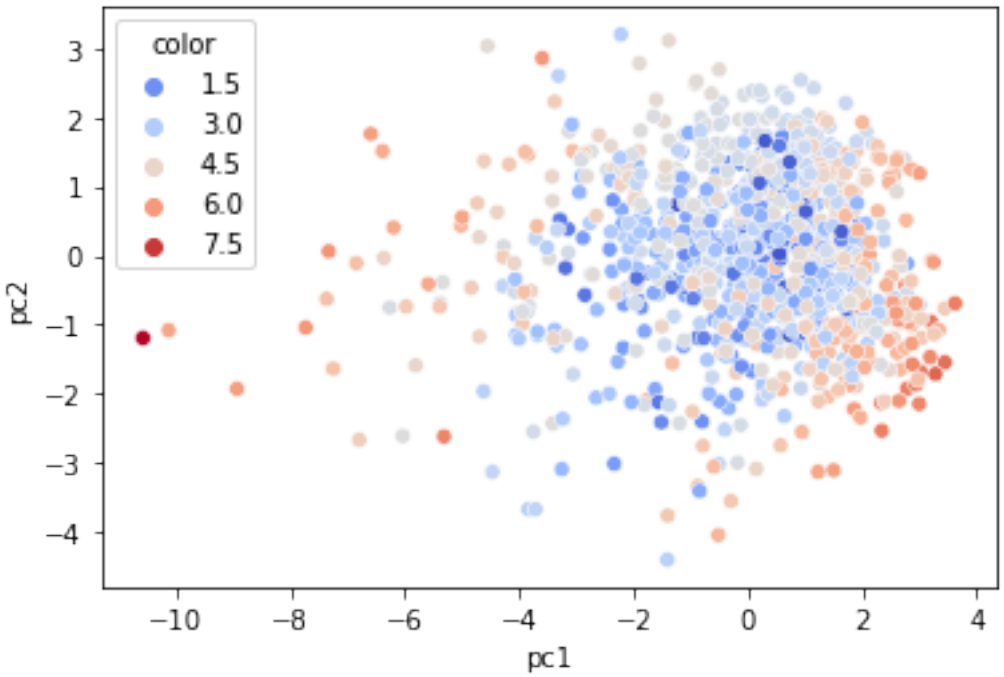
0.1 Question 2d

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 [37]:
```
u_mid1_centered_scaled, s_mid1_centered_scaled, vt_mid1_centered_scaled = np.linalg.svd(mid1_gr
mid1_1st_2_pcs = pd.DataFrame(u_mid1_centered_scaled * s_mid1_centered_scaled)[[0, 1]]
mid1_1st_2_pcs.rename(columns = {0: 'pc1', 1: 'pc2'}, inplace=True)
sns.scatterplot(data = colorize_midterm_data(mid1_1st_2_pcs), x = "pc1", y = "pc2", hue = "colo
```



