Group-9A
DCPP Group Assignment Report
Domain - Birds

Shephali Bhardwaj – 12110010

Varshini Modulla – 12110052

Niharika Miriyala - 12110013

Hem Kumar Reddy Maryada – 12110072

EXECUTIVE SUMMARY

Problem Statement:

- Create an end-to-end data collection and pre-processing pipeline for the domain Birds
- The resulted dataset should be an information-rich and reliable dataset with ample attributes

Proposed solution:

- Search for meaningful data sources either structured or unstructured
- Extract data by scraping the web using packages like Wikipedia and BeautifulSoup
- Integrate the data collected through multiple sources and clean the dataset

Brief understanding of the challenges:

- The seed data sources on the web either had incredible number of rows with very less attributes or good number of attributes with very few rows.
- Most of the websites had neither adequate attributes nor a good collection of birds
- Majority of the websites were based on bird-watching. So, they did have many physical bird attributes like dimensions of the wings, weight of the bird

THOUGHT PROCESS FOR CHOOSING BIRDS

A Generic Domain

Lots of excel/csv structured data sources available on the web

Wikipedia and many other sites have reliable amount of data

Can scrape, crawl, integrate and pre-process the data

Results an information-rich and meaningful dataset

STRUCTURED AND UNSTRUCTURED SOURCES

- We searched through plentiful seed sources and websites

POTENTIAL SOURCES	TYPE OF SOURCE	CHALLENGES
http://datazone.birdlife.org/species/taxonomy	Structured	Excel file with ample rows and attributes but the attributes made no sense
https://figshare.com/articles/dataset/Data_from_The_Global_Avian_Invasions_Atlas_-_A_database_of_alien_bird_distributions_worldwide/4234850	Structured	Excel file with ample rows and attributes but the attributes made no sense
https://a-z-animals.com/animals/birds/	Unstructured	Website with 25 attributes but only had data about 124 birds
https://www.worldbirdnames.org/new/ioc-lists/master-list-2/	Unstructured	Website data with sufficient rows and attributes but the attributes made no sense
https://www.kaggle.com/shreyasajal/let-s-infer-statistically-state-of-indian-birds/data?select=State+of+Indias+Birds+-+Essentials.xlsx	Structured	Excel file has 19 attributes but only 800 rows
https://www.kaggle.com/gpiosenka/100-bird-species	Unstructured	Dataset with only images
https://www.birdlist.org/nam/north_america.htm	Structured	Large Data with only 2 attributes

- Also, none of the sources had similar attributes. So, integrating the above sources didn’t result in meaningful data

DATA SCRAPING

Unstructured Source

- We came across a Wikipedia article – 'https://en.wikipedia.org/wiki/List_of_birds_by_common_name' that has all birds listed out alphabetically

Step I - Extracting the links

- From the above link, we extracted the links of all the birds

Step II -Scraping the infobox

- Every bird in the link has its own wiki page. On the right of the page, we have an infobox with the key features of each bird. For example, this is the infobox of Abbott's booby

Step III - Crawling the Wikipedia

- Later, we crawled the web with the bird name and scraped the Country and Habitat of the bird

STEP I – LINK EXTRACTION

We used ‘Beautiful soup’ to scrape all the links present in the Wiki page

On the left is the code snippet of `Link_Extraction.ipynb`. The output of the code is the list of all the links. This list of links are written to an output file for easy access

