Academic Integrity Statement

As a matter of Departmental policy, **we are required to give you a 0** unless you **type your name** after the following statement:

> *I certify on my honor that I have neither given nor received any help, or used any non-permitted resources, while completing this evaluation.*

Mey Sok

Partial Credit

Let us give you partial credit! If you're stuck on a problem and just can't get your code to run:

First, **breathe**. Then, do any or all of the following:

1. Write down everything relevant that you know about the problem, as comments where your code would go.
2. If you have non-functioning code that demonstrates some correct ideas, indicate that and keep it (commented out).
3. Write down pseudocode (written instructions) outlining your solution approach.

In brief, even if you can't quite get your code to work, you can still **show us what you know.**

Part A (30 points)

In this problem, you will create a visualization of gender representation in artwork in the Tate Art Museum (https://github.com/tategallery/collection). Run the code block below to acquire and prepare the data. There's a lot of information that I've removed in the data preparation below, including the name of the artist, their birth and death dates, and various details about each piece. You may wish to explore the full data sets later, but for now, I thought you'd prefer to be able to focus on only the columns needed for today.

In [1]:
```
import pandas as pd
artwork = pd.read_csv('https://raw.githubusercontent.com/rfordatascience/ti
artists = pd.read_csv("https://github.com/tategallery/collection/raw/master

artwork["id"] = artwork["artistId"]
artwork = artwork[["id", "year", "acquisitionYear", "title", "medium"]]
artists = artists[["id", "gender"]]
df = pd.merge(artwork, artists)

def dimension(med_string):
    """
    Assign a dimension to a given piece of artwork based on the description
    of the medium, supplied as a string.
    Media that include the words "paper", "canvas", "oil", or "paint" are a
    2D.
    Media that are not 2d and include the words "bronze", "stone", or "cera
    assumed 3D.
    Otherwise, the media is "Other/Unknown"

    @param med_string: str, the original medium
    @return dim: one of "2D", "3D", or "Other/Unknown" according to the rul
    """
    if type(med_string) != str:
        med_string = str(med_string)
    med_string = med_string.lower()
    if any([w in med_string for w in ["paper", "canvas", "oil", "paint"]]):
        return "2D"
    elif any([w in med_string for w in ["bronze", "stone", "ceramic"]]):
        return "3D"
    else:
        return "Other/Unknown"

df["dimension"] = [dimension(m) for m in df["medium"]]
df = df[["title","acquisitionYear", "gender", "dimension"]]
```

- The `title` column gives the title of each piece.
- The `acquisitionYear` states the year in which the artwork was acquired by the Tate.
- The `gender` column gives the gender of the artist.
- The `dimension` column states whether the piece is two-dimensional (like a drawing or a painting) or three-dimensional (like a sculpture or ceramic). This is determined from a more thorough description of the medium using the simple `dimension()` function from above, although a more careful classification might be beneficial. A number of pieces have "Other/Unknown" in this column.

In [2]:
```
# use this block to inspect the data if you'd like
df
```

Out[2]:

	title	acquisitionYear	gender	dimension
0	A Figure Bowing before a Seated Old Man with h...	1922.0	Male	2D
1	Two Drawings of Frightened Figures, Probably f...	1922.0	Male	2D
2	The Preaching of Warning. Verso: An Old Man En...	1922.0	Male	2D
3	Six Drawings of Figures with Outstretched Arms	1922.0	Male	2D
4	The Circle of the Lustful: Francesca da Rimini...	1919.0	Male	2D
...	...	...	...	...
69190	Venus Mound (from Tampax Romana)	2013.0	Male	Other/Unknown
69191	It's That Time Of The Month (from Tampax Romana)	2013.0	Male	Other/Unknown
69192	Larvae (from Tampax Romana)	2013.0	Male	Other/Unknown
69193	Living Womb (from Tampax Romana)	2013.0	Male	2D
69194	Dancing Scene in the West Indies	2013.0	Male	2D

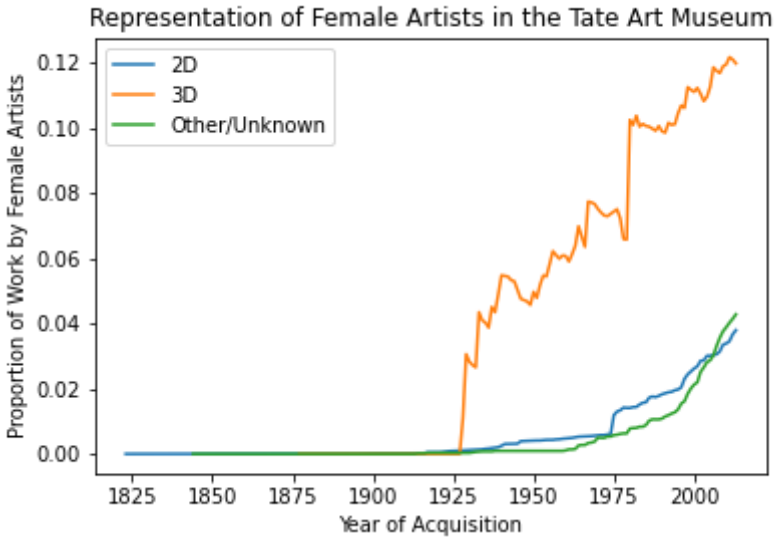
69195 rows × 4 columns

What You Should Do

Create a plot to answer the following question:

