23-02-22

Data Analytics

CSPE 64

Cycle test - 1

106119100

Rajneesh

Question (1)

Month	1	2	3	4	5	6	7	8	9	10	11	12
Demand	12	15	19	23	27	30	32	33	37	41	49	58

(a)

4 month Moving avg. from month 4 to 12 (9 months) total

$m_4 = (23+19+15+12)/4 = 17.25$

~~$m_5 = 27+30+32+33+37$~~

$m_5 = (27+23+19+15)/4 = 21$

$m_6 = (30+27+23+19)/4 = 24.75$

$m_7 = (32+30+27+23)/4 = 28$

$m_8 = (33+32+30+27)/4 = 30.5$

$m_9 = (37+33+32+30)/4 = 33$

$m_{10} = (41+37+33+32)/4 = 35.75$

$m_{11} = (49+41+37+33)/4 = 40$

$m_{12} = (58+49+41+37)/4 = 46.25$

The forecast for month 13 is moving avg. for month before that.

i.e.

the moving avg. for month 12 = $M_{12} = 46.25$

Hence,

we cannot have fractional demand

so, the forecast for month 13 is 46

(b) Applying exponential smoothing with a smoothing constant 0.2 we get:

$M_1 = Y_1 = 12.$

$M_2 = 0.2Y_2 + 0.8M_1 = 0.2(15) + 0.8(12) = 12.600$

$M_3 = 0.2Y_3 + 0.8M_2 = 0.2(19) + 0.8(12.600) = 13.880$

$M_4 = 0.2Y_4 + 0.8(M_3) = 0.2(23) + 0.8(13.880) = 15.704$

$M_5 = 0.2Y_5 + 0.8M_4 = 0.2(27) + 0.8(15.704) = 17.963$

$M_6 = 0.2Y_6 + 0.8(M_5) = 0.2(30) + 0.8(17.963) = 20.370$

$M_7 = 0.2Y_7 + 0.8(M_6) = 0.2(32) + 0.8(20.370) = 22.696$

$M_8 = 0.2Y_8 + 0.8(M_7) = 0.2(33) + 0.8(22.696) = 24.757$

$M_9 = 0.2Y_9 + 0.8(M_8) = 0.2(37) + 0.8(24.757) = 27.206$

$M_{10} = 0.2Y_{10} + 0.8(M_9) = 0.2(41) + 0.8(27.206) = 29.96$

$M_{11} = 0.2Y_{11} + 0.8M_{10} = 0.2(49) + 0.8(29.965) = 33.77$

$M_{12} = 0.2Y_{12} + 0.8M_{11} = 0.2(58) + 0.8(33.77) = 38.61$

as, in previous part

forecast for month 13 is just the avg. for month 12 = M12

$= 38.618 = 39$ (as we can't have fractional demand)

(c)

Now,

to compare the two forecast we calculate the MSD (mean squared deviation)

for moving avg:

$$MSD_{mov\,avg} = \frac{(17.25-27)^2 + (21-30)^2 + (24.75-32)^2 + (28-33)^2 + (30.5-37)^2 + (39-41)^2 + (35.75-49)^2 + (40-58)^2}{8}$$

$MSD_{mov\,avg} = 107.43$

for exponetial. smoothed avg.

$$MSD_{expon.} = \frac{(12.6-19)^2 + (13.8-25)^2 + (15.7-23)^2 + (17.9-27)^2 + (20.3-30)^2 + (22.6-32)^2 + (24.7-33)^2 + (27.2-37)^2 + (29.9-41)^2 + (33.7-49)^2 + (33.7-58)^2}{11}$$

$MSD_{expo} = 176.05$

overall, then we see that four month moving avg appears to give ↑ best one month ahead forecast as it has lower MSD.

Hence,
we can prefer the forecast of 46. that has been produced by the four month moving avg.

also, other factors:
seasonal demand, advertising, price change general economic situation, new technology.

