eBirdCleanorg

April 5, 2024

1 Chicago Luxury Effect

1.1 Section 1. Project Introduction

This project explores the relationship between socio-economic indicators and bird diversity in Chicago communities. Utilizing a dataset containing metrics such as housing conditions, poverty rates, and per capita income alongside bird diversity measures, we aim to investigate whether affluent neighborhoods exhibit higher bird diversity.

1.2 Section 2. Data Cleaning

[24]:
```
import eda, ml, pandas as pd, numpy as np, geopandas as gpd, warnings; warnings.
↪filterwarnings("ignore")
```

1.2.1 2.1 Read and Filter eBird Dataset

[59]:
```
# Filter columns
df = pd.read_csv('data/ebd_US-IL_200801_201212_relJan-2024.txt',sep='\t')
req_cols = ['CATEGORY', 'COMMON NAME', 'SCIENTIFIC NAME', 'OBSERVATION COUNT',␣
↪'EXOTIC CODE', 'LATITUDE', 'LONGITUDE', 'OBSERVATION DATE', 'PROTOCOL TYPE',␣
↪'ALL SPECIES REPORTED']
df = df[req_cols]
```

We will keep only species level observations (removing subspecies and genus level observations). We will also filter out incomplete checklists and incidental observations to manage bias towards specific species.

[27]:
```
df = df[(df['CATEGORY']=='species') & (df['PROTOCOL TYPE']=='Traveling') |␣
↪(df['PROTOCOL TYPE']=='Stationary') & (df['ALL SPECIES REPORTED']==1)]
```

1.2.2 2.2 ebird Dataset Transformation

[28]:
```
df['NATIVE'] = df['EXOTIC CODE'].apply(lambda row:0 if row == np.nan else 1) #␣
↪Native column: 1 = is native to chicago, 0 = not native to chicago
df['COUNT'] = df['OBSERVATION COUNT'].apply(lambda row: 1 if row == 'X' else␣
↪row) # Assume all 'X' observations have a count of 1 bird
req_cols = ['COMMON NAME', 'SCIENTIFIC NAME', 'NATIVE', 'COUNT', 'LATITUDE',␣
↪'LONGITUDE', 'OBSERVATION DATE']
```

```
df = df[req_cols] # remove unnecessary columns
```

1.2.3 2.3 Aggregate eBird data based on neighborhood

[29]:
```
com_areas = gpd.read_file('data/neighborhoods/
 ↪geo_export_f5325bf0-9c6d-49a5-a5d9-0e5bf24fa856.shp')
ebird_gdf = eda.join_datasets(df,com_areas)
```

1.2.4 2.4 Creation of Consolidated eBird dataset at Community level

[31]:
```
ebird_gdf = eda.modify_ebird_dataset(ebird_gdf)
grouped = eda.comm_level_dataset(ebird_gdf)
```

1.2.5 2.5 Read census dataset and merge with consolidated eBird dataset

[32]:
```
census_df = pd.read_csv("data/
 ↪Census_Data_-_Selected_socioeconomic_indicators_in_Chicago__2008___2012_20240228.
 ↪csv")
census_df['COMMUNITY AREA NAME'] = census_df['COMMUNITY AREA NAME'].str.upper()
census_df = census_df.rename(columns={'PER CAPITA INCOME ': 'PER CAPITA␣
 ↪INCOME'})
final_df = census_df.merge(grouped, left_on='COMMUNITY AREA NAME',␣
 ↪right_on='community')
```

1.2.6 2.6 Transform final dataset

[ ]:
```
final_df["PER CAPITA INCOME IN K"] = final_df.apply(lambda x: x["PER CAPITA␣
 ↪INCOME"] / 1000, axis=1)
final_df["PovertyFlag"] = final_df.apply(lambda x: "Poor" if x["PER CAPITA␣
 ↪INCOME"] < 40000 else "Rich", axis=1)
final_df["shannon_index"] = final_df["COUNT"].apply(eda.shannon_index)
final_df.head(2)
```

1.3 Section 3: EDA

- **eBird Dataset**
  - The dataset contains bird observation data in Illinois between 2008 and 2012 at the level of individual observation points defined by latitude and longitude coordinates and attributes such as common name, scientific name, native status, count, observation date.
- **Neighborhood Dataset**
  - A geometry dataset which contains the boundary shape details of all Chicago communities
- **Final Dataset**
  - The granularity of the final dataset is at the level of individual Chicago community areas.The data is obtained by merging both eBird and census datasets

[ ]:
```
for col in ["COMMON NAME","community"]:
    eda.distribution(ebird_gdf,col)
eda.monthly_distribution(ebird_gdf)
```

