The MIRA Project: A Study of Internet Resiliency among Countries in Africa

Blaise Viateur Niyigena
bniyigen@andrew.cmu.edu

Esther Kamau
emkamau@andrew.cmu.edu

Isaac Manzi
isaacm@andrew.cmu.edu

Pasqua Ruggiero
pasquar@andrew.cmu.edu

May 17, 2021

Abstract

The heightened importance of the Internet has indicated the need for a solution to determine which countries are in higher need of more secure Internet infrastructure to prevent significant damages from power outages. The nonprofit organization AFRINIC has created the MIRA project to construct a methodology to measure Internet resiliency for countries in Africa and to thus be able to provide recommendations to achieve higher resiliency. This framework was constructed by taking weighted sums of selected metrics (pertinent to measuring Internet resiliency) and using this model as the country's Internet resiliency score. These scores were calculated and displayed in a dashboard to facilitate the comparison of scores between countries. Though there is a future need for more data to make more conclusions regarding the countries that are in most need to improve their Resiliency scores, the data that is available reveals that there is a vast difference between countries' scores. This prototype reveals that there are countries such as Ethiopia that are in greater need of interventions to improve their Internet resiliency, hence revealing the benefits of constructing such an index.

1 Introduction

The security and reliability of Internet connectivity is of utmost priority to many today, especially since the COVID-19 pandemic has revealed the importance of stable connection to each other during a time of limited physical social interactions (Deloitte, 2020). Though there has been a large emphasis on the strengthening of the Internet during these times, the distribution of these measures has not been equal across all countries.

In particular, many low-income countries do not get to benefit from as stable of Internet connectivity as many others in the world due to facing various issues such as under-provisioned networks, lack of proper cable infrastructure, or even redundant interconnection systems. These areas are more prone to widespread and high impact internet outages when their infrastructure is compromised, in the case of a cable break or a power failure. The results from the power outages are incredibly detrimental and impact the entire Internet ecosystem, leading to revenue loss to their digital economy (AFRINIC, 2021). The impact of the Internet outages could be drastically decreased if these countries had the ability to thoroughly audit their Internet Infrastructure and were to implement best practices for building resilient Internet infrastructures.

The prevalence of discrepancy between Internet resiliency among African countries and the desire to provide Internet support to low-income countries is the driving factor that lead to AFRINIC, a nonprofit Internet registry and research organization, to start the MIRA (Measuring Internet Resilience in Africa) project. The MIRA project is the result of a cross-collaborative effort between AFRINIC and Carnegie Mellon University's Pittsburgh and Africa campuses.

The aim of the MIRA project is to create a framework that can evaluate a country's capability to provide reliable means of Internet connectivity and to be able to do so in times of crises (AFRINIC, 2021). The framework is to be created by creating an index composed of various network resiliency metrics. This framework, which will be referred to as the Internet Resiliency Index, will be used to assign a score to each country that reflects the resilience of their physical and logical Internet infrastructure. By doing so, recommendations can be provided to help networks and countries achieve higher resilience (AFRINIC, 2021).

2 Data

The primary types of data to be used for this analysis is recent data collected for the various resilience metrics that have been determined to be most important for creating the resilience index (this is explained further in the Methods section). Most data was obtained through open Internet data sources, though data that was not readily available was acquired through either requests to private organizations for their internal data, or by utilizing AFRINIC's internet probes as a primary source for gathering this information. Table 1 dictates the different metrics that were selected to be included in the Internet Resiliency Index. As indicated below, there were 17 metrics that were selected to be included in this framework. Further details regarding data procurement can be found in Technical Appendix 1.

Before these datasets could be incorporated into our framework, various levels of preprocessing were carried out to ensure that the data were tidy, normalized, and sufficiently representative of countries in Africa. The ideal cleaned datasets would consist of 57 rows while those with most cases of missing data consisted of at least more than 30 entries - one row per country of interest as well as the subsequent Internet measures. A more detailed explanation of the preprocessing steps are included in Technical Appendix 2.

Category	**Metric**	**Measurement**
QoS	Throughput - Download Speed Fluctuations	Fluctuations in throughput - download speeds
	Throughput - Download Speed	download speeds
	Throughput - Upload Speed Fluctuations	Fluctuations in throughput - upload speeds
	Throughput - Upload Speed	upload speeds
	Latency Fluctuations	Latency to local services fluctuations
	Latency	Latency to local services
Security	IPv6 capability	IPv6 capability of the ISP network
	MANRS score (Routing regulations)	% of prefixes covered by IRR object
	AS hegemony	Compute the AS dependency of network
	DDos Potential	Level of risks posed to other countries
	Spam Infection	% spam infection 17
Infrastructure	IXP efficiency	% of ASes present at the IXP
	Upstream	Number of upstream providers
	Cable landing stations	Number of cable landing stations per capita/km2
	reach	% of population within 10-KM reach
	degree distribution	Degree distribution of cable entering/leaving a country/city
Affordability	Affordability	How affordable is Internet services in this country

Table 1: Table of metrics selected to be incorporated for calculating a country's Internet resiliency Score.

