Photo by Abbie Bernet

Landscapes

At the Metropolitan Museum of Art

Amanda Anderson-You

Background

- Multiple concepts before landing here
- Inspired by scope of landscape keywords in CSV
- Worked with Pandas to format the dataset
- Used plotly.js for visualizing, which is D3.js based

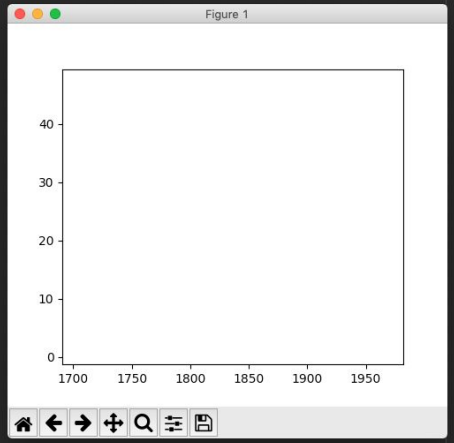
Process

- Reviewed CSV and filtered by keyword “landscapes”
  - Includes objects where landscape is included anywhere in the list of tags, not just as a primary tag
- Reviewed departments and chose to include only departments with over 25 objects, keeping 8 of 16 departments.
- Used object end date for year created, decided to limit to years AD and remove 13 records (of almost 5K total) of years BC

Met_QuantVisual.csv | Met_StructureData.py

```
# import csv
# input_file = csv.DictReader(open("MetQuant_GoodData.csv"))
# for row in input_file:
#     print(row)

import csv
import collections
import matplotlib.pyplot as plt
# import pandas as pd


with open('Departments/American_DecoArts.csv') as csvfile:
    readCSV = csv.reader(csvfile, delimiter=',')
    years = []
    for row in readCSV:
        year = row[1]
        years.append(year)

    # print(years)


dictionary = (collections.Counter(years))
# print(dictionary)

keys = dictionary.keys()
values = dictionary.values()

# print(keys)
# print(values)

keys = [int(i) for i in keys]
values = [int(i) for i in values]
print(keys)
print(values)


x = [keys]
y = [values]
plt.plot(x,y)
plt.show()
```

Figure 1



with open('Departments/American... › for row in readCSV

Met_StructureData

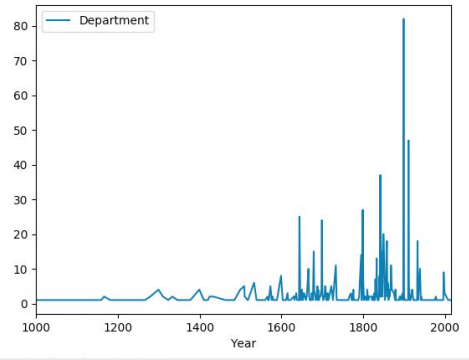
```
/usr/local/bin/python3.7 /Users/a.andersonyou/Documents/School/Parsons/MajorStudio_1/Met_StructureData.py
[1704, 1727, 1730, 1755, 1758, 1760, 1761, 1776, 1778, 1790, 1795, 1796, 1798, 1800, 1806, 1810, 1815, 1816, 1820, 1823, 1825, 1828
[1, 1, 3, 1, 1, 1, 1, 1, 2, 5, 4, 1, 1, 5, 1, 4, 4, 15, 2, 2, 5, 2, 37, 1, 5, 2, 3, 3, 4, 14, 28, 11, 5, 3, 1, 4, 4, 1, 3, 10, 7, 1
```

Work in Progress...

Met_PythonVis.py | Met_StructureData.py | testADA.json

```
# import csv
# input_file = csv.DictReader(open("MetQuant_GoodData.csv"))
# for row in input_file:
#     print(row)

import csv
import collections
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

# Use pandas to read the .csv file with approximately 5k object records of landscape art at the Met.
# Columns include department and end date year, I'll later use counter to record the quantity of pieces
# per year per department.
data = pd.read_csv('MetQuant_GoodData.csv', delimiter=',')
# create variables that hold individual tables of date for each of the 8 total departments.
american_decoarts = data[data.Department == 'American Decorative Arts']
asian_arts = data[data.Department == 'Asian Art']

american_decoarts = american_decoarts.groupby('Year').count()
asian_arts = asian_arts.groupby('Year').count()

print(asian_arts)

# american_decoarts.plot.line()
asian_arts.plot.line()
plt.show()
```

