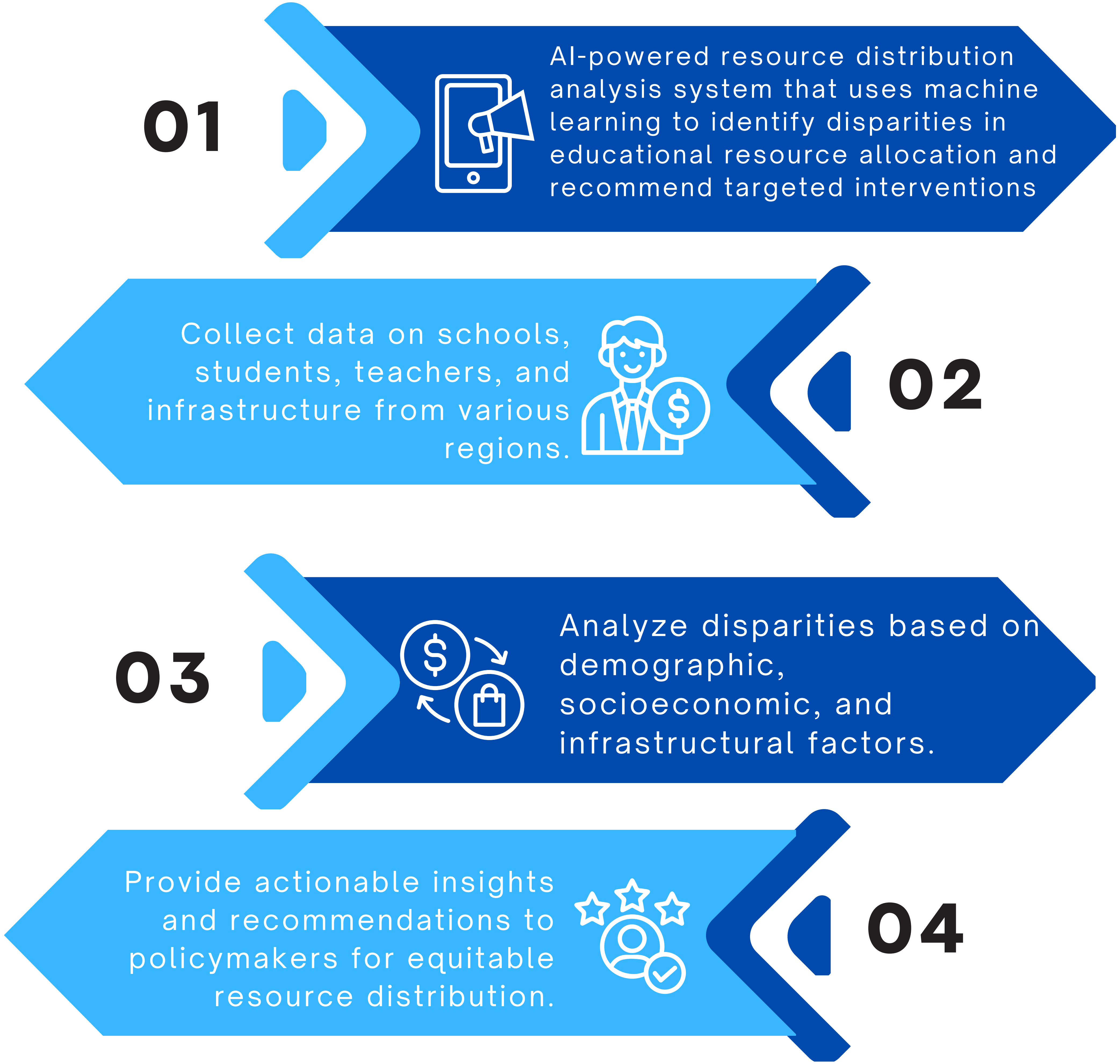
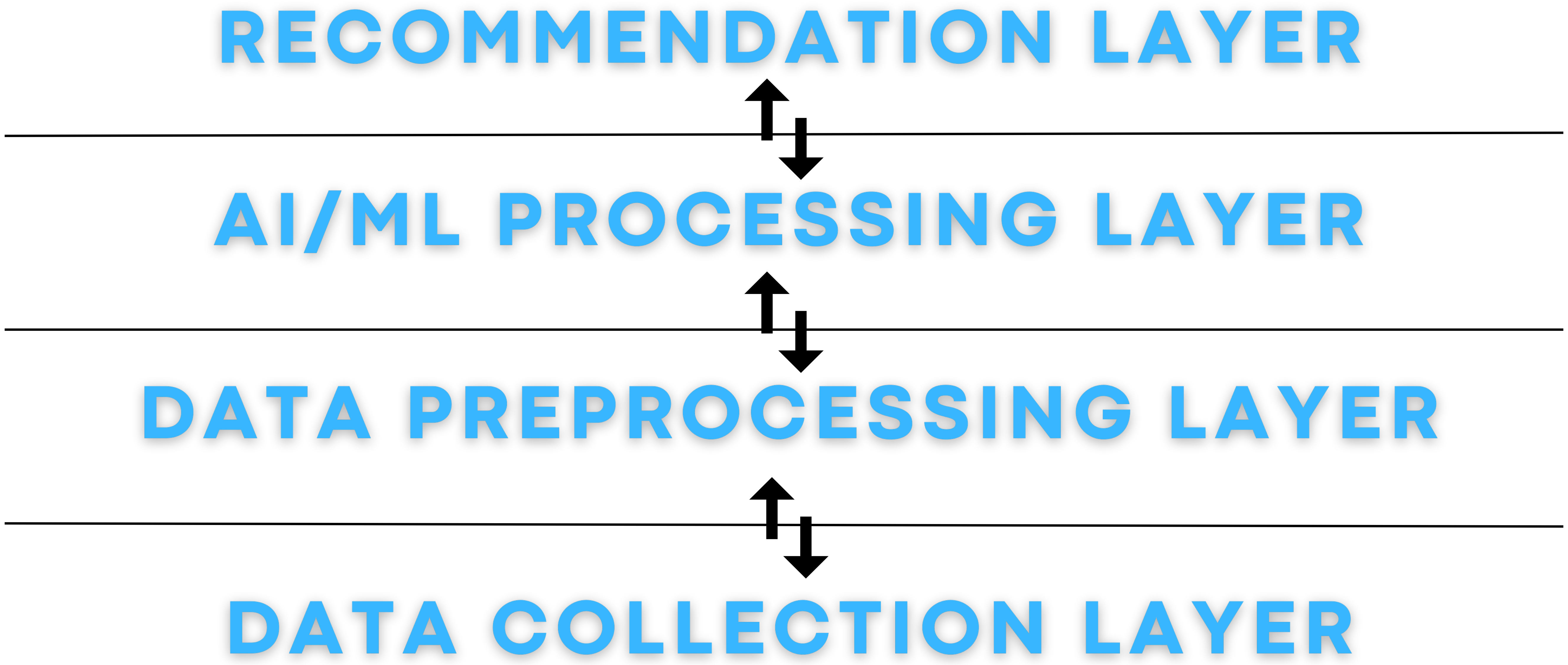
Team Prottush

