cluster_evasao_escolar_censo

April 5, 2025

1 ATIVIDADE 05/04/2025: POS-MDE-SEDU

1.1 MODELOS DESCRITIVOS

1.1.1 Professor: Sérgio Nery Simões

ATENÇÃO: ao final do notebook, há alguns exercícios de análise e interpretação de clusters. Leia o notebook com atenção procurando entender o que é realizado em cada passo e, ao final, procure resolver os exercícios.

2 Análise Descritiva e Clusterização de Escolas por Evasão Escolar

Este notebook tem como objetivo realizar uma análise descritiva e aplicar técnicas de clusterização para identificar perfis de escolas com base na evasão escolar e características de infraestrutura.

2.1 Importação de bibliotecas

Vamos importar as bibliotecas necessárias para manipulação de dados, visualizações e modelagem com K-means.

[1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
from sklearn.decomposition import PCA
```

2.2 Carregamento dos dados

Aqui carregamos o dataset simulado contendo informações sobre escolas, como localização, tipo de administração, infraestrutura e taxa de evasão.

[2]:
```
df = pd.read_csv("../../datasets/evasao_escolar_simulado.csv")  # Substitua␣
↪pelo caminho correto se necessário
print("Dimensão do dataset:", df.shape)
df.head()
```

```
Dimensão do dataset: (100, 10)
```

[2]:
```
   id_escola        regiao localizacao dependencia_adm  infra_biblioteca  \
0          1           Sul       Rural       Municipal                 0
1          2  Centro-Oeste       Rural       Municipal                 1
2          3       Sudeste      Urbana        Estadual                 0
3          4  Centro-Oeste      Urbana       Municipal                 1
4          5  Centro-Oeste      Urbana       Municipal                 1

   infra_internet  infra_laboratorio  total_matriculados  evasao_percentual  \
0               1                  0                1183          12.007803
1               1                  0                 268          18.234745
2               1                  0                 745          17.397170
3               1                  1                 252          19.930682
4               0                  1                 404           2.384619

   evasao_absoluta
0              142
1               48
2              129
3               50
4                9
```

2.3 Análise descritiva por região

Vamos calcular a média da evasão percentual por região para entender se há diferenças regionais.

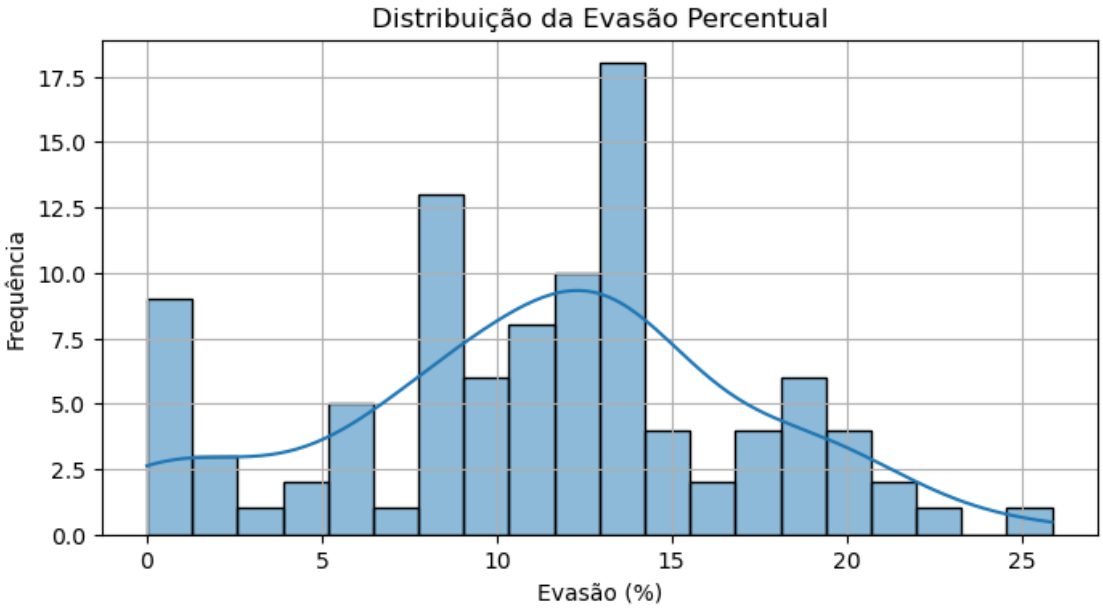
[3]:
```
print("Média de evasão percentual por região:")
print(df.groupby("regiao")["evasao_percentual"].mean().round(2))
```

