Determining the optimal stacking level in Bangladesh using Cost Benefit Analysis

- Wahid Tawsif Ratul

1 Explaining The Golden Circle

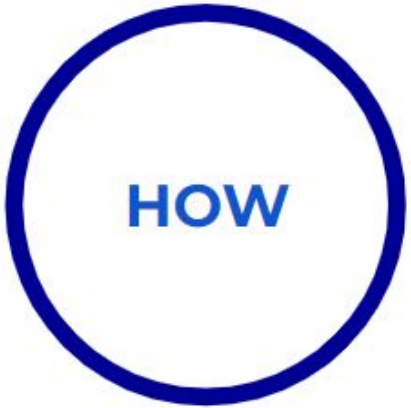
Optimizing Stacking

➔ Determining the optimal " The Algo Order Assignment" stacking

Cost Reduction

➔ Improve
- ◆ Customer Experience
- ◆ Customer Reorder Rates
- ◆ Customer Incident Rates {Extreme Delay, Late Order}
- ◆ Seamless Orders

➔ Lower Costs
- ◆ Reactivation & Proactive Compensation
- ◆ Refund & Wallet Compensation
- ◆ Rider Earnings Per Order



Statistical Modeling

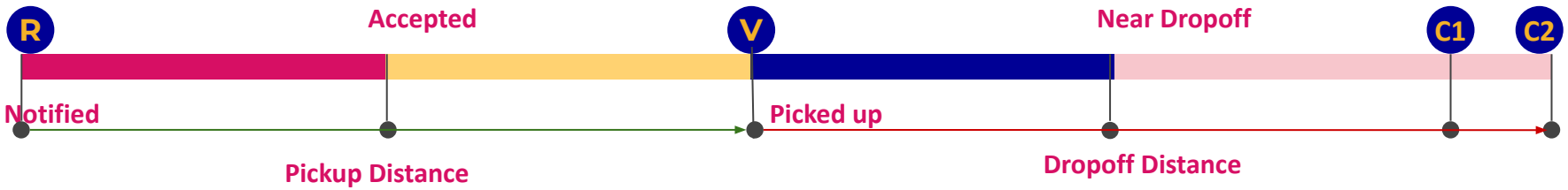
➔ Relative changes in Operational KPIs with increase in stacking %
- ◆ Covariance Test → measure direction of relationship between **stacking & Top Line metrics**
- ◆ Pearson Correlation Test → measure strength of **relationship between stacking & other metrics**
- ◆ Prediction Modeling → measure **impact of X% increase in stacking**

2 What is stacking?

Multiple customers placed orders from the same vendor within the same zone

Data Science Model assigns the same rider who is closest to the vendor (Pickup Location)

Rider will pickup both the orders, and head towards C1 then C2



Why Stacking?

Save up ~25% of the rider cost per order by saving up the pickup distance for stacked orders

Increase # of orders completed by a rider per hour to boost up their earnings per hour

Economies of Scale

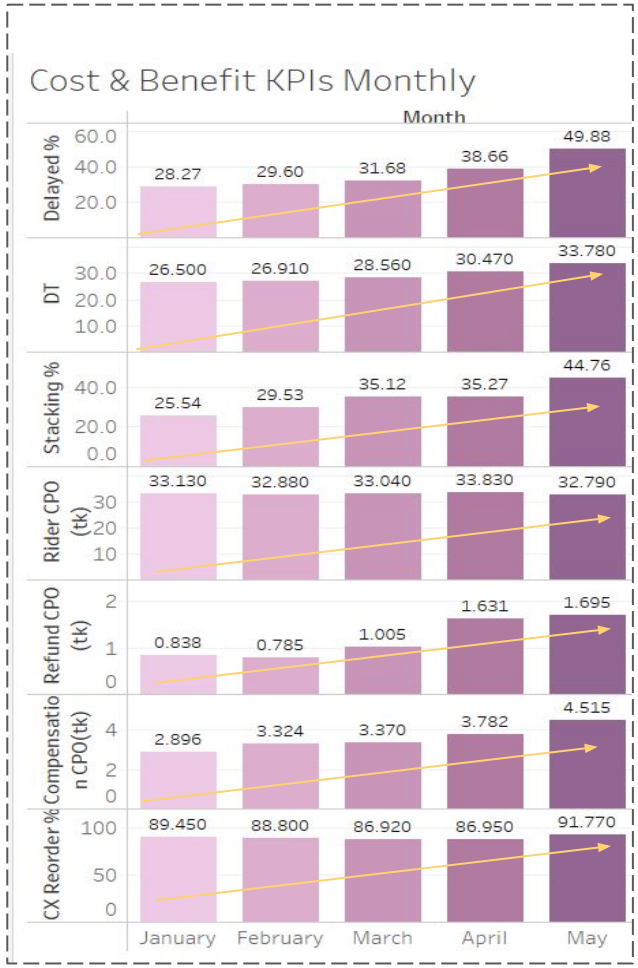
3 Purpose of the Project

- ➔ Central Team **(Delivery Hero)** chose a Cost Optimization Approach across all APAC markets (12 countries) in **Q1 & Q2**
  - ◆ How?
    - Increasing **"The Algo Order Assignment"** prioritization to stack more orders regardless
    - Increasing **Rider Efficiency** by increasing **overall stacking orders %**
    - Reduce **Rider Cost Per Order**

- ➔ What happened?
  - ◆ **Economies of scale came at a huge cost** as different markets had heterogeneous impact (for eg– Singapore had a positive impact, whereas densely populated countries such as **Bangladesh & Pakistan suffered with bad Customer Experience → AKA customers churning out**

