Business case, Siarhei Zavaryn

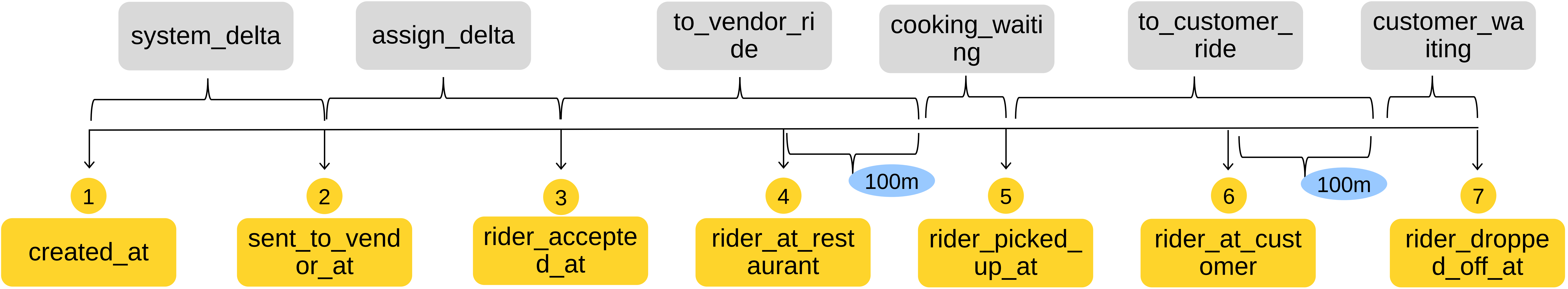
- Delivery timeline: Metrics
- Delivery timeline: Aggregations
- System_delta: Infrastructure
- Cooking_waiting: Automation
- Customer_waiting: Automation
- UTR: Conclusions

Delivery timeline: Metrics

Following metrics are based on primary data and make up delivery time:

1. system_delta = sent_to_vendor_at - created_at
2. assign_delta = rider_accepted_at - created_at
3. to_vendor_ride = (rider_at_restaurant - rider_accepted_at) + 10 sec – consider 10 sec is time for 100m
4. cooking_waiting = rider_picked_up_at – (rider_at_restaurant + 10 sec)
5. to_customer_ride = (rider_at_customer - rider_picked_up_at) + 10 sec
6. customer_waiting = rider_dropped_off_at – (rider_at_customer + 10 sec)

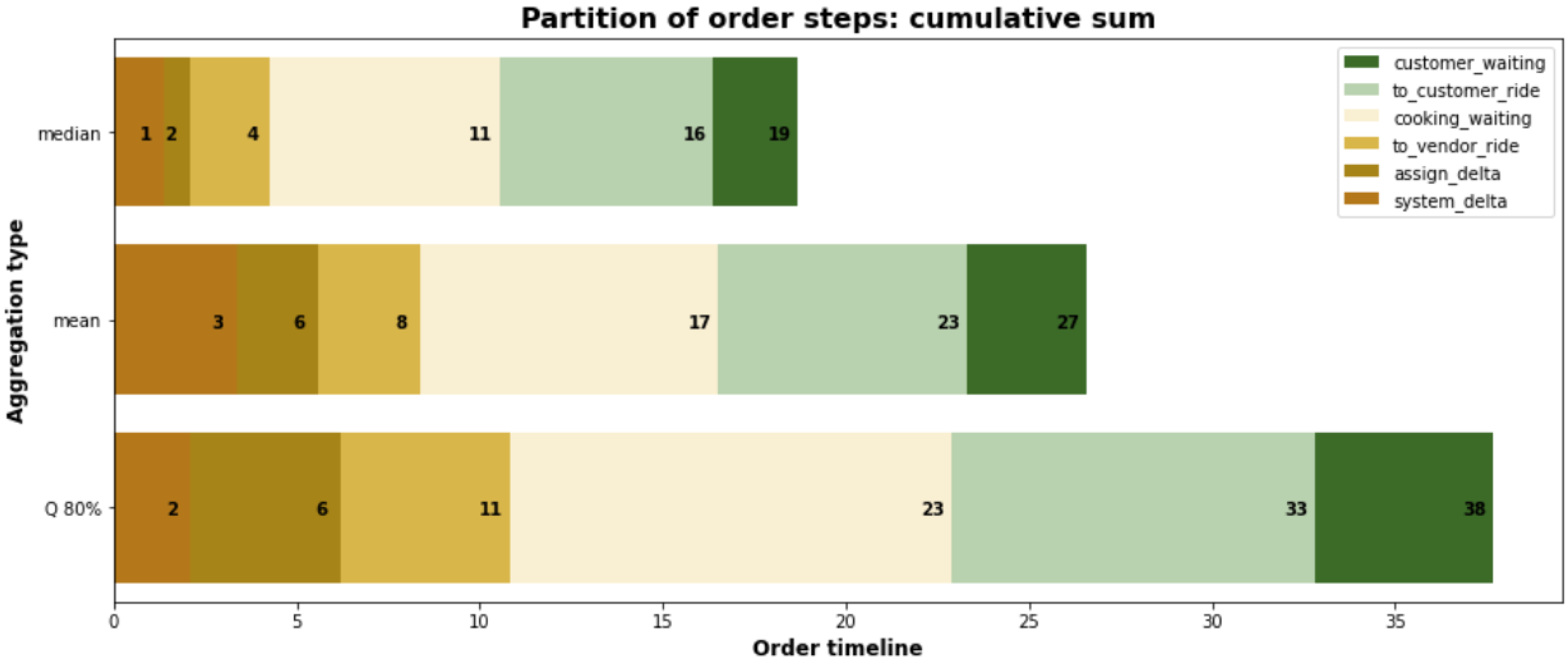
Dataset consists of two days (2020-08-23 and 2020-08-24) performance in one city with over 10k orders which were delivered by 459 riders.



