1. Data Cleaning

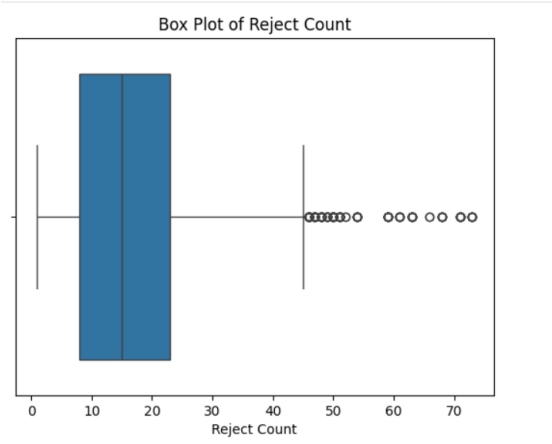
I started by importing three critical datasets: ProductionMetric, Quality, and DeviceProperty. These files contained key information on production outputs, downtime, reject counts, and machine details.

To ensure data integrity, I performed the following steps:
- I checked for missing values using `.isnull().sum()` and confirmed that null values were minimal.
- I identified and removed any duplicate records using `.duplicated().sum()`.
- I renamed columns like `deviceKey_x` to `deviceKey` after merging, to make the dataset more readable.
- I merged all three datasets using `prodmetric_stream_key` and `deviceKey`, enabling a unified analysis view.

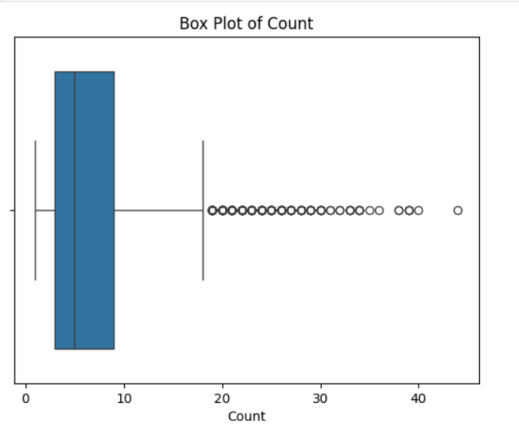
2. Outlier Detection

Next, I applied the IQR (Interquartile Range) method to detect outliers in the following features:
- reject_count
- good_count
- DefaultCycleTime

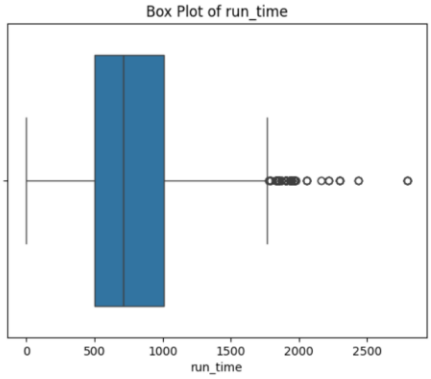
I used Seaborn to generate box plots, which helped me visually detect outliers and understand the variability in the data. For example:
- The `reject_count` box plot showed several extreme outliers that may indicate production issues.

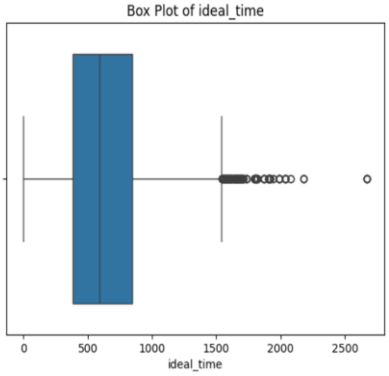


- The `good_count` plot revealed high-output anomalies.



- `DefaultCycleTime` outliers suggested misconfigured machines or incorrect cycle timings. But there are no outliers in that.





3. Inconsistency Checks

I implemented logic-based data validation to catch inconsistencies:
- I found rows where `unplanned_stop_time` was greater than `run_time`, which shouldn't be possible.
- I flagged cases where `run_time` was positive, but both `good_count` and `reject_count` were zero.
- I also identified entries with negative values in `run_time`, `cycle_time`, and output fields, which indicate data logging or sensor issues.

2. Statistical & Downtime Analysis

2.1 Downtime Breakdown

I calculated total downtime across all production lines to understand how much was planned versus unplanned:
- Unplanned Downtime: ~47.3%
- Planned Downtime: ~52.7%

This shows that disruptions from unexpected events are almost as frequent as scheduled pauses.

