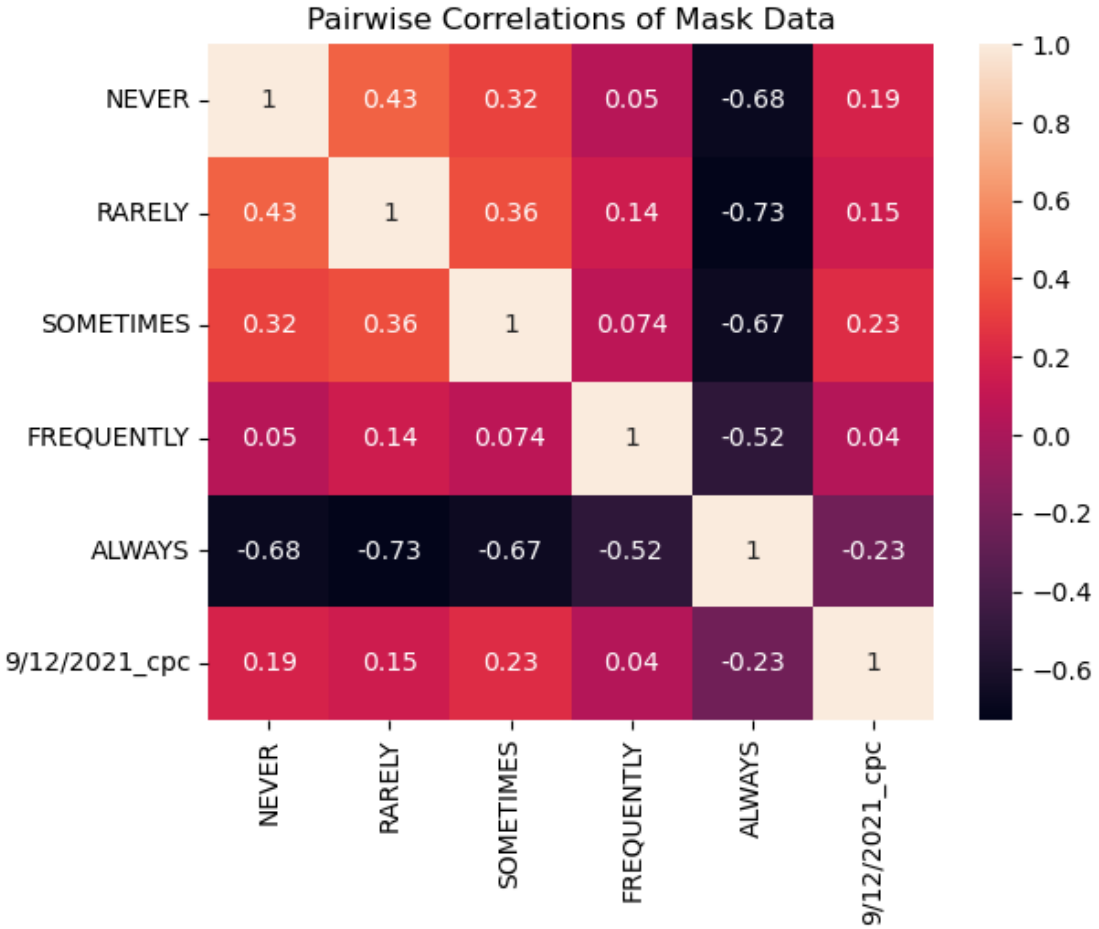
0.0.1 Question 1c

Our goal is to use county-wise mask usage data to predict the number of COVID-19 cases per capita on September 12th, 2021 (i.e., the column `9/12/2021_cpc`). But before modeling, let's do some EDA to explore the multicollinearality in these features, and then we will revisit this question in part 4.

Create a visualization that shows the pairwise correlation between each combination of columns in `mask_data`. For 2-D visualizations, consider Seaborn's heatmap. Remember to add a title to your plot.

Hint: You should be plotting 36 values corresponding to the pairwise correlations of the six columns in `mask_data`.

```
In [37]: sns.heatmap(data=mask_data.corr(), annot=True)
         plt.title('Pairwise Correlations of Mask Data');
```



0.0.2 Question 1d

(1) Describe the trends and takeaways visible in the visualization of pairwise correlations you plotted in Question 1c. Specifically, how does the correlation between pairs of features (i.e. mask usage) look like? How does the correlation between mask usage and cases per capita look like?

(2) If we are going to build a linear regression model (with an intercept term) using all five mask usage columns as features, then what problem will we encounter?

(1) As the frequency changes from never to always, the pairwise correlation decreases. The same is true for cases per capita paired with each frequency level: the correlation decreases from as frequency increases. Also observe that anything paired with 'ALWAYS' had a negative correlation (aside from it paired with itself).

(2) We can observe that the correlations are weak to moderate, so our regression model wouldn't be a very good fit. Since some correlations are stronger than others, including the weaker correlated mask usages would weaken our model.