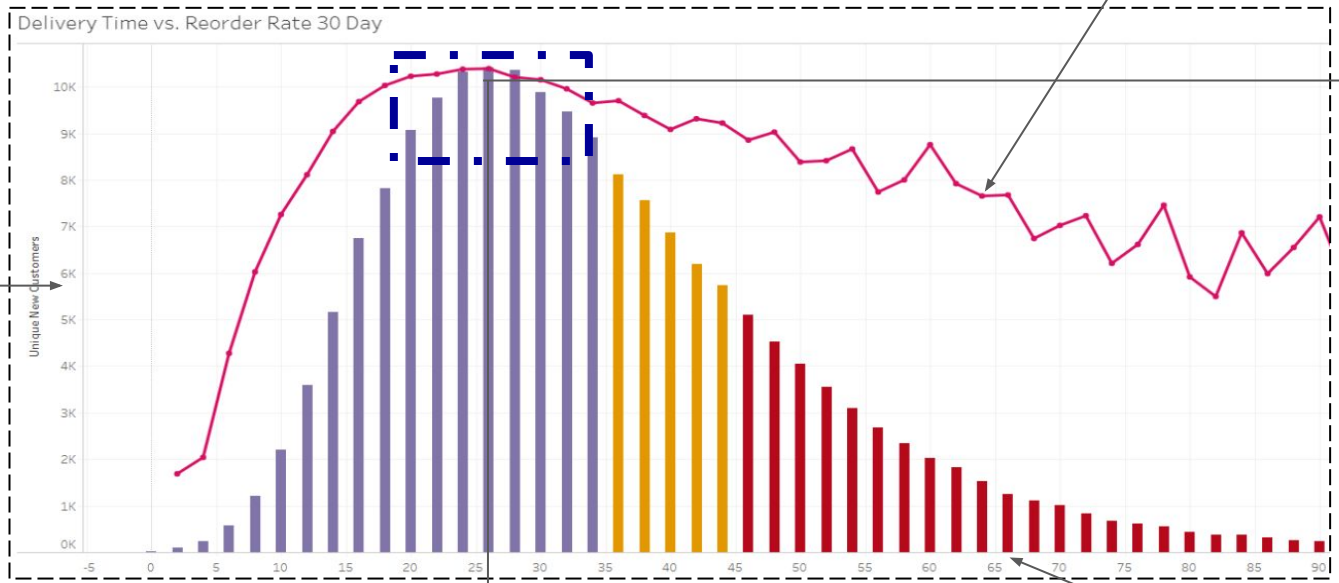
    - Rider Cost Per Order was stagnant
    - Customer Compensation Per Order went up by **~17%**
    - Customer Refund Per Order went up by **~56%**
      - Due to Late Orders Complain
    - Delayed Orders went up by **38%** in a span of 5 months

- ➔ **The million $ question** – **What is the optimal stacking should we configure to minimize cost & maintain a healthy customer experience?**



January 2022-June 2022

4 Delivery Time vs Reorder Rate Case Study



CX Reorder Rate

Optimal DT to drive Customer Experience & Growth

Unique New Customers

Helps us understand what Delivery Time helped us sustain New Customers

Delivery Time

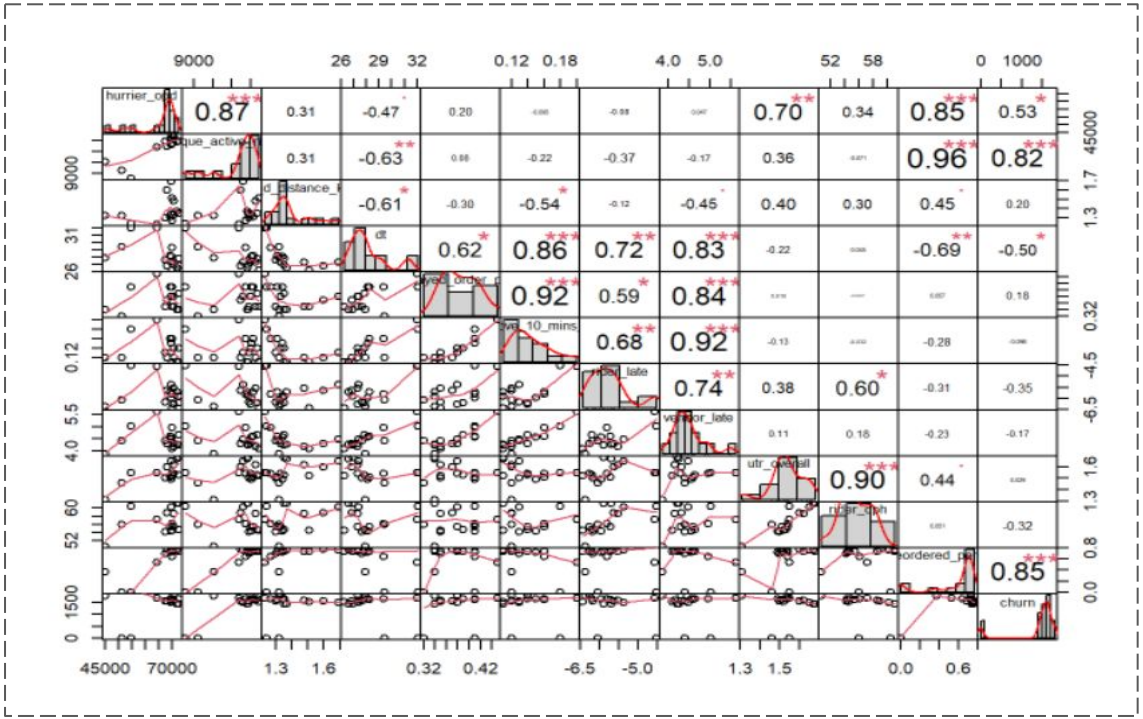
- First glance at the Business Problem –
  - How do we ensure proper customer experience?
  - What's driving bad delivery experience? HINT – **High Delivery Time & Unhealthy Stacking!!**

- **Northstar Business KPIs to focus in order to quantify the negative impact of stacking onto the business :-**
  - Cost Levers ⟶ Rider Payments, Customer Compensations, Refund Compensations
  - Benefit Levers ⟶ Customer Reorder Rates, Delayed Orders, Delivery Time

6 Data Deep Dive

Variables include (27) :-

- ➔ Hurrier_orders
- ➔ Unique_active_rider
- ➔ Delivery Time
- ➔ Delayed_order%
- ➔ Rider_late
- ➔ Distance (km)
- ➔ Vendor Late (mins)
- ➔ Rider Fill Rate %
- ➔ Rider Efficiency Rate
- ➔ Rider Working Hours
- ➔ Stacked deliveries%
- ➔ Weighted Surge Time%
- ➔ Weighted Shrinkage Time%
- ➔ Nps
- ➔ Csat%
- ➔ Seamless Orders %
- ➔ Auto Compensation CPO
- ➔ Refund Compensation CPO
- ➔ Total Rider Earnings (€)
- ➔ Rider CPO
- ➔ Rider CPH
- ➔ Reordered Rate %
- ➔ Reordered 7day %,
- ➔ Reordered 30day%,
- ➔ Reordered 60day%,
- ➔ Churn %
- ➔ Per Rider Weekly Income (€)



