nltk_names_words

2019년 8월 7일

1 nltk 모듈을 이용한 이름 분석

1.1 이름 코퍼스 초기 설정

```
In [2]: import nltk
        import matplotlib.pyplot as plt
        %matplotlib inline
        names = nltk.corpus.names
        names.fileids()
```

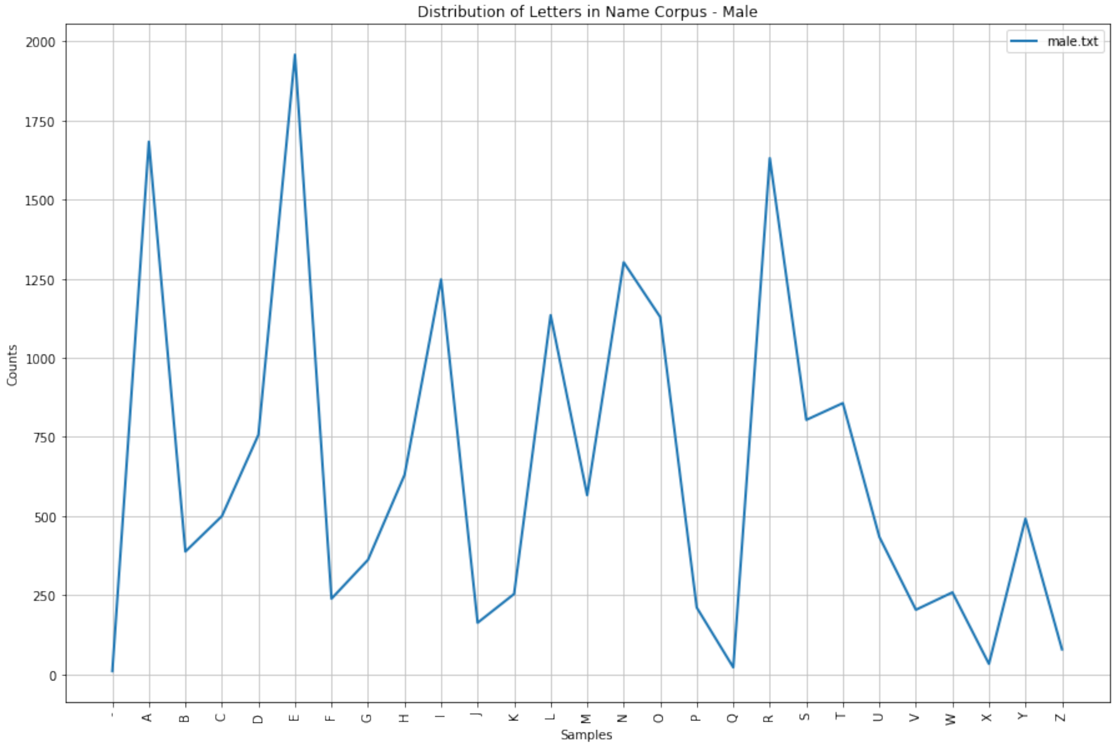
```
Out[2]: ['female.txt', 'male.txt']
```

1.2 상남자 이름 분석

```
In [3]: male_names = names.words('male.txt')
        female_names = names.words('female.txt')
        # [w for w in male_names if not w in female_names]
```