```
Média de evasão percentual por região:
regiao
Centro-Oeste    10.64
Nordeste         9.61
Norte           12.61
Sudeste         11.47
Sul             12.08
Name: evasao_percentual, dtype: float64
```

2.4 Visualização da distribuição da evasão

Geramos um histograma para visualizar como a evasão percentual está distribuída entre as escolas.

[4]:
```
plt.figure(figsize=(8, 4))
sns.histplot(df['evasao_percentual'], bins=20, kde=True)
plt.title("Distribuição da Evasão Percentual")
plt.xlabel("Evasão (%)")
plt.ylabel("Frequência")
plt.grid(True)
plt.show()
```



2.5 Codificação de variáveis categóricas

Transformamos variáveis categóricas em variáveis numéricas usando one-hot encoding, necessário para os algoritmos de clustering.

[5]:
```
df_encoded = pd.get_dummies(df, columns=['regiao', 'localizacao',␣
↪'dependencia_adm'], drop_first=True)
```

[6]:
```
df_encoded
```

[6]:
```
    id_escola  infra_biblioteca  infra_internet  infra_laboratorio  \
0           1                 0               1                  0
1           2                 1               1                  0
2           3                 0               1                  0
3           4                 1               1                  1
4           5                 1               0                  1
..        ...               ...             ...                ...
95         96                 1               1                  1
96         97                 0               1                  1
97         98                 1               0                  0
98         99                 0               1                  1
99        100                 0               0                  1

    total_matriculados  evasao_percentual  evasao_absoluta  regiao_Nordeste  \
0                 1183          12.007803              142            False
1                  268          18.234745               48            False
2                  745          17.397170              129            False
```

```
3                  252          19.930682                50            False
4                  404           2.384619                 9            False
..                 …                …                   …              …
95                 308          10.911203                33            False
96                1369           0.000000                 0            False
97                 310           8.072309                25             True
98                 427          16.652193                71             True
99                 310          19.996586                61            False

    regiao_Norte  regiao_Sudeste  regiao_Sul  localizacao_Urbana  \
0          False           False        True               False
1          False           False       False               False
2          False            True       False                True
3          False           False       False                True
4          False           False       False                True
..           …               …           …                   …
95         False            True       False                True
96         False           False       False               False
97         False           False       False                True
98         False           False       False                True
99          True           False       False               False

    dependencia_adm_Federal  dependencia_adm_Municipal
0                     False                       True
1                     False                       True
2                     False                      False
3                     False                       True
4                     False                       True
..                      …                          …
95                    False                      False
96                    False                       True
97                    False                      False
98                    False                      False
99                    False                      False

[100 rows x 14 columns]
```

2.6 Seleção e normalização de variáveis

Selecionamos as variáveis mais relevantes para o agrupamento e aplicamos a padronização para evitar viés de escala.

[7]:
```
features = [
    'infra_biblioteca', 'infra_internet', 'infra_laboratorio',
    'total_matriculados', 'evasao_percentual'
] + [col for col in df_encoded.columns if any(prefix in col for prefix in␣
 ↪['regiao_', 'localizacao_', 'dependencia_adm_'])]
```

```
X = df_encoded[features]

scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

[8]:
```
X_scaled[0]
```

[8]:
```
array([-1.12815215,  0.56195149, -0.96076892,  0.94122889,  0.12371553,
       -0.51558005, -0.46852129, -0.43643578,  1.68705478, -1.82970656,
       -0.31448545,  1.19959343])
```

2.7 Escolha do melhor número de clusters (k)

Testamos diferentes valores de k (de 2 a 9) e calculamos o coeficiente de silhouette para encontrar o número ótimo de clusters.