Normalizing the variables for reliability
Data Extacted → Jan 2022- April 2022

7 Pearson Correlation Matrix

The table of correlation coefficients

- ➔ **p-values corresponding to the significance levels of the correlations**
  - (+) represents positive correlation
  - (-) represents negative correlation
  - Null Hypothesis : (p<0.05, meaning the coefficient correlation is statistically significant)
  - Alternate Hypothesis : (p>0.05, meaning the coefficient correlation isn't statistically significant)

- ➔ **Higher the value ~1, stronger the correlation**

- ➔ Metrics that have statistical significant effect through stacking :-
  - Delayed Orders % **{Increase}**
  - Delivery Time **{Increase}**
  - Rider CPO **{Decrease}**
  - Refund CPO **{Increase}**
  - Customer Satisfaction **{Decrease}**

Row	Column	Correlation	P-Value
Delayed %	Stacked Deliveries %	**0.781**	**0.0035****
Churn	Stacked Deliveries %	**-0.133**	0.62
Delivery Time	Stacked Deliveries %	**0.82**	**0.00082****
Rider CPO	Stacked Deliveries %	**-0.185**	**0.0004****
Refund CPO	Stacked Deliveries %	**0.426**	**0.0019****
Compensation CPO	Stacked Deliveries %	**-0.27**	0.319
CSAT	Stacked Deliveries %	**-0.775**	**0.0026****
Reorder Rate %	Stacked Deliveries %	**-0.225**	0.33

5 Statistical Modeling

- Objective to figure out
  - **Correlation & Covariance** between different Operational KPIs
  - **Prediction** on the impact of the following KPIs if **we increase stacking by X%**
    - DT
    - Rider CPO
    - Compensation CPO
    - Refund Compensation CPO

- Model 1 :- Predicts the **impact on Delivery Time** if we increase/decrease stacking x%
- Model 2 :- Predicts the **impact on Rider CPO** if we increase/decrease stacking x%
- Model 3 :- Predicts the **impact on Refund CPO** if we increase/decrease stacking x%

8 Delivery Time ~ Stacking Model

Model Interpretation :-

- ➔ **R^2 ~0.99 suggests 99% of the variation** in Delivery Time is explained by the variables included in the model **{High Accuracy}**
- ➔ Coefficients -
  - ◆ Double Stacked Orders **decreases** DT
  - ◆ Single Stack Orders & Triple Stack Orders **increases** DT

```
##
## Call:
## lm(formula = dt ~ log(delayed_order_perc) + log(rider_fill_rate_perc) +
##     utr_overall + daily_working_hours + p_d_distance_km + log(stacked_deliveries_perc) +
##     log(single_stacked_deliveries_perc) + log(double_stacked_deliveries_perc) +
##     log(triple_stacked_deliveries_perc) + churn + log(weighted_surge_time_perc) +
##     log(weighted_shrinkage_time_perc), data = data)
##
## Residuals:
##         1         2         3         4         5         6         7         8
## -0.108288  0.085206 -0.017441  0.033085  0.055849  0.008859 -0.093008  0.238296
##         9        10        11        12        13        14        15        16
## -0.224059  0.009979 -0.051890  0.041318  0.017058 -0.016118  0.012086  0.009069
##
## Coefficients:
##                                        Estimate  Std. Error t value Pr(>|t|)
## (Intercept)                          46.02066977  4.15730939  11.070  0.00158 **
## log(delayed_order_perc)               1.39620266  2.86071116   0.488  0.65895
## log(rider_fill_rate_perc)            -3.70803282  3.11222482  -1.191  0.31915
## utr_overall                           1.26816061  1.34078008   0.946  0.41401
## daily_working_hours                  -0.00003032  0.00004472  -0.678  0.54647
## p_d_distance_km                       0.97705241  1.61106222   0.606  0.58702
## log(stacked_deliveries_perc)          1.54069229  5.27887388   0.292  0.78941
## log(single_stacked_deliveries_perc)   3.09951020  5.84349361   0.530  0.63257
## log(double_stacked_deliveries_perc)  -0.72625458  0.88683307  -0.819  0.47283
## log(triple_stacked_deliveries_perc)   2.67158639  1.13424135   2.355  0.09982 .
## churn                                -0.00047742  0.00022876  -2.087  0.12814
## log(weighted_surge_time_perc)        -0.06087584  0.45646732  -0.133  0.90235
## log(weighted_shrinkage_time_perc)     0.95774889  0.26842986   3.568  0.03761 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2195 on 3 degrees of freedom
## Multiple R-squared:  0.9968, Adjusted R-squared:  0.9841
## F-statistic: 78.57 on 12 and 3 DF,  p-value: 0.002076
```

9 Delivery Time ~ Stacking Prediction

week <dbl>	actual_dt <dbl>	predict_dt <dbl>
16	32.08	32.18829
15	29.47	29.38479
14	27.91	27.92744
13	32.03	31.99691
12	29.35	29.29415
11	27.66	27.65114
10	28.14	28.23301
9	28.26	28.02170
8	27.30	27.52406
7	27.85	27.84002

Condition IF	Model Prediction	Confidence Interval (95%)
Stacking ~24%	26.2	**(24.3, 28.3)**
Stacking (+5%)	27.2	**(26.1,28.9)**
Stacking(+10%)	28.5	**(26.4, 30.7)**
Stacking(+15%)	29.5	**(25.9, 33.1)**
Stacking(-5%)	24.8	**(21.1, 28.6)**

Confidence Interval Test

Assuming other variables are maintaining at status quo level, ONLY increasing OVERALL :-

- ➔ **stacking by 5%** would result **DT ~26.2 mins** within (24.3, 28.3) mins range
- ➔ **stacking by 10%** would result **DT ~27.2 mins** within (26.1, 28.9) mins range
- ➔ **stacking by 15%** would result **DT ~28.5 mins** within (26.4, 30.7) mins range
- ➔ **stacking by -5%** would result **DT ~24.8 mins** within (21.1, 28.6) mins range