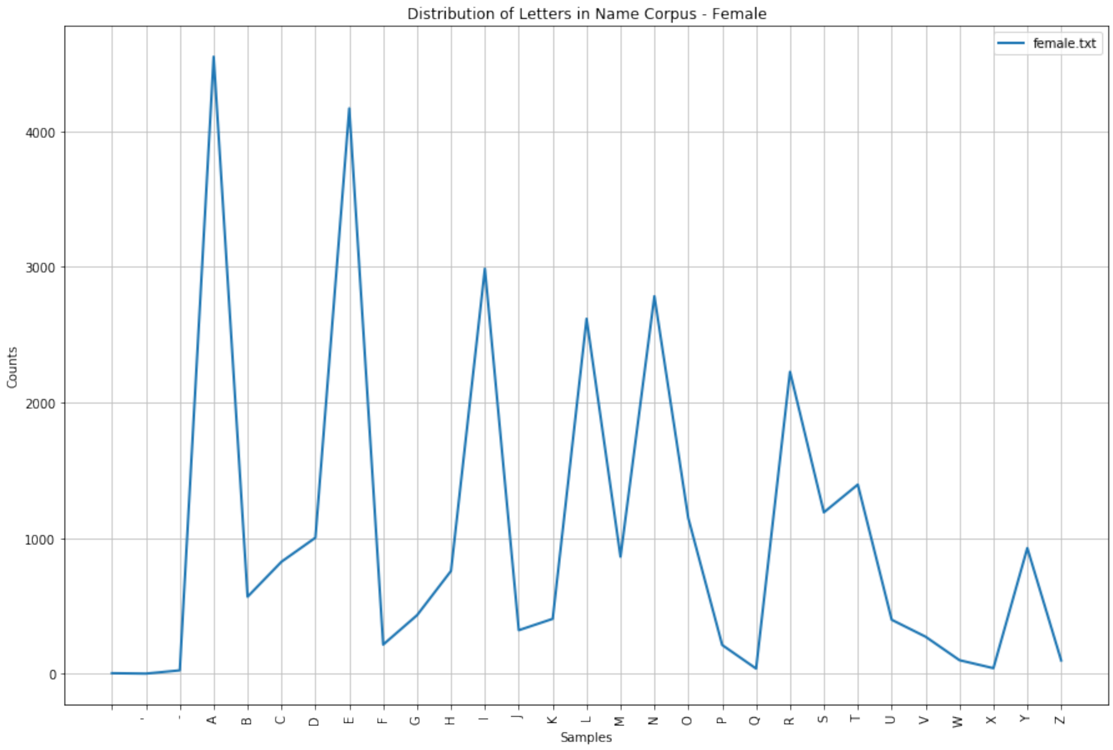
1.3 이름 알파벳 분포 분석 (남성)

```
In [4]: cfd = nltk.ConditionalFreqDist(('male.txt', letter.upper()) \
                                       for name in names.words('male.txt') \
                                       for letter in name)
        plt.figure(figsize=(15, 10))
        cfd.plot(title='Distribution of Letters in Name Corpus - Male')
```



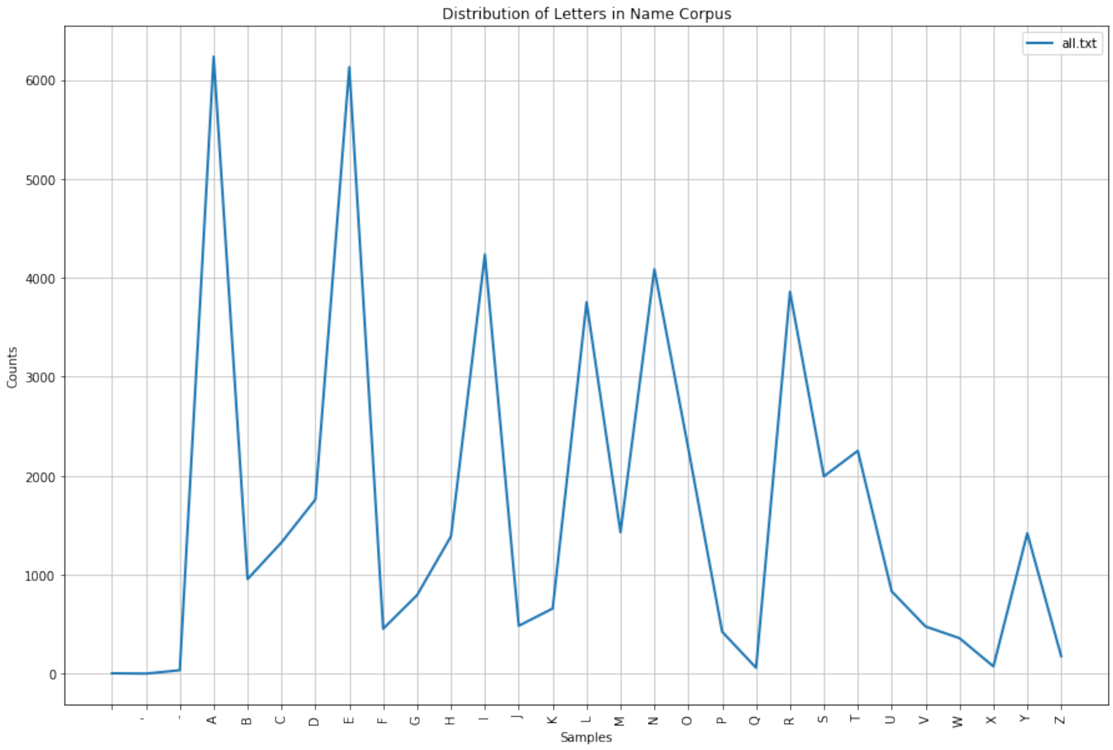
1.4 이름 알파벳 분포 분석 (여성)

```
In [5]: cfd = nltk.ConditionalFreqDist(('female.txt', letter.upper()) \
                                        for name in names.words('female.txt') \
                                        for letter in name)
        plt.figure(figsize=(15, 10))
        cfd.plot(title='Distribution of Letters in Name Corpus - Female')
```