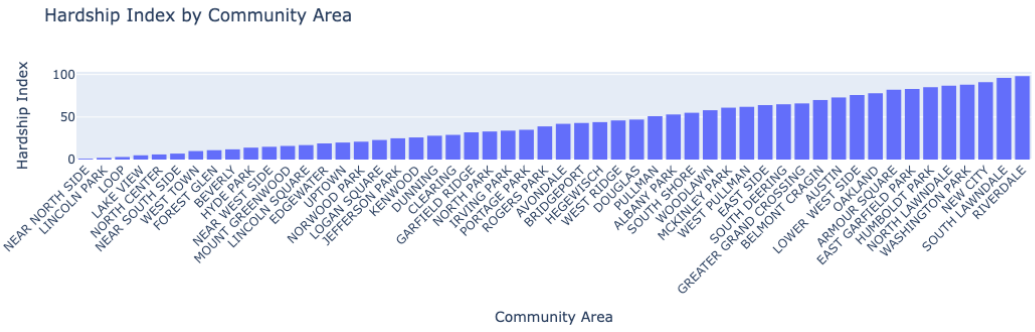
1.3.1 3.1 Insights from Datasets

- **Ring-billed Gull** and **European Starling** are the most commonly seen birds in Chicago .
- Approximately **60%** of bird observations are concentrated in only **five** communities, with **uptown** accounting for nearly 27% of these observations.
- **April, May**, and **June** have higher bird observations due to breeding and migration patterns in the summer.

1.3.2 3.2 Hardship Index By Community

[36]:
```
import matplotlib.pyplot as plt, matplotlib.patches as mpatches, seaborn as␣
↪sns, plotly.express as px, altair as alt
```

[37]:
```
px.bar(final_df.sort_values(by="HARDSHIP INDEX"), x="COMMUNITY AREA NAME",␣
↪y="HARDSHIP INDEX",
       title="Hardship Index by Community Area", labels={"HARDSHIP INDEX":␣
↪"Hardship Index",
       "COMMUNITY AREA NAME": "Community Area"}, hover_data={"COMMUNITY AREA␣
↪NAME": True,
       "HARDSHIP INDEX": True}, color_discrete_sequence=["#636EFA"]).
↪update_layout(xaxis_tickangle=-45,
       xaxis=dict(type='category')).update_traces(marker_line_width=0).show()
```



1.4 Section 4: Vizualizations

1.5 4.1 (*Does Income play a role in bird diversity of a community?*)

(*The graph depicts bird diversity across Chicago's communities, revealing a correlation between higher per capita income and increased bird diversity, while also highlighting the fluctuating levels of diversity within low-income neighborhoods.*)

[58]:
```
px.scatter(final_df[["shannon_index", "PER CAPITA INCOME IN K",␣
↪"PovertyFlag"]], x='PER CAPITA INCOME IN K', y='shannon_index',␣
↪color='PovertyFlag', color_discrete_map={0: '#636EFA', 1: '#FFA15A'},␣
↪title="Income vs. Bird Diversity", labels={"PER CAPITA INCOME IN K": "Per␣
↪Capita Income Of Community (in K)", "shannon_index": "Shannon Index␣
↪(Diversity)", "PovertyFlag": ""}, hover_data={"PovertyFlag": True},␣
↪width=600, height=400).show()
```

1.6 Section 5: Machine Learning Analysis

[39]:
```
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
```

[40]:
```
features = ['PERCENT OF HOUSING CROWDED',
            'PERCENT HOUSEHOLDS BELOW POVERTY',
            'HARDSHIP INDEX',
            'PER CAPITA INCOME']
target = 'shannon_index'
X = final_df[features]
y = final_df[target]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,␣
↪random_state=42)
```

1.6.1 5.1 Baseline

- The baseline model uses mean of the target data for fitting the data

[41]:
```
print("Mean Squared Error (Baseline - Mean Prediction): {}".format(ml.
↪baseline(y_train,y_test)))
```

```
Mean Squared Error (Baseline - Mean Prediction): 6.226041012212022
```

1.6.2 5.2 Decision Tree Regressor vs Baseline

- Decision Tree Regressor's MSE of 13.741 indicates substantial prediction errors.Baseline MSE of 6.226 shows the model's performance is worse than a simple mean predictor.Possible overfitting is suggested by the high MSE of the baseline model, might need for model refinement.

[56]:
```
print("Mean Squared Error (Decision Trees): {}".format(ml.decisiontree(X_train,␣
↪y_train,X_test,y_test)))
```

```
Mean Squared Error (Decision Trees): 13.741316951941537
```

1.6.3 5.3 Random forest regressor vs baseline

- Random Forest leverages ensemble learning, combining multiple decision trees for improved accuracy.It captures non-linear relationships in data, unlike the Baseline, which predicts a constant mean.Random Forest automatically assesses feature importance, focusing on relevant predictors, reducing prediction errors.

[57]:
```
print("Mean Squared Error (Random Forest): {}".format(ml.randomforest(X_train,␣
↪y_train,X_test,y_test)))
```

```
Mean Squared Error (Random Forest): 4.330921826772657
```

1.7 Section 6: Reflection

1.7.1 What is hardest part of the project that you've encountered so far?

Identifying community of a bird observation using the bird coordinates and the community boundaries.Obtaining the correct geographical dataset of Illinois counties with exact coordinates.

1.7.2 What are your initial insights?

Rich communities definitely have the higher bird diversity but the vice versa is not exactly true.