Gender Distributions of Collection Authors in the Rubenstein Library's Main Entry Card Catalog

These demographics were computed using the Python Gender Guesser package. As part of our exploratory data analysis, we hope to gain a better understanding of *who* is represented in the catalog. We recognize that gender is not considered as strictly binary in current times, but hope to use this analysis to evaluate the identities of the people that past librarians deemed important enough to catalog the work of in a time when gender *was* treated as more black and white than it is now. The intention of this analysis is to explore the extent of gender discrepancies in the card catalog to further study the history of the library's treatment of minority groups.

The authors of collections in the file are typically either a person or an organization. We are evaluating the gender typically associated with only the *people*. The Gender Guesser package classifies genders as one of 6 groups. Male and female result from names that are traditionally associated with one of those genders. Mostly male and mostly female result from names that are less cut and dry in regards to the gender they are associated with. "Andy," or androgynous, means that a name is not traditionally strongly associated with either gender and unknown means that the package was unable to classify a name into any of the other categories. These tended to be non-person organizations or places that would not have a gender and thus were dropped for the visualizations of the results.

We also used the numpy, pandas, and matplotlib packages, which provided us with the tools to manipulate and graph the data easily.

In [ ]:
```
!pip install gender-guesser
```

In [1]:
```
import gender_guesser.detector as gender
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:
```
# Collect first names from dataset
df = pd.read_csv("catalog/csv_iterations/all_sorted_collection.csv")

# Holds all first names
first_names = []
# Holds indices of names for use in wordclouds
indices = []

for index, row in df.iterrows():
    if row['Coll_head'] == 1:
        # if name contains a comma, it's likely a name and not an organization
        if ", " in str(row['Name']):
            name = str(row['Name'].split(" ")[1])
            # Find first name after Sir or Mrs.
            if ("Sir" in name or "Mrs" in name or "Dr." in name or "Capt" in name or "Miss" in name or "Lord" i
                name = str(row['Name'].split(" ")[2])
            # For initials, try to find whole name
            if len(name) < 2 and len(row['Name'].split(" ")) > 2 and len(str(row['Name'].split(" ")[2])) > 2:
                name = str(row['Name'].split(" ")[2])
            if not "Archive" in name and not "Army" in name and not "University" in name and not "Family" in na
                name = name.strip(" \r\n,().")
                # Add valid names to list
                if not "." in name and len(name) > 3:
                    first_names.append(name)
                    indices.append(index)
```

In [3]:
```
# Detect gender for each main entry author
d = gender.Detector()

gender_counts = {
    "unknown": 0,
    "andy": 0,
    "male": 0,
    "female": 0,
    "mostly_male": 0,
    "mostly_female": 0
}

female_indices = []
male_indices = []

# Determine gender of each name
for i in range(len(first_names)):
    name = first_names[i]
    gen = d.get_gender(name)
    if gen == "unknown":
        # Common unclassified name
        if name == "Duff":
            gen = "male"
        # Check for trailing e's (common in names from OCR error) and recheck those names
        elif "e" == name[len(name)-1]:
            name = name[0:len(name)-1]
            gen = d.get_gender(name)
    if "female" in gen:
        female_indices.append(i)
    elif "male" in gen:
        male_indices.append(i)
    gender_counts[gen] += 1