1.5 이름 알파벳 분포 분석 (전체)

```
In [6]: cfd = nltk.ConditionalFreqDist(('all.txt', letter.upper()) \
                                       for fileid in names.fileids() \
                                       for name in names.words(fileid) \
                                       for letter in name)
        plt.figure(figsize=(15, 10))
        cfd.plot(title='Distribution of Letters in Name Corpus')
```



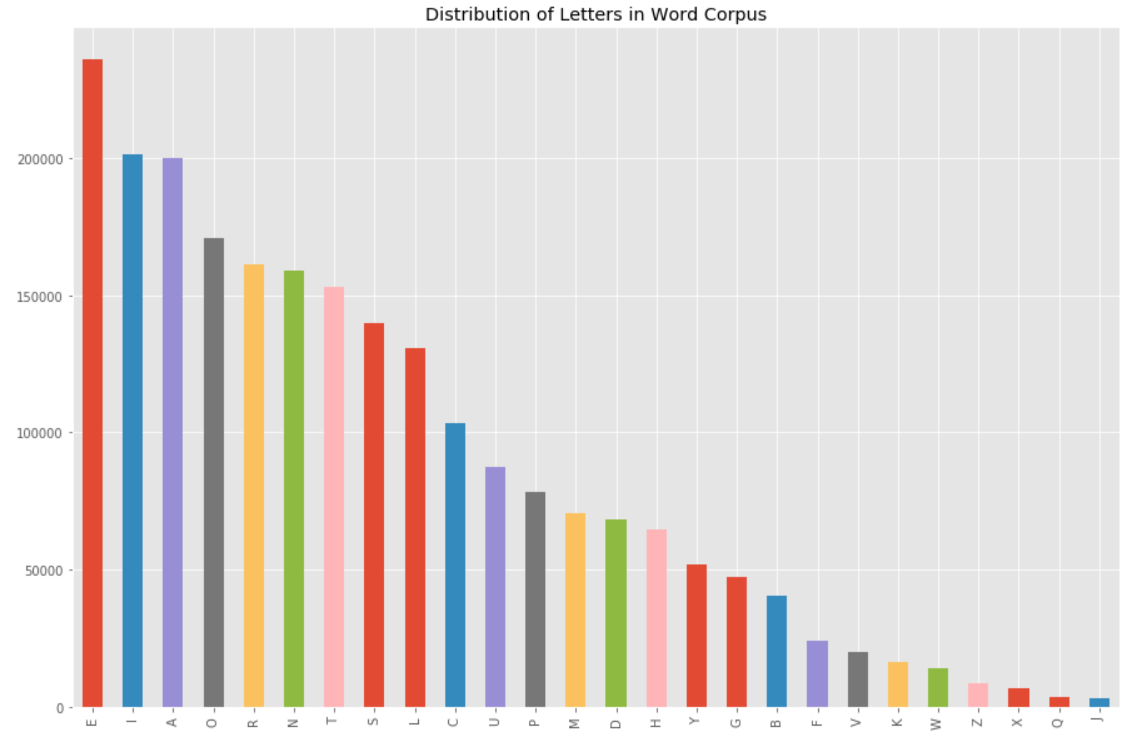
2 nltk, pandas를 이용한 이름, 어휘 분석 및 시각화

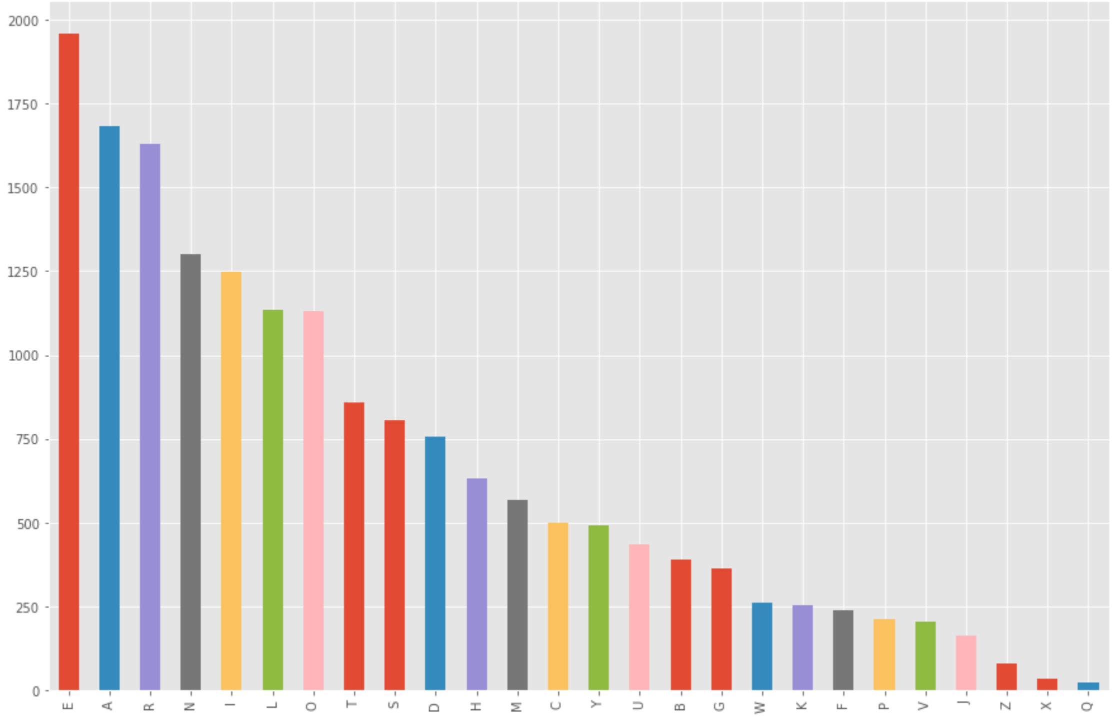
2.1 어휘, 이름(남성, 여성, 전체) 알파벳 분포 분석 (pandas)

```
In [7]: import pandas as pd
        import numpy as np
        import nltk
        import matplotlib.pyplot as plt
        %matplotlib inline
        plt.style.use('ggplot')

        names = nltk.corpus.names
        abclist = ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O',

        male_names = names.words('male.txt')
        male_letters = [l.upper() for w in male_names for l in w if l.upper() in abclist]
```