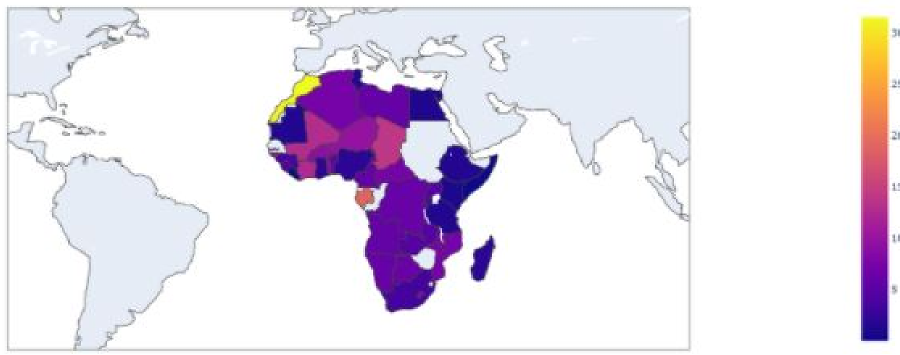

Figure 1: Choropleth map for Affordability created using Python's Plotly library.

Another step that was taken before creating the aggregate index was to perform an initial exploratory data analysis to determine if there were any potential patterns regarding country representation for all datasets, as well as visualize the metrics' measurements across Africa. This was done by calculating the correlations between metric measurements. Additionally, geographic maps were created, which served to be an important and useful tool for comparing individual metric measurements between African countries. For the data to be aggregated, the raw values for each indicator were converted to an equivalent scale and unit using a min/max normalization for quantitative measurements.

Figure 1 indicates one example of the geographic maps. This particular map represents the measurements for cheapest prepaid mobile plan (Affordability). In this map, there are various trends prevalent, such as the most expensive plans being in Western and North-Western Africa, while the cheapest plans mostly lie along the Eastern coastline. More details on the implementation of these analyses, as well as descriptions and results of further exploratory analyses can be found in Technical Appendix 2.

3 Methods

The best method to measure the Internet resiliency of African countries was determined to be the creation of an index that was composed of various aggregated network resiliency metrics. The objective of this aggregation is to enable us to construct a composite score that effectively communicates how a country's Internet Resilience ranks relative to others on the African continent .

The first step towards creating this tool consisted of extensive research on potential metrics. This procedure required not only formal research on various Internet metrics, but also included research into other Internet security indexes that have been created and previous suggestions from AFRINIC personnel. Additionally, it was crucial to ask for the opinion of Subject Matter Experts in the selection of these various metrics since no statistical methodology for index computation is independent of the theoretical aspect. This primarily took the form of structured discussions between the researchers and the experienced personnel at AFRINIC.

There were various criteria that were considered when choosing the most relevant metrics of measurement to include in the Internet Resiliency Index. There are many types of Internet measurement metrics that could be useful, but considering certain criteria before selecting the metrics helped in selecting only the highest quality forms of measurements were used. At the end, there were seventeen metrics selected to be included. The qualities of consideration were as follows.

1. Measurements that are easily attainable, whether it be through open source data or through attainment of internal data. Though the theoretical foundation of the Internet Resilience Index is very important, there is a need for real data to be able to test and display the scores produced by this framework.

2. Given time restraints, the data must include the corresponding countries of measurements, or at the very least be able to be easily joined with other datasets of country descriptions to map one metric measure per country. As the goal of this framework is to eventually prescribe recommendations for improving Internet infrastructure and security for low-income countries, it is important to have a basis for how the countries score on each metric.

3. The metrics must have data available that is recent (within the last two years). This way, the most accurate scores can be calculated per country. This also provides incentive to collect more updated data in future years when the metric scores are updated.

4. The metric must have data available that has sufficient coverage of African countries. Few metrics that were initially considered had less than fifty percent coverage of African countries in the available data. Incorporating this much missing data would not be ideal and would make the process of comparing metrics more sparse and thus more difficult.

5. The metrics must not be duplicates of each other. This is to avoid double-penalizing countries that may rank low on certain metrics (that have similar measures incorporated into the framework) and to give space for the inclusion of more insightful metrics to be incorporated instead.

6. Most importantly, the metrics are indicative of a certain aspect of a country's Internet resiliency.

After selecting the most relevant, available and representative metrics to create the framework, the metrics were grouped into various categories based on their type of measurement. This was used to determine the levels of weights that are necessary to consider in creating the Internet Resiliency Index. This selection of categories was primarily influenced by the introductory MIRA paper (AFRINIC, 2021), which highlighted important categories of consideration. Further information regarding the selection of the metrics can be found in Technical Appendix 3.

After structuring the metrics into categories, one of the final steps in the creation of the theoretical framework itself was to assign weights to each category and metric. Various objective and subjective approaches were considered when selecting the weight for each metric. The two candidate approaches were PCA (Principal Component Analysis) and the ad-hoc weighting scheme explained in "The Inclusive Internet Index 2020 Methodology Report" (The Economist Intelligence Unit, 2021). The aforementioned "Inclusive Internet" aggregate index assigns weights based on the Internet Life Cycle, which is defined as a ranking of the most important aspects to Internet development in a region. Their ranking followed a 40-30-20-10 rule, with the categories being Internet Availability, Internet Affordability, Internet Accessibility and Internet Readiness (The Economist Intelligence Unit, 2021).

