4 - Modelagem com Machine Learning

Propomos as seguintes perguntas:

P1: É possível predizer a taxa de poluição emitida no ar nos próximos anos?
P2: A estação do ano influencia a quantidade de poluentes no ar de alguma forma?

Porém só tralhamos na P1 pois a H1 trabalhada na fase anterior resultou em uma correlação positiva entre a estação do ano e a quantidade de poluentes no ar. Logo, não houve necessidade de fazer uma regressão para a P2.

Feature Engineering

Para feature engineering, analisamos dados do NO2, PM2.5 e 03, agrupando os valores por ano, de forma a ignorar as estações. Pensamos em realizar a regressão com dois tipos de parâmetros:

- LAGs de cada um dos poluentes, para identificar se é possível predizer o poluente com base em dados antigos.
- LAGs de outros poluentes, para identificar se é possivel predizer um poluente baseado em outro.

Logo, separamos o dataset em dois, um que continha apenas os LAGs (1, 2, 3) e YEAR baseados no próprio poluente. E outro que possuia os LAGs (1, 2, 3) e YEAR de todos os poluentes. Chamamos o primeiro dataset de simples, e o segundo de complexo.

SQL para preparar o dataset simples

```
WITH base AS (
    SELECT
        indicatorid,
        name,
        year,
        AVG(datavalue) AS DataValue
    FROM data
    GROUP BY indicatorid, name, year
)
SELECT
    indicatorid,
    name,
    year,
    DataValue AS Current,
    LAG(DataValue, 1) OVER (
            PARTITION BY indicatorid
            ORDER BY year
        ) AS Lag1,
    LAG(DataValue, 2) OVER (
            PARTITION BY indicatorid
            ORDER BY year
        ) AS Lag2,
    LAG(DataValue, 3) OVER (
            PARTITION BY indicatorid
            ORDER BY year
        ) AS Lag3
FROM base
ORDER BY year;
```

SQL para preparar o dataset complexo

```
WITH base AS (
    SELECT
        indicatorid,
        name,
        year,
        AVG(datavalue) AS DataValue
```

```
    FROM data
    GROUP BY indicatorid, name, year
),
lag_no2 AS (
    SELECT
        indicatorid,
        name,
        year,
        DataValue,
        LAG(DataValue, 1) OVER (ORDER BY year) AS Lag1,
        LAG(DataValue, 2) OVER (ORDER BY year) AS Lag2,
        LAG(DataValue, 3) OVER (ORDER BY year) AS Lag3
    FROM base
    WHERE indicatorid = 375
),
lag_pm25 as (
    SELECT
        indicatorid,
        name,
        year,
        DataValue,
        LAG(DataValue, 1) OVER (ORDER BY year) AS Lag1,
        LAG(DataValue, 2) OVER (ORDER BY year) AS Lag2,
        LAG(DataValue, 3) OVER (ORDER BY year) AS Lag3
    FROM base
    WHERE indicatorid = 365
),
lag_o3 as (
    SELECT
        indicatorid,
        name,
        year,
        DataValue,
        LAG(DataValue, 1) OVER (ORDER BY year) AS Lag1,
        LAG(DataValue, 2) OVER (ORDER BY year) AS Lag2,
        LAG(DataValue, 3) OVER (ORDER BY year) AS Lag3
    FROM base
    WHERE indicatorid = 386
)
SELECT
    b.indicatorid,
    b.name,
    b.year,
    b.DataValue AS Current,
    no2.Lag1 AS Lag1_NO2,
    no2.Lag2 AS Lag2_NO2,
    no2.Lag3 AS Lag3_NO2,
    pm25.Lag1 AS Lag1_PM25,
    pm25.Lag2 AS Lag2_PM25,
    pm25.Lag3 AS Lag3_PM25,
    o3.Lag1 AS Lag1_O3,
    o3.Lag2 AS Lag2_O3,
    o3.Lag3 AS Lag3_O3
FROM base b
INNER JOIN lag_no2 no2 ON b.year = no2.year
INNER JOIN lag_pm25 pm25 ON b.year = pm25.year
INNER JOIN lag_o3 o3 ON b.year = o3.year
ORDER BY b.name, b.year;
```

OBS: Para lidar com o conjunto de features com Boro basta adicionar a coluna GeoPlaceName, junto de IndicatorID nas linhas de partition da query simples.

Feature selection

Para selecionar as melhores features para o conjunto Citywide, ou seja, aquelas que podem predizer melhor o futuro, optamos pelo seletor `SelectKBest`, usando o algoritmo *f_regression* do `scikit-learn`. Além disso ao lidarmos com o conjunto dos Boroughs (distritos) utilizamos o RFECV com uma simplificação do classificador a ser utilizado, por exemplo, ao selecionar features para o Linear Regression, também foi treinado o RFECV com o mesmo.

Treino

Optamos por testar as regressões com os dataframes separados de duas formas diferentes, quando me refiro à citywide, estou considerando entradas no dataframe de origem os quais tiveram a coluna "GeoPlaceName" filtradas pela string "Citywide", o que resulta em dados considerando a média da cidade toda. Já "Borough" são as denominações para os 5 distritos os quais Nova Iorque é divido: Brooklyn, Queens, Staten Island, Bronx e Manhattan; Estamos imaginando que tal divisão pelo chamado "Boro" deveria resultar em dados mais precisos.

