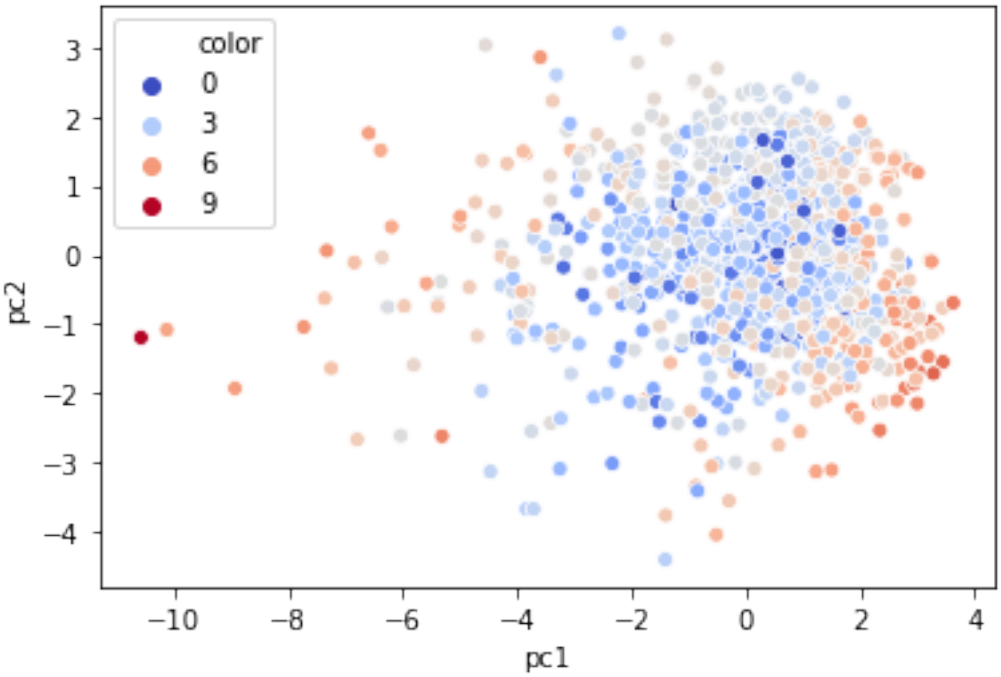
0.1 Question 2d

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_2d_1st_2_pcs`. Your code will be very similar to the code from problems 2a and 2b.

In [86]:
```
u_2d, s_2d, vt_2d = np.linalg.svd(mid1_grades_centered_scaled, full_matrices = False)

mid1_2d_1st_2_pcs = (mid1_grades_centered_scaled @ vt_2d.T).iloc[:, :2].rename(columns={0 : 'p
sns.scatterplot(data=colorize_midterm_data(mid1_2d_1st_2_pcs), x='pc1', y='pc2', hue='color', p
```

Out[86]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f04acaa9fd0>
```



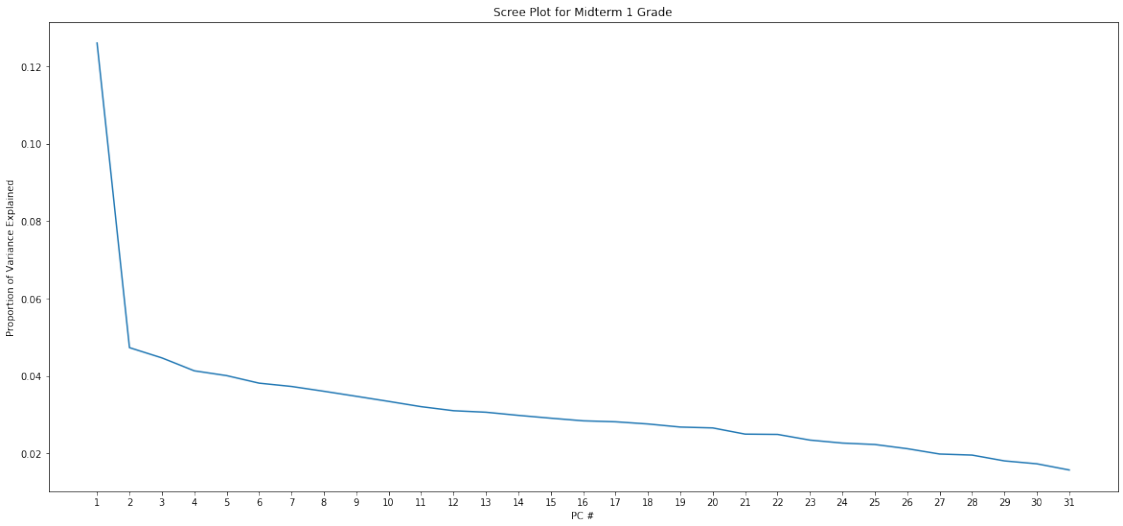
0.2 Question 2e

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 PC #i 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 [143]: plt.figure(figsize= (20, 9))
          plt.plot(np.arange(1, 32), s_2d**2 / sum(s_2d**2))
          plt.xticks(np.arange(1, 32), np.arange(1, 32))
          plt.xlabel('PC #')
          plt.ylabel('Proportion of Variance Explained')
          plt.title('Scree Plot for Midterm 1 Grade')
```

```
Out[143]: Text(0.5, 1.0, 'Scree Plot for Midterm 1 Grade')
```



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.