Figure 1



Run: Met_PythonVis

```
/usr/local/bin/python3.7 /Users/a.andersonyou/Documents/School/Parsons/MajorStudio_1/Departments/Met_PythonVis.py
      Department
Year
999            1
1060           1
1090           1
1133           1
1159           1
...          ...
1997           9
1999           3
2008           1
2009           1
2015           1

[266 rows x 1 columns]
```

```
                    Department  Year
0     American Decorative Arts  1704
1     American Decorative Arts  1727
2     American Decorative Arts  1730
3     American Decorative Arts  1730
4     American Decorative Arts  1730
..                         ...   ...
757   American Decorative Arts  1941
758   American Decorative Arts  1944
759   American Decorative Arts  1957
760   American Decorative Arts  1962
761   American Decorative Arts  1969

[762 rows x 2 columns]
```

```
# when department == "department name"
df['count'] = df.groupby('Year')['Department'].transform('count')
```

```
/usr/local/bin/python3.7 /Users/a.andersonyou/D
                    Department  Year  count
0     American Decorative Arts  1704      2
1     American Decorative Arts  1727      2
2     American Decorative Arts  1730     10
3     American Decorative Arts  1730     10
4     American Decorative Arts  1730     10
...                        ...   ...    ...
4861  Robert Lehman Collection  1958      6
4862  Robert Lehman Collection  1963      3
4863  Robert Lehman Collection  1964      4
4864  Robert Lehman Collection  1965      3
4865  Robert Lehman Collection  1969     19

[4866 rows x 3 columns]

Process finished with exit code 0
```

```
data = pd.read_csv('MetQuant_GoodData.csv', delimiter=',', names=['Department', 'Year', 'Count'])
# create variables that hold individual tables of date for each of the 8 total departments.
american_decoarts = data[data.Department == 'American Decorative Arts']
year_counts = pd.value_counts(american_decoarts.values.ravel())
# year_counts = year_counts[1:]
# print(type(year_counts))
df = pd.DataFrame([year_counts])
print(df)
```

```
   American Decorative Arts  1870  1890  1904  ...  1796  1842  1941  1921
0                       762    47    46    39  ...     1     1     1     1

[1 rows x 128 columns]
```

Work in Progress...

```
import pandas as pd
import plotly.express as px
import plotly.graph_objects as go

app = dash.Dash(__name__)

# Use pandas to read the .csv file with approximately 5k object records of landscape art at the Met.
# Existing csv columns used include departments & end date year. CSV contains additional object level information in
# case needed.
df = pd.read_csv('MetLandscape_Details.csv', delimiter=',')

# Create a tidy Pandas DataFrame from the csv file.
# Use reset_index to make an Artworks column totaling the unique instances of both department and year.
df = df.groupby(['Department', 'Year']).size().reset_index(name='Artworks')
# print(df)

# Using vectors to plot all points at once
# Plotly uses D3, the user is able to hover over each point for additional detail
```

Structuring the Data

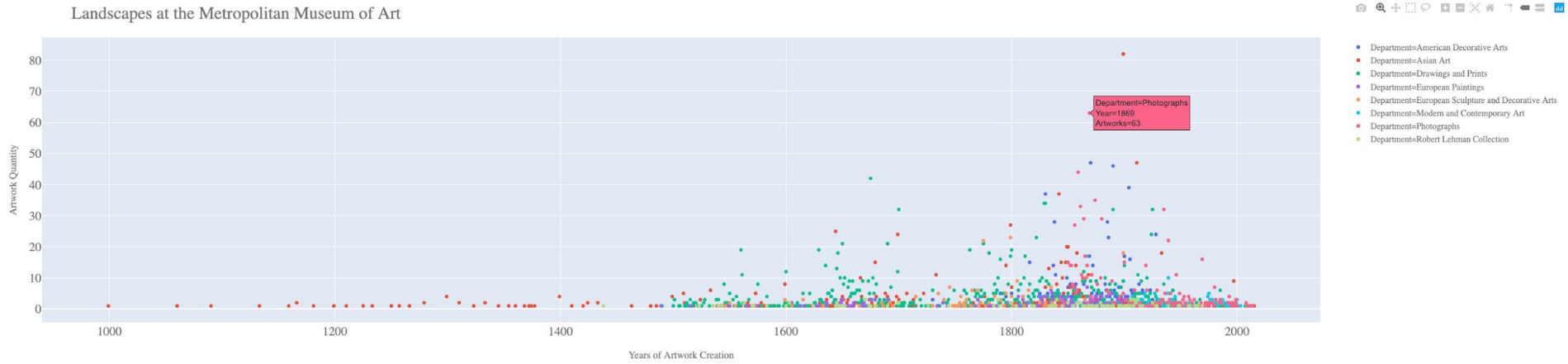
```python
# Plotly Scatter Plot:
fig = px.scatter(df, x="Year", y="Artworks", color="Department")
fig.update_layout(
    title='Landscapes at the Metropolitan Museum of Art',
        font=dict(
            family="Times, monospace",
            size=20,
            color="#7f7f7f"
    )
)
fig.update_layout(

    xaxis=go.layout.XAxis(
        title=go.layout.xaxis.Title(
            text="Years of Artwork Creation",
            font=dict(
                family="Times, monospace",
                size=16,
                color="#7f7f7f"
            )
        )
    ),
    yaxis=go.layout.YAxis(
        title=go.layout.yaxis.Title(
            text="Artwork Quantity",
            font=dict(
                family="Times, monospace",
                size=16,
                color="#7f7f7f"
            )
        )
    )
)

fig.update_layout(

    legend=go.layout.Legend(
        traceorder="normal",
        font=dict(
            family="Times, monospace",
            size=14,
            color="#7f7f7f"
        ),
    )
)

fig.show()
```

