Matrix Estimation on Time Series data

Matrix estimation on Time Series Data:

In the week, Practical Data Science, we have learned about time-series data, that it is sequential data varying with time. We also learned different characteristics of time-series data like the trend, seasonality, etc. along with different forecasting techniques.

Now, we will see how we can build a recommendation system using the time-series data.

Till now we know that filling missing values in the interaction matrix is an important task for building the recommendation system. We use different methods for doing that. Let's see how they can be applied to the time-series data.

Suppose you are given the following time-series data:

X(1), X(2), ?, ?, X(5)………X(L)…….. X(T), X(T+1), X(T+2)….

The first question is, we make recommendation systems using a matrix and this is linear data. So, the first requirement is to convert this linear data into a matrix form. For this, we take some value L, say 3, and create a matrix in the following manner:

X(1)	?	.	.	.	.	.	.	X(T+1)
X(2)	X(5)	.	.	.	.	.	.	X(T+2)
?		.	.	.	.	.	X(T)	X(T+3)

We take the first L(here 3) values and put them as a column in the matrix and do this for all the data points. Looking at this matrix, now it looks similar to the matrix we are using till now to build the recommendation systems.

We can apply different matrix estimation methods to fill in the missing values and then forecast the next L values in the dataset **X(T+1), X(T+2),** and **X(T+3).** The good thing about this process is, unlike the traditional forecasting methods which assume stationarity, we don't have any assumptions here and we can apply this method for time-series forecasting.

How to choose L?

In an ideal condition, we take L as, **L~sqrt(T),** where T is the number of elements in the time-series data. For example, if the series contains 100 records then L should be ~10.