10 Rider CPO ~ Stacking Modeling

Model Interpretation :-

- ➔ **R^2 ~0.91 suggests 91% of the variation** in Rider CPO is explained by the variables included in the model **[High Accuracy]**
- ➔ Coefficients -
  - Overall Stacking & Double Stacking **decreases** Rider CPO
  - Triple Stack **increases** Rider CPO

```
##
## Call:
## lm(formula = rider_cpo ~ order_per_rider_per_day + unique_active_rider +
##     utr_overall + per_rider_weekly_income_eur + daily_working_hours +
##     p_d_distance_km + log(stacked_deliveries_perc) + log(single_stacked_deliveries_perc) +
##     log(double_stacked_deliveries_perc) + log(triple_stacked_deliveries_perc) +
##     churn, data = data)
##
## Residuals:
##         1         2         3         4         5         6         7         8
##  0.018019 -0.015892  0.019297 -0.044953  0.053229 -0.126655 -0.022823  0.172919
##         9        10        11        12        13        14        15        16
## -0.040547 -0.008211 -0.022164  0.055104 -0.114604  0.107998  0.026616 -0.057335
##
## Coefficients:
##                                    Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                      36.3053725  2.3034608  15.761 0.0000947 ***
## order_per_rider_per_day           0.9220413  1.4318989   0.644    0.5547
## unique_active_rider               0.0004618  0.0009150   0.505    0.6403
## utr_overall                      -5.3155780  6.2628104  -0.849    0.4438
## per_rider_weekly_income_eur       0.3482046  0.1657174   2.101    0.1035
## daily_working_hours              -0.0002333  0.0002543  -0.917    0.4108
## p_d_distance_km                   0.1457138  0.6276895   0.232    0.8278
## log(stacked_deliveries_perc)     -2.9640654  3.5109520  -0.844    0.4461
## log(single_stacked_deliveries_perc)  2.3505792  2.9957699   0.785    0.4765
## log(double_stacked_deliveries_perc) -0.5361111  0.6454316  -0.831    0.4529
## log(triple_stacked_deliveries_perc)  0.7590942  0.3223340   2.355    0.0781 .
## churn                             0.0002953  0.0001957   1.509    0.2057
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1469 on 4 degrees of freedom
## Multiple R-squared:  0.9163, Adjusted R-squared:  0.686
## F-statistic:  3.98 on 11 and 4 DF,  p-value: 0.0971
```

11 Rider CPO ~ Stacking Prediction

week <dbl>	actual_rider_cpo <dbl>	pred_rider_cpo <dbl>
16	33.73	33.71198
15	33.57	33.58589
14	33.81	33.79070
13	33.50	33.54495
12	33.41	33.35677
11	33.17	33.29665
10	33.50	33.52282
9	33.51	33.33708
8	33.11	33.15055
7	32.90	32.90821

Condition IF	Model Prediction	Confidence Interval (95%)
Stacking ~24%	32.37	**(26.55, 38.20)**
Stacking (+5%)	32.28	**(26.49, 38.07)**
Stacking(+10%)	32.20	**(26.44, 37.97)**
Stacking(+15%)	32.18	**(26.37, 37.99)**
Stacking(-5%)	32.49	**(26.59 38.38)**

Confidence Interval Test

Assuming other variables are maintaining at status quo level, ONLY increasing OVERALL :-

- ➔ **stacking by 5%** would result **Rider CPO ~ 32.28 tk** within (26.49, 38.70) tk range.
- ➔ **stacking by 10%** would result **Rider CPO ~32.20 tk** within (26.44, 37.97) tk range.
- ➔ **stacking by 15%** would result **Rider CPO ~32.18 tk** within (26.37, 37.99) tk range.
- ➔ **stacking by -5%** would result **Rider CPO ~32.49 tk** within (26.59, 38.38) tk range.

12 Refund CPO ~ Stacking Model

Model Interpretation :-

- ➔ **R^2 ~0.97 suggests 97% of the variation** in Refund CPO is explained by the variables included in the model **[High Accuracy]**
- ➔ Coefficients -
  - ◆ Double Stacking, Triple Stacking **increases** Refund-Comp CPO

.

```
##
## Call:
## lm(formula = refund_comp_cpo ~ dt + log(delayed_order_perc) +
##     utr_overall + daily_working_hours + log(stacked_deliveries_perc) +
##     log(single_stacked_deliveries_perc) + log(double_stacked_deliveries_perc) +
##     log(triple_stacked_deliveries_perc) + log(reordered_perc +
##     0.00001) + log(reordered_7day_perc + 0.00001) + log(reordered_30_day_perc +
##     0.00001), data = data)
##
## Residuals:
##         1         2         3         4         5         6         7         8
## -0.015403  0.015755 -0.004621 -0.011283  0.033286  0.009831 -0.009001  0.023266
##         9        10        11        12        13        14        15        16
## -0.056035  0.020650 -0.025964 -0.028817  0.004882  0.006318 -0.012692  0.049828
##
## Coefficients:
##                                      Estimate  Std. Error t value Pr(>|t|)
## (Intercept)                         9.90091437  3.89905684   2.539   0.0640 .
## dt                                 -0.06812802  0.05346262  -1.274   0.2716
## log(delayed_order_perc)            -0.27887243  0.74707486  -0.373   0.7279
## utr_overall                        -1.29191932  0.78041079  -1.655   0.1732
## daily_working_hours                -0.00005067  0.00001765  -2.870   0.0455 *
## log(stacked_deliveries_perc)       -0.04904068  0.87661578  -0.056   0.9581
## log(single_stacked_deliveries_perc)  0.82287462  0.76319320   1.078   0.3416
## log(double_stacked_deliveries_perc)  0.00950618  0.19506382   0.049   0.9635
## log(triple_stacked_deliveries_perc)  0.55223149  0.36004647   1.534   0.1999
## log(reordered_perc + 0.00001)        2.95737779  3.26477946   0.906   0.4162
## log(reordered_7day_perc + 0.00001)  -1.14640374  1.36953512  -0.837   0.4496
## log(reordered_30_day_perc + 0.00001) -1.82046835  3.14919021  -0.578   0.5942
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0506 on 4 degrees of freedom
## Multiple R-squared:  0.9752, Adjusted R-squared:  0.9071
## F-statistic: 14.32 on 11 and 4 DF,  p-value: 0.01018
```