And on the right is the screenshot of the output

```
import requests
import bs4
import wikipedia
```

```
URL = 'https://en.wikipedia.org/wiki/List_of_birds_by_common_name'
```

```
# Fetch all the HTML source from the url
response = requests.get(URL)
```

```
soup = bs4.BeautifulSoup(response.text, 'html.parser')
links = soup.select('a')
```

```
list_links = []
for link in links:
  if link.get('href') != None:
    if 'https://' in link.get('href'):
      list_links.append(link.get('href'))
    else:
      list_links.append('https://en.wikipedia.org' + link.get('href'))
```

```
list_links
```

```
'https://en.wikipedia.org/wiki/Abbott%27s_babbler',
'https://en.wikipedia.org/wiki/Abbott%27s_booby',
'https://en.wikipedia.org/wiki/Abbott%27s_starling',
'https://en.wikipedia.org/wiki/Abbott%27s_sunbird',
'https://en.wikipedia.org/wiki/Abd_al-Kuri_sparrow',
'https://en.wikipedia.org/wiki/Abdim%27s_stork',
'https://en.wikipedia.org/wiki/Aberdare_cisticola',
'https://en.wikipedia.org/wiki/Aberrant_bush_warbler',
'https://en.wikipedia.org/wiki/Abert%27s_towhee',
'https://en.wikipedia.org/wiki/Abyssinian_catbird',
'https://en.wikipedia.org/wiki/Abyssinian_crimsonwing',
'https://en.wikipedia.org/wiki/Abyssinian_ground_hornbill',
'https://en.wikipedia.org/wiki/Abyssinian_ground_thrush',
'https://en.wikipedia.org/wiki/Abyssinian_longclaw',
'https://en.wikipedia.org/wiki/Abyssinian_owl',
'https://en.wikipedia.org/wiki/Abyssinian_roller',
'https://en.wikipedia.org/wiki/Abyssinian_scimitarbill',
'https://en.wikipedia.org/wiki/Abyssinian_slaty_flycatcher',
'https://en.wikipedia.org/wiki/Abyssinian_thrush',
```

STEP II – INFOBOX SCRAPING

- We used ‘Wikipedia’ package to retrieve the infobox on each of the link extracted before
- The following attributes are extracted from the infobox and added to a dataframe:
  1. Name
  2. Conservation Status
  3. Kingdom
  4. Phylum
  5. Class
  6. Order
  7. Family
  8. Genus
  9. Species
  10. Binomial Name
  11. Synonyms
- A get function is written to retrieve each of the above attribute. An example in on the right
- The excel snapshot shows the output of this file

```
import pandas as pd
import wikipedia
import numpy as np
```

```
for i in pages:
    infoboxes = pd.read_html(i, attrs={"class":"infobox biota"})
    if len(infoboxes)==1:
        extract_data(infoboxes[0])
print(birds)
```

```
def get_conservation_status(df1):
    for i in range(0,len(list(df1.index))):
        if list(df1.index)[i] == 'Conservation status':
            return (list(df1.index)[i+1])
    if 'Conservation status' not in list(df1.index):
        return np.NaN
```

```
def get_kingdom(df1):
    if 'Kingdom:' in list(df1.index):
        return (df1.loc['Kingdom:'][0])
    else:
        return np.NaN
```

Name	Conservation status	Kingdom	Phylum	Class	Order	Family	Genus	Species	Binomial Name	Synonyms
Iago sparrow	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Passeridae	Passer	P. iagoensis	Passer iagoensis(Gould, 1837)	Malimbus ibadanensisElgood, 1958

STEP III - DATA CRAWLING

- After extracting the common name of the bird from the infobox, all the Wikipedia pages are crawled for any information regarding that bird
- And we were able to scrape the following attributes through crawling

1. Description
2. Categories
3. References
4. Wiki URL
5. Image

- We also scraped the below two attributes by using pre-defined lists

1. Country – pycountry package has a list of all the countries in this world. If any of those countries are mentioned in the description of the birds, then it is scraped
2. Habitat – we custom created a list of all the bird habitats. If any of those habitats are mentioned in the description of the birds, then it is scraped

```
def get_summary(name):
    return(wikipedia.summary(name, sentences=2))
```

```
def get_categories(name):
    categories = wikipedia.page(name)
    return(categories.categories[0:3])
```

```
def get_references(name):
    refs = wikipedia.page(name)
    return(refs.references[0:3])
```

```
def get_url(name):
    url = wikipedia.page(name)
    return(url.url)
```

```
def get_image(name):
    image = wikipedia.page(name)
    return(image.images[0])
```

```
def findCountry(data):
    countries = sorted([country.name for country in pycountry.countries] , key=lambda x: -len(x))
    for country in countries:
        if country.lower() in data.lower():
            return country
    return None
```

```
#habitat
habitats_list = ['forest', 'forests', 'woods', 'woods', 'woodland', 'woodlands', 'bog', 'bogs', 'f
```

```
def findHabitat(data):
    for habitat in habitats_list:
        if habitat in data.lower():
            return habitat
    return None
```

