Spray-Tek Work Report

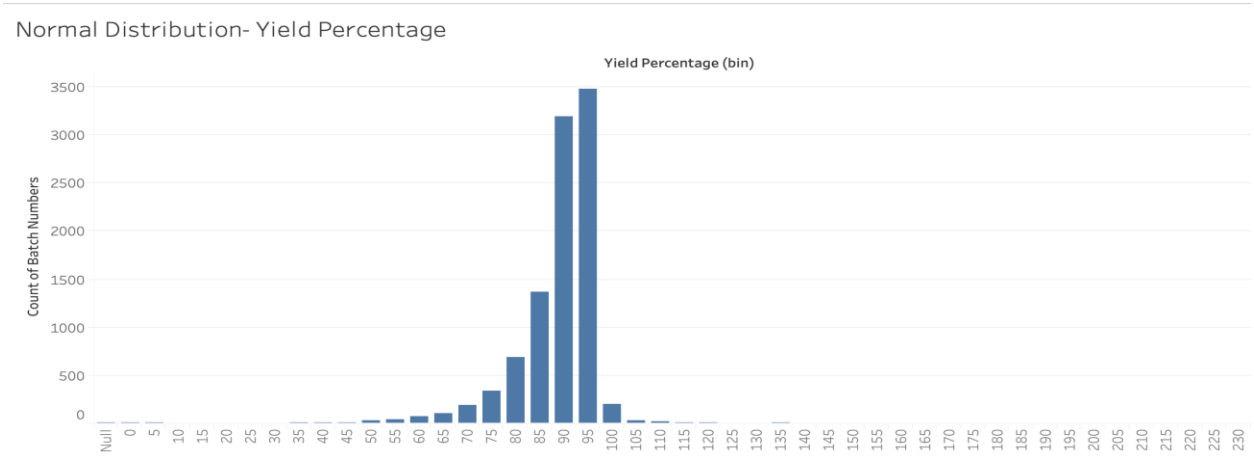
[df is Original Raw Data]

I think columns like Flow, Hygroscopicity, Bulk Density & Moisture might not be of much use for our project. But this is just an assumption so instead of throwing them away, we create **df1** without these columns & continue our EDA.

Looking at the stats for **df1** we see high standard deviation for ActualDryQty which is justified as this totally depends on the size of the order coming towards SprayTek.

	ScheduledDryQty	ActualDryQty	YieldPercentage	DryingTime_Hrs	CleanTime_Hrs	DownTime_Hrs	TotalRunTime_Hrs	ttlBatches
count	9969.000000	9956.000000	9961.000000	9970.000000	9970.000000	9970.000000	9970.000000	9970.000000
mean	8944.787058	8348.359213	91.214466	21.752849	6.339228	4.596861	32.653591	5.036409
std	13974.411561	13350.619864	13.582797	23.249582	6.556783	7.568556	29.983541	6.930778
min	5.680500	0.100000	0.000000	0.000000	-9.200000	0.000000	0.000000	1.000000
25%	1559.935000	1423.374925	88.700000	5.725000	3.500000	0.600000	12.200000	1.000000
50%	4400.696400	4263.649950	93.300000	13.900000	5.300000	1.900000	22.500000	2.000000
75%	10878.646600	10034.850000	96.600000	30.600000	7.700000	5.500000	42.800000	6.000000
max	301195.720000	293923.198200	864.900000	221.500000	172.100000	127.400000	284.700000	120.000000