13 Refund CPO ~ Stacking Prediction

week <dbl>	actual_refund_comp_cpo <dbl>	pred_refund_comp_cpo <dbl>
16	1.10	1.1154029
15	1.03	1.0142447
14	1.19	1.1946206
13	1.00	1.0112830
12	0.85	0.8167140
11	0.79	0.7801687
10	0.74	0.7490011
9	0.78	0.7567338
8	0.72	0.7760355
7	0.73	0.7093496

Condition IF	Model Prediction	Confidence Interval (95%)
Stacking ~24%	0.93	**(0.43, 1.43)**
Stacking (+5%)	1.18	**(0.35, 2.01)**
Stacking(+10%)	1.38	**(0.23, 2.53)**
Stacking(+15%)	1.58	**(0.11, 3.05)**
Stacking(-5%)	0.64	**(0.20, 1.07)**

Confidence Interval Test

Assuming other variables are maintaining at status quo level, ONLY increasing OVERALL:

- ➔ **stacking by 5%** would result **Refund-Comp CPO ~ 0.93 tk** within (0.43, 1.43) tk range.
- ➔ **stacking by 10%** would result **Refund-Comp CPO ~1.18 tk** within (0.35, 2.01) tk range.
- ➔ **stacking by 15%** would result **Refund-Comp CPO ~1.58tk** within (0.11, 3.05) tk range.
- ➔ **stacking by -5%** would result **Refund-Comp CPO ~0.64 tk** within (0.20, 1.07) tk range.

14 Key Findings from Predictive Analytics

- **Strong positive correlation between Stacking and DT**
  - **If stacking is increased by 10%, DT is predicted to increase from 26.2 min to 28.5 min**
- **Strong negative correlation between Stacking and Rider CPO**
  - **If stacking is increased by 10%, Rider CPO is predicted to decrease from 32.37 tk to 32.20 tk**
- **Strong positive correlation between Stacking and Refund-Comp CPO**
  - **If stacking is increased by 10%, Refund-Comp CPO is predicted to increase from 0.93 tk to 1.38 tk**

Moral of the story :-

- While rider cost is predicted to go down due to stacking, performance is predicted to get worse with higher Delivery Time and more Refund Compensations

15 What's Next?

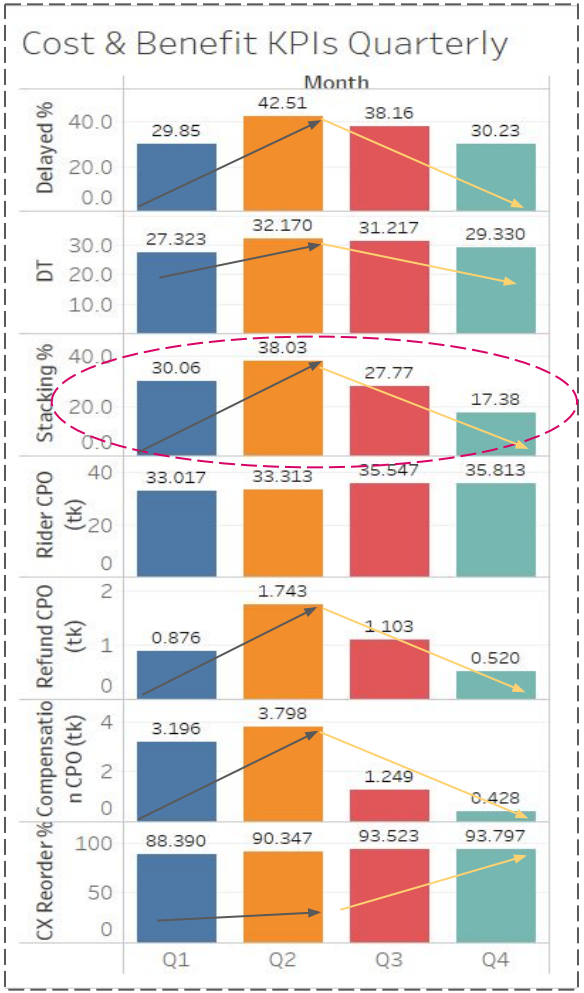
Before the project quest →

➔ **The million $ question** – **What is the optimal stacking should we maintain to minimize cost & maintain a healthy customer experience?**

After the project quest →

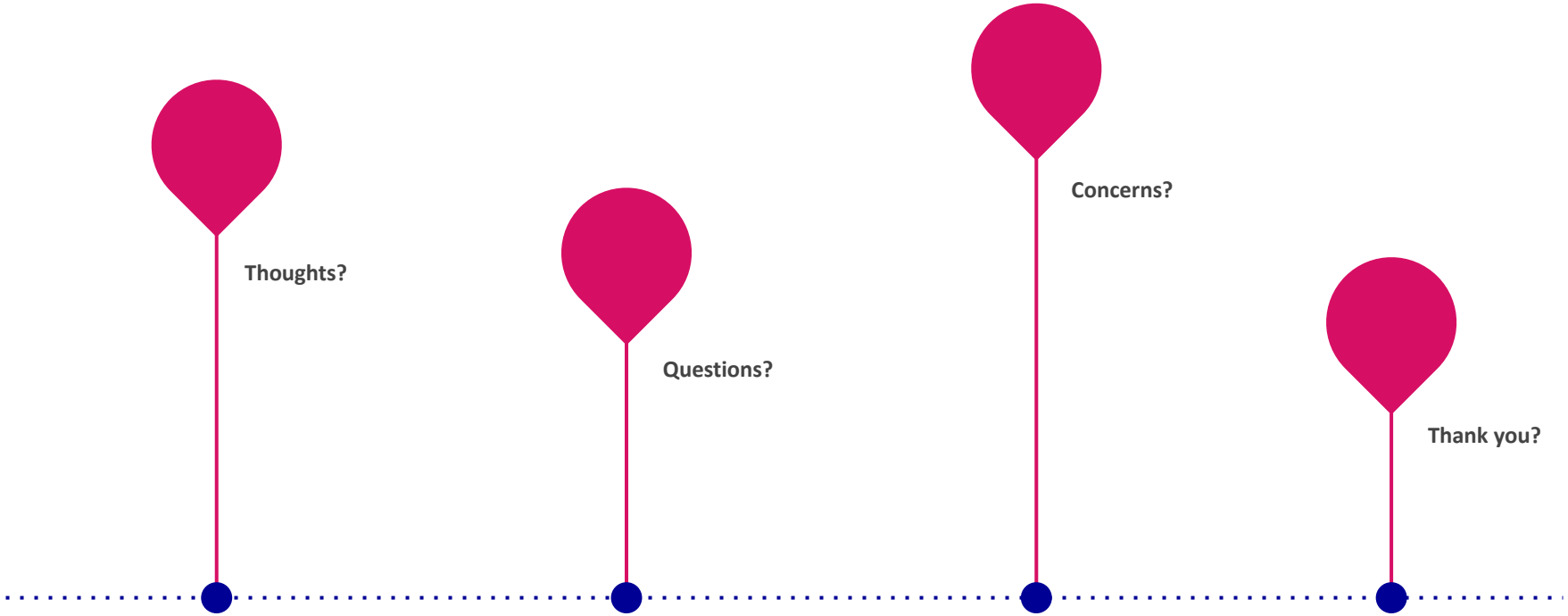
➔ We should **maintain ~20% to 25% overall stacking** to find a healthy balance between cost & logistics performance!