Below is the snapshot of output

Summary	Categories	References	Wiki URL	Image	Country	Habitat
The Iago sparrow (Passer iagoensis), also known as the Cape Verde or rufous-backed sparrow, is a passerine bird of the sparrow family Passeridae. It is endemic to the Cape Verde archipelago, in the eastern Atlantic Ocean near western Africa.	['ARKive links', "Articles with 'species' microformats", 'Articles with short description']	['http://www.avescanarias.com/pdfs/CO%20Garcia-del-Rey%20POSTER%20(PASIAG).pdf', 'http://ibc.lynxeds.com/species/cape-verde-sparrow-passer-iagoensis', 'http://malimbus.free.fr/articles/V26/26034037.pdf']	https://en.wikipedia.org/wiki/Iago_sparrow	https://upload.wikimedia.org/wikipedia/commons/c/c4/PasserIagoensis_cropped.jpg	Nigeria	forest

STRUCTURING DATA

- In total, we accumulated 18 attributes through crawling and scraping and we collected the data of **6867** birds.

- All the data we scraped were added to a dataframe and in the end, the dataframe is written to an excel file and the final output looks like this

```
#Exporting data to Excel
#Give File name here
file_name = 'Birds_Dataset.xlsx'

# writing to excel
birds.to_excel(file_name)
print('DataFrame is written to Excel File successfully.')
```

```
DataFrame is written to Excel File successfully.
```

S.No	Name	Conservation status	Kingdom	Phylum	Class	Order	Family	Genus	Species	Binomial Name	Synonyms	Summa
0	Caatinga antwren	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Thamnophilidae	Herpsilochmus	H. sellowi	Herpsilochmus sellowiWhitney, Pacheco, Buzzetti & Parrini, 2000		The Caatinga a
1	Chestnut wood quail	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Galliformes	Odontophoridae	Odontophorus	O. hyperythrus	Odontophorus hyperythrusGould, 1858		The chestnut
2	Cinereous tyrant	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Tyrannidae	Knipolegus	K. striaticeps	Knipolegus striaticeps(d'Orbigny & Lafresnaye, 1837)		The cinereous
3	Cone-billed tanager	Endangered (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Thraupidae	Conothraupis	C. mesoleuca	Conothraupis mesoleuca(Berlioz, 1939)		The cone-bille
4	Caatinga cacholote	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Furnariidae	Pseudoseisura	P. cristata	Pseudoseisura cristata(Spix, 1824)		The Caatinga c
5	Chestnut woodpecker	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Piciformes	Picidae	Celeus	C. elegans	Celeus elegansMuller, 1776		The chestnut
6	Cinereous vultureTemporal range: Miocene-recent [	Near Threatened (IUCN 3.1)[2]	Animalia	Chordata	Aves	Accipitriformes	Accipitridae	Aegypius	A. monachus	Aegypius monachus(Linnaeus, 1766)	Vultur monachus Linnae	The cinereous
7	('Congo bay owl', 'Conservation status')		Animalia	Chordata	Aves	Strigiformes	Tytonidae	Phodilus	P. prigoginei	Phodilus prigogineiSchouteden, 1952		The conservat
8	Caatinga parakeet	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Psittaciformes	Psittacidae	Eupsittula	E. cactorum	Eupsittula cactorum(Kuhl, 1820)		The Caatinga p
9	Cinereous warbling finch	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Thraupidae	Microspingus	M. cinereus	Microspingus cinereusBonaparte, 1851		The cinereous
10	Congo martin	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Hirundinidae	Riparia	R. congica	Riparia congica(Reichenow, 1887)		The Congo ma
11	Caatinga puffbird	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Piciformes	Bucconidae	Nystalus	N. maculatus	Nystalus maculatus(Gmelin, 1788)		Nystalus macu
12	Chestnut-backed antshrike	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Thamnophilidae	Thamnophilus	T. palliatus	Thamnophilus palliatus(Lichtenstein, 1823)		The chestnut-
13	Cinereous-breasted spinetail	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Furnariidae	Synallaxis	S. hypospodia	Synallaxis hypospodiaSclater, 1874		The cinereous
14	Congo moor chat	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Muscicapidae	Myrmecocichla	M. tholloni	Myrmecocichla tholloni(Oustalet, 1886)		The Congo mo
15	Cabanis's bunting	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Emberizidae	Emberiza	E. cabanisi	Emberiza cabanisi(Reichenow, 1875)		Cabanis's bunt
16	('Chestnut-backed buttonquail', 'Conservation status')		Animalia	Chordata	Aves	Charadriiformes	Turnicidae	Turnix	T. castanotus	Turnix castanotus(Gould, 1840)		The conservat
17	('Cinnabar boobook', 'Conservation status')		Animalia	Chordata	Aves	Strigiformes	Strigidae	Ninox	N. ios	Ninox ios(Rasmussen, 1999)		The conservat
18	Congo peafowl	Vulnerable (IUCN 3.1)[1]	Animalia	Chordata	Aves	Galliformes	Phasianidae	AfropavoChapin,	A. congensis	Afropavo congensisChapin, 1936		The Congo pea
19	Cabanis's greenbul	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Pycnonotidae	Phyllastrephus	P. cabanisi	Phyllastrephus cabanisi(Sharpe, 1881)	Criniger cabanisi Phyllas	Cabanis's gree
20	Chestnut-backed chickadee	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Paridae	Poecile	P. rufescens	Poecile rufescens(Townsend 1837)	Parus rufescens	The chestnut-
21	Cinnamon attila	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Tyrannidae	Attila	A. cinnamomeus	Attila cinnamomeus(Gmelin, 1789)		The cinnamon
22	Cabanis's ground sparrow	Near Threatened (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Passerellidae	Melozone	M. cabanisi	Melozone cabanisi(PL Sclater & Salvin, 1868)	Melozone biarcuata cab	Cabanis's grou
23	Chestnut-backed jewel-babbler	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Cinclosomatidae	Ptilorrhoa	P. castanonota	Ptilorrhoa castanonota(Salvadori, 1875)		The chestnut-
24	Cinnamon becard	Least Concern (IUCN 3.1)[1]	Animalia	Chordata	Aves	Passeriformes	Tityridae	Pachyramphus	P. cinnamomeus	Pachyramphus cinnamomeusLawrence, 1861		The cinnamon
25	('Congo sunbird', 'Conservation status')		Animalia	Chordata	Aves	Passeriformes	Nectariniidae	Cinnyris	C. congensis	Cinnyris congensisvan Oort, 1910	Nectarinia congensis	The conservat
26	("Cabanis's seedeater", 'Scientific classification')		Animalia	Chordata	Aves	Passeriformes	Cardinalidae	Amaurospiza	A. concolor	Amaurospiza concolorCabanis, 1861		In biology, tax