Question (2)

age values

13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30
33, 33, 35, 35, 35, 36, 40, 45, 46, 52, 70.

(a) five-number summary:

→ The five number summary of a distribution consist of the minimum value, first quartile median value, third quartile, and maximum value.

It provides a good summary of the shape of the distribution and for this data is: **13, 20, 25, 35, 70**

As,

→ first quartile (Q_1) = 25th percentile of the data

$= (25 * 26) / 100$

$= 6.5$

Q_1 = 7th value. = 20.

→ median value = $\frac{(n+1)/2 \text{ th value} + (n/2)^{th} \text{ value}}{2}$

as 26 values.

mid of $\{n/2 \text{ and } \frac{(n+1)}{2}\}$ $= (26+1)/2$ th value $= 14^{th}$ value $+ 13^{th}$ value

$= 14$th value $+ 13^{th}$ value.

Median = 25 $= \frac{25+25}{2}$

→ third quartile (Q_3) = 75th percentile of the data

$= \frac{75 \times 26}{100} = 19.5$

$= 20^{th}$ value.

$Q_3 = 35$

→ Maximum = 70

→ Minimum = 13

(b) first quantile (Q_1) = 25th percentile of data

$$= \frac{25 \times 26}{100} = 6.5$$

7th value $\Rightarrow$ $\boxed{Q_1 = 20}$

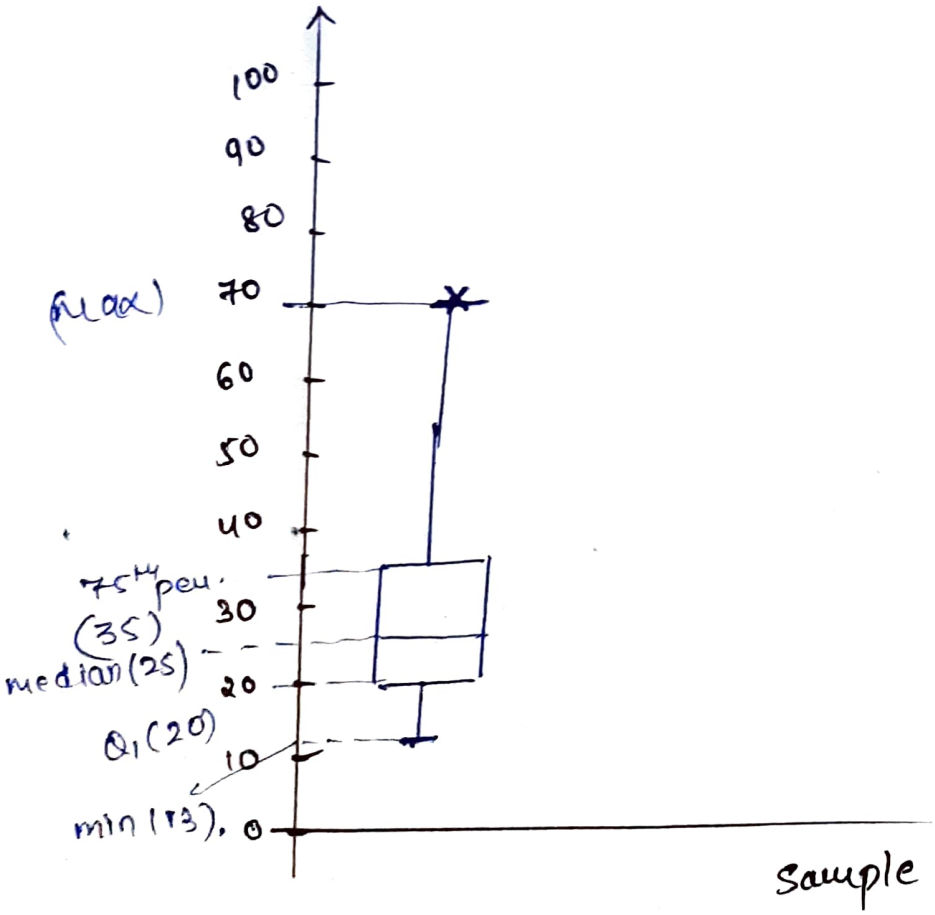
third Quantile (Q_3)

= 75th percentile of data

$$= \frac{75 \times 26}{100} = 19.5$$

20th value = $\boxed{Q_3 = 35}$

(c) Boxplot of data.