[9]:
```
scores = []
for k in range(2, 10):
    kmeans = KMeans(n_clusters=k, random_state=42)
    labels = kmeans.fit_predict(X_scaled)
    score = silhouette_score(X_scaled, labels)
    scores.append((k, score))

# Exibe os resultados
for k, s in scores:
    print(f"Silhouette para k={k}: {s:.4f}")

best_k = max(scores, key=lambda x: x[1])[0]
print(f"\nMelhor número de clusters: {best_k}")
```

```
Silhouette para k=2: 0.0905
Silhouette para k=3: 0.1418
Silhouette para k=4: 0.1794
Silhouette para k=5: 0.1810
Silhouette para k=6: 0.1485
Silhouette para k=7: 0.1433
Silhouette para k=8: 0.1638
Silhouette para k=9: 0.1486

Melhor número de clusters: 5
```

2.8 Aplicação do K-means com o melhor k

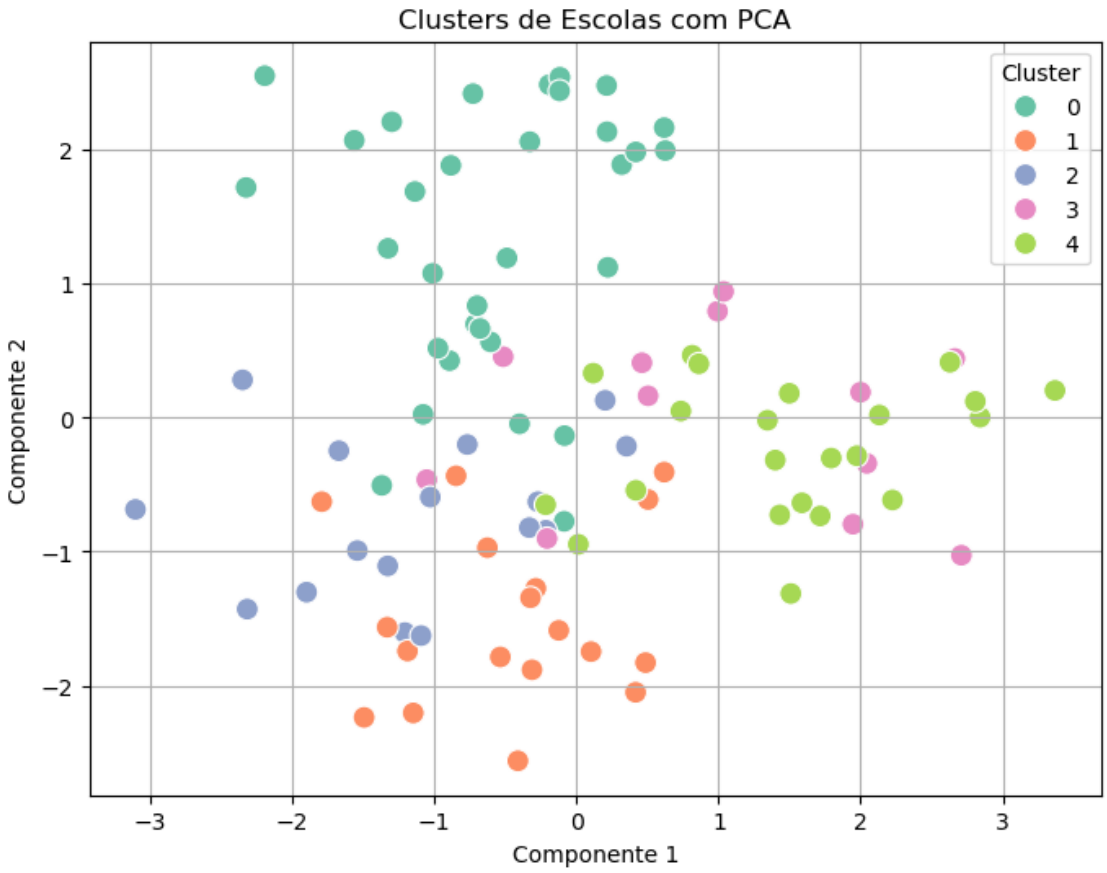
Executamos o algoritmo K-means com o número ideal de clusters encontrado anteriormente.