Sheet1

DATA CLEANING AND PRE-PROCESSING

Handling Missing Values:

- The attribute ‘Synonyms’ had more than 20% missing values. It doesn’t add much to our further EDA analysis and hence, we dropped the attribute
- Since we had 18 attributes, we set a missing value threshold of 3. Any row having more than 3 missing values will be dropped

```
df = df.drop('Synonyms',axis = 1)
```

```
df = df.dropna(thresh=3, axis = 0)
```

Data Cleaning:

- When the infobox of a bird doesn’t have an image, the name of the bird is scraped incorrectly. And the code snippet below cleans it

```
def clean_Name(df):
    sub_list = ["Conservation", "status", "Scientific", "classification"]
    for i in range(0, len(df)):
        for sub in sub_list:
            df.Name[i] = df.Name[i].replace(sub, ' ')
        df.Name[i] = df.Name[i].translate(str.maketrans('', '',string.punctuation))
        df.Name[i] = df.Name[i].strip()
        print(df.Name[i])
```

Name
("Grauer's cuckooshrike", 'Conservation status')

→

Name
Grauers cuckooshrike

DATA CLEANING AND PRE-PROCESSING cont.

- Also, we cleaned the Conservation Status to be more meaningful

```
def clean_Conservation_Status(df):
    for i in range(0, len(df)):
        df['Conservation status'][i] = df['Conservation status'][i].split('(',1)[0]
        print(df['Conservation status'][i])
```

Conservation status
Least Concern (IUCN 3.1)[1]

→

Conservation status
Least Concern

- Lastly, attributes References and Categories are extracted in the form of a list. So, they are cleaned to comma separated values

```
def clean_Categories_and_References(df):
    remove_list = '[]\"\''
    for i in range(0, len(df)):
        df.Categories[i] = df.Categories[i].translate(str.maketrans('', '', remove_list))
        df.References[i] = df.References[i].translate(str.maketrans('', '', remove_list))
        print(df.Categories[i])
        print(df.References[i])
```