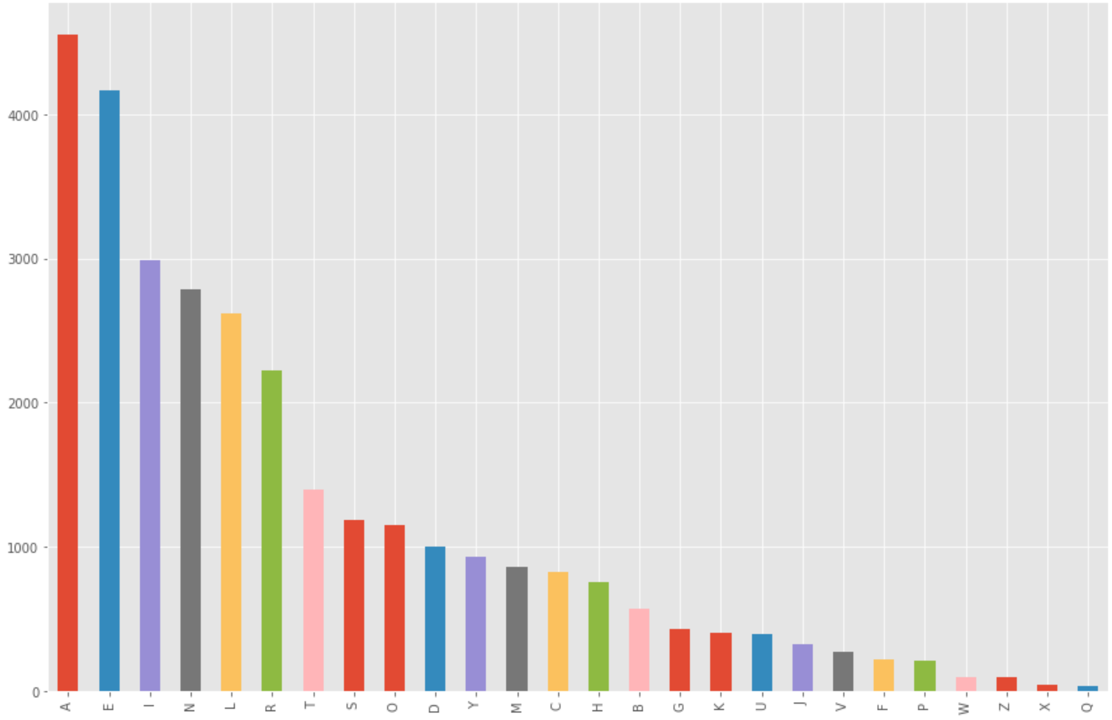
```python
female_names = names.words('female.txt')
female_letters = [l.upper() for w in female_names for l in w if l.upper() in abclist]

names_list = [w for w in male_names] + [w for w in female_names if not w in male_names]
names_letters = [l.upper() for w in names_list for l in w if l.upper() in abclist]
# print(names_letters)

words_list = nltk.corpus.words.words()
words_letters = [l.upper() for w in words_list for l in w if l.upper() in abclist]

df0 = pd.Series(words_letters)
plt.figure(figsize=(15, 10))
df0.value_counts().plot.bar()
plt.title('Distribution of Letters in Word Corpus')
plt.savefig('dist0.png', dpi=400, bbox_inches='tight')
plt.show()



df1 = pd.Series(male_letters)
plt.figure(figsize=(15, 10))
df1.value_counts().plot.bar()
plt.title('Distribution of Letters in Name Corpus - male')
plt.savefig('dist1.png', dpi=400, bbox_inches='tight')
plt.show()

df2 = pd.Series(female_letters)
plt.figure(figsize=(15, 10))
df2.value_counts().plot.bar()
plt.title('Distribution of Letters in Name Corpus - female')
plt.savefig('dist2.png', dpi=400, bbox_inches='tight')
plt.show()

df3 = pd.Series(names_letters)
plt.figure(figsize=(15, 10))
df3.value_counts().plot.bar()
```

```
plt.title('Distribution of Letters in Name Corpus - all')
plt.savefig('dist3.png', dpi=400, bbox_inches='tight')
plt.show()
```