1. Identifying the problem
2. Consulting with relevant stakeholders
3. Data Analytics to pinpoint the problem
4. Drive the narrative for a change with empirical evidence
5. Enjoy the results



January 2022-December 2022

16 Q&A



Return Order Initiative to minimize Wastage Cost

- Wahid Tawsif Ratul

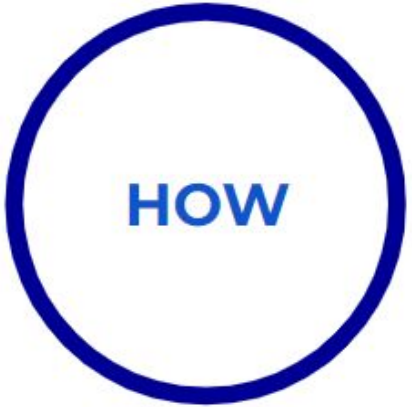
1 Explaining The Golden Circle

Return Cancelled Orders

- ➔ Streamlining a Standard Procedure between Foodpanda and Restaurant owners
- ➔ Vendors receives the cancelled order in return of what Foodpanda would pay
- ➔ Saving **40%** of GMV per cancelled order

Minimize Wastage Costs

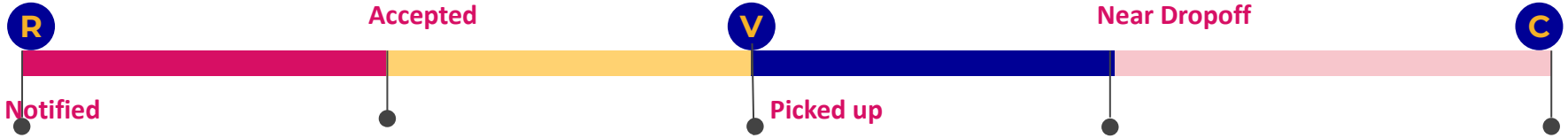
- ➔ To prevent CASH LOSS (% of GMV) due to **~4%** cancellations every month
- ➔ Improve Vendor Retention
- ➔ Provide **competitive edge** with **ZERO** wastage policy Like **PATHAO!**



Vendor Onboarding & Return Trip

- ➔ Revisiting vendor contracts, onboarding eligible vendors through selection criteria ***{Cuisine, Wastage Contribution, & Vendor ratings}***
- ➔ Momentarily generating Automatic Return Trip against cancelled orders
- ➔ Monitoring Return Trip Journey through Rider GPS

2 Previous Scenario

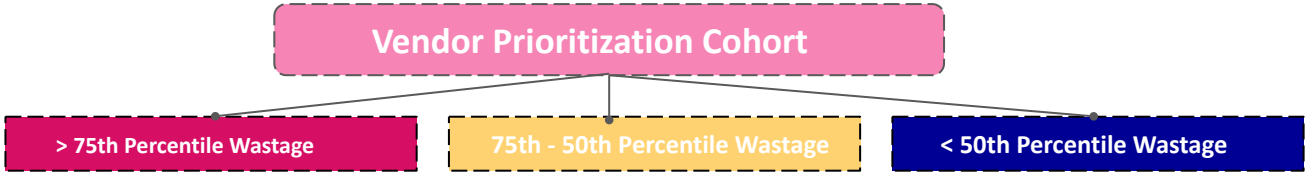


- ➔ **No Return Trip Procedure** – Riders would take the cancelled orders to the Local Hubs
- ➔ No visibility on the Return Trip Journey
- ➔ Foodpanda had incurred **~4.5%** of Wastage on GMV monthly
- ➔ Resentment around Small & Medium sized vendors due to high commissions & Wastage Cost
- ➔ High Barrier to Entry for small businesses due to intrinsic costs for operating in the platform

Month	Wastage Orders %	Wastage GMV (€)	Wastage CPO (€)	Cancellations %	Wastage Vendor %
2021-01-01	5.00	30,000	0.36	5.02	56.1
2021-02-01	5.10	26,170	0.31	5.11	54.5
2021-03-01	5.40	30,994	0.30	5.38	48.4
2021-04-01	6.30	38,245	0.31	6.32	53.4
2021-05-01	5.70	39,828	0.32	5.66	39.8
2021-06-01	3.70	34,523	0.37	3.70	33.9
2021-07-01	3.60	40,112	0.34	3.65	27
2021-08-01	3.00	32,442	0.37	2.97	28.2
2021-09-01	2.90	29,485	0.40	2.85	32.2
2021-10-01	3.30	32,817	0.37	3.27	30.2
2021-11-01	3.30	31,286	0.39	3.34	32.8
2021-12-01	3.20	34,080	0.42	3.23	31.2

3 What's my Solution?

1 Identify Vendors with High Contributing Overall Wastage Cost and share it with Account Management Team to initiate Return Policy to their contract **[Timeline → 3weeks]**