References
['http://www.inbo.be/content/homepage_en.asp', 'http://www.cosepac.gc.ca/eng/sct5/index_e.cfm', 'http://www.ymparisto.fi/default.asp?node=6053&lan=en']

↓

References
http://www.inbo.be/content/homepage_en.asp, http://www.cosepac.gc.ca/eng/sct5/index_e.cfm, http://www.ymparisto.fi/default.asp?node=6053&lan=en

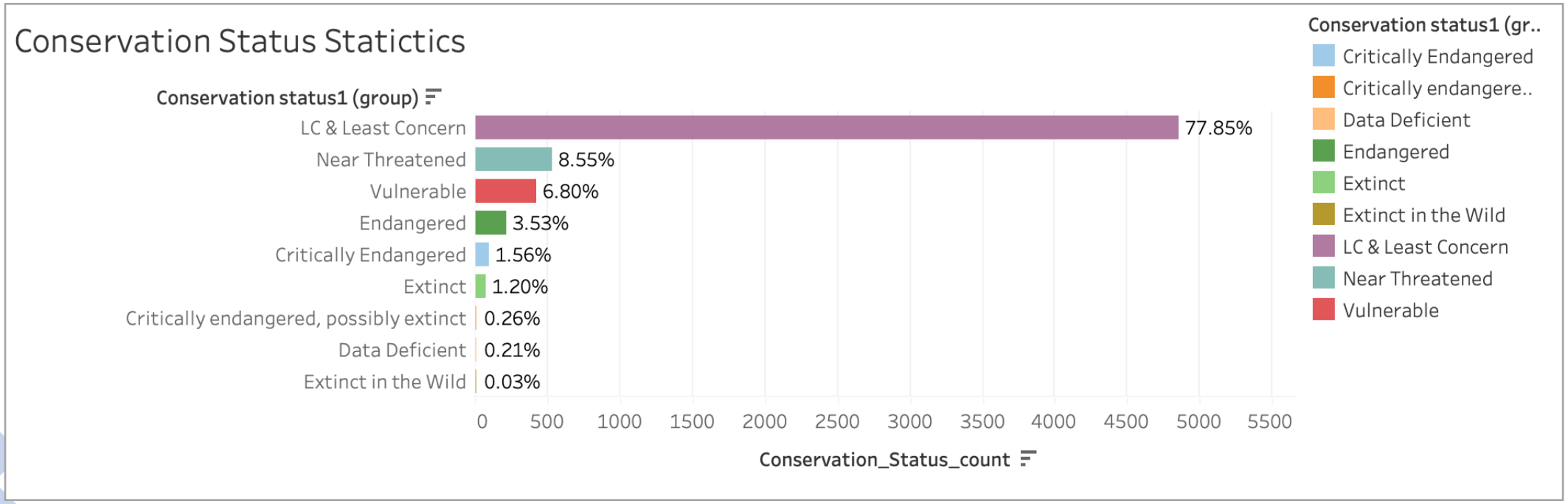
Categories
['All articles containing potentially dated statements', 'All articles with dead external links', 'Articles containing potentially dated statements from January 2008']

↓

Categories
All articles containing potentially dated statements, All articles with dead external links, Articles containing potentially dated statements from January 2008

