In an effort to find the best model that had the greatest reduction in forecasting errors, we experimented with four different models: a multiplicative model based on day of week of the departure date, a multiplicative model based on number of days prior to the departure date, an additive model, and an additive model based on day of week of the departure date.

All four models begin by subtracting the departure date column from the booking date column to calculate how many days prior each booking date is to its respective departure date. The final demand per departure date is then found by looking at cumulative bookings where days prior equals zero.

The next model <mul_model_dow> was a multiplicative model with the historical booking rate based on what day of the week the departure date fell on. The historical booking rate was calculated by dividing cumulative bookings by final demand. From there, day of week of departure date was factored in by grouping this booking rate by days prior and day of week and then calculating the mean.

Then, this historical booking rate was merged with the validation data by days prior and day of week and the forecast column was created by dividing the cumulative bookings from the validation data by the historical booking rate.

The third model <mul_model_dop>, a multiplicative model based on days prior to departure date, is identical to the day of week multiplicative model, except when the booking rate is calculated. The mean of the booking rate is grouped by just days prior (instead of by day of week and days prior, as in the previous model).

The additive model <add_model> begins by calculating remaining tickets for each booking date in the training data by subtracting the cumulative bookings column from the final demand column.

Average demand for each day prior up to 28 days was then calculated by grouping the training data by the days prior and calculating the mean. Then a slice was taken of only the first 29 rows.

This average demand per days prior was merged with the validation data, and the forecast column was created by finding the sum of the column of remaining tickets to the column of mean cumulative bookings per day prior.

The fourth model <add_model_dow> is an additive model that calculated average demand by grouping the training data by both days prior and day of week and calculating the mean cumulative booking. This mean cumulative booking was then added to remaining demand to compute the forecast.

Finally, the overall function <airlineForecast> takes in all four functions and returns the MASE and forecast table for the model with the lowest MASE. The fourth model <mix_model> was the best, returning a MASE of 0.786. Compared to the naive model, our model reduced errors by 21.4 percent.