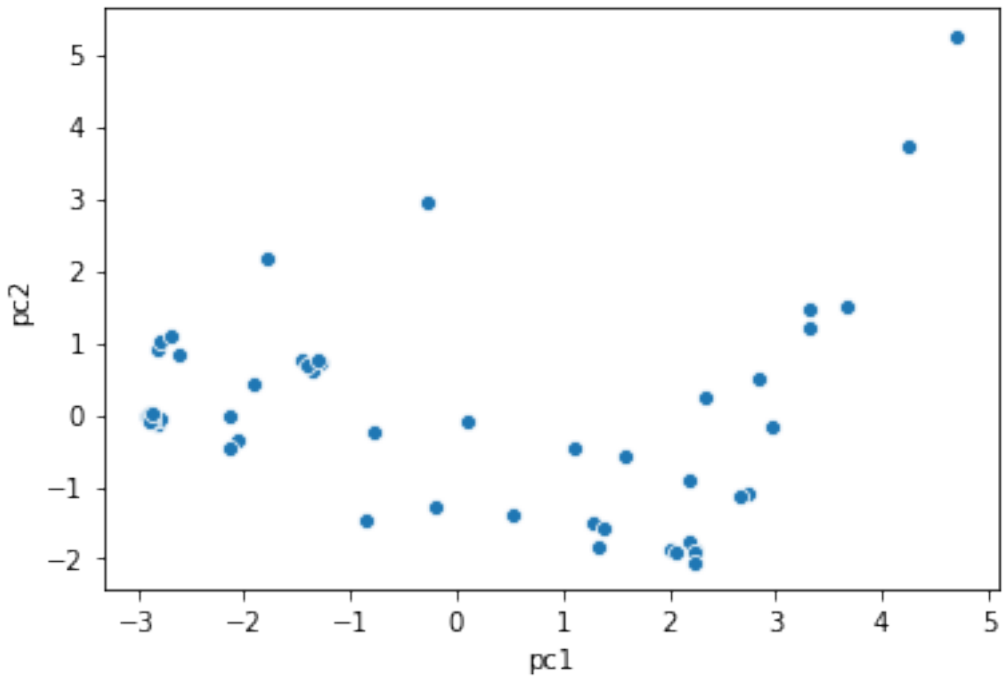
In the cell below, create a new dataframe `first_2_pcs_jittered` with a small amount of random noise added to each principal component. In this same cell, create a scatterplot.

The amount of noise you add should not significantly affect the appearance of the plot, it should simply serve to separate overlapping observations.

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

```
In [134]: first_2_pcs_jittered = first_2_pcs + np.random.normal(0, 0.1, size = (len(first_2_pcs), 2))
          sns.scatterplot(data = first_2_pcs_jittered, x = 'pc1', y = 'pc2')
```

```
Out[134]: <matplotlib.axes._subplots.AxesSubplot at 0x7f04a5bc9210>
```



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.'.

One cluster that votes in the same way is at -2, 2 and has the states Illinois, Connecticut, Maine, Vermont, California, and New Jersey. These states always tend to vote Democrat so this cluster doesnt surprise me.

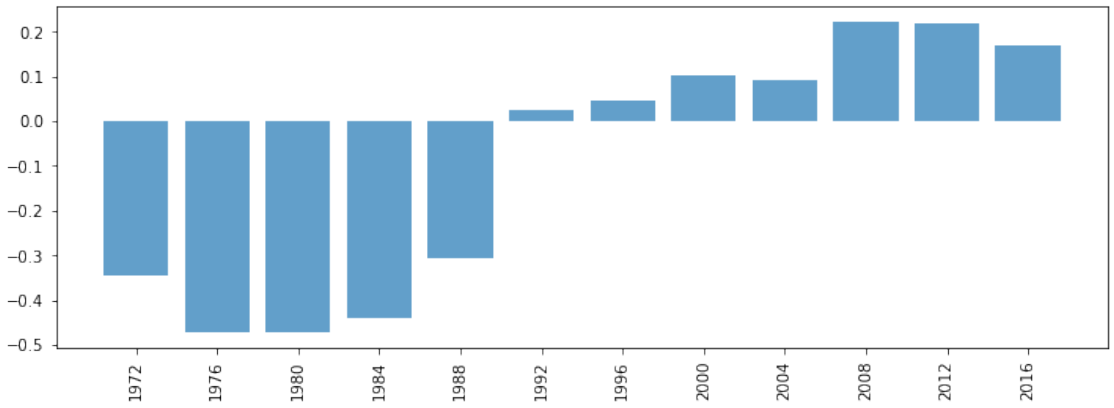
In the cell below, write down anything interesting that you observe by looking at this plot. You will get credit for this as long as you write something reasonable that you can take away from the plot.

Swing states like Ohio are near (0, 0). This is interesting as it isn't very well captured by either principal component, and also lies halfway between the cluster of Democratic States and the cluster of Republican voting states. This is perhaps due to its inconsistent voting patterns as states like Ohio tend to vote differently in different years, alternating between Democrat and Republican.

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

Hint: You are just copying and pasting code from the cell above and then changing one number.

```
In [140]: with plt.rc_context({"figure.figsize": (12, 4)}):
              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).

A large pc1 value indicates that a state tends to vote Democrat. pc 2 values are a little hareder to interpret. After looking into some states with higher pc2 values. It seems like they are primarily states that used to vote Republican (prior to about 1992) and more recently tend to vote Democrat. This is reflected in the plot of pc 2 above as it seems that the sign associated with the contribution from the year of the elections changes at about 1992.

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 ith 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 [145]: plt.plot(np.arange(1, 13), s_q3**2 / sum(s_q3**2))
          plt.xticks(np.arange(1, 13), np.arange(1, 13))
          plt.xlabel('PC #')
          plt.ylabel("Proportion of Variance Explained")
          plt.title("Scree Plot for Election Data")
```

```
Out[145]: Text(0.5, 1.0, 'Scree Plot for Election Data')
```