

## More Ways to Analyze Financial Data- Rolling values & Average

→ Add new sheet. Matrix ko drag krna. Rows me Date aur values me Total\_FTP add krni.

Report ko filter me drag krna aur P&L select krna ha.

→ What is the **problem in P&L** with this particular data?

When we have calculated for the period values, we know that this is the sum of all the P&L-related values for the month of January, February, and so on. The problem is that when we are performing analytics, we may like to smooth out any temporary fluctuations and exceptional values, or you can say any outliers. For example, you can see there are higher sales or maybe we can say higher profit in April only.

→ So, looking at one specific month, isolated from others, may lead to a little wrong judgment.

→ So what we can do instead we can work on a **90-day rolling or a three-month rolling value rather than looking at a standalone month value.**

→ Similarly, we can calculate a moving average based on the previous three months as well.

→ **So you can consider a three-month value, which will be called a rolling value.**

→ Rolling means when I'm going to march the value will be the sum of Jan, Feb, and March.

When I go to July, the value will be the sum of July, June, and May.

→ What is the **problem in the Balance Sheet** with this particular data?

Hm existing sheet ko duplicate kry gy, aur filter me se BALANCE SHEET select krna aur sath hi FTP ki jgha TTD drag krna.

→ Ab jo November me value a rhi wo start se November tk ha.

→ So the problem with these values are these values are calculated at the end of this particular period, that is month or quarter, or a year, whatever that is. And you would agree with me that the ending value of the period does not always reflect that you were maintaining the same level of assets or liabilities throughout the period.

→ So, for example, if I'm talking about cash and we say that the cash value at the end of the month, I mean the last day of the month, is \$1 million, it does not mean that you always have \$1 million in your month. It is possible that you only have \$10 in the month and all the remaining.

→ Tou ab hm yha p average values calculate kry gy. 2 ways:

1- simple avg formula, starting + Closing / 2

2- A better way is to calculate the balance for each day of the month and then take the average by summing all of them and dividing it by the total number of days in the month.

## P&L Rolling values and Moving average:

### 90 Days rolling value:

Total\_FTP\_90\_day\_Rolling = CALCULATE([Total\_FTP], DATESBETWEEN(tbl\_Calendar[Date], [MaxDate]-90, [MaxDate]))

hm isy matrix me drag kry gy. aur jan kaliye Ftp, and ye same ha. Feb me jan plus feb aur isi trah march me b. Mazeed smjhny kaliye hm isy days level tk expand kr k export to excel kr k b smjh skty. 90 days k mtlb zroori k k suppose April kaaliye wo 1<sup>st</sup> feb se start ho hmny jb excel me import kr k check kiya tou ye Jan 30 se start ho rha. Aur jan 30 – April 30 FTP ka sum aayega.

### 30 Days moving average based on 90 days Rolling values:

$30\_Day\_Moving\_Average = [Total\_FTP\_90\_day\_Rolling] / 90 * 30$

Year	Total_FTP	Total_FTP_90_day_Rolling	30_Day_Moving_Average	T
2018				
Qtr 1				
January	15710	15,710	5,237	
February	-20312	-4,602	-1,534	
March	-13696	-18,298	-6,099	
Qtr 2				
April	24313	-72,391	-24,130	
May	9221	27,613	9,204	
June	12555	46,089	15,363	
Qtr 3				
July	79435	107,407	35,802	
August	73838	165,828	55,276	
September	63934	217,207	72,402	
Qtr 4				
October	89786	238,700	79,567	
November	164511	318,231	106,077	
December	124561	379,132	126,377	
2019	1303147	700,639	233,546	
2020	1289945	824,428	274,809	

### 3 Month Rolling Value:

$Total\_FTP\_3\text{-Month}\_Rolling = CALCULATE([Total\_FTP], DATESINPERIOD(tbl_Calendar[Date], [MaxDate], -3, MONTH))$

**DATESINPERIOD**- instruct on which period we want to work. For example, in this, we mention the month that we want to work on at the month level.

How many intervals are you looking for = -3 Backward jaye aur Month pr kaam kry.