[10]:
```
kmeans = KMeans(n_clusters=best_k, random_state=42)
df['cluster'] = kmeans.fit_predict(X_scaled)
```

2.9 Visualização dos clusters com PCA

Usamos PCA para reduzir a dimensionalidade dos dados e visualizar os grupos formados.

[11]:
```
pca = PCA(n_components=2)
components = pca.fit_transform(X_scaled)
df['pca1'], df['pca2'] = components[:, 0], components[:, 1]

plt.figure(figsize=(8,6))
sns.scatterplot(data=df, x='pca1', y='pca2', hue='cluster', palette='Set2',␣
↪s=100)
plt.title('Clusters de Escolas com PCA')
plt.xlabel('Componente 1')
plt.ylabel('Componente 2')
plt.grid(True)
plt.legend(title='Cluster')
plt.show()
```



2.10 Interpretação dos clusters

Por fim, analisamos a média das variáveis principais por grupo para interpretar os perfis encontrados.

[12]:
```
print("Médias por cluster:")
df.groupby('cluster')[['infra_biblioteca', 'infra_internet',
                       'infra_laboratorio', 'total_matriculados',
                       'evasao_percentual']].mean().round(2)
```

```
Médias por cluster:
```

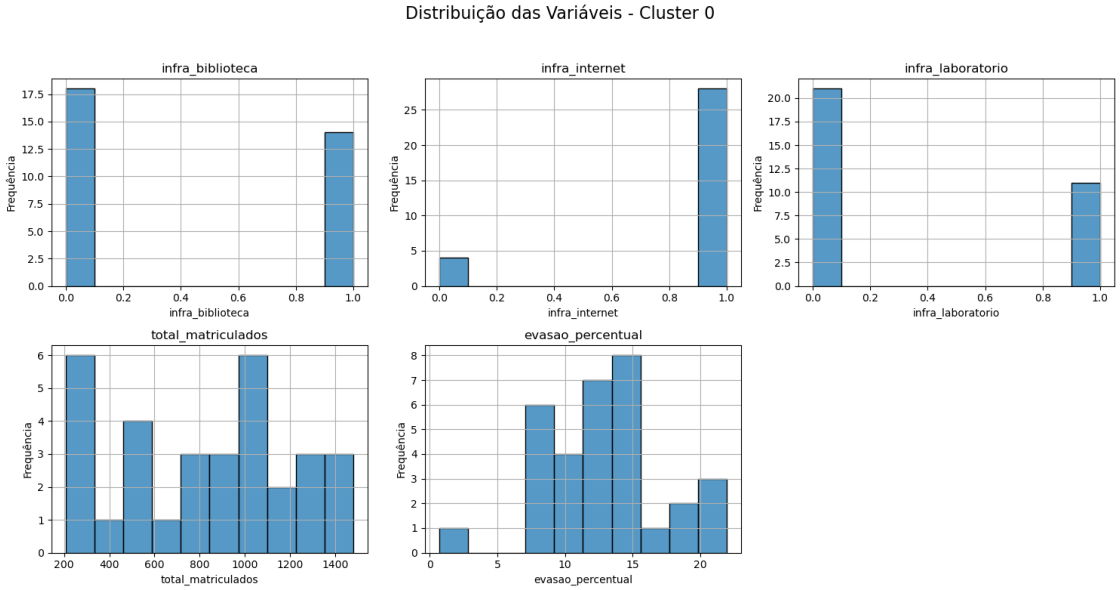
[12]:

cluster	infra_biblioteca	infra_internet	infra_laboratorio	total_matriculados	evasao_percentual
0	0.44	0.88	0.34	825.72	12.86
1	0.61	0.72	0.72	828.94	12.61
2	0.38	0.81	0.62	628.88	11.47
3	0.50	0.67	0.50	758.08	10.70
4	0.86	0.64	0.36	873.45	8.11

