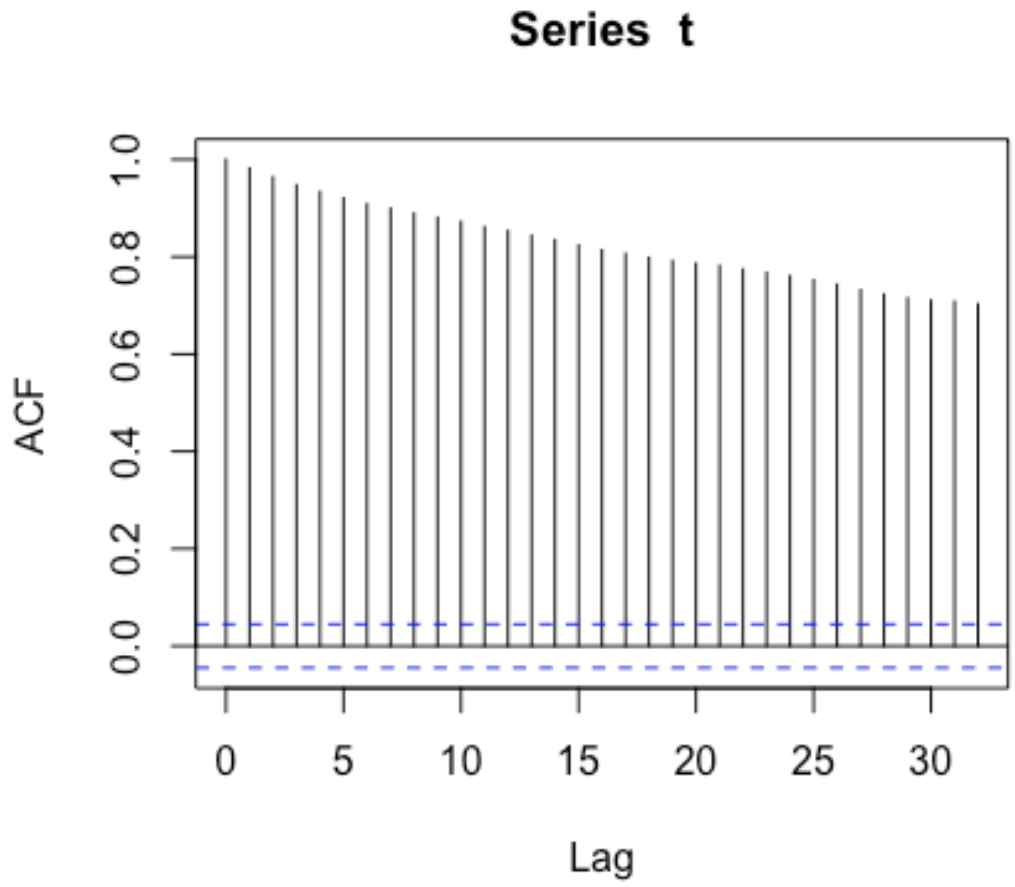
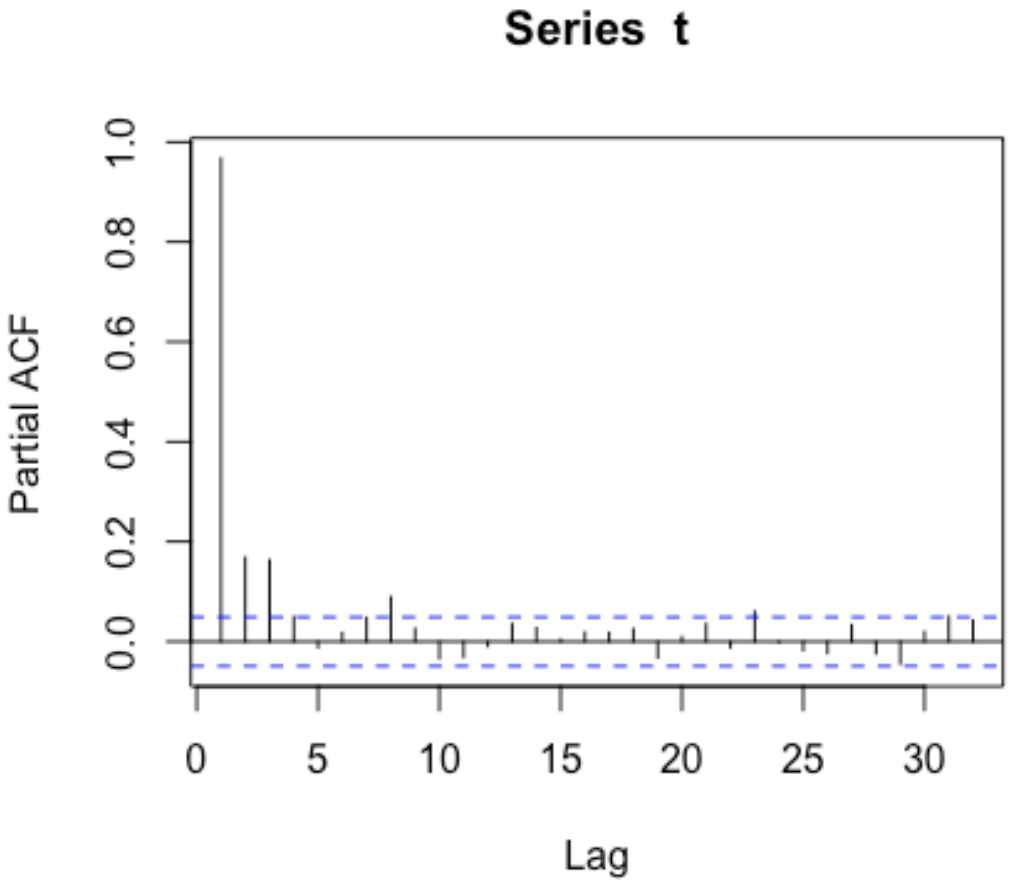
Below is a brief summary of what I have done with this dataset. It's the first time I have encountered a Time Series Forecasting problem. So firstly let me walk you through my understanding of this problem statement.