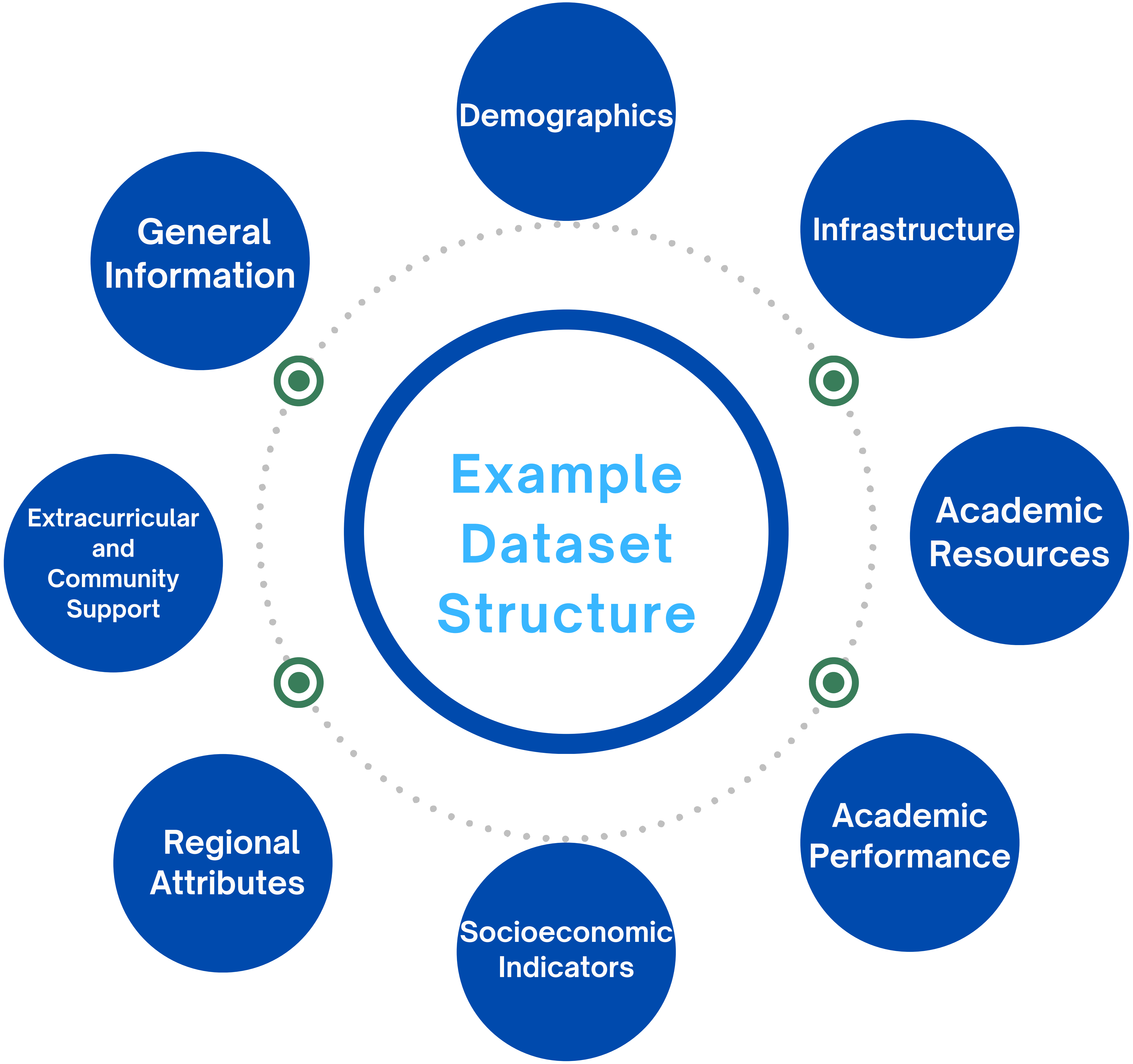
EDUCATIONAL RESOURCE ALLOCATION FOR BANGLADESH

NUSRAT JAHAN LIA (1306)
LABIB MUNTASIR(1319)
TASNIM MAHFUZ NAFIS(1327)


01
AI-powered resource distribution analysis system that uses machine learning to identify disparities in educational resource allocation and recommend targeted interventions
02
Collect data on schools, students, teachers, and infrastructure from various regions.
03
Analyze disparities based on demographic, socioeconomic, and infrastructural factors.
04
Provide actionable insights and recommendations to policymakers for equitable resource distribution.

High-Level System Architecture




Example Dataset Structure
Demographics
Infrastructure
Academic Resources
Academic Performance
Socioeconomic Indicators
Regional Attributes
Extracurricular and Community Support
General Information

Government databases: BANBEIS((Bangladesh Bureau of Educational Information and Statistics),) school profiles, Ministry of Education reports.

```python
institution = {
    'EIIN': fake.unique.bothify("#######"),
    'Institution_Name': f"Ins_{random.randint(1, 1000)}" + (" School" if inst_type == 'Secondary School' else
                                                            " College" if inst_type == 'College' else " Madrasah"),
    'Institution_Type': inst_type,
    'Management': np.random.choice(['Public', 'Private'], p=[0.15, 0.85]),
    'Establishment_Year': np.random.randint(1950, 2020),
    'Division': division,
    'District' : np.random.choice(bangladesh_districts[division]),

    # Madrasah Specific
    'Madrasah_Type': np.random.choice(['Dakhil', 'Alim', 'Fazil', 'Kamil'], p=[0.7, 0.15, 0.1, 0.05]) if inst_type == 'Madrasah' else None,

    # College Specific
    'College_Type': np.random.choice([
        'School and College', 'Higher Secondary',
        'Degree (Pass)', 'Degree (Honors)', 'Masters'
    ], p=[0.3, 0.25, 0.2, 0.15, 0.1]) if inst_type == 'College' else None,

    # Initial Quality Indicators (2010 baseline)
    'Computer_Facility_2010': np.random.choice([0,1], p=[0.4, 0.6]) if inst_type == 'College' else np.random.choice([0,1], p=[0.7, 0.3]),
    'Internet_2010': 0  # Will evolve over years
}
institutions.append(institution)
```