0.0.3 Question 2b

To visualize the model performance from part (a), let's make the following two visualizations:

(1) The predicted values vs. observed values on the test set,

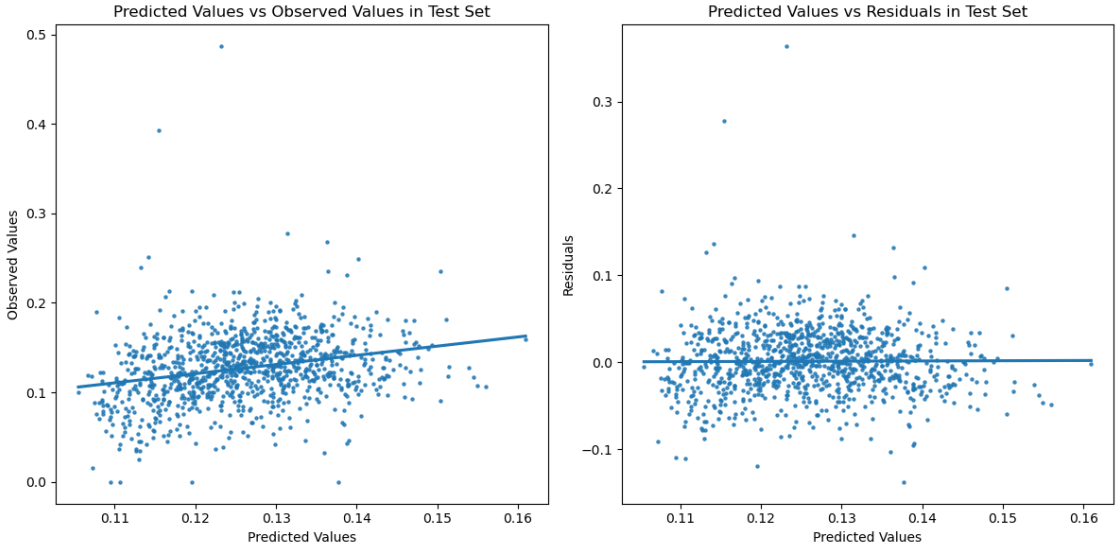
(2) The residuals plot. (Note: in multiple linear regression, the residual plot has predicted values vs. residuals)

Note: * We've used `plt.subplot` (documentation) so that you can view both visualizations side-by-side. For example, `plt.subplot(121)` sets the plottable area to the first column of a 1x2 plot grid; you can then call Matplotlib and Seaborn functions to plot that area, before the next `plt.subplot(122)` area is set. * Remember to add a guiding line to both plots where $\hat{y} = y$, i.e., where the residual is 0. * Please add descriptive titles and axis labels for your plots!

```
In [40]: plt.figure(figsize=(12,6))      # do not change this line
         plt.subplot(121)                # do not change this line
         # (1) predictions vs. observations
         sns.regplot(x=y_predicted, y=y_test, ci=0, scatter_kws={'s':5})
         sns.lineplot()
         plt.title("Predicted Values vs Observed Values in Test Set")
         plt.xlabel("Predicted Values")
         plt.ylabel("Observed Values")

         plt.subplot(122)               # do not change this line
         # (2) residual plot
         sns.regplot(x=y_predicted, y=y_test - y_predicted, ci=0, scatter_kws={'s':5})
         plt.title("Predicted Values vs Residuals in Test Set")
         plt.xlabel("Predicted Values")
         plt.ylabel("Residuals")

         plt.tight_layout()             # do not change this line
```


Predicted Values vs Observed Values in Test Set
Observed Values
Predicted Values
0.0
0.1
0.2
0.3
0.4
0.5
0.11
0.12
0.13
0.14
0.15
0.16
Predicted Values vs Residuals in Test Set
Residuals
Predicted Values
−0.1
0.0
0.1
0.2
0.3
0.11
0.12
0.13
0.14
0.15
0.16

0.0.4 Question 2c

Describe what the plots in part (b) indicate about this linear model.

The left plot indicates that there is a positive association between predicted values and observed values, but the correlation is weak since there are several points far from the regression line. The right plot shows a scatter shaped like a cloud, meaning that the linear model is a good fit for the data.

0.0.5 Question 3d

Interpret the confidence intervals above for each of the θ_i, where θ_0 is the intercept term and the remaining θ_i's are parameters corresponding to mask usage features. What does this indicate about our data and our model?

Describe a reason why this could be happening.

Hint: Take a look at the design matrix, heatmap, and response from Question 1!

We can see that 0 is included in all of our parameters' confidence intervals. In other words, it is possible each feature has no correlation with the covid cases per capita. Therefore we cannot reject the null hypothesis, which means that there is no correlation between X and y. The features themselves are correlated with each other, making predictions difficult.

0.0.6 Question 4b

Comment on the ratio `ratio`, which is the proportion of the expected square error on the data point captured by the model variance. Is the model variance the dominant term in the bias-variance decomposition? If not, what term(s) dominate the bias-variance decomposition?

Note: The Bias-Variance decomposition from lecture:

$$\text{model risk} = \sigma^2 + (\text{model bias})^2 + \text{model variance}$$

where σ^2 is the observation variance, or "irreducible error".

Since the ratio between model variance and model risk is very small, we can infer that the model variance does not have much of an impact on the model risk. Thus the observation variance and model bias dominate the bias-variance decomposition.

0.0.7 Question 4d

Propose a solution to reducing the mean square error using the insights gained from the bias-variance decomposition above.

Assume that the standard bias-variance decomposition used in lecture can be applied here.

In order to reduce the mean squared error (which is the model risk), we must reduce the squared model bias. To do this, we can increase the model complexity (add more features) without overfitting the data. We can avoid overfitting by optimizing the test error to a minimum value.