The data is available from InstaCart in the form of six comma separated variable (csv) files. Pandas easily reads this files into a DataFrame for data wrangling. A quick look confirms that the data is fairly clean. Only one column has missing values in the form of 'NaN', which will have to be addressed. It is in the column 'days since prior'. Referencing the InstaCart source data, this column captures days since the customer previous order. The column does not go above 30. It seems that InstaCart is only interested in orders with in a month time frame. A zero in this column indicates that customers made duplicate orders in one day. If we substitute zero for the NaN columns here these duplicate orders in one day would be lost. However, if the NaN is left, many computations with this column will not function properly. Therefore, it was chosen to lose this information and substitute zero for the NaN . Calculations with this column will then only capture purchases at least a day apart.

Now that all the missing values are addressed, the next step is getting the data in a tidy format, that is where rows represent observations and columns represent features. Pandas reads the six csv files into 6 different pandas data frames, which need to be combined to gather all the features we want to consider. The key that relates these relational databases is 'order_id', which we will use as the 'on' indicator for the pandas merge, concat, and join functions in combining these data frames. Special attention is required, since

an inner join works as an intersection in set logic and an outer join works as an union in set logic. Now that we have all the data together in a tidy format and all the original features (variables) together, data exploration and statistical analysis can begin.