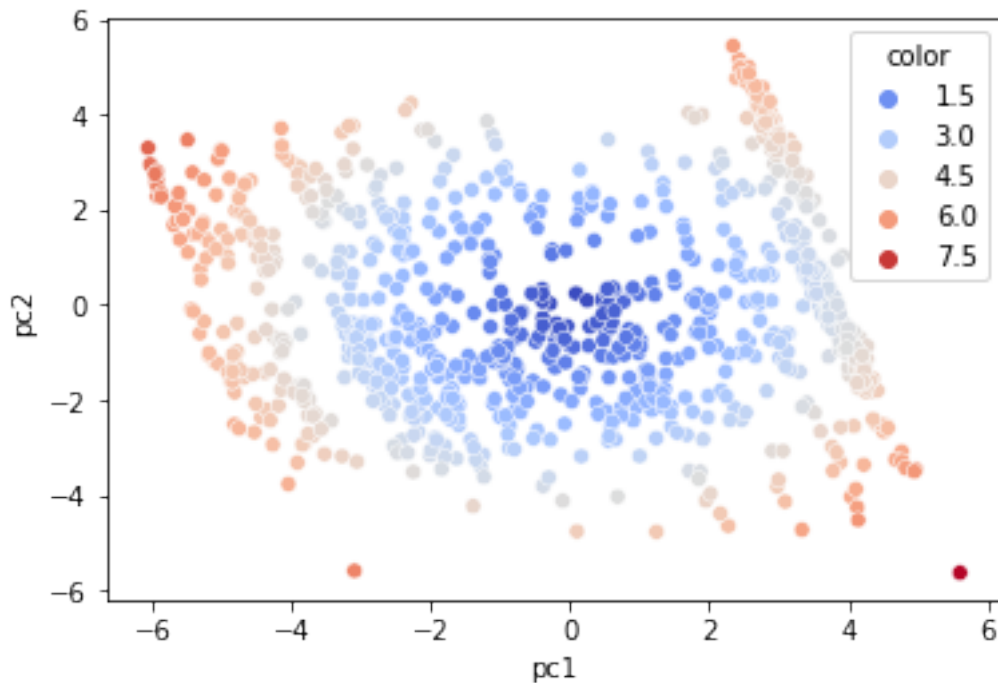


0.1 Question 2d

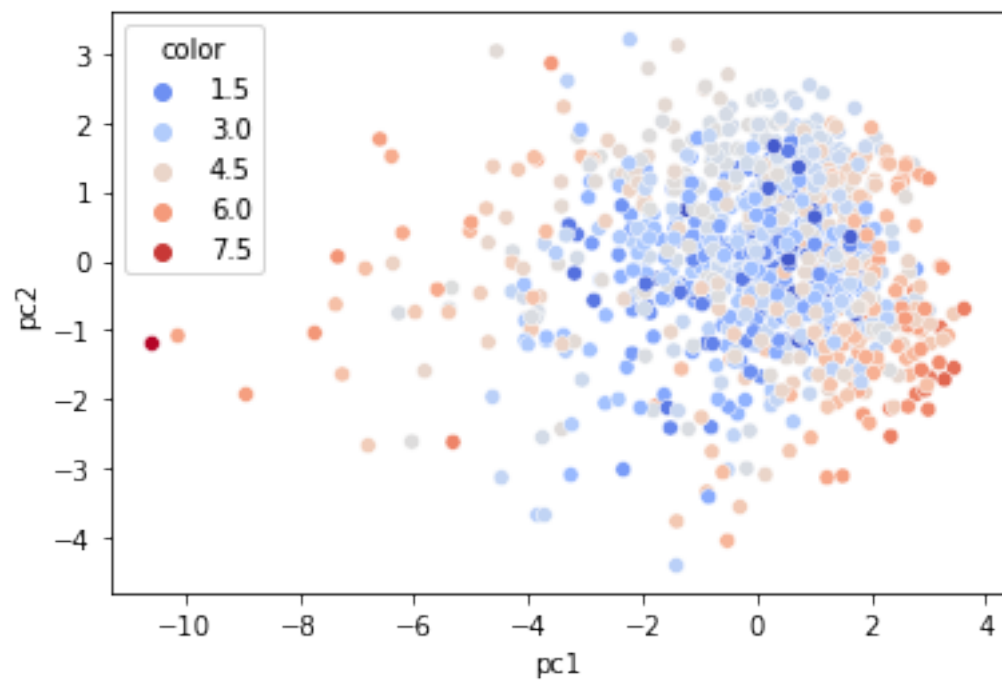
Create a 2D scatterplot of the first two principal components of `mid1_grades_centered_scaled`. Use `colorize_midterm_data` to add a color column to `mid1_1st_2_pcs`. Your code will be very similar to the code from problems 2a and 2b.

```
In [115]: sns.scatterplot(data = colorize_midterm_data(mid1_1st_2_pcs), x = "pc1", y = "pc2", hue = "color")
```



```
In [116]: u_2c, s_2c, vt_2c = np.linalg.svd(mid1_grades_centered_scaled, full_matrices=False)
pcas_2 = (u_2c*s_2c)
mid1_2d_1st_2_pcs = pd.DataFrame(data={'pc1':pcas_2[:,0], 'pc2':pcas_2[:,1]})

sns.scatterplot(data=colorize_midterm_data(mid1_2d_1st_2_pcs), x='pc1',y='pc2',hue='color',pa
```