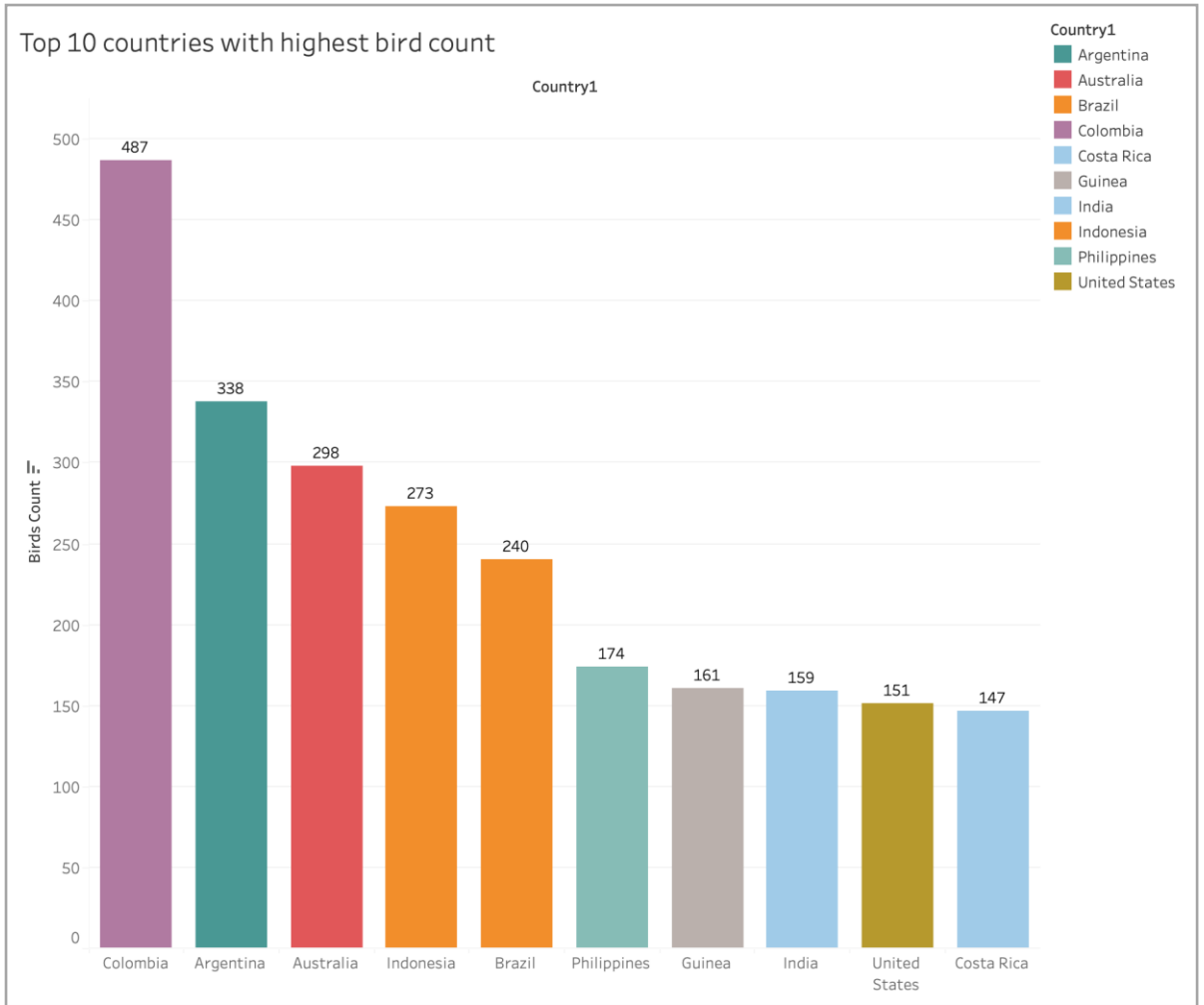
EXPLORATORY DATA ANALYSIS

We observed that there are 77.85% birds whose conservation status is of least concern and 1.20% bird species are already extinct