BANBEIS 2021, APSC Report, DPE Statistics

```python
for year in range(start_year, end_year + 1):
    # Base population growth
    students = max(100, int(np.random.normal(
        500 if inst['Institution_Type'] == 'Secondary School' else 800,
        200
    ) * (1.03 ** (year - 2010))))

    # Gender ratios with temporal improvement
    female_student_pct = np.clip(np.random.normal(
        54.67 if inst['Institution_Type'] == 'Secondary School' else
        50.37 if inst['Institution_Type'] == 'College' else 33.42,
        5 + (year-2010)*0.3  # Reducing variance over time
    ), 30, 70)


  prob = 0.9492 + 0.005 * (year - 2010)
  prob = min(max(prob, 0), 1)


  # Quality indicator progression
  quality = {
      'TSR': np.clip(np.random.normal(
          34 - (year-2010)*0.5 if inst['Institution_Type'] == 'Secondary School' else
          27 + (year-2010)*0.3 if inst['Institution_Type'] == 'College' else 20,
          2
      ), 15, 60),

      'Computer_Facility': inst['Computer_Facility_2010'] * (1 + 0.1*(year-2010)),
      'Internet_Access': min(1, inst['Internet_2010'] + 0.12*(year-2010)),

      'Electricity_Access': np.clip(np.random.normal(0.7 + 0.03*(year-2010), 0.1), 0, 1),
      # 'Girls_Toilet': np.random.choice([0,1], p=[
      #     1 - (0.9492 + 0.005*(year-2010)),
      #     0.9492 + 0.005*(year-2010)
      # ])
      'Girls_Toilet': np.random.choice([0, 1], p=[1 - prob, prob])
```

```python
# Academic performance
dropout_rate = np.clip(np.random.normal(
    35.66 - 0.8*(year-2010) if inst['Institution_Type'] == 'Secondary School' else
    21.14 - 0.5*(year-2010) if inst['Institution_Type'] == 'College' else 40.00 - 1.2*(year-2010),
    3
), 5, 50)


yearly_data = {
    **inst,
    'Year': year,
    'Total_Students': students,
    'Female_Students': int(students * female_student_pct/100),
    'Disabled_Students': int(np.random.normal(students*0.05, students*0.01)),
    'Teachers': max(5, int(students / quality['TSR'])),
    'TSR': quality['TSR'],
    'Dropout_Rate': dropout_rate,
    'Completion_Rate': 100 - dropout_rate,
    'Pass_Rate': np.clip(np.random.normal(
        83.04 + 0.3*(year-2010) if 'School' in inst['Institution_Type'] else
        77.78 + 0.4*(year-2010), 3
    ), 60, 99),
    'Computer_Facility': quality['Computer_Facility'],
    'Internet_Access': quality['Internet_Access'],
    'Electricity_Access': quality['Electricity_Access'],
    'Girls_Toilet': quality['Girls_Toilet'],
    'Budget': budget,
    'Community_Contribution': budget * np.random.uniform(0.01, 0.1),
    # 'SPI': np.random.normal(500, 100),  # Student Performance Index
    # 'TPI': np.random.normal(30, 5),     # Teacher Performance Index
    'Solar_System': np.random.choice([0,1], p=[0.9, 0.1]) if year < 2015 else np.random.choice([0,1], p=[0.7, 0.3])
}
```