> How has the amount of artwork **by female artists** increased with time, as a fraction of all artwork owned by the Tate? Are women better represented in the Tate through certain forms of artistic expression than others?

To answer this question, create the following plot:



The vertical axis is the percentage of all artwork created by female artists which was acquired on or before the stated date. You may assume that artwork, once acquired, remains permanently with the Tate (i.e. it is not lost or sold).

Specs

- There are multiple good approaches. A solution using a `for` - or `while` -loop can receive up to 27/30 points. For full credit, no explicit loops!
- It is not necessary for your output to exactly match mine -- feel free to change colors, modify the labels, etc. However, you should ensure that you include axis labels and the legend.
- Comments and docstrings are not necessary in this problem.
- You are free to use any Python tools you find helpful in order to create this plot.

"What if my plot looks different?"

Your final product should closely resemble the supplied example. You may make reasonable alternative choices that lead your plot to look slightly different in small details. You can receive full credit as long as your result looks quantitatively similar and has the same qualitative interpretation.

Hints

- `np.cumsum()` . You'll need to appropriately sort `df` first in order to get a good result.

In [3]:
```
# your solution here
import numpy as np
from matplotlib import pyplot as plt
from sklearn import preprocessing

# year and gender

# dropping NaN values and title
df = df.dropna()

cols = ['acquisitionYear', 'gender', 'dimension']
df = df[cols]


# encoding sex
# female = 0
# male = 1
le = preprocessing.LabelEncoder()
df['gender'] = le.fit_transform(df['gender'])

# df['cum_sum'] = df.groupby('gender').unique.cumsum() / df.gender.sum()

df1 = df.copy()
```

In [18]:
```
# split data frame into male and female to figure out cumsum
x = df1.groupby(['acquisitionYear', 'gender']).size().unstack().fillna(0)

gender_frame = pd.DataFrame(x)
y = gender_frame

gender_frame

x1 = df1.groupby(['acquisitionYear','gender']).count().reset_index()
```

Out[18]:

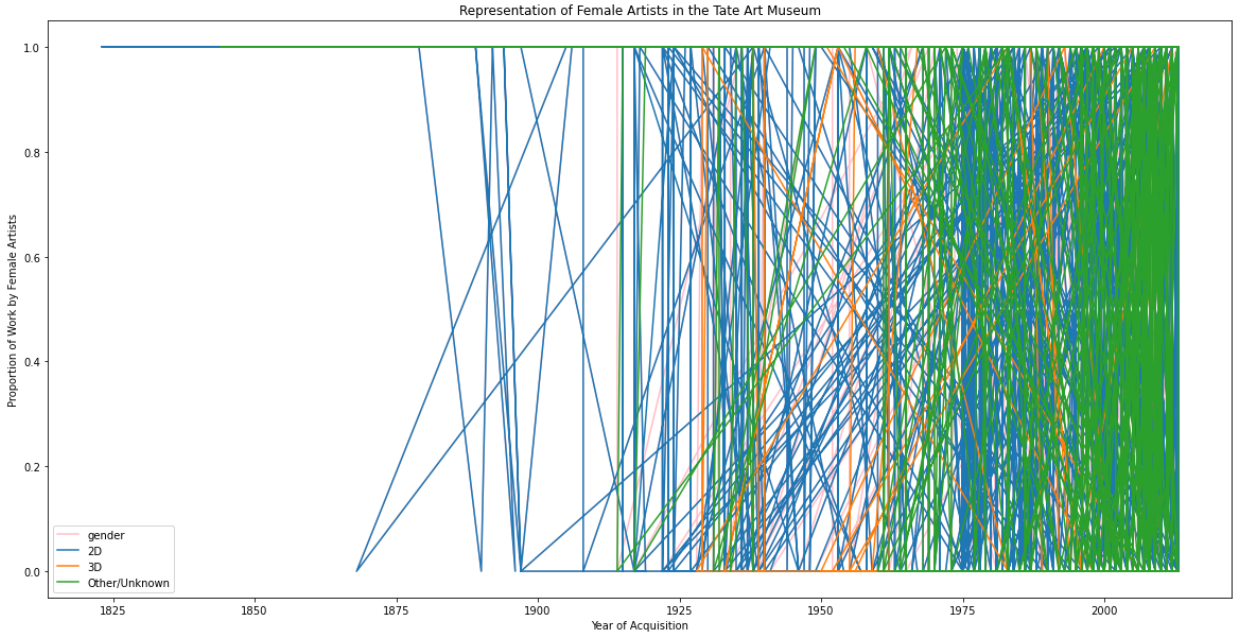
	acquisitionYear	gender	dimension
0	1823.0	1	1
1	1824.0	1	2
2	1826.0	1	4
3	1827.0	1	1
4	1828.0	1	3
...	...	...	...
272	2011.0	1	249
273	2012.0	0	152
274	2012.0	1	345
275	2013.0	0	105
276	2013.0	1	352