Para ambos os tipos de modelos resolvemos dividir seus respectivos datasets da seguinte forma: Dataset_treino=df[df['Year'] < 2020] Dataset_teste=df[df['Year'] >= 2023]

Para o treino dos baselines utilizamos a regressão linear, já para o treino dos modelos em si foi utilizado a regressão linear, a random forest e o gradient_boosting

Treino - Modelo Citywide

OBS: Algoritmo utilizado para seleção de features foi o SelectKBest do scikit-learn

Para o modelo simples, foi feita a seleção entre as 4 features eenquanto 3 delas foram aplicadas no conjunto chamado features_simples, que contém as seguintes features: ['Lag1', 'Lag2', 'Lag3', 'Year'], Note que os lags aqui são referentes ao próprio identificador, algo que será diferente para o modelo complexo.

Para o modelo complexo, fizemos a seleção entre 10 features, chamadas features complexas: ['Lag1_NO2', 'Lag2_NO2', 'Lag3_NO2', 'Lag1_PM25', 'Lag2_PM25', 'Lag3_PM25', 'Lag1_O3', 'Lag2_O3', 'Lag3_O3', 'Year'] Perceba que aqui, além de procurarmos por relações entre os lags do próprio indicador também estamos lidando com valores de outros indicadores, por isso temos que fazer a identificação dos lags.

Como baseline, utilizamos o modelo de Regressão Linear(RL) com o TOP 1, escolhido com o SelectKBest, do conjunto de features simples.

Testes - Citywide

A partir do modelo Baseline é possível ver um indício de que é possível prever um R^2 para o gás NO2, já que foi o único que teve um resultado signficativo quando comparado com o resto:

Resultados RL - Baseline | Fine particles (PM 2.5) (Citywide)

Período Treino: 2012 - 2019 | Teste: 2020 - 2023 R-quadrado (R^2): -2.948 RMSE: 0.732 Coeficiente usado: Year(-0.85887)

Resultados RL - Baseline | Nitrogen dioxide (NO2) (Citywide)

Período Treino: 2012 - 2019 | Teste: 2020 - 2023 R-quadrado (R^2): 0.188 RMSE: 0.469 Coeficiente usado: Year(-0.83841)

Resultados RL | Ozone (O3) (Citywide)

Período Treino: 2012 - 2019 | Teste: 2020 - 2023 R-quadrado (R^2): -0.024 RMSE: 2.488 Coeficiente escolhido: Lag3(-0.86644)

Sendo assim, isso já nos deu uma dica de que ao menos o NO2 seria possível prever algo utilizando modelos de Machine Learning, e ao executar os testes com múltiplos parâmetros tivemos resultados ainda mais animadores para o NO2, porém PM2.5 e 03, os resultados continuaram ruins, portanto mostraremos apenas os significativos:

Resultados RL - Conjunto de Features Simples | Nitrogen dioxide (NO2) (Citywide)

Período Treino: 2012 - 2019 | Teste: 2020 - 2023 R-quadrado (R^2): 0.542 RMSE: 0.353 Features: Lag1(-0.814175), Lag2(-0.874518), Year(-2.167129)

Resultados RL - Conjunto de Features Complexas | Nitrogen dioxide (NO2) (Citywide)

Período Treino: 2012 - 2019 | Teste: 2020 - 2023 R-quadrado (R^2): 0.488 RMSE: 0.373 Features:Lag1_PM25(-0.174009), Lag1_NO2(-0.353214), Year(-1.271726)

Observe que para o conjunto Citywide, tentar relacionar NO2 com Lags de outros indicadores foi um problema, como pode ser visto no R^2 de features complexas, que acabou por selecionar o Lag1 de PM2.5 para tentar prever o modelo de NO2 e isso reduziu a precisão do mesmo. Portanto as melhores features para tal foram as selecionadas no conjunto de features simples.

Treino - Borough

Aqui temos apenas um conjunto de features, que ria equivalente às features simples do modelo Citywide: [‘Lag1’, ‘Lag2’, ‘Lag3’, ‘Year’]

Algoritmo utilizado para seleção de features foi o RFECV do scikit-learn

Como baseline, utilizamos o modelo de Regressão Linear(RL) com o TOP 1 escolhido através do RFECV do conjunto de features.

Testes - Borough

Para testar também utilizamos o algoritmo RFECV e infelizmente não tivemos nenhum resultado bom em nenhum dos Boros/Indicadores, nem nos treinos baselines nem nos testes aplicados. Sendo assim nem apresentaremos aqui nessa seção, os resultados estarão no arquivo borough.txt

Conclusão

Como já foi falado, os testes por Boro não melhoraram em nada a predição dos modelos, portanto a melhor e única predição que podemos fazer é de NO2 no dataset citywide. Acreditávamos que iriamos conseguir melhor precisão ao utilizar os dados com uma granulação menor, porém isso não se apresentou, a média inteira da cidade foi realmente o melhor e utilizando o modelo de Regressão Linear.