11966	1179271	Ins_71 Madr	Madrasah	Private	1996	Mymensingh	Netrakona	Kamil		1	0	2010	983	334	38	42	23.10110352	35.42893045	64.57106954	73.79760313	1
11967	1179271	Ins_71 Madr	Madrasah	Private	1996	Mymensingh	Netrakona	Kamil		1	0	2011	849	254	45	39	21.27103745	37.03046524	62.96953475	79.43445388	1
11968	1179271	Ins_71 Madr	Madrasah	Private	1996	Mymensingh	Netrakona	Kamil		1	0	2012	644	193	39	38	16.92434877	45.10464644	54.89535355	78.6513886	1
11969	1179271	Ins_71 Madr	Madrasah	Private	1996	Mymensingh	Netrakona	Kamil		1	0	2013	555	202	26	27	19.88846704	36.57051377	63.42948622	80.23048613	1
11970	1179271	Ins_71 Madr	Madrasah	Private	1996	Mymensingh	Netrakona	Kamil		1	0	2014	948	334	46	47	19.98577844	35.17945332	64.82054667	78.55075086	1
11971	1179271	Ins_71 Madr	Madrasah	Private	1996	Mymensingh	Netrakona	Kamil		1	0	2015	733	219	41	34	21.39164226	35.09504496	64.90495503	79.80496409	1
11972	1179271	Ins_71 Madr	Madrasah	Private	1996	Mymensingh	Netrakona	Kamil		1	0	2016	573	247	29	26	21.6827194	28.61016289	71.38983710	77.08776066	1
11973	1179271	Ins_71 Madr	Madrasah	Private	1996	Mymensingh	Netrakona	Kamil		1	0	2017	544	163	29	28	18.76318500	33.3298454	66.67015456	82.56620853	1
11974	1179271	Ins_71 Madr	Madrasah	Private	1996	Mymensingh	Netrakona	Kamil		1	0	2018	1062	344	46	54	19.44085621	32.55678919	67.44321080	79.32361692	1
11975	1179271	Ins_71 Madr	Madrasah	Private	1996	Mymensingh	Netrakona	Kamil		1	0	2019	769	230	36	37	20.58676372	32.17946404	67.82053595	81.2230236	1
11976	1179271	Ins_71 Madr	Madrasah	Private	1996	Mymensingh	Netrakona	Kamil		1	0	2020	1401	511	54	58	24.02203650	24.74386099	75.25613900	83.86657171	1
11977	1179271	Ins_71 Madr	Madrasah	Private	1996	Mymensingh	Netrakona	Kamil		1	0	2021	1456	436	76	63	23.09272348	29.65726534	70.34273465	81.70040733	1
11978	3217458	Ins_284 Colle	College	Public	1970	Barishal	Bhola		School and C	0	0	2010	808	425	35	27	29.30359049	19.02679544	80.97320455	80.75695094	0
11979	3217458	Ins_284 Colle	College	Public	1970	Barishal	Bhola		School and C	0	0	2011	891	402	42	32	27.10854581	23.44662790	76.55337209	77.26218264	0
11980	3217458	Ins_284 Colle	College	Public	1970	Barishal	Bhola		School and C	0	0	2012	808	491	35	30	26.33933273	18.19977410	81.80022589	77.44985332	0
11981	3217458	Ins_284 Colle	College	Public	1970	Barishal	Bhola		School and C	0	0	2013	785	311	52	24	31.83729926	21.72991415	78.27008584	82.63312928	0
11982	3217458	Ins_284 Colle	College	Public	1970	Barishal	Bhola		School and C	0	0	2014	565	276	25	19	29.30278101	19.12691373	80.87308626	74.44525227	0
