

Manufacturing processes for any product is like putting together a puzzle. Products are pieced together step by step and it's important to keep a close eye on the process.

For this project, you're supporting a team that wants to improve the way they're monitoring and controlling a manufacturing process. The goal is to implement a more methodical approach known as statistical process control (SPC). SPC is an established strategy that uses data to determine whether the process is working well. Processes are only adjusted if measurements fall outside of an acceptable range.

This acceptable range is defined by an upper control limit (UCL) and a lower control limit (LCL), the formulas for which are:

$$ucl = avg_height + 3 * \frac{stddev_height}{\sqrt{5}}$$

$$lcl = avg_height - 3 * \frac{stddev_height}{\sqrt{5}}$$

Using SQL window functions, you'll analyze historical manufacturing data to define this acceptable range and identify any points in the process that fall outside of the range and therefore require adjustments. This will ensure a smooth running manufacturing process consistently making high-quality products.

The data

The data is available in the `manufacturing_parts` table which has the following fields:

- `item_no` : the item number
- `length` : the length of the item made
- `width` : the width of the item made
- `height` : the height of the item made
- `operator` : the operating machine

```
In [1]: SELECT *  
FROM manufacturing_parts
```

Out[1]:

	item_no	length	width	height	operator
0	1	102.67	49.53	19.69	Op-1
1	2	102.50	51.42	19.63	Op-1
2	3	95.37	52.25	21.51	Op-1
3	4	94.77	49.24	18.60	Op-1
4	5	104.26	47.90	19.46	Op-1
...
495	496	101.24	49.03	20.96	Op-20
496	497	98.37	52.12	19.68	Op-20
497	498	96.49	48.78	19.19	Op-20
498	499	94.16	48.39	21.60	Op-20
499	500	102.35	51.24	21.47	Op-20

500 rows x 5 columns

```

In [2]: -- Flag whether the height of a product is within the control limits
SELECT
    b.*,
    CASE
        WHEN
            b.height NOT BETWEEN b.lcl AND b.ucl
        THEN TRUE
        ELSE FALSE
    END as alert
FROM (
    SELECT
        a.*,
        a.avg_height + 3*a.stddev_height/SQRT(5) AS ucl,
        a.avg_height - 3*a.stddev_height/SQRT(5) AS lcl
    FROM (
        SELECT
            operator,
            ROW_NUMBER() OVER w ,
            height,
            AVG(height) OVER w AS avg_height,
            STDDEV(height) OVER w AS stddev_height
        FROM manufacturing_parts
        WINDOW w AS (
            PARTITION BY operator
            ORDER BY item_no
            ROWS BETWEEN 4 PRECEDING AND CURRENT ROW
        )
    ) AS a
    WHERE a.row_number >= 5
) AS b

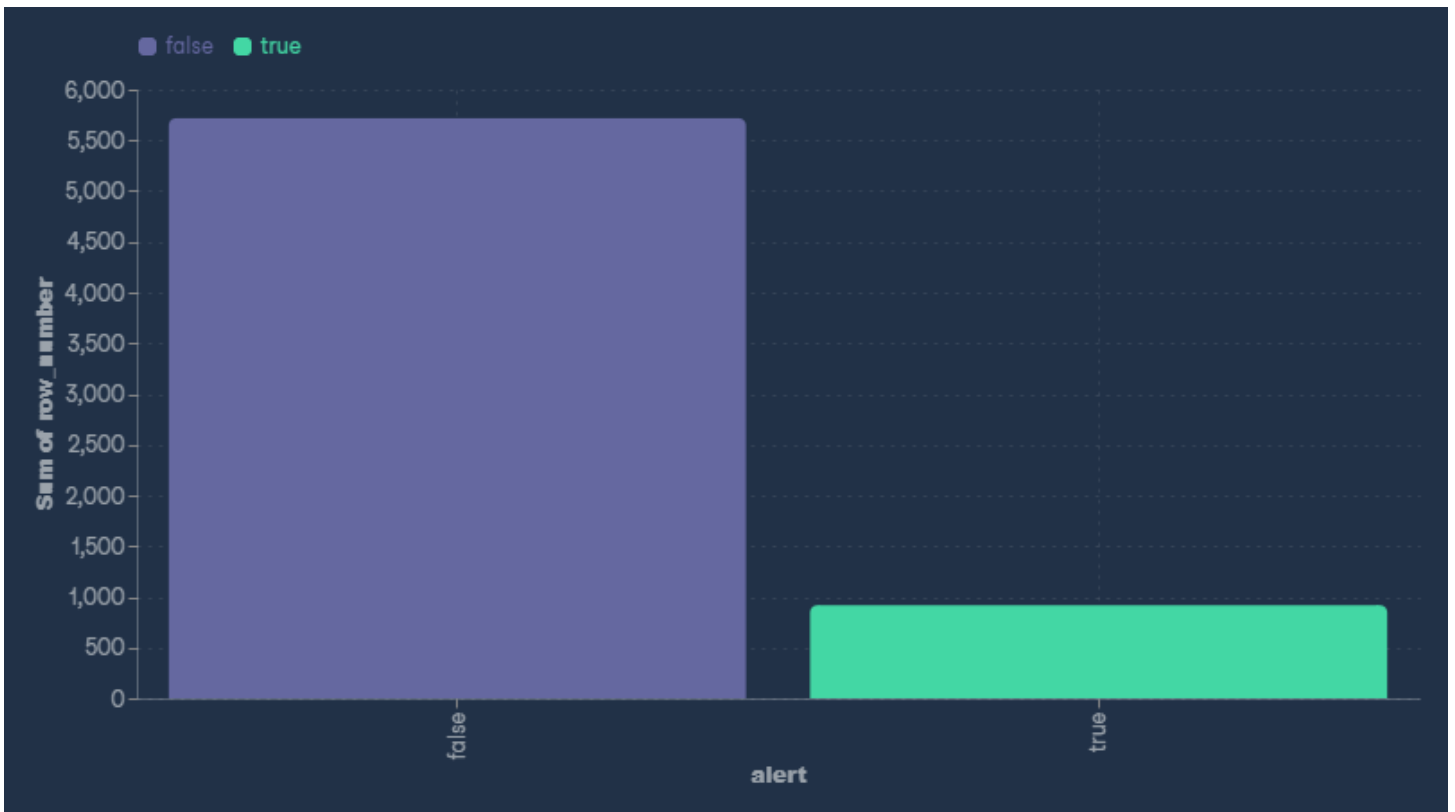
```