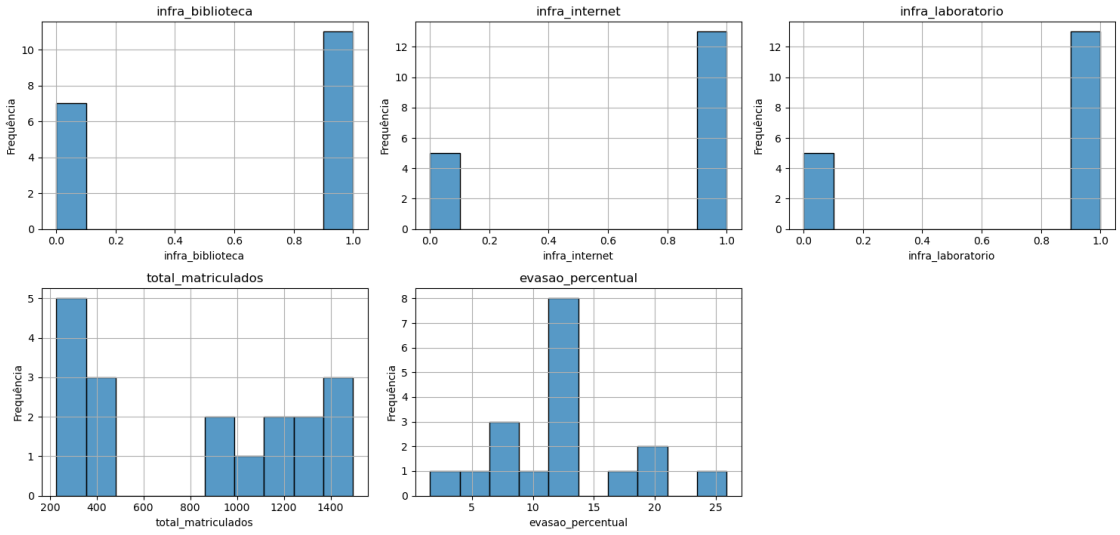
[13]:
```
# Visualização detalhada da distribuição para cada cluster com subplots 2x3
variables_to_plot = ['infra_biblioteca', 'infra_internet',
                     'infra_laboratorio', 'total_matriculados',
                     'evasao_percentual']

for cluster_id in sorted(df['cluster'].unique()):
    cluster_df = df[df['cluster'] == cluster_id]
    print(f"\nDistribuições para o Cluster {cluster_id}:")
    fig, axes = plt.subplots(2, 3, figsize=(15, 8))
    axes = axes.flatten()
    for i, var in enumerate(variables_to_plot):
        sns.histplot(cluster_df[var], bins=10, kde=False, ax=axes[i])
        axes[i].set_title(f'{var}')
        axes[i].set_xlabel(var)
        axes[i].set_ylabel('Frequência')
        axes[i].grid(True)
    for j in range(len(variables_to_plot), len(axes)):
        fig.delaxes(axes[j])  # Remove subplot vazio se houver
    fig.suptitle(f'Distribuição das Variáveis - Cluster {cluster_id}',␣
 ↪fontsize=16)
```

```
    plt.tight_layout(rect=[0, 0, 1, 0.95])
    plt.show()
```