Delivery timeline: Aggregations

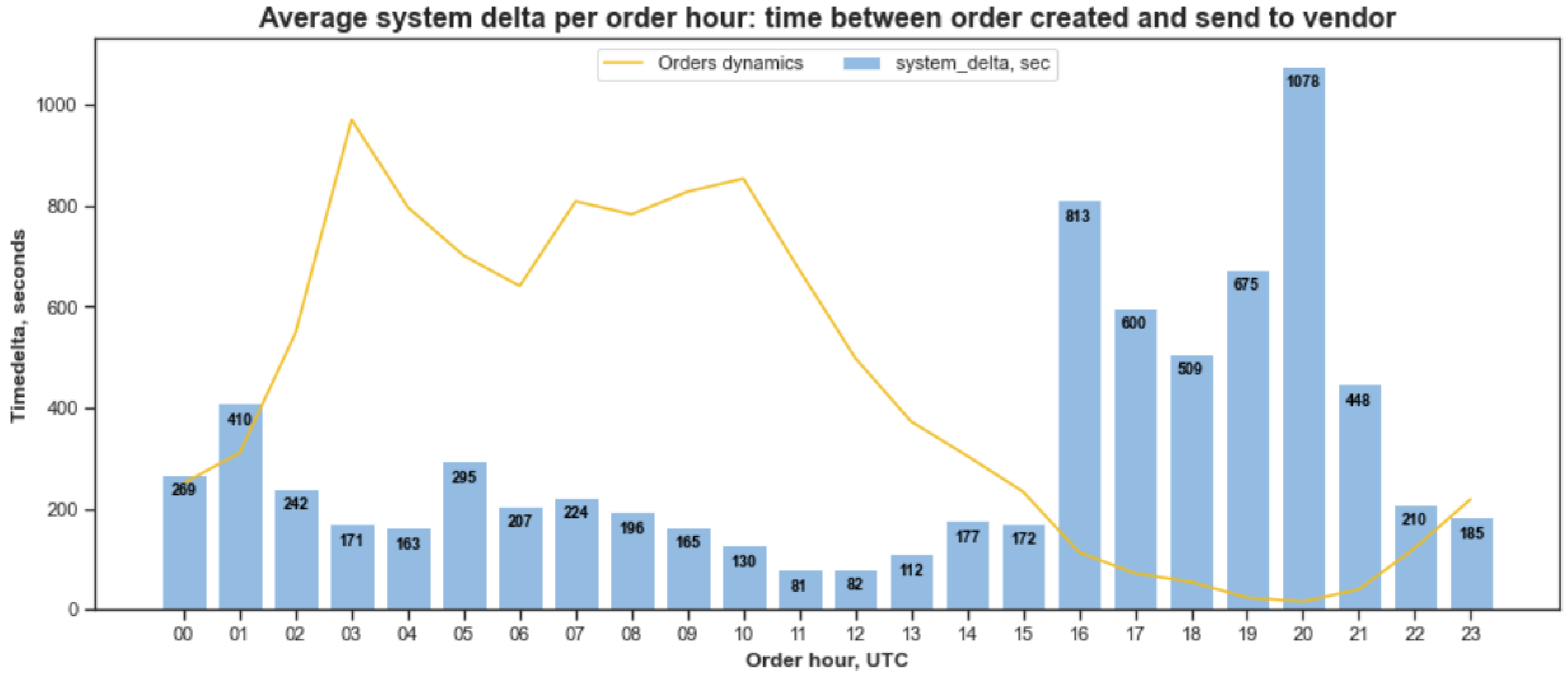
That's the orders split on average (mean), median and Q80% distribution values of each step