Question (5)

Stream → 3, 1, 4, 1, 3, 4, 2, 1, 2

frequency moment of a stream is calculated by using:

$$f_m = \sum_{i}^{N} f_i^m$$, m is order of moment and f is number of occurences of ith element

element	occurance	1st moment	2nd mome	3rd mo
1	3	3	9	27
2	2	2	4	8
3	2	2	4	8
4	2	2	4	8
		$f_m = 9$	$f_m = 21$	$f_m = 51$

Second moment $f_m = 21$

The third moment of this stream is 51

According to the Alon - Matian - Szegedy algorithm we have the following table

starting position i	Xi element	Xi value.
1	3	2
2	1	3
3	4	2
4	1	2
5	3	1
6	4	1
7	2	2
8	1	1
9	2	1

Question (4)

(a)

In this we need to find tail length and estimate of distinct element if hash function is provided

- According to FRAJOLET-MARTIN ALGORITM, we can estimate the

we can estimate the number of distinct elements by hashing the element of universal set to a bit string.

The length of the bit string must be sufficient.

Algo: Apply Hash function & convert to bit stream

→ count nubers of trailing zeros

→ find max of trailing zero.

→ answer will be 2^k distinct element

Hash value	$2x+1 \bmod 32$	$3x+7 \bmod 32$	$4x \bmod 32$
3	7 mod 32 = 7 = 00111	16 mod 32 = 16 10000	12 mod 32 = 12 01100
1	3 mod 32 = 3 = 00011	10 mod 32 = 10 01010	4 mod 32 = 4 00100
4	9 mod 32 = 9 01001	19 mod 32 = 19 10011	16 mod 32 = 16 10000
5	11 mod 32 = 11 01011	22 mod 32 = 22 10110	20 mod 32 = 20 10100
9	19 mod 32 = 19 10011	34 mod 32 = 2 00010	36 mod 32 = 4 00100
2	5 mod 32 5 00101	13 mod 32 = 13 01101	8 mod 32 = 8 01000
6	13 mod 32 = 01101	25 mod 32 11001	24 mod 32 = 24 11000

R = 0
(Because No string is ending with 0)

R = 4
(Hash value 3 is ending 4 0's)

R = 4.
(Hash value is ending with 4 0's)

(i) $2x+1 \bmod 32$: Resulting estimate

$2^R = 2^0 = 1$

(ii) $3x+7 \bmod 32$

Resulting estimate

$2^R = 2^4 = 16$

(iii) $4x \bmod 32$ = Resulting estimate

$= 2^R = 2^4 = 16.$

(6) total 4th bucket can be made using DGIM algo.

Step 1
Original:
Stream - 10010 11 0 111 01

Step (2)
bucket size 1 = $2^0 = 1$

First bucket = $2^0 = 1$ (Bucket = 1)

10010110111 0 (1) → first

step (3)
Next bit from right of size=1, $2^0 = 1$

100 10 11 0 11 (1) 0 (1)
second ↓ first

Stemp (4)
bucket size = 2, size $2^1 = 2$
Third bucket (11)

100 10 11 0 (11) (1) 0 (1)
third ↓ second ↓ first

Step (5)
bucket size = 4, $2^2 = 4$ only one way
fourth buck

(1001011) 0 (11) (1) 0 (1)
4th ↓ third ↓ 2nd ↓ first

Question (3)

Stream processor architecture for twitter data:

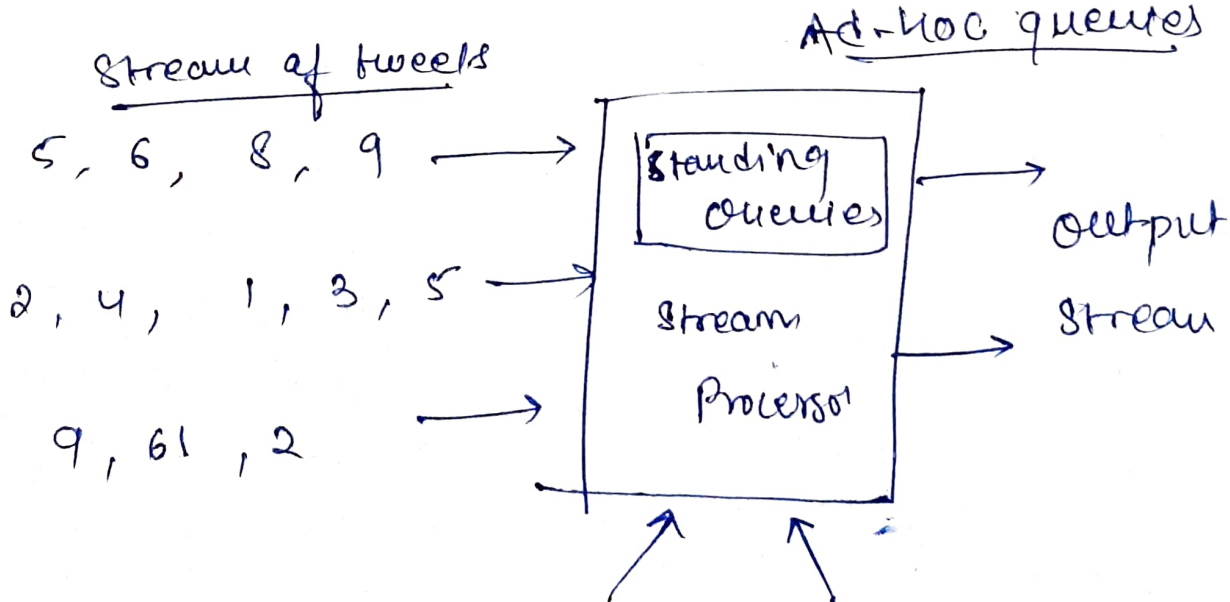
Twitter is a social media platform, billions of people tweet every day, on a regular basis.

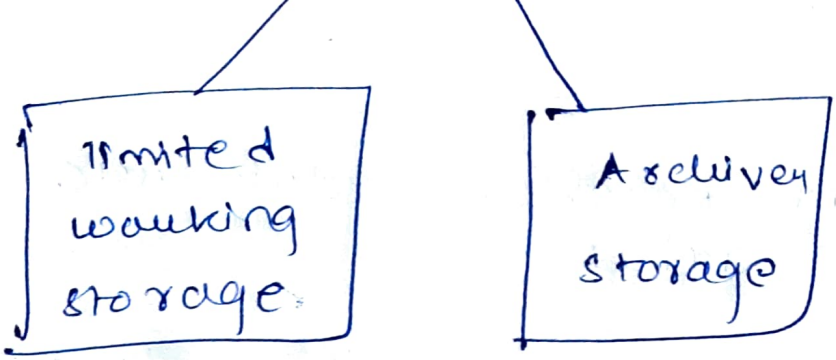
Due to large amount of data its difficult to process queries on stored data.

So, in order to process the billions of data. Twitter can not use normal DB.