→ This value is three month rolling value. So even if I change the context from months to quarters, this value should remain three month rolling.

→ What I mean by that is if you are focusing on the June value for a moment, let us focus on

this value. 691598. If I change the context from months to quarters, this value, and of course, each quarter-end value, should remain the same.

## Monthly Moving Average:

**Total\_FTP\_Monthly\_Moving\_Average = [Total\_FTP\_3-Month\_Rolling] / 3**

Year	Total_FTP	Total_FTP_90_day_Rolling	30_Day_Moving_Average	Total_FTP_3-Month_Rolling	Total_FTP_Monthly_Moving_Average
2018					
Qtr 1					
January	218386	218,386	72,795	218,386	72,795
February	135042	353,428	117,809	353,428	117,809
March	181765	535,193	178,398	535,193	178,398
Qtr 2					
April	232865	549,672	183,224	549,672	183,224
May	235894	650,524	216,841	650,524	216,841
June	222839	691,598	230,533	691,598	230,533
Qtr 3					
July	372250	830,983	276,994	830,983	276,994
August	358720	953,809	317,936	953,809	317,936
September	344653	1,075,623	358,541	1,075,623	358,541
Qtr 4					
October	380572	1,083,945	361,315	1,083,945	361,315
November	501714	1,226,939	408,980	1,226,939	408,980
December	390728	1,273,014	424,338	1,273,014	424,338
2019	<b>5697845</b>	<b>2,026,612</b>	<b>675,537</b>	<b>2,026,612</b>	<b>675,537</b>
2020	<b>7835369</b>	<b>2,783,743</b>	<b>927,914</b>	<b>2,783,743</b>	<b>927,914</b>

## Average values for Balance Sheet:

Hmryo as is wkt jo TTD values ha hmny isny se starting/ opening month ki values calculate krni phr hm dono ko add kr 2 pe divide kry gy.

**Total\_TTD\_Opening = [Total\_TTD] - [Total\_FTP]**

→ Whatever the period and balance, deduct whatever happened in that period, so we will reach the opening value automatically.

→ Jo January ka closing balance hai wo Feb ka opening, aur isi trah jo Q1 ka closing balance wo Q2 ka opening.

**Total\_TTD\_Average = ([Total\_TTD\_Opening] + [Total\_TTD]) / 2**

Ab agr hmny specific value dekhni suppose Assets ki tou usy filter me drag krna.

Calculate an average balance for the balance sheet-related ledgers. But in that average, we want to consider every single day. My system should first calculate the closing balance for every single day. And then it should sum up all these individual values and divide with the number of days in the period. This is what we want to do.

## We cannot use the simple average function to achieve this. Why?

Because the simple average will actually sum all the transactions and divide by the number of transactions. So we don't want to work on the number of transactions.

We want to work with the number of days and not with the transaction balances.

→ I cannot refer my average function to my GL table where there are amount columns. This is because the amount is for the transaction. So I have to refer my average function to a table where there is already a day-level summary of closing balance available.

→ Now you know that that is not available in GL. GL is simply a list of all the transactions. So I first need to make a table like that. And don't worry, I'm not going to add a new table to my data model, and I'm not going to add a new table anywhere in this particular file. What I'm going to do is I'm going to create a virtual table within the measure so that my table is available to the measure.

**→ Write a summarize function to create this table virtually and Average Function:**

Average\_TDD\_DL = AVERAGEX(SUMMARIZE(tbl\_Calendar, tbl\_Calendar[Date]), [Total\_TTD])

→ We have to start with the average and it has to be average X because we want to work at the day level, before calculating the average. The first argument for this average function is a table.

→ But rather than giving any table, I want to say that please make your own table by using the summarize function. And in that summary, please consider the calendar table. And in that calendar table, you are grouped by the level of the data. The summary level of the data should be the calendar date. Now we can close the bracket for this summarize function.

→ If I go one step up this quarter, one average has been calculated as every single day in the quarter, that closing balance for every single day divided by that number of you know, once we calculate every single day balance, we sum them up and then we divide that with the number of days in the quarter.