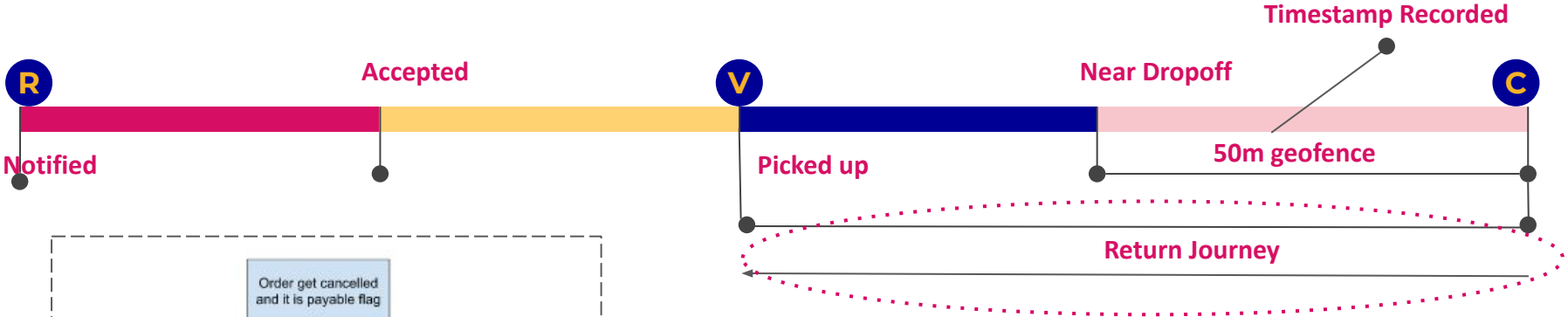
Vertical	Cuisine	Vendor Name	Vendor Code	Wastage Orders	Wastage Orders %	Wastage Amount (€)	Fail Rate %
restaurants	Fast Food	Pizza Burg - Wari	jswd	163	4.3	374.9	6.96
restaurants	Rice Dishes	7dayz	t8ba	148	1.2	266.2	7.30
restaurants	Burgers	Burger Xpress - Banani	vyi8	140	1.2	373.6	5.91
restaurants	Bangladeshi	Palki Restaurant	t4sx	135	1.4	194.3	7.01

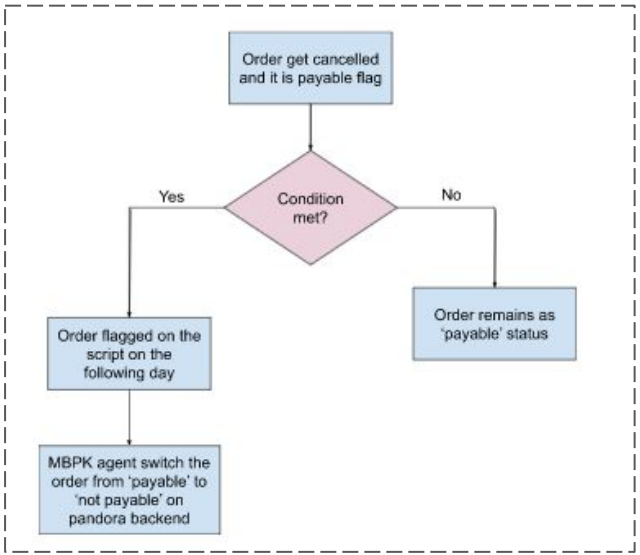
2 → Creating Standard Operating Procedure → Cuisine selection {excluding vendors which consists major perishable times}
→ Backend Configuration → Automation of the Return Trip for specific vendors with a CSV upload

3 Automated Dashboard to provide visibility for the Leadership Team & relevant stakeholders with Return Trip Success Metrics

4 Re-iterate the process 1st of EVERY month

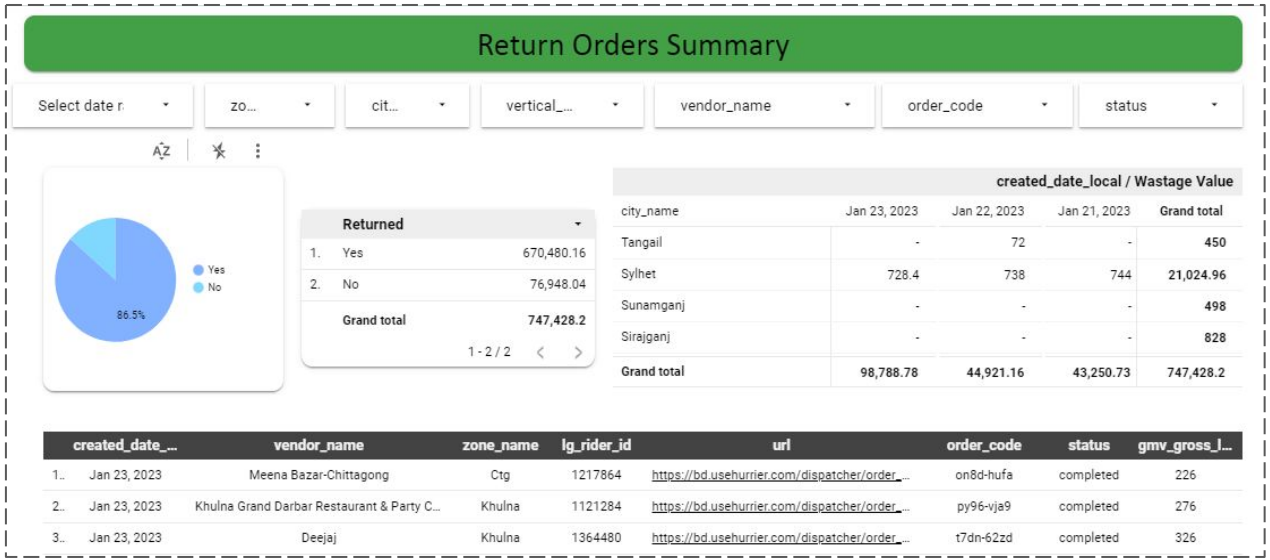
4 The New Profound Rider Journey





- ➔ Timestamp is registered as soon as the Rider cross the dropoff location **50m geofence**
- ➔ Rider has to provide **Proof of Return** to complete the delivery
- ➔ Regional SSC team would **(X+1) day** later manually change the billing status to NON-BILLABLE, hence preventing 40% cash loss against cancelled orders