Finally, to facilitate the comparison of scores between countries, the team decided that the best way to visualize the Internet resiliency score was through the creation of a dashboard. Various frameworks and packages were considered such as RShiny, D3, Python Dash and Highcharts, while factors such as setup requirements, data loading and code integration, handling, interactivity, and customization helped determine which visualization tool was the final selection.

4 Results

4.1 Metric Selection and Aggregation

After extensive consideration of various weighting methods, the ad-hoc weighting approach was selected in favor of PCA. This method was preferred because it groups parameters with no significant association relative to their correlation coefficients. PCA would have abstracted away key indicators in our rather limited feature set of the 17 final indicators due to the high correlation factor of certain metrics.

After selection of the weighing schematic and thorough consideration of each metric, category and necessary weights, the metrics were grouped and assigned various levels of weights to be used to calculate the Internet Resiliency score. Table 2 shows the final metrics that were selected, the category they were assigned to, and the weights per category and individual metrics. The four final categories that were selected were Quality of Service, Security, Infrastructure and Affordability. Though these categories were not the categories that were included in The Inclusive Internet Index 2020 Methodology Report, the idea of the availability of Internet services being more important than its affordability (after all, the question of affordability can not be even considered if services are

$$Y = w_{c_1} * (w_{m_1 1} * m_1 2 + \dots) + w_{c_2} * (w_{m_2 1} * m_2 1 + \dots) + ...$$

Figure 2: Formula derived to calculate Internet Resiliency Scores. The variable c represents category, while m represents the individual metric.

Category	Metric	Metric Weight
Quality of Service (25%)	Throughput - Download Speed Fluctuations	8.35%
	Throughput - Download Speed	8.35%
	Throughput - Upload Speed	8.35%
	Latency Fluctuations	16.7%
	Latency	16.7%
	IPv6 capability	33.3%
Security (25%)	MANRS score (Routing regulations)	25%
	AS hegemony	25.0%
	DDos Potential	25.0%
	Spam Infection	25.0%
	IXP efficiency	25.0%
	Upstream	25.0%
Infrastructure (35%)	Cable landing stations	12.5%
	reach	25.0%
	degree distribution	12.5%
Affordability (15%)	Affordability	100%

Table 2: Final weights and metrics selected to be used for calculating the Internet Resiliency Score.

not available) was used to assign weights to each category.

Assigning weights to the individual metrics in each category was more difficult, but it followed a similar process to that of selecting the metrics to be included in the Internet Resiliency Index. These metrics were ranked among others in their category based on data availability and coverage, as well as their importance to describing the encompassing category.

Upon assignment and validation of weights, the aggregated metric is mapped to numerical scores with the formula indicated in Figure 2. This was calculated for each African country and would become its Internet resiliency score. Further details regarding aggregation selection can be found in Technical Appendix 4.

4.2 Dashboard

After thoroughly analyzing the benefits and drawbacks of each proposed visualization software, Python Dash was selected as the best platform to display the Internet resiliency scores. It is easy to integrate with our analyses, which were already implemented in Python. There would be minimal time used for setting up the environment, and loading the data into the dashboard. This tool

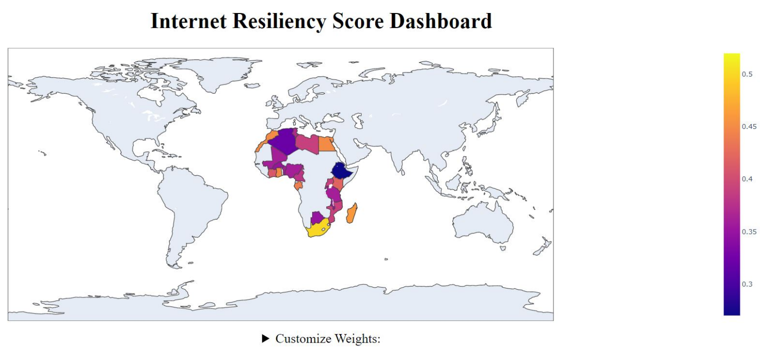

Figure 3: Final Dashboard created using Python Plotly and Dash.

also was able to produce detailed and highly interactive visualizations, which allowed users to easily compare Internet resiliency scores of African countries to each other.

The final dashboard prototype produced by this team includes various features to facilitate the comparison of Internet Resiliency scores between African countries. A choropleth map was used to display each country, and the countries are all filled in with the color corresponding to its score. The colors are ordered by the magnitude of each score (heatmap). The tooltip for each country contains the country ISO3 code (identifying feature), as well as the exact quantitative score as calculated by the Internet Resiliency Index. The default score that is displayed by the dashboard is the Internet Resiliency score calculated using the weights that the research team had selected. This dashboard also was created with the opportunity for users to be able to change the category and metric weights and facilitate the comparison of new calculated scores based on the weights they selected. A capture of the dashboard can be viewed in Figure 3, while further technical details regarding the creation of the dashboard are discussed in Technical Appendix 5.