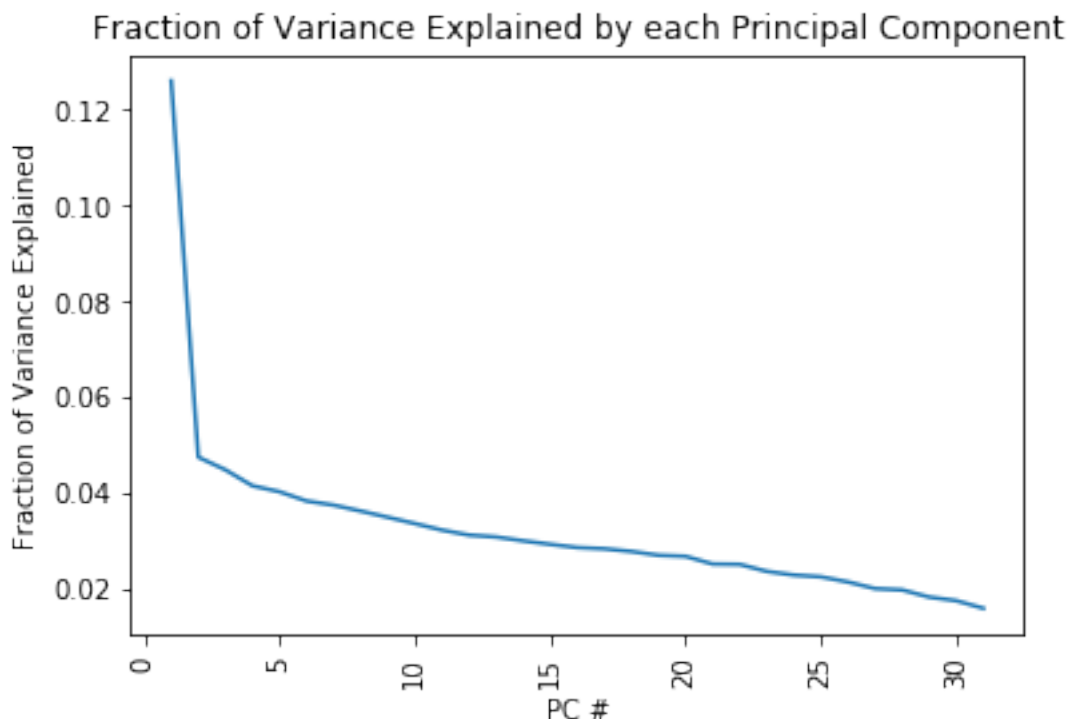
0.2 Question 2e

If you compute the fraction of the variance captured by this 2D scatter plot, you'll see it's only 17%, roughly 12% by the 1st PC, and roughly 5% by the 2nd PC. **In the cell below, create a scree plot showing the fraction of the variance explained by each principle component using the data from 2d.**