Hence they use stream processor for this as a database management system

Example



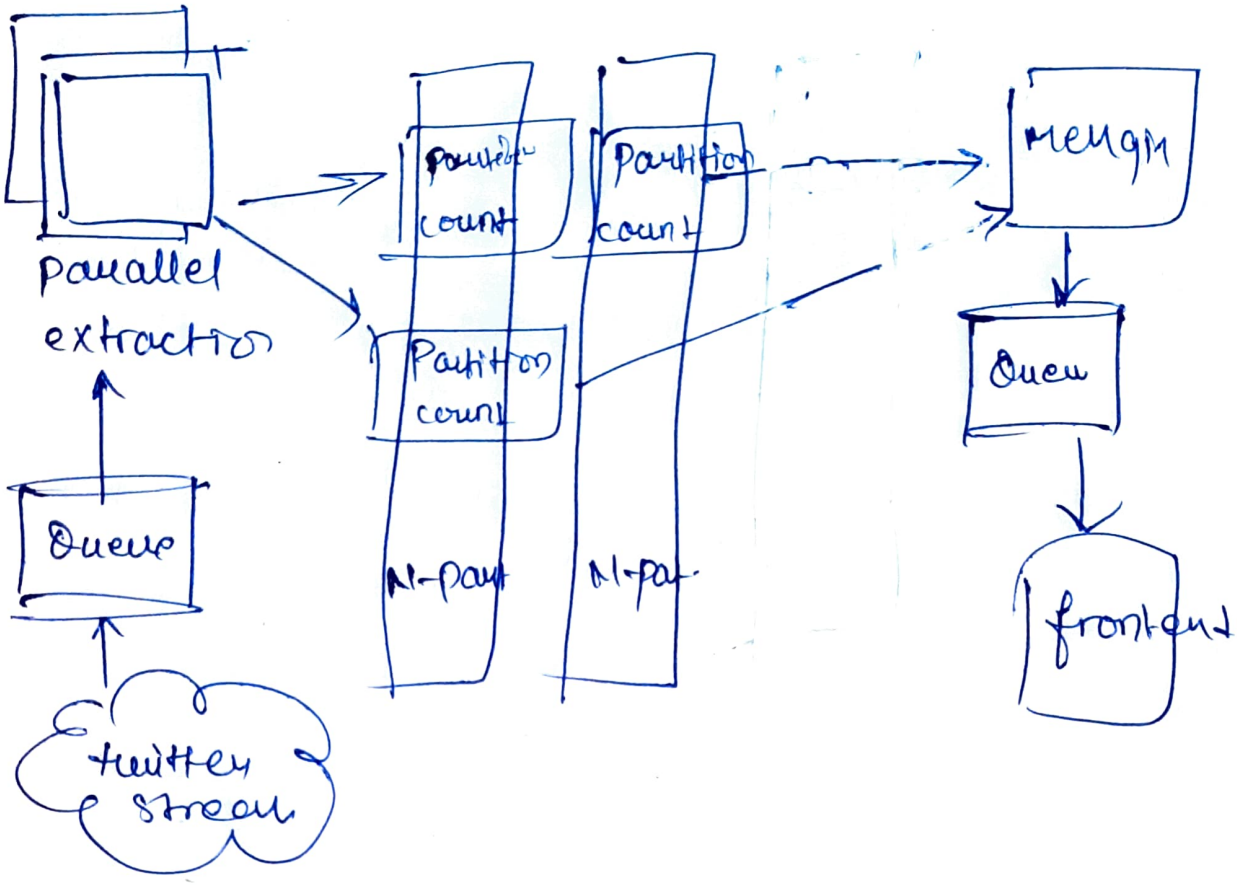


Sometimes when the tweets are a lot it is not possible to answer all queries, so data stored in large archival stage. and retrieving it from it is a time consuming process

The memory that is used by twitter to handle query is limited memory storage

queries are two type

- Ad-Hoc
  - ask once about the current state of the stream.
  either stream needs to be stored and element to answer
- standing
  - queries execute permanently in the system and produce output at appropriate time



for eg.

twitter requires to know no. of unique users in a month this adhoc query can be implemented if twitter stores data of all unique users in a sliding windows with appropriate time stamps.

—