→ And now let me show you that how this balance has been calculated manually as well, so that you can trust that what I am telling you is absolutely right. This balance has been calculated correctly for this.

→ We are going to take this data in Excel and let me show you the values manually. So let us save this file first. And once it is done, we can open the file.

And in this file, first of all, let me do the numbers formatting so that we can do work with it easily,

**→ So here you can see for this particular column, total to date column, we have the day level summary the closing balance mean for every single day.**

→ Now if you have to work for the month of January, we can. Select all the days from January and you can see your closing balance showing bottom. That is 3070258.

**→ Now, of course, this value won't match here. This is the average for various single day, but I have to compare that with average for the month so I can go back to my power bi file. And here in power BI you can see my balance is 3070258.**

→ So system is telling you power BI is telling you that your average balance for the month of January in 2008 is 3070258. And that is exactly what we can see here in Excel as well.

→ Now can we do that for the year as well?

Sure we can Let us do it for a year. And for example, we can work with 2019 data here.

Let me go back to my Excel and let me directly apply the filter of 2019.

So I would say, please only bring me 2019 data.

And now this is all the individual days of 2019, right from January to December.

Once we have selected all of them, the average balance is 8357436 and this is 8357436.

So we have correctly calculated average balance For the balance sheet.

And we can apply for any single account of the balance sheet considering the day level of the balance.

Year	Total_TTD	Average_TDD_DL
□ 2018		
□ Qtr 1		
⊕ January	6,166,510	6,094,067
⊕ February	6,131,378	6,269,839
⊕ March	6,156,378	6,191,926
⊕ Qtr 2	<b>6,321,282</b>	<b>6,297,386</b>
⊕ Qtr 3	<b>7,028,400</b>	<b>6,767,874</b>
⊕ Qtr 4	<b>7,751,604</b>	<b>7,323,420</b>
□ 2019		
⊕ Qtr 1	<b>15,868,050</b>	<b>15,935,837</b>
⊕ Qtr 2	<b>15,919,438</b>	<b>16,107,929</b>
⊕ Qtr 3	<b>17,154,358</b>	<b>16,669,037</b>
⊕ Qtr 4	<b>18,476,192</b>	<b>17,718,771</b>
□ 2020		
⊕ Qtr 1	<b>22,524,412</b>	<b>22,326,911</b>
⊕ Qtr 2	<b>22,476,588</b>	<b>22,283,024</b>
⊕ Qtr 3	<b>23,785,332</b>	<b>22,712,358</b>
⊕ Qtr 4	<b>24,640,002</b>	<b>23,815,322</b>

Year	Total_TTD	Total_TTD_Opening	Total_TTD_Average
<b>2018</b>			
<b>Qtr 1</b>			
January	6,166,510	0	3,083,255
February	6,131,378	6,166,510	6,148,944
March	6,156,378	6,131,378	6,143,878
<b>Qtr 2</b>			
April	6,183,306	6,156,378	6,169,842
May	6,246,062	6,183,306	6,214,684
June	6,321,282	6,246,062	6,283,672
<b>Qtr 3</b>			
July	6,375,894	6,321,282	6,348,588
August	6,805,190	6,375,894	6,590,542
September	7,028,400	6,805,190	6,916,795
<b>Qtr 4</b>			
October	7,055,198	7,028,400	7,041,799
November	7,488,472	7,055,198	7,271,835
December	7,751,604	7,488,472	7,620,038
<b>2019</b>			
<b>Qtr 1</b>	<b>15,868,050</b>	<b>7,751,604</b>	<b>11,809,827</b>
<b>Qtr 2</b>	<b>15,919,438</b>	<b>15,868,050</b>	<b>15,893,744</b>
<b>Qtr 3</b>	<b>17,154,358</b>	<b>15,919,438</b>	<b>16,536,898</b>
<b>Qtr 4</b>	<b>18,476,192</b>	<b>17,154,358</b>	<b>17,815,275</b>
<b>2020</b>	<b>24,640,002</b>	<b>18,476,192</b>	<b>21,558,097</b>

“Video me subclass sales select ki ha tb hi values lakho me a rhi”