11983	3217458	Ins_284 Colle	College	Public	1970	Barishal	Bhola		School and C	0	0	2015	880	455	35	30	29.22964422	13.19138024	86.8086198	79.17245560	0
11984	3217458	Ins_284 Colle	College	Public	1970	Barishal	Bhola		School and C	0	0	2016	1190	517	63	38	30.53040091	17.63307061	82.36692938	77.55129004	0
11985	3217458	Ins_284 Colle	College	Public	1970	Barishal	Bhola		School and C	0	0	2017	1024	510	59	33	30.81110050	15.18432892	84.81567107	84.30197232	0
11986	3217458	Ins_284 Colle	College	Public	1970	Barishal	Bhola		School and C	0	0	2018	1251	585	56	34	36.53693699	8.621606432	91.37839356	77.11908779	0
11987	3217458	Ins_284 Colle	College	Public	1970	Barishal	Bhola		School and C	0	0	2019	1535	655	104	50	30.34635295	15.86161771	84.13838228	78.86074494	0
11988	3217458	Ins_284 Colle	College	Public	1970	Barishal	Bhola		School and C	0	0	2020	1348	558	65	42	31.52147762	14.27006662	85.72993337	76.936447	0
11989	3217458	Ins_284 Colle	College	Public	1970	Barishal	Bhola		School and C	0	0	2021	795	405	51	25	30.60572969	15.79472905	84.20527094	86.78578784	0
11990	1522842	Ins_793 Scho	Secondary Sc	Public	2001	Sylhet	Sylhet			0	0	2010	485	241	20	14	32.87050011	38.1846385	61.8153615	80.2531291	0
11991	1522842	Ins_793 Scho	Secondary Sc	Public	2001	Sylhet	Sylhet			0	0	2011	687	344	39	19	34.72449049	38.17998559	61.82001440	85.24270802	0
11992	1522842	Ins_793 Scho	Secondary Sc	Public	2001	Sylhet	Sylhet			0	0	2012	641	332	38	20	32.03680075	34.39655096	65.60344903	84.07317856	0
11993	1522842	Ins_793 Scho	Secondary Sc	Public	2001	Sylhet	Sylhet			0	0	2013	654	345	37	19	33.7988525	37.61872876	62.38127123	87.42829467	0
11994	1522842	Ins_793 Scho	Secondary Sc	Public	2001	Sylhet	Sylhet			0	0	2014	560	275	31	18	30.56151614	33.1810527	66.81894728	84.95156164	0
11995	1522842	Ins_793 Scho	Secondary Sc	Public	2001	Sylhet	Sylhet			0	0	2015	466	261	23	15	30.42005061	33.07875992	66.92124007	79.42792575	0
11996	1522842	Ins_793 Scho	Secondary Sc	Public	2001	Sylhet	Sylhet			0	0	2016	1059	668	53	31	33.69286778	33.78999573	66.21000426	87.14026759	0
11997	1522842	Ins_793 Scho	Secondary Sc	Public	2001	Sylhet	Sylhet			0	0	2017	908	576	52	31	28.64841704	33.8996331	66.1003669	84.2475159	0
11998	1522842	Ins_793 Scho	Secondary Sc	Public	2001	Sylhet	Sylhet			0	0	2018	468	220	23	14	32.6221637	30.91953137	69.08046862	82.38107245	0
11999	1522842	Ins_793 Scho	Secondary Sc	Public	2001	Sylhet	Sylhet			0	0	2019	1046	660	25	36	28.77134664	34.45302539	65.54697460	85.1760556	0
12000	1522842	Ins_793 Scho	Secondary Sc	Public	2001	Sylhet	Sylhet			0	0	2020	816	390	41	25	31.63609325	28.47055110	71.5294489	87.3646592	0

