicates the file is "PML > Rest_Endpoints > score.py", the file type is "Python", and the file size is 4 spaces.