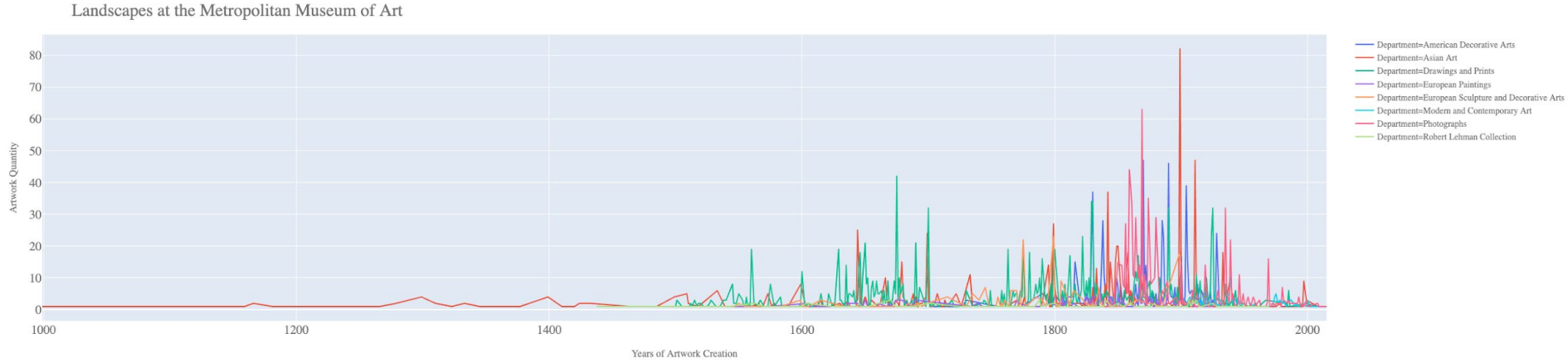
```python
# Plotly Line Plot:
fig = px.line(df, x='Year', y='Artworks', color='Department')
fig.update_layout(
    title='Landscapes at the Metropolitan Museum of Art',
        font=dict(
            family="Times, monospace",
            size=20,
            color="#7f7f7f"
    )
)
fig.update_layout(

    xaxis=go.layout.XAxis(
        title=go.layout.xaxis.Title(
            text="Years of Artwork Creation",
            font=dict(
                family="Times, monospace",
                size=16,
                color="#7f7f7f"
            )
        )
    ),
    yaxis=go.layout.YAxis(
        title=go.layout.yaxis.Title(
            text="Artwork Quantity",
            font=dict(
                family="Times, monospace",
                size=16,
                color="#7f7f7f"
            )
        )
    )
)
fig.update_layout(

    legend=go.layout.Legend(
        traceorder="normal",
        font=dict(
            family="Times, monospace",
            size=14,
            color="#7f7f7f"
        ),
    )
)
fig.show()
```