Random Forest for Prediction

```python
import pandas as pd
import numpy as np
import joblib
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split

df = pd.read_csv("/Users/macbook/Documents/GitHub/Educational-Resource-Allocation-for-Bangladesh/backend/data/education_data.csv")

features = ["Total_Students", "Female_Students", "Disabled_Students", "Teachers", "Budget", "Pass_Rate", "Internet_Access"]
target_dropout, target_completion, target_tsr = "Dropout_Rate", "Completion_Rate", "TSR"

X = df[features]
y_dropout, y_completion, y_tsr = df[target_dropout], df[target_completion], df[target_tsr]

X_train, X_test, y_dropout_train, y_dropout_test = train_test_split(X, y_dropout, test_size=0.2, random_state=42)
X_train, X_test, y_completion_train, y_completion_test = train_test_split(X, y_completion, test_size=0.2, random_state=42)
X_train, X_test, y_tsr_train, y_tsr_test = train_test_split(X, y_tsr, test_size=0.2, random_state=42)

dropout_model = RandomForestRegressor(n_estimators=100, random_state=42).fit(X_train, y_dropout_train)
completion_model = RandomForestRegressor(n_estimators=100, random_state=42).fit(X_train, y_completion_train)
tsr_model = RandomForestRegressor(n_estimators=100, random_state=42).fit(X_train, y_tsr_train)

joblib.dump(dropout_model, "/Users/macbook/Documents/GitHub/Educational-Resource-Allocation-for-Bangladesh/backend/data/models/dropout_model.pkl")
joblib.dump(completion_model, "/Users/macbook/Documents/GitHub/Educational-Resource-Allocation-for-Bangladesh/backend/data/models/completion_model.pkl")
joblib.dump(tsr_model, "/Users/macbook/Documents/GitHub/Educational-Resource-Allocation-for-Bangladesh/backend/data/models/tsr_model.pkl")
```

Usage of Prediction

Dropout Rate Prediction:

Features like total students, teachers, and budget can help predict dropout rates, highlighting the importance of institutional resources and student demographics in retention.

Completion Rate Prediction:

Factors such as internet access, female students, and teacher-student ratio play a significant role in predicting completion rates, offering insights into how educational support structures affect graduation.

TSR (Teacher-Student Ratio) Optimization:

By analyzing factors like teachers and total students, the model can predict how effectively an institution is managing its educational workforce, which is critical for improving teaching quality and student outcomes.

Clustering with K-Means

```python
import pandas as pd
import joblib
from sklearn.cluster import KMeans
from sklearn.preprocessing import MinMaxScaler


df = pd.read_csv("/Users/macbook/Documents/GitHub/Educational-Resource-Allocation-for-Bangladesh/backend/data/education_data.csv")
X = df[["Total_Students", "Budget", "Teachers", "Internet_Access"]]

num_clusters = 5
kmeans = KMeans(n_clusters=num_clusters, random_state=42).fit(X)

df["Cluster"] = kmeans.labels_
infra_features = ["Internet_Access", "Budget", "Teachers"]

# Normalize using Min-Max Scaling
scaler = MinMaxScaler()
df[infra_features] = scaler.fit_transform(df[infra_features])

weights = {
    "Internet_Access": 0.2,  # 20%
    "Budget": 0.5,           # 50%
    "Teachers": 0.3          # 30%
}

df["Infrastructure_Score"] = (
    df["Internet_Access"] * weights["Internet_Access"] +
    df["Budget"] * weights["Budget"] +
    df["Teachers"] * weights["Teachers"]
)
# df["Infrastructure_Score"] = df["Infrastructure_Score"].clip(0, 10)


joblib.dump(kmeans, "/Users/macbook/Documents/GitHub/Educational-Resource-Allocation-for-Bangladesh/backend/data/models/kmeans_model.pkl")
df.to_csv("/Users/macbook/Documents/GitHub/Educational-Resource-Allocation-for-Bangladesh/backend/data/education_data_clustered.csv", index=False)
```

Usage of Clustering

Infrastructure Grouping: The algorithm clusters schools based on their infrastructure characteristics (internet access, budget, and teachers), which provides a clearer picture of resource distribution across different types of schools. For example, schools with similar infrastructure needs can be grouped together, helping policymakers allocate resources more efficiently.

Cluster-Based Infrastructure Scores: By incorporating infrastructure scores that weigh internet access, budget, and teachers, this model helps assess which schools have the most robust infrastructure. This score can guide decisions on which schools need immediate attention or which clusters might benefit from targeted investments.