5 Discussion

There are various important preliminary conclusions and implications that can be generated from the creation of the Internet Resiliency Index, the dispersion of Internet Resiliency scores across the continent and the dashboard itself.

The creation of the dashboard allows for any user, whether it be a visitor that has no cybersecurity knowledge to the AFRINIC website or a subject matter expert to have the opportunity to see the effect of different scoring methodologies in determining the Internet Resiliency score of African countries. By utilizing a heatmap choropleth map to compare scores, users are able to easily compare countries' Internet resiliencies and also determine which metrics contribute most to the resiliency scores. This way, using the dashboard can help determine which measures are most crucial to be improved upon to increase a country's resiliency score. Additionally, on the software development side, the ability of users to select custom weights additionally will save time for selecting weights per categories and metrics. Developers can use this tool to easily test new weights before updating these weights in the Internet Resiliency Index.

The utilization of a choropleth map reveals that most countries that have an available Internet resiliency score (from the data that was gathered) are those on the coast. Given this sample, it is evident that the country with the largest resilience score is South Africa, while other highly developed countries such as Morocco and Egypt also have high scores. Countries such as Ethiopia and Algeria tend to have lower scores. Given this, it may be assumed that recommendations may be needed for these countries to improve their Internet Resiliency. Yet, given the limited amount of data used, it would not be recommended to act on recommendations given the work that has been completed at this stage.

As mentioned above, one drawback of this tool is that there are about 50% of total African countries that can be displayed on this dashboard, limited conclusions can be made regarding the effects of the dashboard and the Internet Resiliency scores across Africa. Thus, there are various future steps that this research group would like to consider to improve upon the Internet Resiliency Index and dashboard created.

The most important step is to procure more data for the currently missing measurements. Since Internet resiliency scores were only able to be calculated for approximately 50% of African countries, more data needs to be collected to allow for a more accurate comparison across the African continent. This may be done either through further research to find available open source data or facilitating discussions between private Internet measurements companies to obtain data, or at worst utilizing AFRINIC's Internet probes to collect the missing measurements themselves. Based on the visualizations produced, it is obvious that shoreline countries were mostly likely to have had all metrics' worth of data available. Thus, there should be an emphasis on collecting data from landlocked countries in the future.

Another step would be the implementation of a database for recalibration. Creation of a database, preferably in Apache Superset, would streamline the process of updating model parameters and data.

Additionally, the consideration of other metrics, especially those that may not be directly tied to Internet resiliency (such as literacy rate) may provide more accurate and detailed insight into countries' Internet resiliency and allow for improvement in recommendations for improvement.

Whether additional metrics are incorporated into the index or not, another consideration would be to incorporate a tool that would facilitate the process of metric weight validation. The team was introduced to a prototype tool known as the "Subject Harnessing Tool", which creates an environment that allows users (in this situation, Internet measurement subject matter experts) to compare the importance of metrics to each other. The insights collected from this tool would be used to validate the magnitude of weights selected based on subject matter experts' opinions.

Finally, to increase the impact and usability of the Internet Resiliency Index, this dashboard should be deployed to the AFRINIC website and also be expanded to facilitate comparisons between countries outside of Africa. This would allow more experienced scientists to be included in the collaborative effort of comparing Internet resiliency, and allow the possibility for other countries to be helped in improving their resiliency.

Regardless of the limited amount of Internet measurement data available to developers and the work that needs to be done to improve the Internet Resiliency Index, it is clear the methodology and code developed for this project is going to be incredibly useful for prescribing changes to lower performing countries and help them develop more resilient Internet services. The ability to accurately pinpoint the changes needed to be made will cause less Internet outages and allow for users to become much more easily and strongly connected to each other in the countries of impact, especially during these times when Internet connection is of utmost importance. Everyone deserves to have reliable and consistent Internet services, and this tool is one step forward in the direction that ensures that everyone has this commodity.

6 References

AFRINIC (2021), *Measuring Internet Resilience in Africa (MIRA)*. Africa.

AFRINIC, *State of Internet Measurement in Africa - A Survey*. Africa.

AFRINIC (2021), *AFRINIC ASN Statistics*. URL https://stats.afrinic.net/asn.

Amazon Web Services (2021), *Speedtest by Ookla Global Fixed and Mobile Network Performance Maps*. URL https://registry.opendata.aws/speedtest-global-performance/

APNIC Labs (2020), *IPv6 Resource Allocations*. URL https://labs.apnic.net/dists/v6cc.html.

A4AI (2020), *Affordability Drivers Index*. URL https://a4ai.org/affordability-report/data/?_year=2020indicator=INDEX.

CAIDA (2017), *CAIDA Data Server: Index of /datasets/as-relationships*. URL https://publicdata.caida.org/datasets/as-relationships/.