Out[2]:

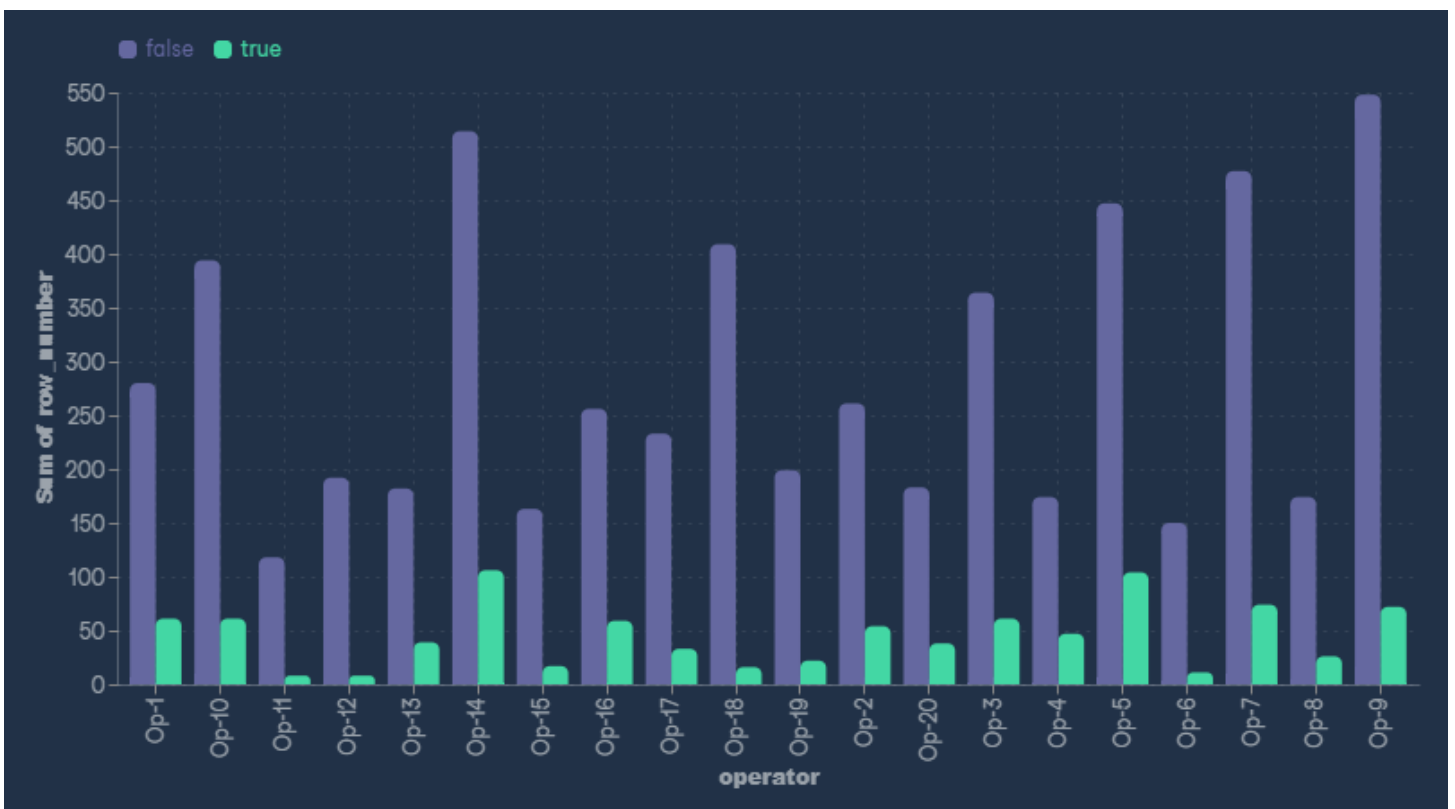
	operator	row_number	height	avg_height	stddev_height	ucl	lcl	alert
0	Op-1	5	19.46	19.778	1.062812	21.203912	18.352088	False
1	Op-1	6	20.36	19.912	1.090812	21.375477	18.448523	False
2	Op-1	7	20.22	20.030	1.084574	21.485108	18.574892	False
3	Op-1	8	21.03	19.934	0.931225	21.183369	18.684631	False
4	Op-1	9	19.78	20.170	0.598832	20.973418	19.366582	False
...
415	Op-9	31	19.01	18.904	0.203052	19.176422	18.631578	False
416	Op-9	32	18.57	18.864	0.250260	19.199759	18.528241	False
417	Op-9	33	20.91	19.266	0.952276	20.543613	17.988387	True
418	Op-9	34	21.24	19.678	1.291112	21.410208	17.945792	False
419	Op-9	35	21.66	20.278	1.392828	22.146675	18.409325	False

420 rows x 8 columns

In [1]: --The 16.03% of the productions are out of accepted range



In [2]: -- Production Not out of range (F) vs Production out of range (v) by Operator



In [4]: --Total production out of range by amount of row numbrs