Informally, we can say that our midterm scores matrix has a high rank. More formally, we can say that a rank 2 approximation only captures a small fraction of the variance, and thus the data are not particularly amenable to 2D PCA scatterplotting.

```
In [117]: plt.plot(np.arange(1, 32, 1), s_2c**2 / sum(s_2c**2));  
plt.xticks(np.arange(0, 32, 5), np.arange(0, 32, 5));  
plt.xticks(rotation=90)  
plt.xlabel('PC #');  
plt.ylabel('Fraction of Variance Explained');  
plt.title('Fraction of Variance Explained by each Principal Component')
```

```
Out[117]: Text(0.5, 1.0, 'Fraction of Variance Explained by each Principal Component')
```



Unfortunately, we have two problems:

1. There is a lot of overplotting, with only 27 distinct dots. This means that at least some states voted exactly alike in these elections.
2. We don't know which state is which because the points are unlabeled.

Let's start by addressing problem 1.

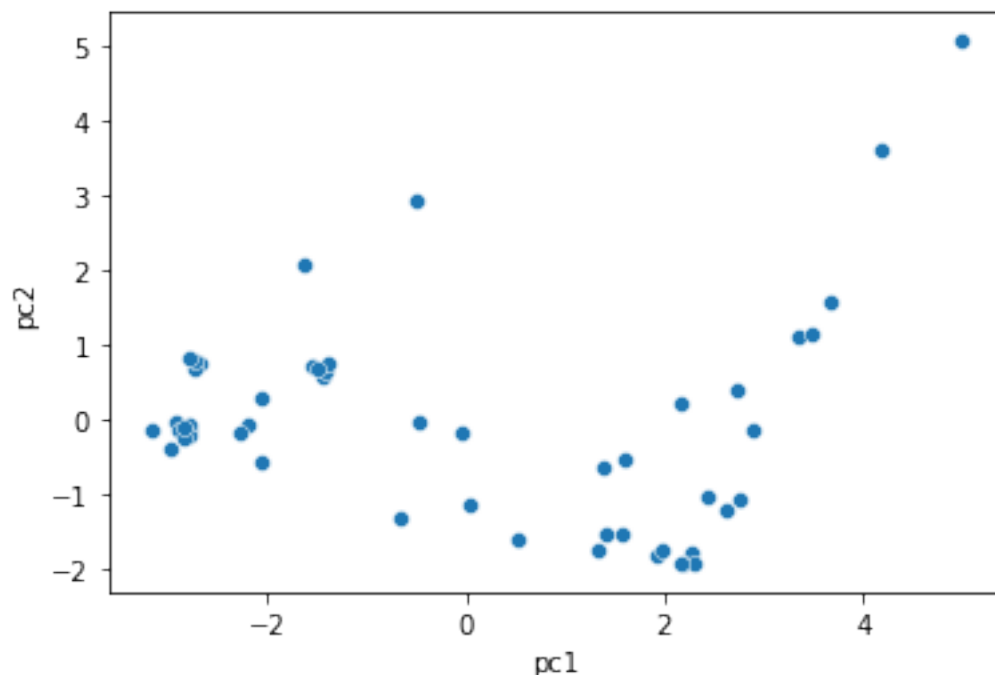
In the cell below, create a new dataframe `first_2_pcs_jittered` with a small amount of random noise added to each principal component. In this same cell, create a scatterplot.

The amount of noise you add should not significantly affect the appearance of the plot, it should simply serve to separate overlapping observations. Don't get caught up on the exact details of your noise generation, it's fine as long as your plot looks roughly the same as the original scatterplot.

Hint: See the pairplot from the intro to question 2 for an example of how to introduce noise.

```
In [129]: first_2_pcs_jittered = first_2_pcs + np.random.normal(0, 0.1, (len(first_2_pcs), 2))  
          sns.scatterplot(data = first_2_pcs_jittered, x = "pc1", y = "pc2")
```

```
Out[129]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7f584951f0>
```



Give an example of a cluster of states that vote a similar way. Does the composition of this cluster surprise you? If you're not familiar with U.S. politics, it's fine to just say 'No, I'm not surprised because I don't know anything about U.S. politics.'

