**1. Creating time series graphs with matplotlib**

Great work on the exercises! Now that we have a familiarity working with date and time values, we will leverage this knowledge to visualize trends in user data.

**2. Conversion rate over time**

Earlier we looked at user conversion rates across various cohorts. Now we will take this one step further and look at this value over time.

**3. Monitoring the impact of changes**

Most companies like ours are constantly making changes, and we may have some conceptualization that these things could help or hurt some of our KPIs. One way to monitor the impact of these changes, and the company health overall is track these metrics over time and evaluate them as we make the changes.

**4. Week one conversion rate by day**

Let’s return to working with week one conversion rate, looking now at a recently released subscription product from the past year. Repeating what we have done before, we can start by calculating our `sub\_time` column as we did at the end of chapter one.

**5. Conversion Rate by Day**

Next, we can group by the registration date values and then aggregate using our conversion rate function we wrote in chapter one. Looking at our results, we now have the week one conversion rate metric broken out by registration day.

**6. Plotting Daily Conversion Rate**

The best way to view this daily data is as a graph. In pandas, plotting the values in a dataFrame can be very easy. We can call the `plot()` method on our conversion rate dataFrame passing in the name of our x-axis and our y-axis, in this case our `reg\_date` and `sub\_time` columns respectively.

**7. Plotting Daily Conversion Rate**

Finally, we call the show method of the matplotlib library to reveal our graph. Interestingly, there is a dip in conversion rate near the end of December. Perhaps our services are more useful with the stress of the Holidays but these users do not need the service after this period and thus don't subscribe.

**8. Trends in different cohorts**

Further segmenting time series graphs can make them even more powerful. This can provide insight into whether these changes are impacting all users equally or different cohorts in different ways. For many businesses splitting by country and device are particularly useful as these groups can often have drastically different experiences with a product.

**9. Trends across time and user groups**

To finish, lets look to see if this holiday dip is occurring across all countries, as this could reasonably not be the case. To plot multiple series, we will need to reformat our data slightly. Here is a set of our data, with the additional grouping of country added. We will use the `pivot\_table()` method, to reformat our data.

**10. Conversion rate by country**

To plot this by country we need the data formatted such that there is a column with the conversion rate for each country, rather than the country value in a column of its own. First, we pass our dataFrame to the pivot\_table method. Next we set the `values` field to the column we want as our table values. In this case our conversion rate. Next we pass in the value we want to have as columns in our resulting table, which is the country. Finally, we specify the value we want as our rows as the `index`, which in this case is the `sub\_date`. Calling this function and reformatting slightly, we can see our transformed data.

**11. Plotting trends in different cohorts**

To finish we can call `plot()` on our resulting table. with the x-value as our dates and the y-axis as a list of our countries. The dip appears in all of our countries, though interestingly seems to be slightly more prominent in some countries than in others, though it is hard to tell directly. This graph will be useful for monitoring going forward and investigating this issue further.

**12. Let's practice!**

Great work, now let’s practice!