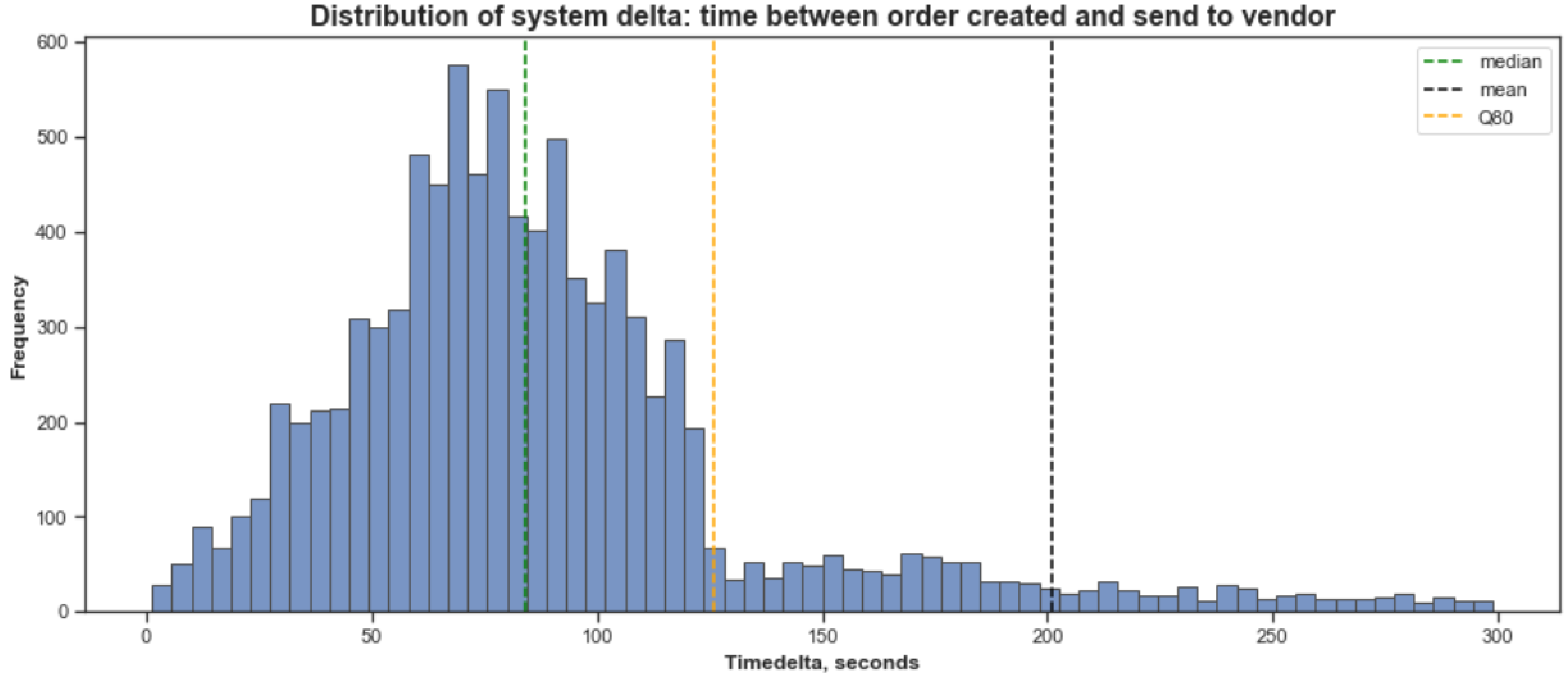
- Just in case: mean delivery time is sum of all mean steps, but it doesn't work for any percentiles.
- All values are shown in minutes.
- In average **only 39%** of delivery time is spent on rides.