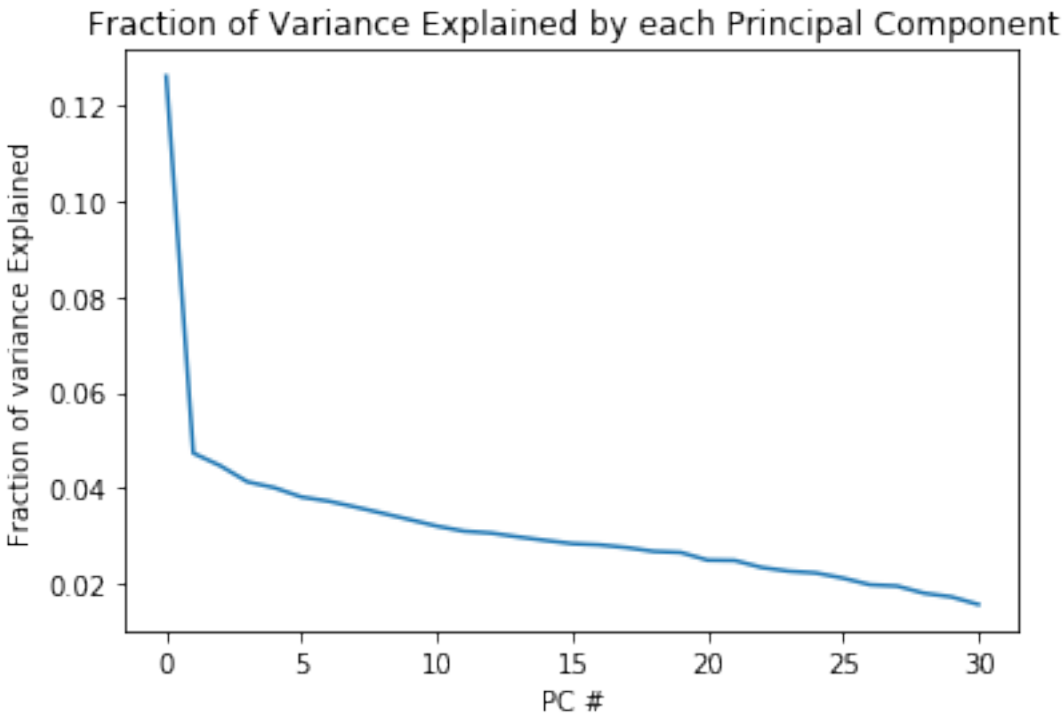
0.2 Question 2e

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 componant 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 [38]:
```
plt.plot(np.arange(len(s_mid1_centered_scaled)), (s_mid1_centered_scaled**2 / sum(s_mid1_cente
plt.xlabel('PC #');
plt.ylabel('Fraction of variance Explained');
plt.title('Fraction of Variance Explained by each Principal Component')
```

Out[38]:
```
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.

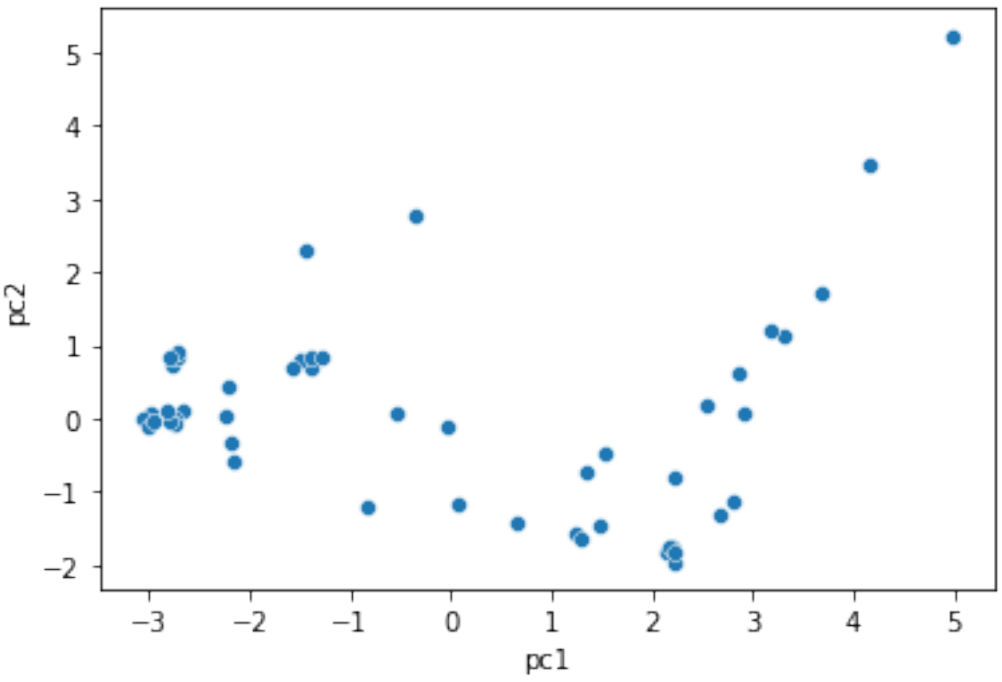
In the cell below, create a new dataframe `first_2_pcs_jittered` with a small amount of random noise added to each principal component. In this same cell, create a scatterplot.

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 [50]: 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[50]: <matplotlib.axes._subplots.AxesSubplot at 0x7efc70a80bb0>
```



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.'.