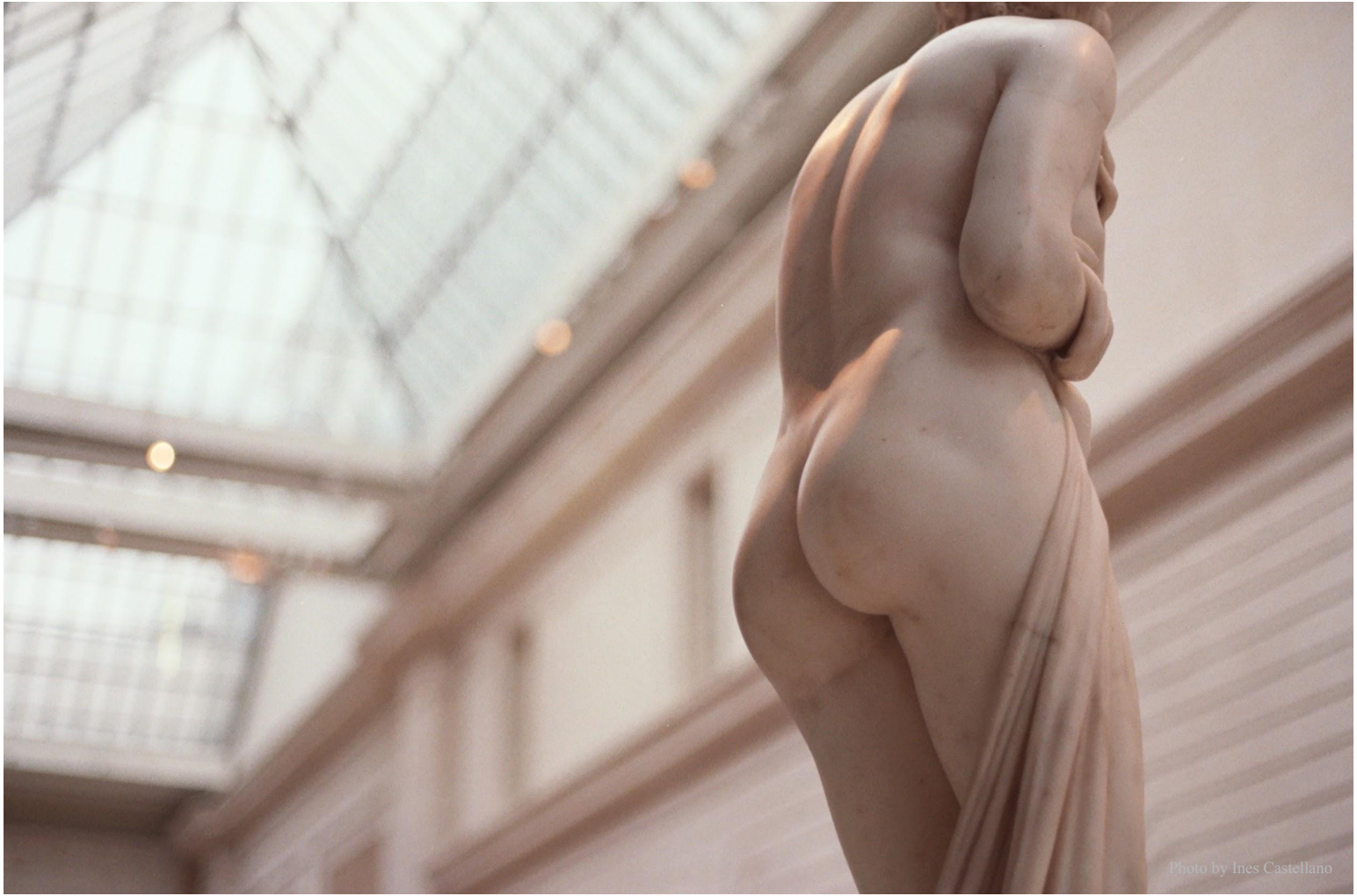
Plotting the Scatter and Line Graphs

Click to view and interact:
Scatter & Line Plots

Tips
Hover over points to see more information
Select specific departments to turn them on/off


Landscapes at the Metropolitan Museum of Art
Department=Photographs
Year=1869
Artworks=63
Artwork Quantity
Years of Artwork Creation
Department=American Decorative Arts
Department=Asian Art
Department=Drawings and Prints
Department=European Paintings
Department=European Sculpture and Decorative Arts
Department=Modern and Contemporary Art
Department=Photographs
Department=Robert Lehman Collection


Landscapes at the Metropolitan Museum of Art
Artwork Quantity
Years of Artwork Creation
Department=American Decorative Arts
Department=Asian Art
Department=Drawings and Prints
Department=European Paintings
Department=European Sculpture and Decorative Arts
Department=Modern and Contemporary Art
Department=Photographs
Department=Robert Lehman Collection



Thank you!