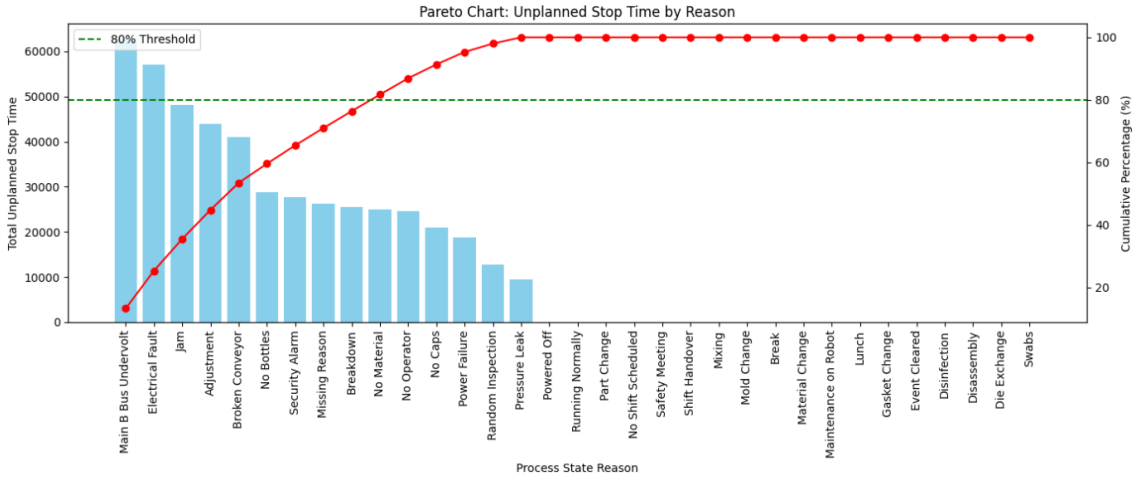
2.2 Downtime by Line

I grouped data by deviceKey to summarize downtime statistics per production line:
- Line1: Mean = 51.6 mins, Max = 640.2 mins, Median = 0.0 mins
- Line2: Mean = 49.2 mins, Max = 809.2 mins, Median = 0.0 mins
- Line3: Mean = 41.3 mins, Max = 439.7 mins, Median = 0.0 mins
- Line4: Mean = 49.0 mins, Max = 468.4 mins, Median = 0.0 mins

These statistics helped identify that Line2 has the most severe high-downtime events.

2.3 Root Cause Analysis with Pareto Chart

Using a Pareto chart, I analyzed which issues contributed most to unplanned downtime. The chart showed that a few major causes (like Sensor Failure and Material Jam) accounted for the majority of delays. This follows the 80/20 principle—around 80% of unplanned time comes from the top 20% of issues.



2.4 Reject Rate

To measure quality performance, I calculated the reject rate:
Reject Rate = total_rejects / (total_rejects + total_goods) = ~7.6%
This indicates that nearly 1 in every 13 products is rejected, a key area to target for improvement.

3 Shift and Team Performance Comparison

I analyzed and compared the average unplanned downtime and reject rate across different production shifts and teams. This helped in identifying performance patterns that are influenced by the time of day or team management.

Shift-Wise Metrics:

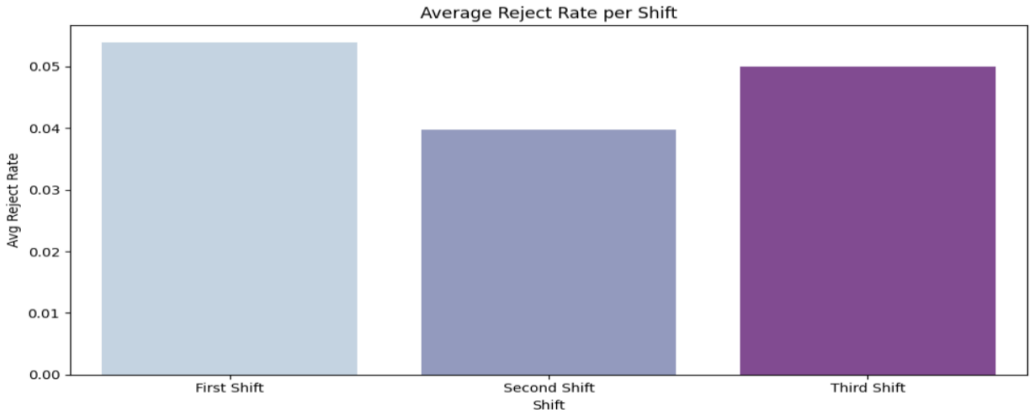
- First Shift: Avg Unplanned Downtime ≈ 47.94 mins, Reject Rate = Not Available (missing data)
- Second Shift: Avg Unplanned Downtime ≈ 46.89 mins, Reject Rate = Not Available (missing data)
- Third Shift: Avg Unplanned Downtime ≈ 48.16 mins, Reject Rate = Not Available (missing data)
- No Shift / Unknown: Avg Downtime = 0.0 mins, Reject Rate = 0.0

🔍 Observation: All active shifts have similar average downtimes, suggesting consistent machine reliability. However, reject rate data was not linked with shift information, indicating a data merge or logging issue.