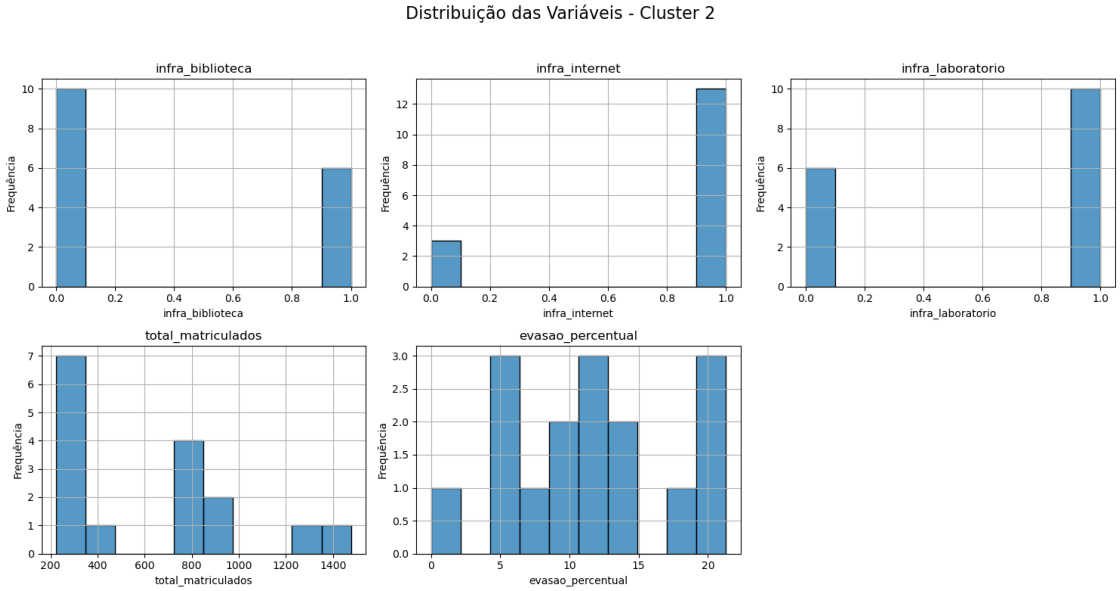
Distribuições para o Cluster 0:



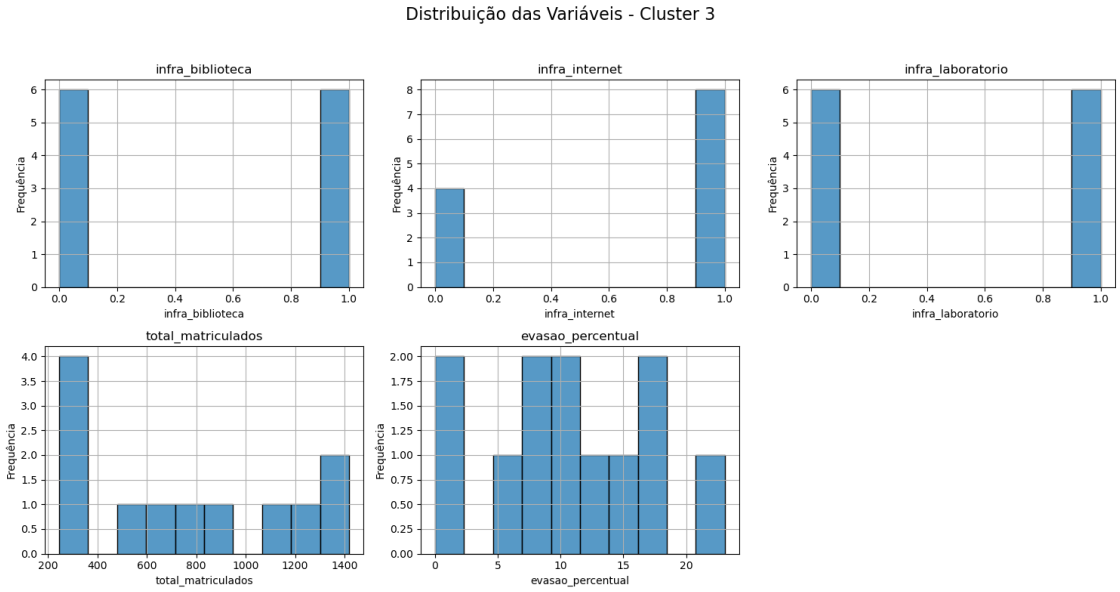
Distribuições para o Cluster 1:



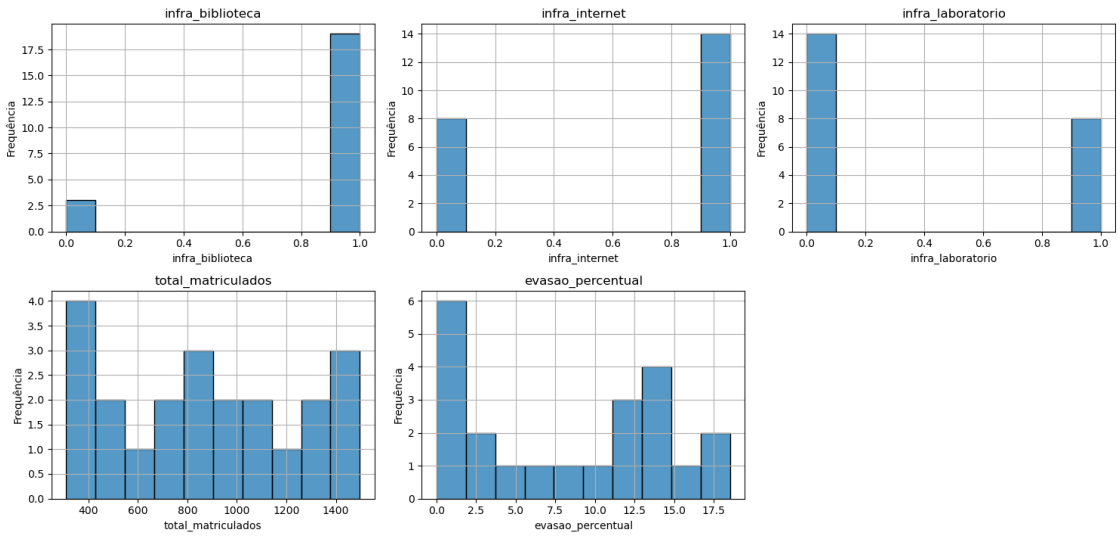
Distribuições para o Cluster 2:



Distribuições para o Cluster 3:



Distribuições para o Cluster 4:



3 Perguntas para Análise e Interpretação dos Clusters

3.0.1 Análise Técnica dos Clusters

1. Quais variáveis mais se destacam em cada cluster? (ex: evasão, infraestrutura, número de alunos)
2. Há variáveis que variam pouco entre os clusters? O que isso indica?
3. Em qual cluster estão concentradas as escolas com maior evasão?
4. Qual cluster apresenta maior variabilidade interna? O que isso pode significar?
5. As distribuições das variáveis são simétricas ou assimétricas dentro dos clusters?
6. Existe algum padrão geográfico (ex: região, zona urbana/rural) associado a cada cluster?
7. Alguma infraestrutura aparece de forma consistente nos clusters com menor evasão?

RESPOSTAS:

...

...

...

...

...

...

3.0.2 Análise Gerencial e Tomada de Decisão

8. Se você tivesse recursos limitados, qual cluster você priorizaria para intervenção imediata? Por quê?
9. Que tipo de ação (pedagógica, tecnológica, estrutural) seria mais adequada para o cluster com maior evasão?

10. O cluster com boas condições e baixa evasão pode servir de modelo? O que ele pode ensinar aos demais?
11. Que critérios objetivos você usaria para priorizar escolas dentro de um mesmo cluster?
12. Como os perfis encontrados poderiam orientar o planejamento de políticas públicas ou programas regionais?

RESPOSTAS:

...
...
...
...
...
...

Desafio Final (Reflexão): Se você fosse apresentar esses resultados à equipe gestora da Secretaria da Educação, que 3 recomendações faria com base nos clusters encontrados?

.
.
.
.
.

Bom trabalho!