5 Compliance Tracking



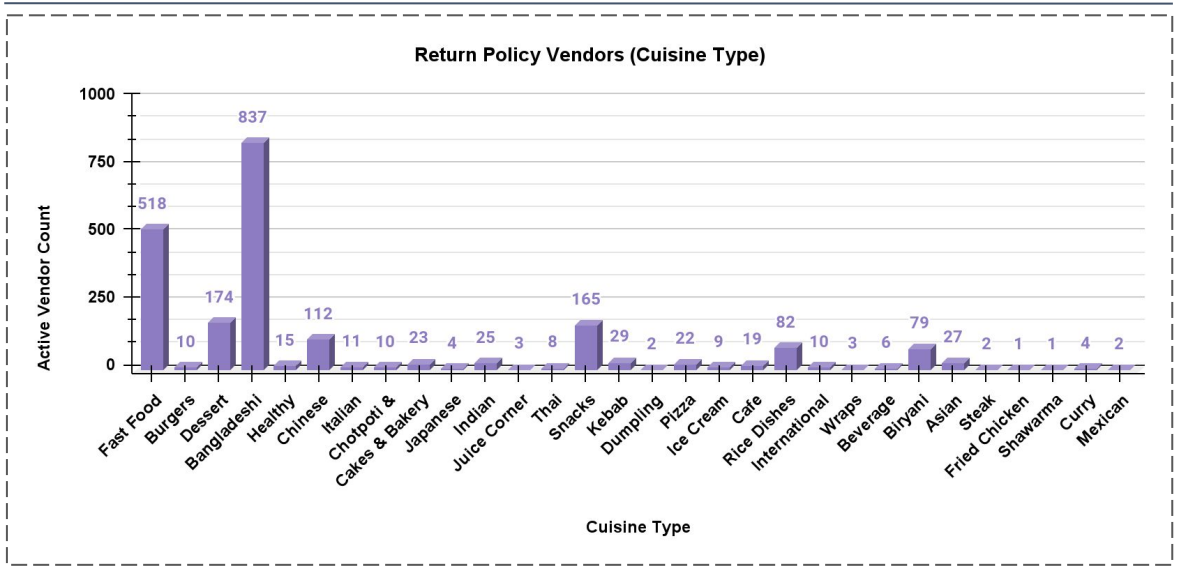
- ➔ Made in Google Data Studio
- ➔ Extracts from Google BigQuery
- ➔ Automatically updates every day @ 8am
- ➔ Flags orders that are **Not Returned** (No geofence breach timestamp)

Current Status :-

- ➔ For **~13.5%** of cancelled orders not returned, riders are penalized with GMV value adjusted from rider wallet weekly
- ➔ Cancelled food from **EXCLUDED** vendors are deliberately donated to an exclusive charitable foundation

6 Vendor Demographics

of Vendors by Cuisine Type



- ➔ **~23%** of the active vendors are on the Return Policy Initiative
- ➔ **~4.8%** of Overall Wastage Cost are saved

- ➔ Roughly about **2214 vendors** are active on the **Return Policy Initiative**
- ➔ Onboarding happens once a month after every payment cycle
- ➔ Cuisine selection is **EXTREMELY** critical to avoid repercussions from **NEGETIVE** Customer Experience
- ➔ Small chains with **Low Order Volume & High Commissions PREFER** the policy to minimize their Wastage Cost

7 Business Impact

Major KPIs being monitored

Month	Wastage Orders %	Wastage GMV Saved (€)	Wastage CPO (€)	Food Quality Complaints	Active Vendors
2022-01-01	0.02	13,590	0.14	1,373	9770
2022-02-01	0.03	23,446	0.28	1,293	9903
2022-03-01	0.03	33,611	0.34	1,573	10280
2022-04-01	0.41	54,616	0.95	1,138	9478
2022-05-01	0.49	58,350	1.02	1,480	8984
2022-06-01	0.28	49,988	0.92	1,296	9020
2022-07-01	0.37	38,184	0.74	960	8620
2022-08-01	0.35	36,255	0.51	255	8662
2022-09-01	0.33	36,037	0.42	310	8448
2022-10-01	0.29	37,349	0.48	298	8418
2022-11-01	0.20	28,727	0.21	273	8120
2022-12-01	0.23	28,907	0.28	540	7805

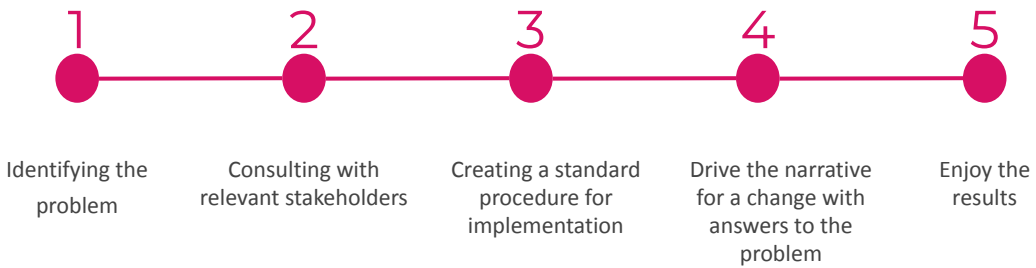
High Vendor Churn

- ➔ **Extremely Low Vendor Lifespan** for Medium & Small Vendors if Wastage Costs are significantly high coupled with High Commission fees
- ➔ **~377,060 €** saved in year 2022, **~3.4%** of GMV saved from the initiative
- ➔ **Customer Incident Rates** (complains regarding food quality) from the vendors have gone down by **~60.6%**
- ➔ However, **Wastage Cost Per Order** has gone down by **0.12€ in year 2022**

8 Project Hindsight

Highlighting some of the intrinsic challenges

- ➔ **Holding strong command on riders RETURNING cancelled food to vendors**
- ➔ **Creating visibility for unmonitored part of the journey where riders are donating food to local NGOs**
- ➔ **Hygiene concerns were raised for the entirety of the process**
- ➔ Overall, **Wastage CPO** fell from **~0.42€** to **~0.28€** **during the initiation of the policy**
- ➔ **However, Wastage CPO from Jan'22 and Dec'22 slightly increased due to change in distribution of Wastage Costs across other vendors (with increase in Average Food Value) platform wide**
  - ◆ **94%** of the vendors in the Return Policy Initiative were classified as Medium-Small chain restaurants
  - ◆ **Majority of the big chain vendors didn't want to partake into the policy**
    - **As a consequence, a big chunk of wastages were out-of-reach**
- ➔ Cross functional collaboration was a constant struggle
- ➔ **Product limitations to ensure riders actually returned the food for cases where customer geofence were not breached**



9 Q&A