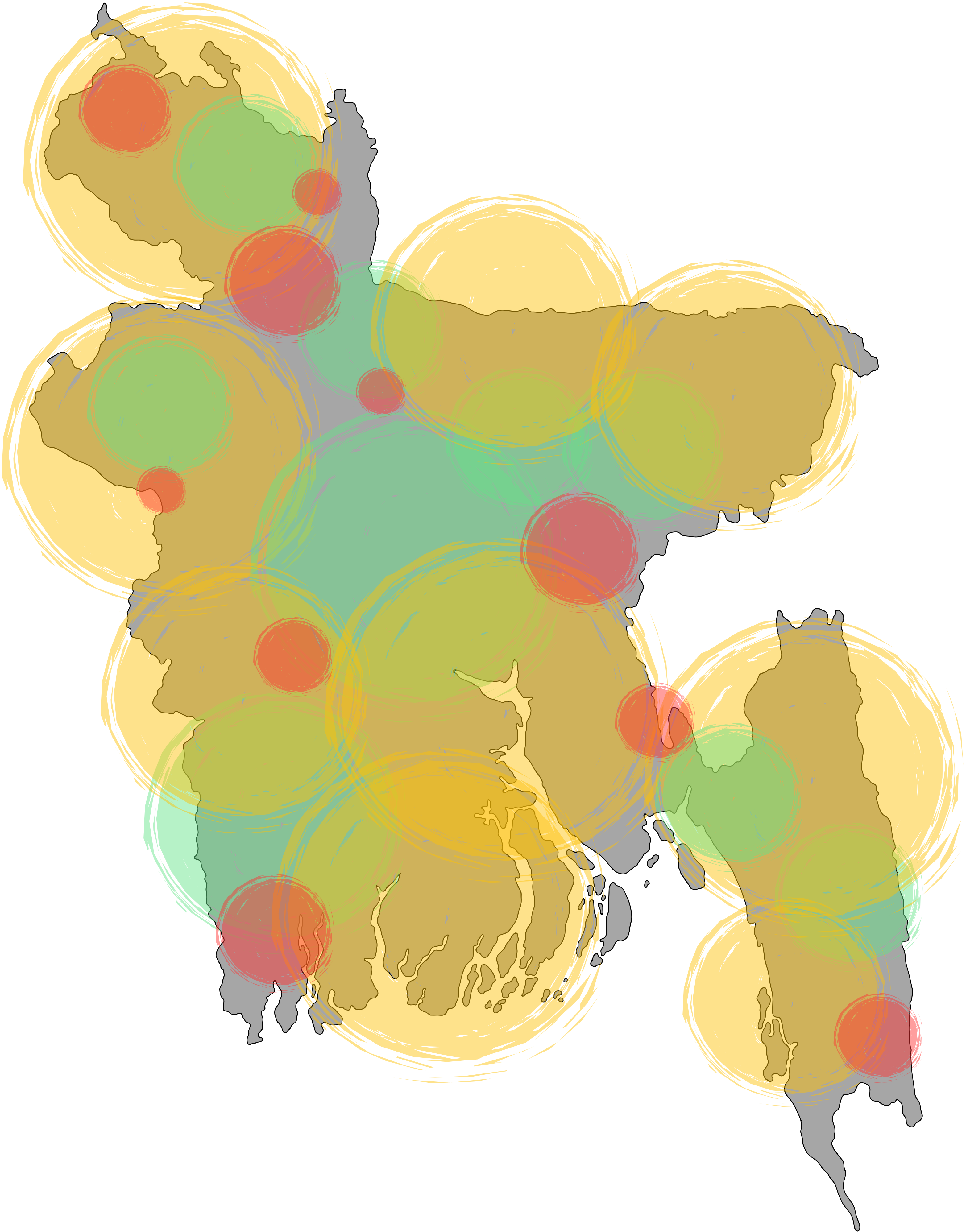
Targeted Interventions: The clustering results can inform tailored interventions for each group, optimizing the allocation of educational resources. Schools with high needs (e.g., low internet access) can be prioritized for digital learning tools, while those with stronger infrastructure may need support in other areas, such as teacher training.

Recommendation based on Clusters

```python
import numpy as np

def generate_recommendations(predicted_dropout, predicted_completion, predicted_tsr):
    """
    Generates recommendations based on predicted values.
    """

    recommendations = []

    # Dropout Rate Recommendations
    if predicted_dropout > 20:
        recommendations.append("Increase student retention programs (mentorship, financial aid, extracurricular activities).")
        recommendations.append("Improve school infrastructure (libraries, clean water, digital classrooms).")

    # Completion Rate Recommendations
    if predicted_completion < 80:
        recommendations.append("Enhance curriculum relevance with real-world applications.")
        recommendations.append("Provide better access to study materials and e-learning resources.")

    # TSR Recommendations
    if predicted_tsr < 30:
        recommendations.append("Hire additional teachers to improve student-to-teacher ratio.")
        recommendations.append("Implement teacher training workshops for better engagement.")

    return recommendations
```