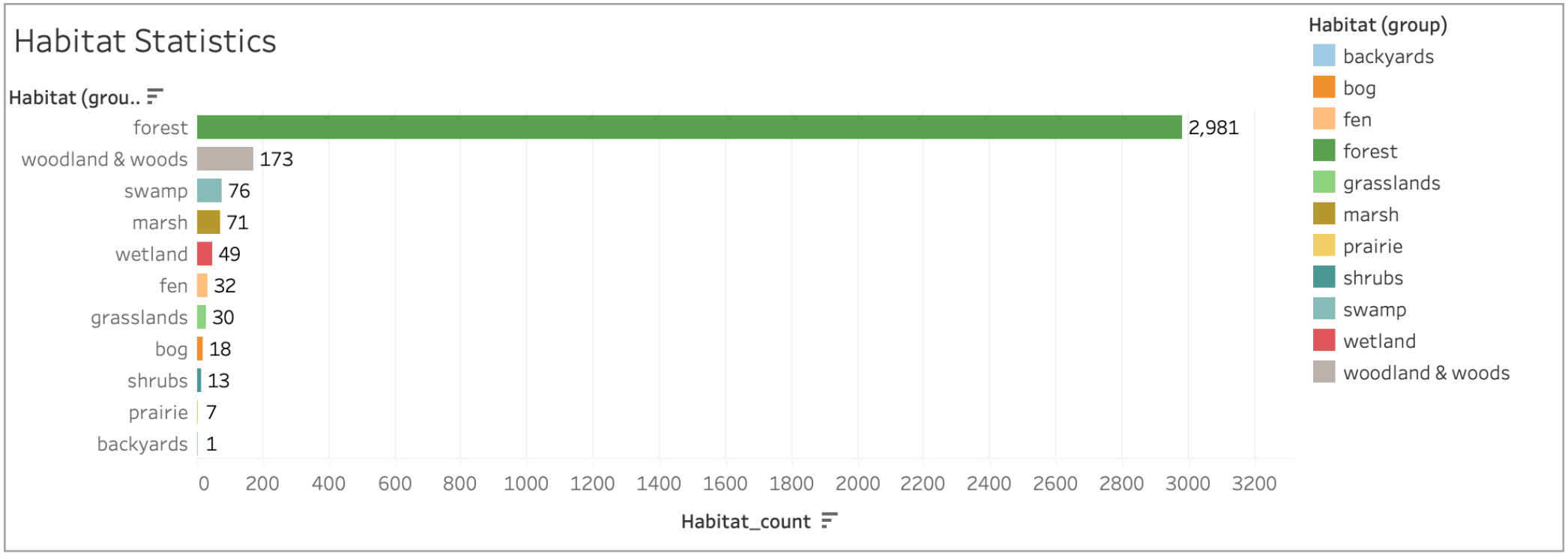
EXPLORATORY DATA ANALYSIS cont.

Columbia stands first in highest bird count chart with 487 birds and India, with 159 birds, stands eighth.



EXPLORATORY DATA ANALYSIS cont.

We observed that `Forest` habitat has the greatest number of birds followed by `woodlands` whereas `backyards` have the least.