277 rows × 3 columns

In [5]:
```
df.plot(x = 'acquisitionYear', y = 'gender',
        kind = 'line', figsize = (20, 10),
        color = ['pink', 'purple', 'blue'])

df.set_index('acquisitionYear', inplace = True)
df.groupby('dimension')['gender'].plot(legend = True)
plt.title("Representation of Female Artists in the Tate Art Museum")
plt.xlabel("Year of Acquisition")
plt.ylabel("Proportion of Work by Female Artists")
```

Out[5]:
```
Text(0, 0.5, 'Proportion of Work by Female Artists')
```



Part B (5 points)

This part is not actually related to Part A above. This part asks you to reflect on dangers related to algorithmic bias in a **hypothetical scenario**.

Imagine that you are a wealthy venture capitalist, considering whether to invest in a new tech startup company. The startup is called TNNT (pronounced "tenant"). The startup aims to accelerate the speed at which landlords are able to find tenants to rent apartments. The founder claims that their proprietary algorithm can be used to identify "high-quality," reliable applicants to whom landlords would like to rent apartments, while screening out "low-quality" applicants. This reduces the number of applications that the landlord will have to read, thus saving the landlord time. The founder hopes that landlords will pay a small fee to use the service.

Here's how the algorithm works:

1. Melanie and Xenith submit an application on TNNT.com, providing basic information about their jobs and income.
2. TNNT's proprietary algorithm produces a *score* for Melanie and Xenith's application, based on the information they provided. Let's say their score is 0.87. This score is intended to reflect the probability that the landlord will approve their application.
3. Only scores above 0.80 are supplied to the landlord for a final decision. So, the landlord will receive and evaluate Melanie and Xenith's application, but not Roberto and Darrell's application which received only a 0.63.

The value for the landlord is that they will receive a smaller number of high-quality applications, as measured by the TNNT algorithm. This saves the landlord time.

A few technical details. The training data for the TNNT algorithm is a large collection of the decisions of landlords from Los Angeles. Each piece of training data includes the name of the applicant(s), information about their job(s), and information about their income. You can think of this information as the predictor data `x`. It also includes an indication about whether the application was ultimately approved by the landlord. You can think of this information as the target data `y`. TNNT collects recent landlord decisions, and uses them to update the model.

Below, please respond briefly to the following questions. **Two sentences each is plenty**, although it's fine if you want to write more.

Note: the definitions of *measurement bias* and *historical bias* are from Rachel Thomas's video (https://youtu.be/S-6YGPrmtYc) that we watched for lecture, "Getting Specific About Algorithmic Bias."

1. What is a potential source of *measurement bias* in the TNNT algorithm?
    - (*Hint*: compare what the founder says the algorithm measures compared to the data the algorithm actually uses).
2. What is a potential source of *historical bias* in the TNNT algorithm?
    - It is not necessary (though it is perfectly fine) to connect your answer to actual history. You may, for example, assume that people with an odd number of letters in English first name constitute a historically oppressed group, if that will help you explain your example.
3. What is an example of a *feedback loop* that might amplify bias in the TNNT algorithm over time?

---

1. A potential source of measurement bias in the TNNT algorithm is that the system will measure the candidates based on whether or not their applications were previously approved. People who have not been approved before (first-time renters, people who were discriminated against, etc.) are most likely going to get lower scores and can cause further bias.
2. The TNNT algorithm is most likely going to be trained on previous approved applications, thus it will most likely discriminate against oppressed groups who have not been approved due to racial or gender bias in the real world. Historically, oppressed groups being those who are categorized as minorities, Black, or people of color have not had a fair shot to obtain housing. This can be seen in the Fair Housing Act that was implemented during the Civil Rights Act of

1968. The implementation of such laws was meant to grant access to housing and rights to former slaves. Being that such groups were historically oppressed, there a chance of historical bias that can lead to fault in the algorithm.

3. Since the algorithm was trained on past data, biases that are present in that past data can carry over into futurn decisions being made when providing a threshold for approving applications. Over time, the machine will keep on feeding into the system, data that will have similar biases that will further amplify those biases.