CAIDA (2017), *CAIDA Data Server: Index of /datasets/peeringdb.* URL https://publicdata.caida.org/datasets/peeringdb/.

Composite Blocking List (2021), *CBL Breakdown by Country Infections, Highest by Count.* URL https://www.abuseat.org/public/countryinfections.html.

CyberGreen (2019), *Level of Risk Posed to Others.* URL https://stats.cybergreen.net/.

Deloitte (2020), *Connectivity Resilience Amidst COVID-19.* Africa.

The Economist Intelligence Unit (2020), *The Inclusive Internet Index 2020 Methodology Report*

GitHub (2021), *Submarine Cable Map.* URL https://github.com/telegeography/www.submarinecablemap.com.

Gotesman, Ryan (2018), *An Introduction to Geographical Data Visualization.* URL towardsdatascience.com.

Internet Health Report (2020), *Country Report.* URL https://ihr.iijlab.net/ihr/en-us/countries/?af=4last=3date=2021-05-13.

ITU (2020), *ITU Interactive Transmission Map.* URL https://www.itu.int/itu-d/tnd-map-public/.

The M-Lab (2020), *NDT Data Set 2020-01-01-2020-03-31.* URL https://measurementlab.net/tests/ndt.

NSRC (2021), *African Undersea and Terrestrial Fibre Optic Cables.* URL https://afterfibre.nsrc.org/close.

Packet Clearing House (2021), *RInternet Exchange Directory.* URL https://www.pch.net/ixp/dir.

PeeringDB (2021), *The Interconnection Database.* URL https://www.peeringdb.com/.

Ripe Network Coordination Centre (2021), *Raw Datasets.*URL https://www.ripe.net/analyse/raw-data-sets.

Ripe Network Coordination Centre (2021), *Routing Information Service (RIS).* URL https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris.

Speedchecker Ltd (2020), *Speedchecker Broadband Internet Speed Tests*. URL https://data.cdrc.ac.uk/dataset/speedchecker-broadband-internet-speed-tests.

Vyncke (2021), *IPv6 Deployment Aggregated Status (IPv6 networks)*. URL https://www.vyncke.org/ipv6status/prefixes.php.

6lab Cisco (2020), *6lab - The Place to Monitor IPv6 Adoption*. URL https://6lab.cisco.com/stats/information.php.

7 Technical Appendix

7.1 Data Procurement and Extraction

This section will provide further details regarding the procurement and extraction of the datasets used to create the Internet Resiliency.

One way that datasets were procured were through API calls. An example of this is the AS Hegemony data that was obtained below.

```
A2_Africa = AfricaISOcodes['A2']

AS_score_dict = dict()
for x in range(0,len(A2_Africa)):
    AS_score_dict[A2_Africa[x]] = x

#collect score per country and per day
#take average of scores per days and per country

month='05'
day='05'

for country in A2_Africa:

    AS_score_overall=0
    url = 'https://ihr.iijlab.net/ihr/api/hegemony/countries/?country=' + country +
    '&timebin=2021-' + month + '-' + day + 'T00:00:00Z'
    resp = requests.get(url)
    if (resp.ok):
        try:
            data = resp.json()
            for i in range(0, len(data['results'])):
                try:
                    with open("~/raw data/AS hegemony/" + country + "/" + day + "_" + month
                    + "_2021.json", 'w') as outfile:
                        json.dump(data, outfile)
                except:
```

```
                continue
            AS_score_overall += data['results'][i]['hege']*data['results'][i]['weight']
        except:
            continue
        AS_score_dict[country] = AS_score_overall
    else:
        AS_score_dict[country] = None
    print(country, AS_score_dict[country])
```

7.2 Data Processing and Cleaning

This section will provide further details regarding the processing and cleaning of the datasets used to create the Internet Resiliency. Figure 4 shows the usage of Pandas Profiling, a python module used to get descriptive statistics on a dataset. This facilitated the data cleanup process.

