Utilizei os mesmos recursos criados durante a trilha AI-900 no Microsoft Learn.

Criei um dataset de arquivos locais com o nome de notas-enem-2019, usando como base o arquivo MICRODADOS\_ENEM\_2019. No esquema selecionei apenas as colunas NU\_NOTA\_CN, NU\_NOTA\_CH, NU\_NOTA\_MT e NU\_NOTA\_LC.

No Designer criei um pipeline com o nome de Enem-regression.

Criei um módulo do dataset notas-enem-2019 e liguei sua saída a um módulo Select Columns in Dataset.

No módulo Select Colums in Dataset selecionei as colunas NU\_NOTA\_MT e NU\_NOTA\_CN. Liguei sua saída a um módulo Clean Missing Data, no qual selecionei a coluna NU\_NOTA\_MT para ser limpa, com razão de valor ausente mínima de 0.0 e máxima de 1.0. o modo de limpeza foi remover a linha inteira. Liguei sua saída a um módulo Normalize Data, com método de transformação MinMax e a coluna NU\_NOTA\_MT selecionada.

Liguei um módulo Split Data à saída de Clean Missing Data e especifiquei seu modo de divisão como dividir linhas, a fração de linhas no primeiro conjunto de dados de saída como 0.5 e divisão aleatória com semente aleatória 45, sem divisão estratificada.

Criei um módulo Linear Regression.

Criei um módulo Train Model e liguei suas entradas às saídas de Linear Regression e Split Data (saída dataset1). No módulo selecionei a coluna NU\_NOTA\_CN.

Criei um módulo Score Model e liguei suas entradas a Train Model e Split Data (saída dataset2).

Criei um módulo Evaluate Model e conectei sua entrada esquerda a Score Model.

Enviei o pipeline criando um novo experimento com ponto de extremidade com nome de Enem-prever-cn-base-mt.

A avaliação do modelo retornou os seguintes resultados, arredondados:

**MAE (erro médio absoluto)**: 47.76

**RMSE (raiz do erro quadrático médio)**: 57.94

**RSE (erro ao quadrado relativo)**: 0.58

**RAE (erro absoluto relativo)**: 0.74

**Coeficiente de determinação (R 2)**: 0.42

Isso demonstra que o modelo não possui alta precisão.

Criei um pipeline de inferência chamado Enem-prever-nota-cn-base-mt. Nesse novo pipeline, deletei o módulo Dataset notas-enem-2019 e o substituí por um módulo Enter Data Manually, com os seguintes dados:

NU\_NOTA\_MT

568

876

364

Removi a coluna NU\_NOTA\_CN do módulo Select Columns in Dataset.

Deletei o módulo Evaluate Model.

Iseri um módulo Execute Python Script entre Evaluate Model e Web Service Output, e inseri nele o seguinte código:

import pandas as pd

def azureml\_main(dataframe1 = None, dataframe2 = None):

    scored\_results = dataframe1[['Scored Labels']]

    scored\_results.rename(columns={'Scored Labels': nota\_cn\_prevista

'},

                        inplace=True)

    return scored\_results

Enviei o pipeline como um novo experimento de nome Enem-inferencia-cn-base-mt.

As notas de ciências da natureza previstas para os três valores de notas de matemática fornecidos foram, respectivamente e arredondados:

498.10

637.33

405.89

Implantei o pipeline criando um ponto de extremidade com o nome de enem-prever-cn-base-mt. O tipo de implantação foi Instância de Contêiner do Azure.

Criei um notebook de nome Teste-Enem-prever-cn.ipynb e inseri o seguinte código python:

endpoint = 'http://17fd0658-305a-4d3e-b03b-96f64264295a.brazilsouth.azurecontainer.io/score'

key = 'Enq6eNiFQ2agfTiAV1WkufDaJuYf6cTe'

import urllib.request

import json

import os

# Prepare the input data

data = {

    "Inputs": {

        "WebServiceInput0":

        [

            {

                    'NU\_NOTA\_MT': 840,

            },

        ],

    },

    "GlobalParameters":  {

    }

}

body = str.encode(json.dumps(data))

headers = {'Content-Type':'application/json', 'Authorization':('Bearer '+ key)}

req = urllib.request.Request(endpoint, body, headers)

try:

    response = urllib.request.urlopen(req)

    result = response.read()

    json\_result = json.loads(result)

    y = json\_result["Results"]["WebServiceOutput0"][0]["nota\_cn\_prevista"]

    print('Nota prevista para ciências da natureza: {:.2f}'.format(y))

except urllib.error.HTTPError as error:

    print("The request failed with status code: " + str(error.code))

    # Print the headers to help debug the error

    print(error.info())

    print(json.loads(error.read().decode("utf8", 'ignore')))

Ao executar a célula a saída recebida foi:

Nota prevista para ciências da natureza: 621.06

Minha conclusão é que, apesar de ser possível criar um modelo de machine learning que preveja a nota de ciências da natureza de uma pessoa a partir de sua nota de matemática na prova do Enem, tal previsão não é precisa o suficiente para ser útil.