Resource Recommendation for Division/District/Schools

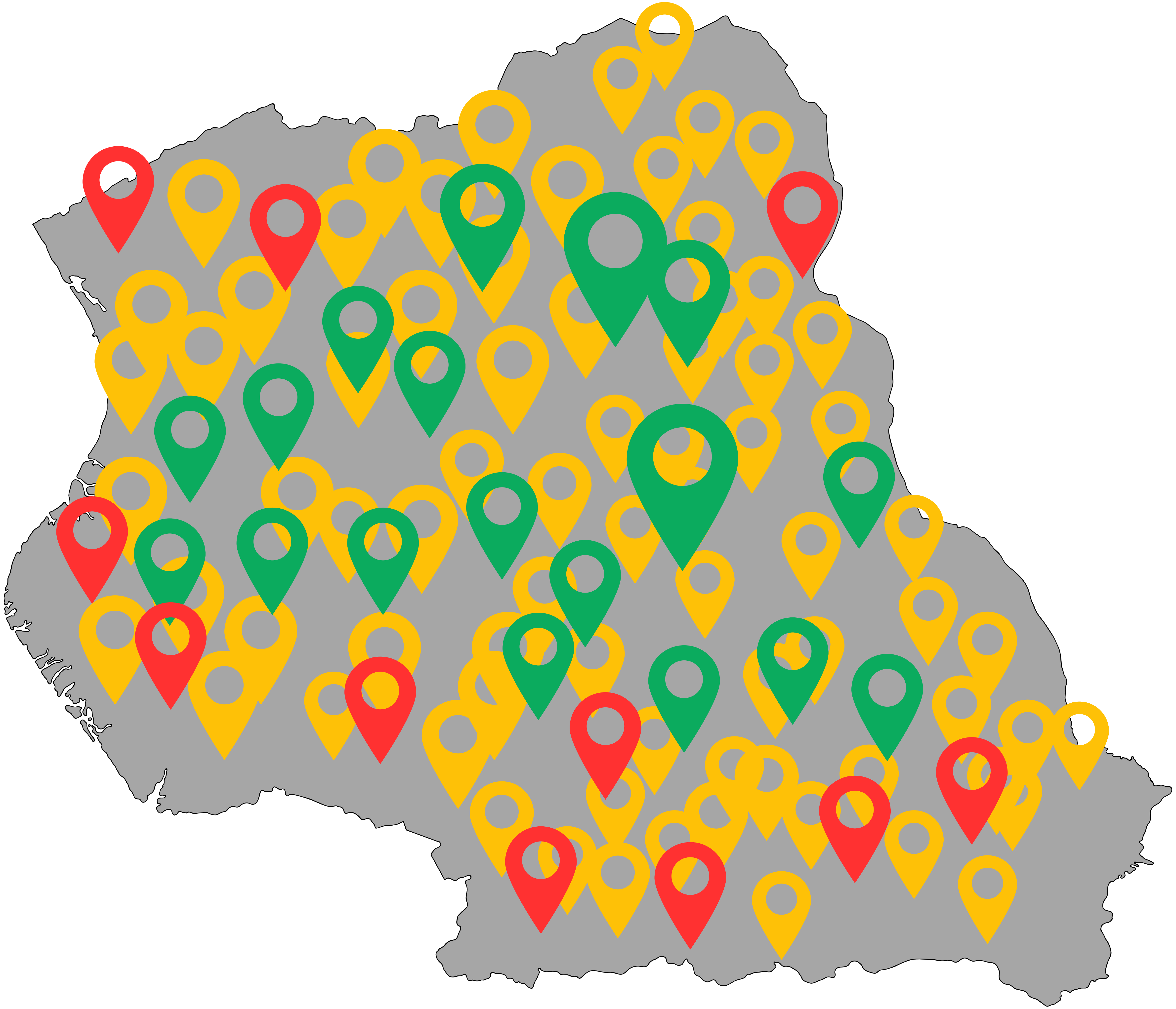
```python
def recommend_resource_allocation(division_data):
    """
    Generates resource allocation recommendations based on division statistics.
    """

    avg_dropout = division_data["Dropout_Rate"].mean()
    avg_completion = division_data["Completion_Rate"].mean()
    avg_tsr = division_data["TSR"].mean()
    avg_infra_score = division_data["Infrastructure_Score"].mean()

    recommendations = []

    if avg_dropout > 20:
        recommendations.append("Increase funding for student support programs in high-dropout areas.")
        recommendations.append("Provide scholarships or financial incentives for students in need.")

    if avg_completion < 80:
        recommendations.append("Improve curriculum with practical learning techniques.")
        recommendations.append("Introduce technology-driven education like smart classrooms.")

    if avg_tsr < 30:
        recommendations.append("Hire additional teachers and conduct teacher training programs.")

    if avg_infra_score < 10:
        recommendations.append("Upgrade school infrastructure (Internet, labs, libraries, sanitation).")

    return recommendations
```



Moderately Concerning

Least Concerning

Most Concerning



Metric	Rajshahi Avg	National Avg	Key Observations
Infrastructure Score	4.5/ 10	6.5 / 10	Below average, infrastructure upgrades needed
Dropout Rate (%)	20.5%	15.0%	Higher than national average, requires intervention
Completion Rate (%)	68%	78%	Low, major issue in rural schools
Teacher-Student Ratio (TSR)	1:102	1:90	Needs more teachers per student
Internet Access (%)	40% of schools	55%	Low digital accessibility, hinders modern learning

THANK YOU

Team Prottush