For instance, the following code was used to process and explore the dataset for the Percent Population within 10-KM Reach metric. The other datasets were cleaned and explored in a similar manner.

```
#read in data and select rows and columns of relevance
capacity_data_pop = pd.read_excel("ITU Broadband Capacity Indicators 2019.xlsx",skiprows=1)
capacity_data_pop = capacity_data_pop[capacity_data_pop.columns[0:9]]
capacity_data_pop.columns = capacity_data_pop.iloc[0]
capacity_data_pop = capacity_data_pop.iloc[2:195]
capacity_data_pop = capacity_data_pop[['Country name', 'IsoCode', '10-km Range']]

#change to correct column types
capacity_data_pop_cleaned['10-km Range'] = capacity_data_pop_cleaned['10-km Range']. \
astype(float)
#merge with ISO3 Codes data
capacity_data_pop_cleaned = pd.merge(capacity_data_pop, AfricaISOcodes, \
left_on ='IsoCode', right_on='A3', how='inner')

#select columns of interest
capacity_data_pop_cleaned = capacity_data_pop_cleaned.drop(['Country', 'A2', 'IsoCode', \
'Region'], axis=1)

#standardize metric score
capacity_data_pop_cleaned["standardized 10-km Range"]  = (capacity_data_pop_cleaned \
['10-km Range']-capacity_data_pop_cleaned['10-km Range'].min())/ \
(capacity_data_pop_cleaned['10-km Range'].max()- \
capacity_data_pop_cleaned['10-km Range'].min())

#Save cleaned dataset
capacity_data_pop_cleaned.to_pickle("~/pickle_files/10 KM Reach.pkl")
```

```
#summarize data
capacity_data_pop_cleaned.describe()

#Check countries where reach = 0
capacity_data_pop_cleaned.loc[capacity_data_pop_cleaned["10-km Range"] == 0]

#create pandas profiling report
profile = ProfileReport(capacity_data_pop, title='capacity_data_pop Profiling Report',\
explorative=True)
profile.to_file("~/Documentation/Data Exploration Reports/capacity_data_pop.html")

#choropleth map of non-standardized score
data = dict (
    type = 'choropleth',
    locations = capacity_data_pop_cleaned["A3"],
    locationmode='ISO-3',
    z=capacity_data_pop_cleaned["10-km Range"])
map = go.Figure(data=[data])
py.offline.plot(map)

#choropleth map of standardized score
data = dict (
    type = 'choropleth',
    locations = capacity_data_pop_cleaned["A3"],
    locationmode='ISO-3',
    z=capacity_data_pop_cleaned["standardized 10-km Range"])
map = go.Figure(data=[data])
py.offline.plot(map)
```

7.3 Metric Selection

The composite Resilience index is a means to assess the overall capacity of the various resilience components; ISP Resilience, Critical Infrastructure Resilience, and Market Resilience, within a country of the African continent (AFRINIC, 2021). The metrics applied in the index construction are a derivation of the aforementioned taxonomy of resilience. Factors that influenced this selection include data reliability, data acquisition time span, and a degree of equivalence to the sub-indicators of the Internet Life Cycle categories.

Table 3 displays the various metrics that were selected.

7.4 Metric Aggregation

The aggregation methods that were considered to construct the composite Resilience index apply formulas based on two mathematical functions; geometric

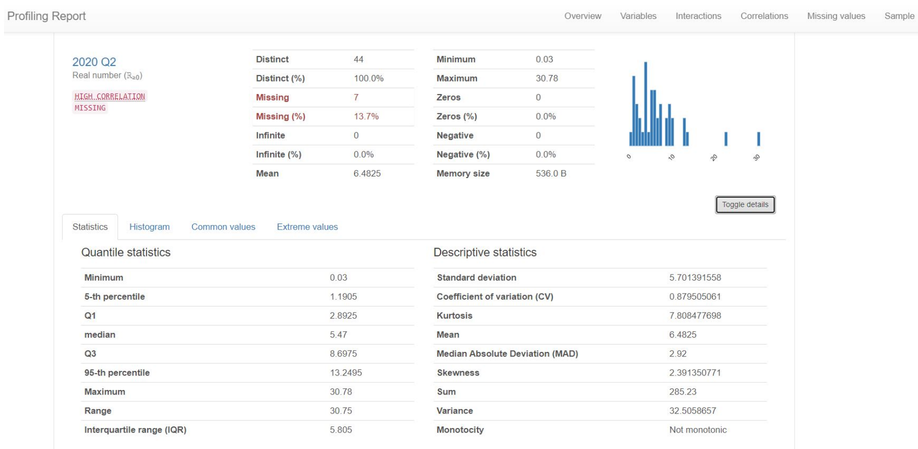

Figure 4: Pandas Profiling report, which was used to do the initial data exploration.

Category	**Metric**	**Metric Weight**
QoS - 25%	Throughput - Download Speed Fluctuations	8.35%
	Throughput - Download Speed	8.35%
	Throughput - Upload Speed Fluctuations	8.35%
	Throughput - Upload Speed	8.35%
	Latency Fluctuations	16.7%
	Latency	16.7%
	IPv6 capability	33.3%
Security - 25%	MANRS score (Routing regulations)	25.0%
Infrastructure - 35%	IXP efficiency	25.0%
	Cable landing stations	12.5%
	reach	25.0%
	degree distribution	12.5%
Affordability - 15%	Affordability	100%

Table 3: Final weights and metrics selected to be used for calculating the Internet Resiliency Score.

and arithmetic aggregation.

The arithmetic mean formula:
$Y = w_{c_1} * (w_{m_1 1} * m_1 2 + \ldots) + w_{c_2} * (w_{m_2 1} * m_2 1 + \ldots) + ...$

The geometric mean formula:
$Y = \sqrt{X_1 * X_2 * X_3, \ldots\ldots\ldots\ldots, X_n}$

Given that our initial analysis includes metrics of different units, such as area, internet bandwidth(speed), and percentages. The geometric mean would be a primary method on the condition that our dataset is exempt of any missing values replaced with zero imputation. Due to an inability to obtain data for all the metrics of each country, some countries have missing values for certain sub-indicators. Considering that at least 25 countries lack values(Not a Number) for any 5 of the 17 metrics, this is represented in the country's sequence of metric values as sudden drops to zero in place of the missing entry. This multiplicative function strongly penalizes "volatility".