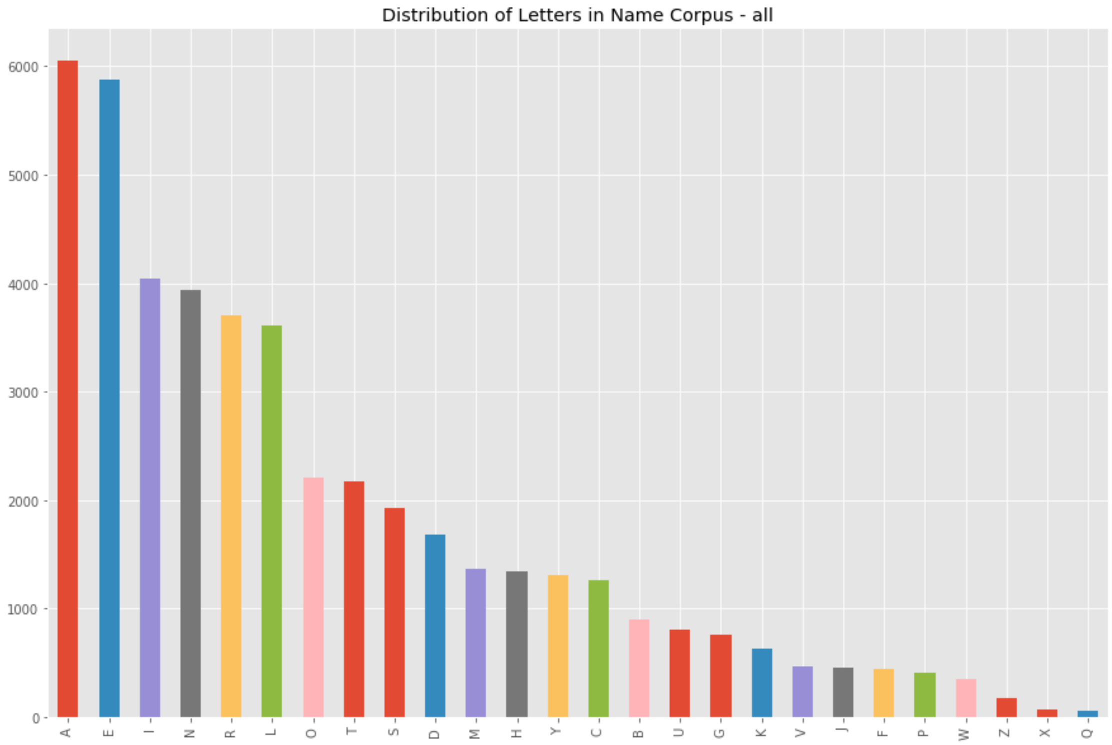




Distribution of Letters in Name Corpus - male



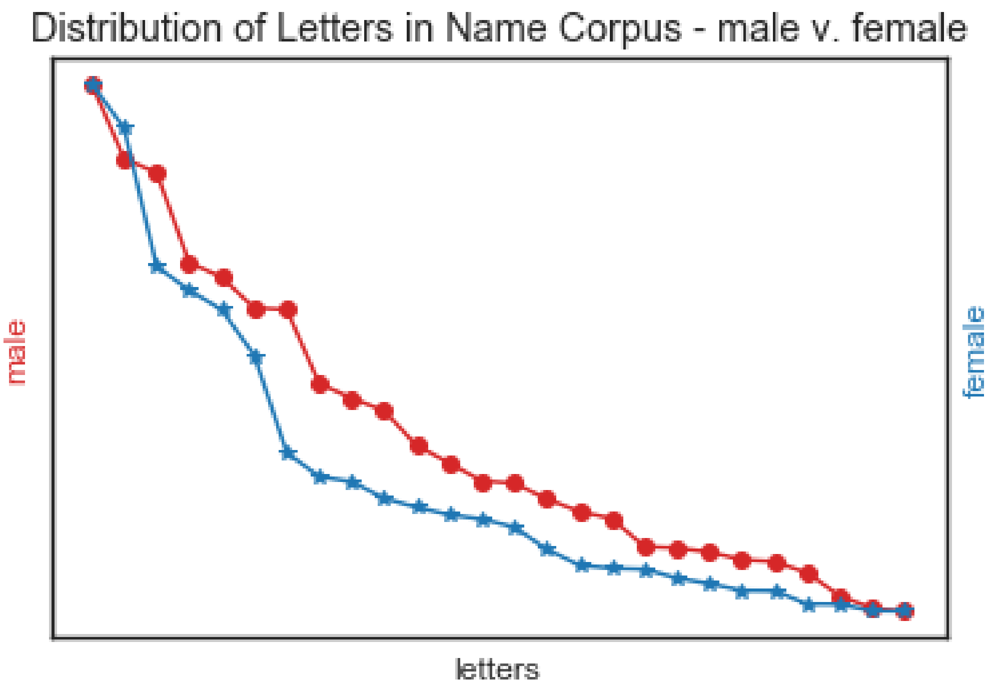
Distribution of Letters in Name Corpus - female



2.2 이름 알파벳 분포 분석 - 성별 비교 (pandas)

```
In [8]: import pandas as pd
        import numpy as np
        import nltk
        import matplotlib.pyplot as plt
        %matplotlib inline
        plt.style.use('seaborn-white')

        # 성별 비교

        names = nltk.corpus.names
        abclist = ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O',

        male_names = names.words('male.txt')
```

```
male_letters = [l.upper() for w in male_names for l in w if l.upper() in abclist]

female_names = names.words('female.txt')
female_letters = [l.upper() for w in female_names for l in w if l.upper() in abclist]

fig, ax1 = plt.subplots()

color = 'tab:red'
ax1.set_xlabel('letters')
ax1.set_ylabel('male', color=color)
ax1.plot(pd.Series(male_letters).value_counts(), label="male", linestyle='-', marker='
ax1.set_xticklabels([])
ax1.set_yticklabels([])

ax2 = ax1.twinx()
color = 'tab:blue'
ax2.set_ylabel('female', color=color)
ax2.plot(pd.Series(female_letters).value_counts(), label="female", linestyle='-', marke
ax2.set_xticklabels([])
ax2.set_yticklabels([])
plt.title('Distribution of Letters in Name Corpus - male v. female')
plt.savefig('dist4.png', dpi=400, bbox_inches='tight')
plt.show()
```