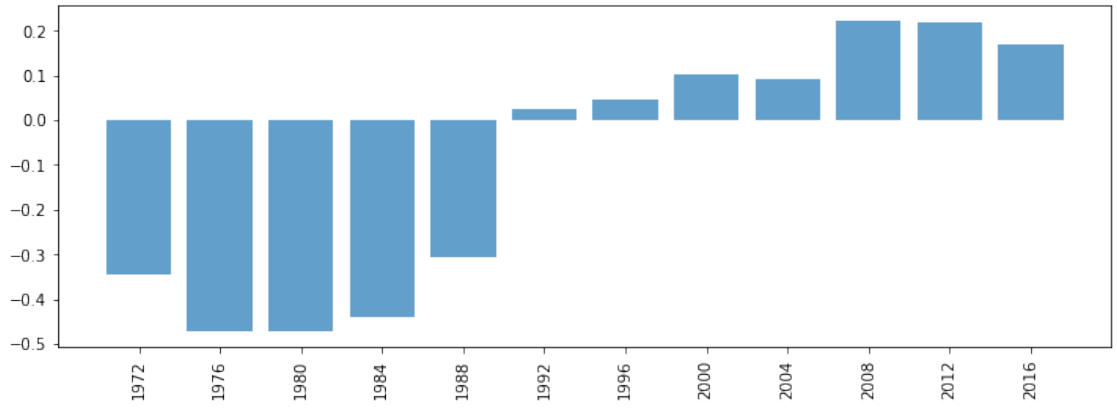
An example of a cluster of states that votes in a similar way are Illinois, California, New Jersey, Vermont, Connecticut. Although I am not too familiar with U.S. politics, this is not surprising because these states are generally viewed as largely liberal. On the other hand, another cluster of states that votes in a similar way are North Dakota, South Dakota, Wyoming, Oklahoma, Utah. This, too, is not surprising because these states are generally regarded as more conservative.

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.

It is interesting that New York does not lie in the cluster of the liberal states including llinois, California, New Jersey, Vermont, Connecticut, etc. I would expect New York to be in this cluster because New York City contains a high proportion of immigrants and is general regarded as one of the most cosmopolitan cities in the world, so I would expect it to lie in the cluster with the rest of the liberal states.

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

```
In [53]: plt.figure(figsize=(12, 4))
         plot_pc(list(df_1972_to_2016.columns), vt_df, 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).

To have a relatively large positive value for pc1 indicates that the state may be a Democratic state. This is because states like California and Illinois have a large positive value for pc1 (and a negative value for pc2).

To have a relative large positive value for pc2 is a lot harder to say since the Republican states are a lot more spread out. Consider the fact that although Minnesota has a high pc2, if we look at the election results from 1972 to 2016, it has been Democratic 11 times and Republican only 1 time (shown below). However, if we look at the cluster of the Republican states consisting of the Dakotas, Wyoming and Alaska, these have a negative value for pc1 and a relatively large positive value for pc1. So we can conclude that a large positive value for pc2 tends to lean more towards a Republican state.

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 [55]: plt.plot(np.arange(len(s_df)), (s_df**2 / sum(s_df**2)));
         plt.xlabel('PC #');
         plt.ylabel('Fraction of Variance Explained');
         plt.title('Fraction of Variance Explained by each Principal Component')
```

```
Out[55]: Text(0.5, 1.0, 'Fraction of Variance Explained by each Principal Component')
```