Consequently, we opt to apply the arithmetic(linear) mean formula instead. For those countries with missing values for certain indicators, the composite index aggregation is not calculated as it would be an inaccurate measurement of the country's score on the Resilience Index.

7.5 Dashboard Creation

This section will provide further technical details regarding the creation of the final dashboard used to display the Internet Resiliency scores per African country.

The code for creating the dashboard is documented below.

```
#Import metrics datasets
merged = pd.read_excel("Cleaned data/composite dataset.xlsx", \
sheet_name="Composite Metrics Dataset")

#import ISO country code dataset
countries_data = pd.read_excel("raw data/ISOafrinic.xlsx")

#merge to get ISO codes
merged = pd.merge(merged, countries_data[["Country", "A3"]])

#rename columns
merged = merged.rename(columns={
    'normalized IPv6Deployment': 'normalized ipv6 counts',
    'MANRS score': 'normalized manrs score',
    'normalized spamInfections': 'normalized spam',
    'Normalized Risk': 'normalized risk',
```

```
    'latency_normalized': 'normalized latency',
    'standardized affordability': 'normalized affordability',
    'standardized AS score': 'normalized AS score',
    'DLkbps_normalized': 'normalized download',
    'UPkbps_normalized': 'normalized upload',
    'providers_scaled': 'normalized upstream',
    'kmReach_normalized': 'normalized 10-km Range',
    'standardized links per node': 'normalized links per node',
    'standardized landing stations': 'normalized landing stations',
    'downloads_scaled': 'normalized download fluctuations',
    'uploads_scaled': 'normalized upload fluctuations',
    'latency_scaled': 'normalized latency fluctuations'
})

#replace values that can be replaced
merged['normalized landing stations'] = merged['normalized landing stations'].fillna(0)
merged['normalized links per node'] = merged['normalized links per node'].fillna(0)
merged['normalized efficiency'] = merged['normalized efficiency'].fillna(0)

#define range values for sliders
range_values = [0,10,20,30,40,50,60,70,80,90,100]


#front end of application

app = dash.Dash(__name__)


app.layout = html.Div([

    #title
    html.H1(children='Internet Resiliency Score Dashboard',
           style={
            'textAlign': 'center',
            'fontSize': '40px'}),

    #space between title and score
    html.P(children='',
             style={'textAlign': 'center',
                    'fontSize': '16px'}),

    #choropleth map
    dcc.Graph(id="choropleth",
            figure={
        "data": [
            {"x": [1, 2, 3], "y": [4, 1, 2], "type": "bar"},
```

```
            {"x": [1, 2, 3], "y": [2, 4, 5], "type": "bar"},
        ],
        "layout": {
            "title": "My Dash Graph",
            "height": 450,  # px
        },
    },
),

    #break
    html.Br(),

    #define weights
    html.Details([
        html.Summary("Customize Weights:",
           style={
            'textAlign': 'center',
            'fontSize': '24px',}
                    ),

    #selection of weights
    #QoS category
    html.P('Quality of Service Category Score:',
          style={'fontSize': '20px'}),
    dcc.Slider(
        id='qos_weight',
        min=0,
        max=100,
        value=25,
        marks={str(i): str(i) for i in range_values},
        step=1,
    ),
    #throughput-download
    html.P('Throughput (Download Speed):',
          style={'fontSize': '20px'}),
    dcc.Slider(
        id='throughputd_weight',
        min=0,
        max=100,
        value=8.35,
        marks={str(i): str(i) for i in range_values},
        step=1
    ),
    #throughput-download fluctuations
    html.P('Throughput (Download Speed) Fluctuations:',
          style={'fontSize': '20px'}),
```

```
dcc.Slider(
    id='throughputdf_weight',
    min=0,
    max=100,
    value=8.35,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#throughput-upload
html.P('Throughput (Upload Speed):',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='throughputu_weight',
    min=0,
    max=100,
    value=8.35,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#throughput-upload fluctuations
html.P('Throughput (Upload Speed) Fluctuations:',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='throughputuf_weight',
    min=0,
    max=100,
    value=8.35,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#latency metric
html.P('Latency:',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='latency_weight',
    min=0,
    max=100,
    value=16.7,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#latency fluctuations metric
html.P('Latency Fluctuations:',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='latencyf_weight',
```

```
    min=0,
    max=100,
    value=16.7,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#iPv6 metric
html.P('IPv6:',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='ipv6_weight',
    min=0,
    max=100,
    value=33,
    marks={str(i): str(i) for i in range_values},
    step=1
),

#line break
html.Br(),

#security category
html.P('Security Category Score:',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='security_weight',
    min=0,
    max=100,
    value=25,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#MANRS Score (Routing Regulations)
html.P('MANRS Score (Routing Regulations):',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='manrs_weight',
    min=0,
    max=100,
    value=25,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#AS hegemony metric
html.P('AS Hegemony:',
      style={'fontSize': '20px'}),