print(gender_counts)
```

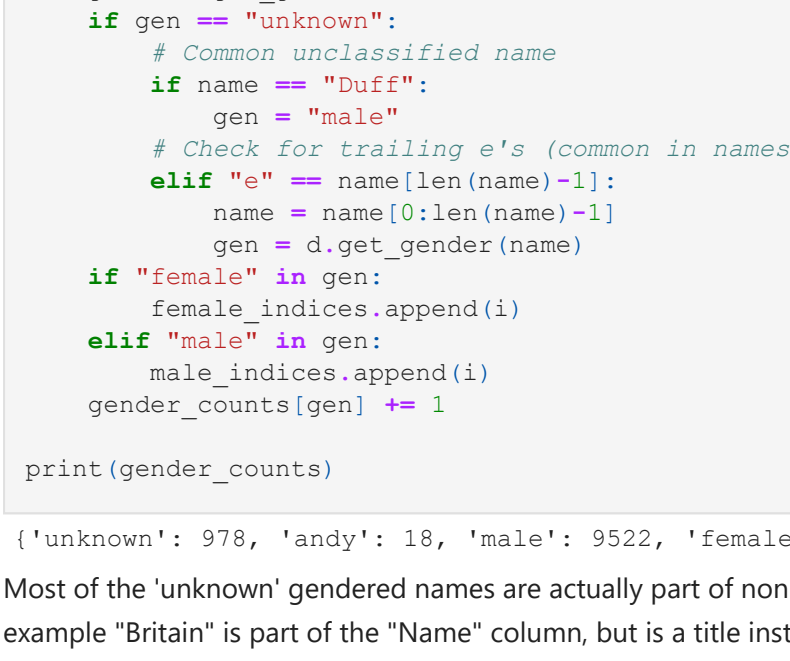
```
{'unknown': 978, 'andy': 18, 'male': 9522, 'female': 932, 'mostly_male': 67, 'mostly_female': 149}
```

Most of the 'unknown' gendered names are actually part of non-person organizations and will be dropped for easier visualization, for example "Britain" is part of the "Name" column, but is a title instead of a person, and thus will not be counted for the distributions.

In [4]:
```
# Remove unknown and andy to create bar chart
known = gender_counts.copy()
known.pop('unknown')
known.pop('andy')
```

Out[4]:
```
18
```

In [7]:
```
# Display bar chart of gender frequencies
plt.bar(*zip(*known.items()), color='#00539B')
plt.xticks(rotation = 45)
plt.title("Gender Frequency of Collection Authors in Card Catalog")
plt.xlabel("Gender")
plt.ylabel("Frequency")
plt.show()
```

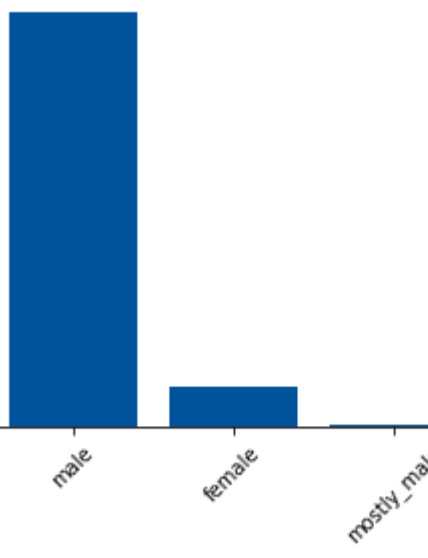


As shown in the above bar chart, the names of the authors present in the library's card collection are overwhelmingly male. This comes as no suprise to anyone who has looked through the cards. Something else of note is the strong presence of binarily gendered names i.e., there are very few "mostly male" or "mostly female" names (and seldom an androgynous one), most are one or the other. Perhaps this is indicative of the kinds of names that were given during the time period represented in the cards.

In [13]:
```
# Set up bar chart comparing male-leaning to female-leaning names
mf = known.copy()
mf['male'] = mf.get('male') + mf.get('mostly_male')
mf.pop('mostly_male')
mf['female'] = mf.get('female') + mf.get('mostly_female')
mf.pop('mostly_female')

colors = ['#339898', '#E89923']

plt.pie(mf.values(), labels = mf.keys(), autopct='%1.1f%%', startangle = 90, colors=colors)
plt.axis('equal')
plt.title("Gender Ratio of Collection Authors")
plt.show()
```



For the pie chart, we combined "mostly male" names with "male" names and "mostly female" names with female names to more easily visualize the gender frequencies. Androgynous names were dropped for the pie chart's sake, due to the fact they compose only about 0.02% of the names. The chart shows that about nine out of ten of the collection authors present in the main entry file were, in fact, male. This confirms our specualtions that men were more often than women represented in the catalog. In addition, this supports the theory that the "head of the household," likely the husband, would be elevated to the "author" of collections that entail multiple individuals, possibly hiding the presence of women in the collections. From this we can hypothesize other reasons for the discepency, but further research should be done into the history of archival records at Duke. Further research could also be done into the common events described in the manuscripts cataloged in the files (e.g., were there many Civil War male soldier accounts that took presidence over a wife's account of staying home and caring for her children?)