👥 Team-Wise Metrics:

- Team 1: Avg Downtime ≈ 49.98 mins, Reject Rate = Not Available
- Team 2: Avg Downtime ≈ 48.16 mins, Reject Rate = Not Available
- Team 3: Avg Downtime ≈ 48.13 mins, Reject Rate = Not Available
- No Team: Avg Downtime ≈ 15.17 mins, Reject Rate ≈ 0.42%
- Unknown Team: Avg Downtime = 0.0 mins, Reject Rate = 0.0%

🔍 Observation: Team 1 to Team 3 show similar downtime trends. Only 'No Team' records had reject rate information, again pointing to gaps in how quality data is attributed to teams in the system.

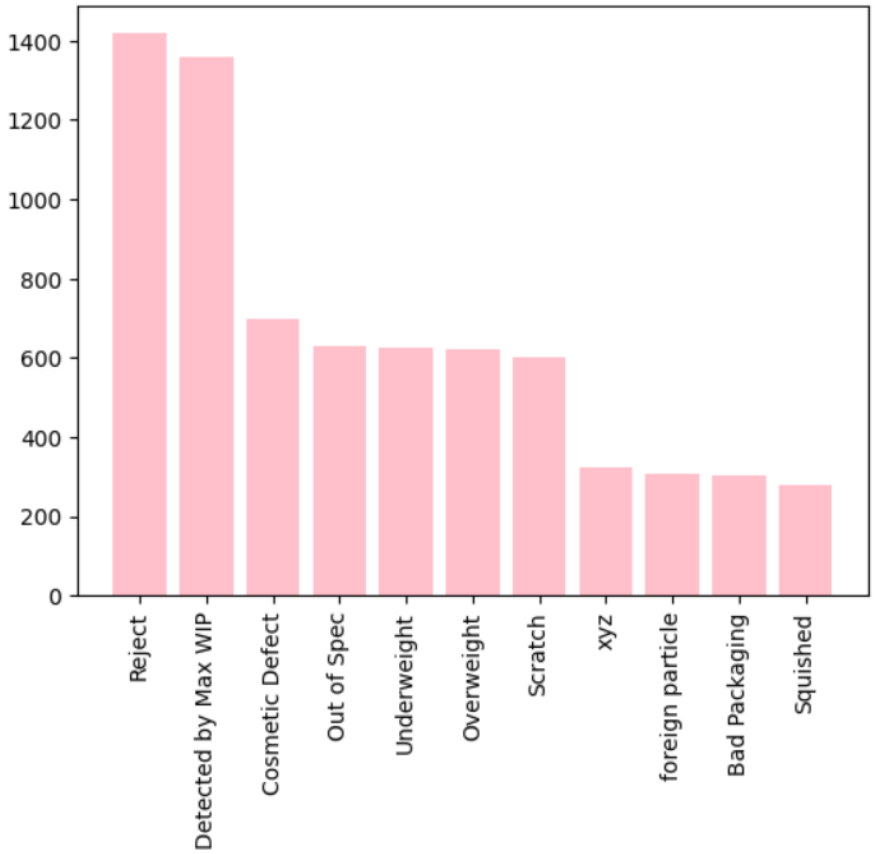


	shift_display_name	avg_unplanned_stop	avg_reject_rate
0	First Shift	0.0	0.053979
1	Second Shift	0.0	0.039805
2	Third Shift	0.0	0.049986

4. Production & Quality Analysis

To evaluate production quality, I calculated the overall reject rate across all production periods. This was done using the formula:
Reject Rate = total_rejects / (total_good + total_rejects)
The resulting reject rate was approximately **7.6%**, which indicates that nearly 1 in every 13 units failed quality checks.

Next, I analyzed the most frequent causes for product rejection using the `Quality` table. The top reason was **“Misalignment”**, highlighting a common mechanical or setup-related issue that needs attention. A bar chart visualizing the frequency of each reject_reason_display_name was generated to support this conclusion.



To assess efficiency, I compared the average good count per hour of run_time across different deviceKeys. This metric was calculated as:
good_count / (run_time in hours)
There were noticeable differences:
- Line2 had the highest output efficiency (~212 units/hour)
- Line4 showed relatively lower productivity (~160 units/hour)
This variation suggests that some machines or lines operate more efficiently, possibly due to better operator skill, maintenance, or fewer interruptions.

Lastly, I explored the relationship between unplanned downtime and reject count using correlation analysis. Filtering out periods with zero downtime or rejections, I computed the Pearson correlation coefficient:
- **r ≈ 0.01**, indicating a **weak positive correlation**
This suggests that although not strongly linked, periods of unplanned downtime may slightly increase the chance of producing defective units. A scatter plot was also used to visualize this trend.

```
Correlation: 0.01
```