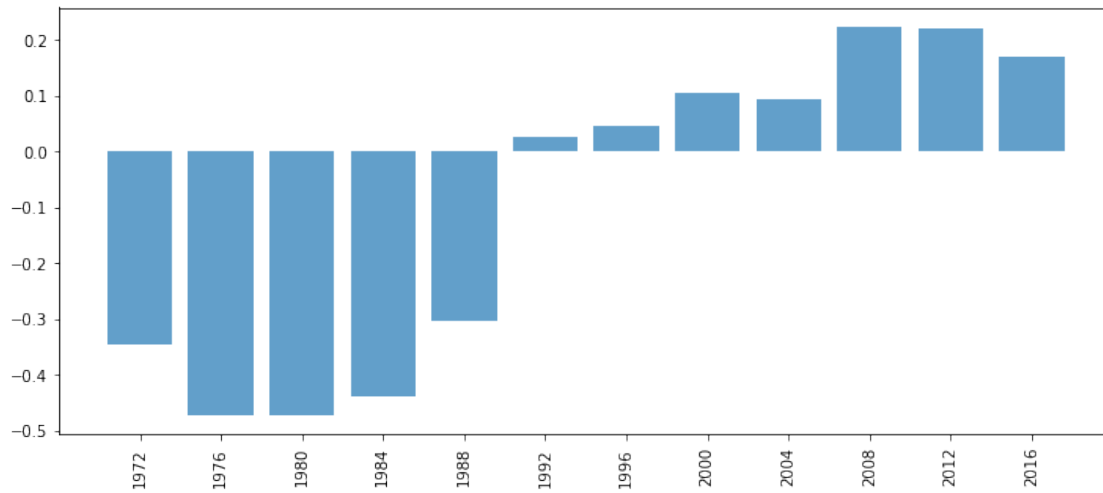
No I'm not surprised because I do not know anything about US politics.

In the cell below, write down anything interesting that you observe by looking at this plot. You will get credit for this as long as you write something reasonable that you can take away from the plot.

Certain clusters of states seen on the plot are geographically located close to each other as well (Massachusetts, Rhode Island, Maryland). However, Washington DC's voting pattern is quite different from the rest of the states.

In the cell below, plot the the 2nd row of V^T .

```
In [132]: plt.figure(figsize=(12, 5))
          plot_pc(list(df_1972_to_2016.columns), vt_q3, 1);
```



0.3 Question 3i

Using your plots from question 3h as well as the original table, give a description of what it means to have a relatively large positive value for **pc1** (right side of the 2D scatter plot), and what it means to have a relatively large positive value for **pc2** (top side of the 2D scatter plot).

In other words, what is generally true about a state with relatively large positive value for **pc1**? For a large positive value for **pc2**?

Note: **pc2** is pretty hard to interpret, and the staff doesn't really have a consensus on what it means either. We'll be nice when grading.

Note: Principal components beyond the first are often hard to interpret (but not always; see question 1 earlier in this homework).

I had to look into the election results for each state and compare that with their position on the plots in 3h and what I gathered was that the states with a higher value for **pc1** had a smaller probability for voting for the Republican party and vice versa if the **pc1** value is small the state was more likely to vote for the Republican party. The reason this is inversed is due to the fact that the **pc1** values are less than 0.

For **pc2**, the interpretation was much harder since there are both positive and negative values. I could guess that there is a direct relation between the **pc2** value and the probability that the state as a whole consistently chooses the same party and does not switch frequently. Consequently, the smaller **pc2** value indicates a state might be more likely to switch.

0.4 Question 3j

To get a better sense of whether our 2D scatterplot captures the whole story, create a scree plot for this data. On the y-axis plot the fraction of the total variance captured by the i th principal component. You should see that the first two principal components capture much more of the variance than we were able to capture when using the Data 100 Midterm 1 data. It is partially for this reason that the 2D scatter plot was so much more useful for this dataset.

Hint: Your code will be very similar to the scree plot from problem 1d. Be sure to label your axes appropriately!

```
In [134]: plt.plot(np.arange(1,13), s_3e**2 / sum(s_3e**2));  
          plt.xticks(np.arange(1,13),np.arange(1,13));  
          plt.xlabel('PC #');  
          plt.ylabel('Fraction of Variance');  
          plt.title('Fraction of Variance Explained by each Principal Component')
```

```
Out[134]: Text(0.5, 1.0, 'Fraction of Variance Explained by each Principal Component')
```