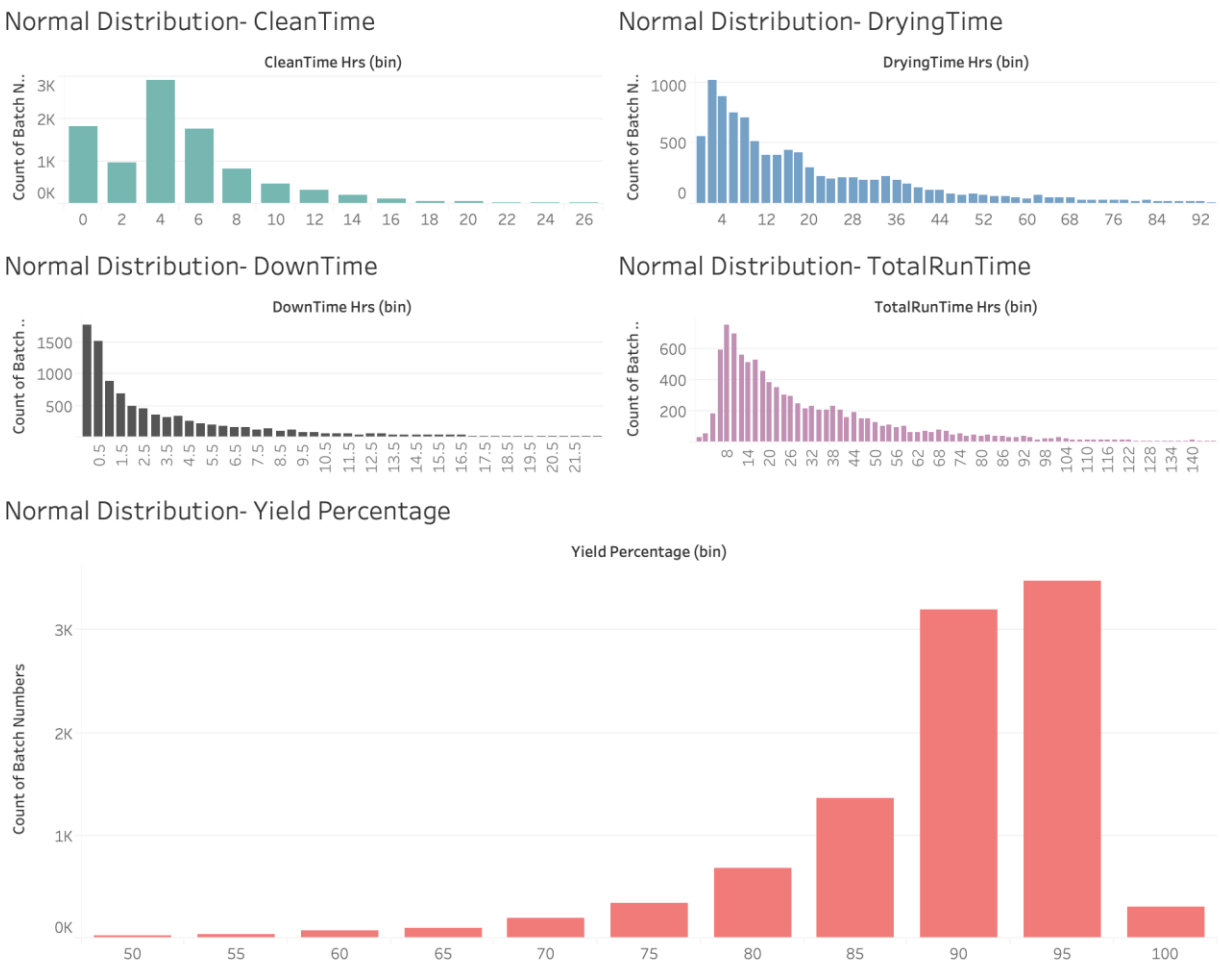
One thing that is unusual is the min Yield Percentage to be 0. It seems peculiar & there is a good possibility that this column might have quite many entries which are unusual.
To understand this better, we use Tableau to find Normal Distribution of Yield Percentage to get some insights.



Now, it doesn't make any sense to consider Yield Percentage greater than 100 for obvious reasons but owing to the fact that the number of apparent faulty rows with Yield percentage greater than 100 is significant, we change those to 100. On the other hand, looking at the distribution, we can eliminate rows with Yield Percentage lesser than 50.

Meanwhile, we also change Missing ProdLine values with its mode.

Although we can see unusual outliers in the stats for our different process times (i.e. CleanTime, DryingTime, Downtime & TotalRunTime) we will not deal with the outliers this early in our analysis. For the time being, we will try to understand the range of hours for which each process time runs the most.



Here, some outliers are taken care of through manual filtering for the purpose of clear representation.

In the ProdLine Column, we see that Flavors have the highest frequency. From the data shown below, we see how much percent of each ProdLine does each dryer run.

ProdLine by Dryer %

	Dryer									
Prod Line	Dryer 01	Dryer 02	Dryer 03	Dryer 04	Dryer 06	Dryer 07	Dryer 08	Dryer 09	Dryer 10	Dryer 11
Chemical	0.85%	5.26%	20.63%	15.50%	35.56%	1.99%	4.41%	9.10%	6.69%	
Cosmetic	1.96%	17.65%	23.53%	45.10%	9.80%			1.96%		
Flavors	37.58%	20.41%	3.96%	0.23%	8.33%	4.03%	8.69%	7.46%	8.99%	0.32%
Food Addit	4.92%	9.84%	30.33%	1.64%	3.28%	4.10%	4.92%	0.82%	38.52%	1.64%
Food Additive	28.63%	4.99%	1.30%	3.90%	15.18%	13.02%	7.16%	9.33%	16.05%	0.43%
Fragrance			40.49%	59.46%			0.05%			
Nutraceutical	0.59%	3.52%	1.17%		7.78%	64.02%	13.95%	0.73%	8.22%	
Pharma		33.33%	10.34%		20.69%	20.69%	10.34%	1.15%	3.45%	
Vitamin	20.51%	1.47%	0.37%		2.93%	16.85%	28.21%	0.73%	28.94%	

% of Total Count ProdLine broken down by Dryer vs. Prod Line.

Dryer 1 runs 37.58% of the Flavors which is the highest. Dryer 2 runs second highest at 20.41%.
It might be insightful to check the covariance between process times & the dryers.

Before we get into that, it would be good to know the correct calculation for the process times:
We create a column which will be CleanTime + DryingTime + DownTime should be equal to TotalRunTime.
As notified by the company, there can be minor difference in times due to dividing consignments into batches before the drying of the previous batch has ended.
Thus, we take difference of CleanTime + DryingTime + DownTime & TotalRunTime, & check if the mean is close to 0.

```
Output: -0.034187601296596294
```

		Covariance
CleanTime_Hrs	Vs Dryer	0.035
DryingTime_Hrs		-0.004
DownTime_Hrs		-0.253
TotalRunTime_Hrs		-0.011

Method: Spearman's correlation coefficient

Further notes:

- Downtime as y variable
- check efficiency through clustering based on ActualDryQty. *[Find relevant attributes for efficiency]* *[Rate = ActualDryQty / TotalRunTime]*