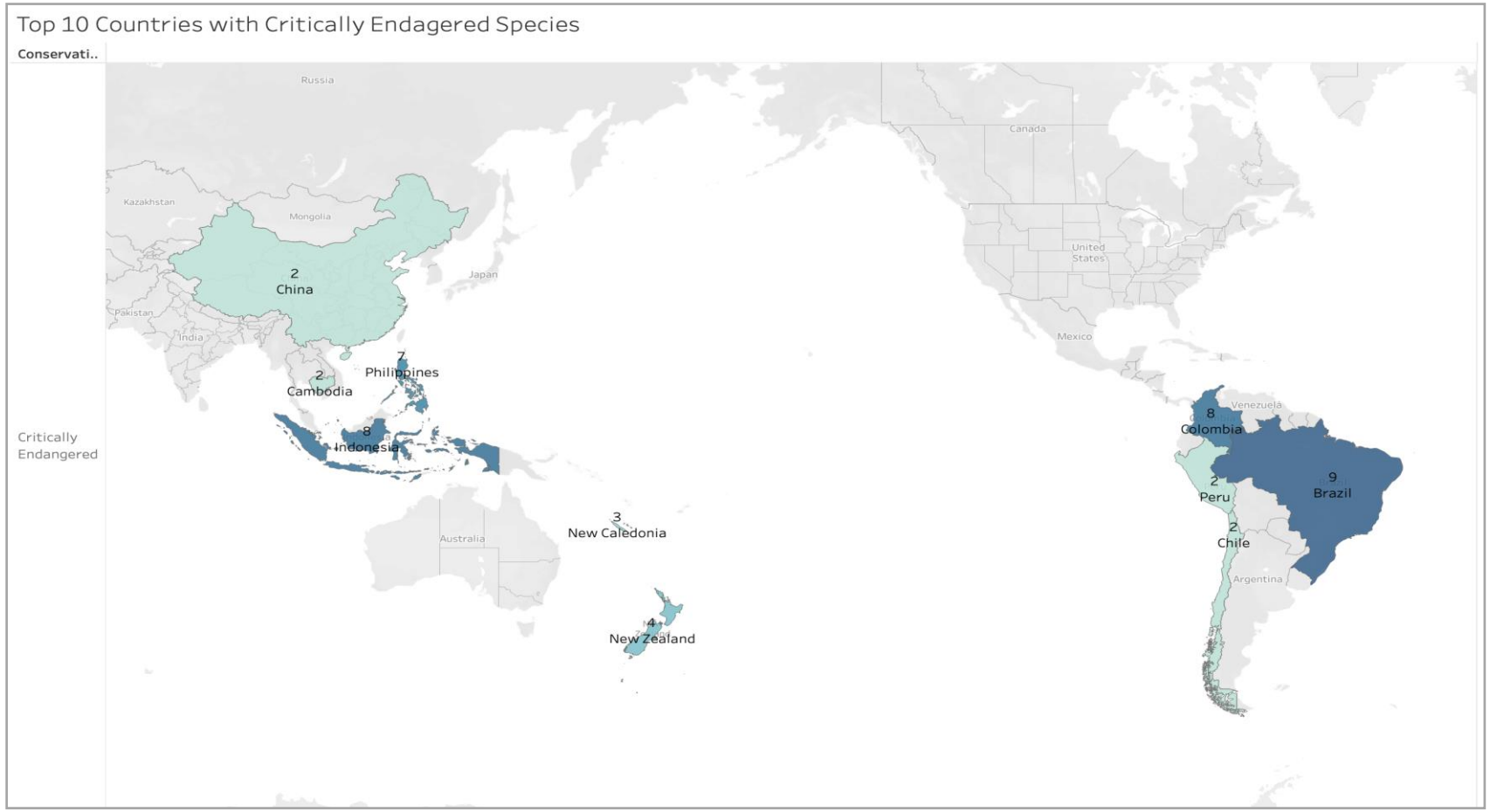


EXPLORATORY DATA ANALYSIS cont.

This chart portrays the world-wide distribution of the birds that are in our dataset. The countries with dark contrast of green has the highest bird count

EXPLORATORY DATA ANALYSIS cont.

This chart depicts the top 10 countries which have highest number of `Critically Endangered` species. Brazil stands first followed by Colombia and Indonesia



STRATEGIES TO ENHANCE THE DATASET

- As the Birds Data is a categorical type and missing data cannot be filled through some assumptions like how we do in case of numerical data, there are other methods through which data can be enhanced. One of such methods is ‘Crowd Sourcing’.
- We have few missing data in attributes – ‘Family’, ‘Species’, ‘Binomial Name’ , ‘Country’ & ‘Habitat’.
- In order to collect information regarding this missing data, our strategy is to find the information from a website – ‘The Cornell Lab of Ornithology- Birds of the World’[https://birdsoftheworld.org/bow/home]
- This webpage has data of 10,824 species of birds. With the help of Text Analytics, we can not only fill the missing values of the above attributes, but we can also scrape many new attributes:

- Demography
- Population
- Breeding, Behaviour
- Sounds & Vocal Behaviour
- Diet and Foraging
- Movements and Migration
- Conservation and Management

REFERENCES AND TECHNOLOGY STACK

References:

- https://medium.com/@Alexander_H/scraping-wikipedia-with-python-8000fc9c9e6c
- https://www.kscottz.com/web-scraping-with-beautifulsoup-and-python/
- https://www.pstanalytics.com/blog/advanced-analytics/python/fetching-text-from-infobox-of-wikipedia-in-python-for-data-science/
- https://towardsdatascience.com/step-by-step-tutorial-web-scraping-wikipedia-with-beautifulsoup-48d7f2dfa52d
- https://scraperwiki.com/2011/12/how-to-scrape-and-parse-wikipedia/
- https://www.py4u.net/discuss/251160
- https://www.dataskunkworks.com/latest-posts/wikipedia-scraping-2020

Technology Stack:

- Anaconda and Jupyter Notebook
- Python
- Wikipedia python package
- BeautifulSoup package
- Atom Text Editor
- Tableau

GITHUB

Git URL: https://github.com/modullavarshini/DCPP_Group_Assignment

List of Files and their description:

File Name	Description
Link_Extraction.ipynb	Extracts all the links from https://en.wikipedia.org/wiki/List_of_birds_by_common_name
Data_Extraction.ipynb	Scraped data from all the infoboxes and crawls all wikipages for more attributes about the birds
Country_and_Habitat_Extraction.ipynb	Crawls and scrapes Country and Habitat of the bird
Preprocessing_and_Cleaning.ipynb	Handles missing values and cleans the data
Birds_Dataset.xlsx	The output dataset
Birds_EDA.twbx	Tableau file used for data visualization

THANK YOU