We have the dataset of 1529 unique stocks for a certain time frame of each stock. Price and Sales for a stock is for a day. So whatever model we built, we will run it on each and every stock separately and then built a data frame for next 180 days (according to the one given in submission file) containing 1529 * 180 rows.

I have used ARIMA modeling to forecast the stock sales and prices for next 6 months. Before using ARIMA function we need to check whether our time series is stationary or not. There are different tests that can use to check whether a given time series is stationary or not. Here I use Autocorrelation function (ACF), Partial autocorrelation function (PACF), Augmented Dickey–Fuller (ADF) test. If the time series is stationary, the ACF will drop to zero relatively quickly, while the ACF of non-stationary time series will decrease slowly.

When I run the acf() and pacf() function on Item_ID = 30288. Graphs below :






ACF and PACF test help us to identify very important parameters. When the values are closer to plus or minus one it indicates a strong correlation. If this value is around zero then we don't have much problem. But here the value decreases gradually indicating that the data is non-stationary. PACF cut off for threshold is after lag 2. So I used AR(2) model and put q = 0. We put q = 0 since we generally do not use both AR and MA terms. Some series are best fitted by AR terms while some by MA terms. If ACF dies gradually and PACF cut offs sharply after few lags then it is a signature that this series will be best fitted by AR terms.

Random Walk Trend where the value at Xt is equal to its value at t-1 plus some random movement. Differencing works here to make it stationary. Our stock data seems like a random walk trend. So we put the value of d = 1 in our arima model.

We can also use auto.arima() function to apply the best ARIMA model and then forecast prices and sales. This function try different model(s) with different order, and use the AIC criterion to search for a better model.