```

```
dcc.Slider(
    id='as_weight',
    min=0,
    max=100,
    value=25,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#ddos metric
html.P('DDoS Potential:',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='ddos_weight',
    min=0,
    max=100,
    value=25,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#spam
html.P('Spam:',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='spam_weight',
    min=0,
    max=100,
    value=25,
    marks={str(i): str(i) for i in range_values},
    step=1
),

#line break
html.Br(),

#infrastructure category
html.P('Infrastructure Category Score:',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='infrastructure_weight',
    min=0,
    max=100,
    value=35,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#IXP efficiency metric
```

```
html.P('IXP Efficiency:',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='ixp_weight',
    min=0,
    max=100,
    value=25,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#reach metric
html.P('% Population Reached within 10 KM Range:',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='reach_weight',
    min=0,
    max=100,
    value=25,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#cable landing stations metric
html.P('Number of Cable Landing Stations:',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='cable_landing_weight',
    min=0,
    max=100,
    value=12.5,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#links per node metric
html.P('Links per Node:',
      style={'fontSize': '20px'}),
dcc.Slider(
    id='links_weight',
    min=0,
    max=100,
    value=12.5,
    marks={str(i): str(i) for i in range_values},
    step=1
),
#upstream
html.P('Number of upstream:',
      style={'fontSize': '20px'}),
```

```
    dcc.Slider(
        id='upstream_weight',
        min=0,
        max=100,
        value=25,
        marks={str(i): str(i) for i in range_values},
        step=1
    ),

    #line break
    html.Br(),

    #affordability category weight
    html.P('Affordability Category Score:',
          style={'fontSize': '20px'}),
    dcc.Slider(
        id='affordability_weight',
        min=0,
        max=100,
        value=15,
        marks={str(i): str(i) for i in range_values},
        step=1
    ),
    ])
])

#define inputs and outputs
@app.callback(
    Output("choropleth", "figure"),
    [Input("qos_weight", "value"),
    Input("security_weight", "value"),
    Input("infrastructure_weight", "value"),
    Input("affordability_weight", "value"),
    Input("reach_weight", "value"),
    Input("as_weight", "value"),
    Input("cable_landing_weight", "value"),
    Input("ddos_weight", "value"),
    Input("links_weight", "value"),
    Input("ipv6_weight", "value"),
    Input("ixp_weight", "value"),
    Input("latency_weight", "value"),
    Input("latencyf_weight", "value"),
    Input("manrs_weight", "value"),
    Input("upstream_weight", "value"),
    Input("spam_weight", "value"),
    Input("throughputd_weight", "value"),
```

```
    Input("throughputu_weight", "value"),
    Input("throughputdf_weight", "value"),
    Input("throughputuf_weight", "value")
    ])

#define output - will be changing z to Resiliency score once have all metrics
def update_choropleth(qos_weight, security_weight, infrastructure_weight, reach_weight,
as_weight, links_weight, affordability_weight, cable_landing_weight, ddos_weight,
ipv6_weight, ixp_weight, latency_weight, manrs_weight, upstream_weight, spam_weight,
throughputd_weight, throughputu_weight, latencyf_weight,
throughputdf_weight, throughputuf_weight):

    qos_score = int(qos_weight)/100*(int(throughputd_weight)/100* \
        merged["normalized download"] + int(throughputdf_weight)/100* \
        merged["normalized download fluctuations"] + int(throughputu_weight)/100* \
        merged["normalized upload"] + int(throughputuf_weight)/100* \
        merged["normalized upload fluctuations"] + int(latency_weight)/100* \
        merged["normalized latency"] + int(latencyf_weight)/100* \
        merged["normalized latency fluctuations"] + int(ipv6_weight)/100* \
        merged["normalized ipv6 counts"])

    security_score = int(security_weight)/100*(int(manrs_weight)/100* \
        merged["normalized manrs score"] + int(as_weight)/100* \
        merged["normalized AS score"] + int(ddos_weight)/100* \
        merged["normalized risk"] + int(spam_weight)/100*merged["normalized spam"])

    infrastructure_score = int(infrastructure_weight)/100*(int(upstream_weight)/100* \
        merged["normalized upstream"] + int(ixp_weight)/100* \
        merged["normalized efficiency"] + int(cable_landing_weight)/100* \
        merged["normalized landing stations"] +
        int(reach_weight)/100* \
        merged["normalized 10-km Range"] +
        int(links_weight)/100* merged["normalized links per node"])

    affordability_score = int(affordability_weight)/100*merged["normalized affordability"]

    calculated =  round((qos_score + security_score + infrastructure_score + \
    affordability_score),2)

    data = dict (
        type = 'choropleth',
        locations = merged["A3"],
        locationmode='ISO-3',
        #hover_name=merged["Country name"],
        z=calculated)
    map = go.Figure(data=[data]).update_layout(
```

```
        autosize = False,
        margin = dict(
                l=0,
                r=0,
                b=0,
                t=0,
                pad=0,
                autoexpand=True
            ),
            width=1650,
            height=500,
    )

    return map

app.run_server(debug=False)
```