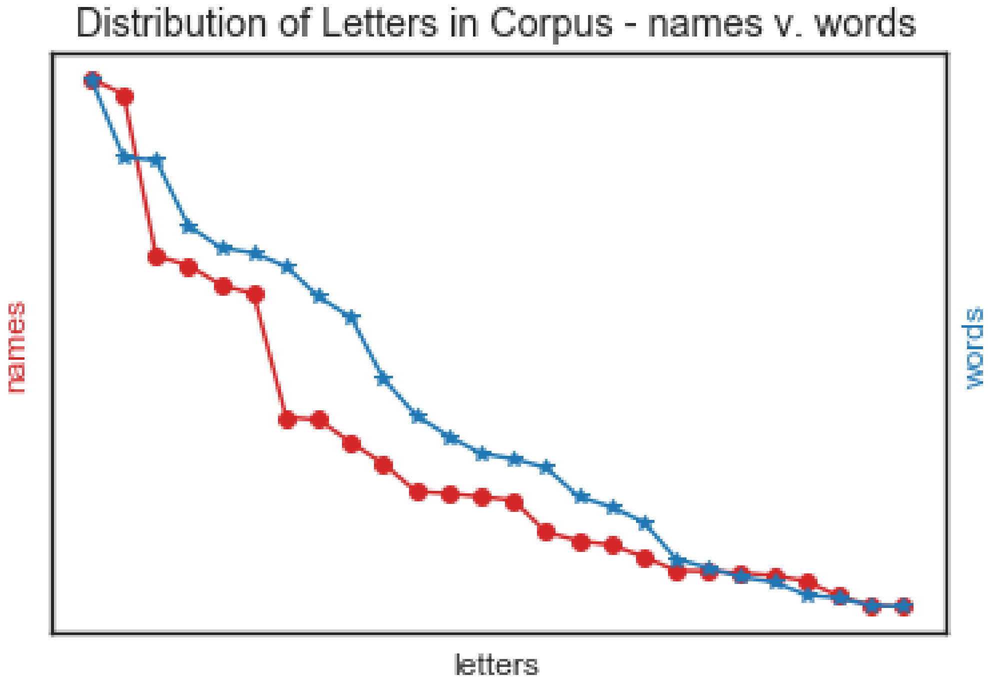
2.3 이름 알파벳 분포 분석 - 명사 어휘 분포와 비교 (pandas)

```
In [10]: import pandas as pd
         import numpy as np
         import nltk
         import matplotlib.pyplot as plt
         %matplotlib inline
         plt.style.use('seaborn-white')

         # 이름 가져오기

         names = nltk.corpus.names
         abclist = ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O',

         male_names = names.words('male.txt')
         female_names = names.words('female.txt')

         names_list = [w for w in male_names] + [w for w in female_names if not w in male_name
```

```
names_letters = [l.upper() for w in names_list for l in w if l.upper() in abclist]

# 단어 가져오기

words_list = nltk.corpus.words.words()
words_letters = [l.upper() for w in words_list for l in w if l.upper() in abclist]

print(len(names_letters), 'letters from', len(names_list), 'names')
print('average', round(len(names_letters) / len(names_list), 2), 'letters per a name')
print(len(words_letters), 'letters from', len(words_list) , 'words')
print('average', round(len(words_letters) / len(words_list), 2), 'letters per a word')



fig, ax1 = plt.subplots()

color = 'tab:red'
ax1.set_xlabel('letters')
ax1.set_ylabel('names', color=color)
ax1.plot(pd.Series(names_letters).value_counts(), label="names", linestyle='-', marke
ax1.set_xticklabels([])
ax1.set_yticklabels([])

ax2 = ax1.twinx()
color = 'tab:blue'
ax2.set_ylabel('words', color=color)
ax2.plot(pd.Series(words_letters).value_counts(), label="words", linestyle='-', marke
ax2.set_xticklabels([])
ax2.set_yticklabels([])
plt.title('Distribution of Letters in Corpus - names v. words')
plt.savefig('dist5.png', dpi=400, bbox_inches='tight')
plt.show()
```

```
45992 letters from 7579 names
average 6.07 letters per a name
2261673 letters from 236736 words
```

```
average 9.55 letters per a word
```


Distribution of Letters in Corpus - names v. words
names
words
letters