Aggregation	Value, minutes
Median	23,1
Mean	26,6
Q80%	32,6

System_delta: Infrastructure





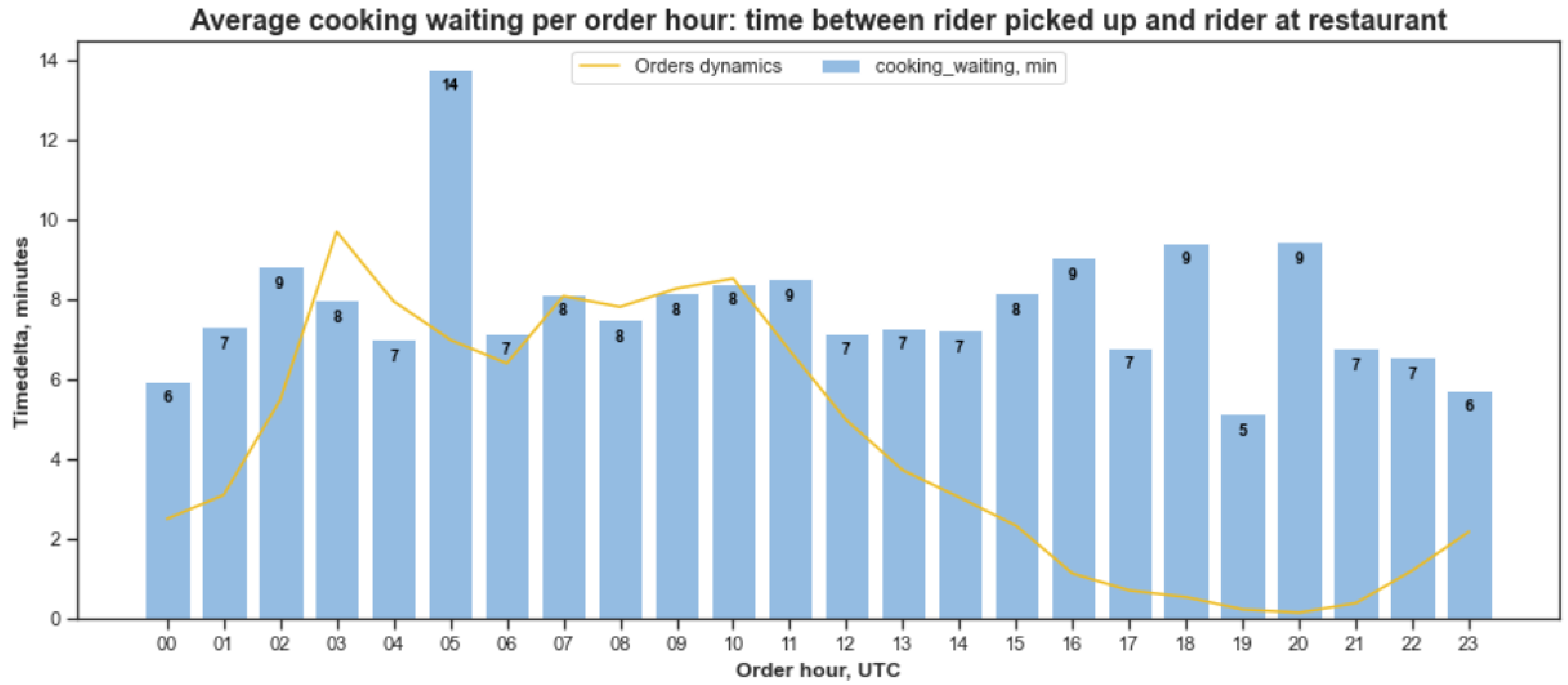
Timedelta between creating the order and sending it to vendor could be reduced by 1-2 minutes.

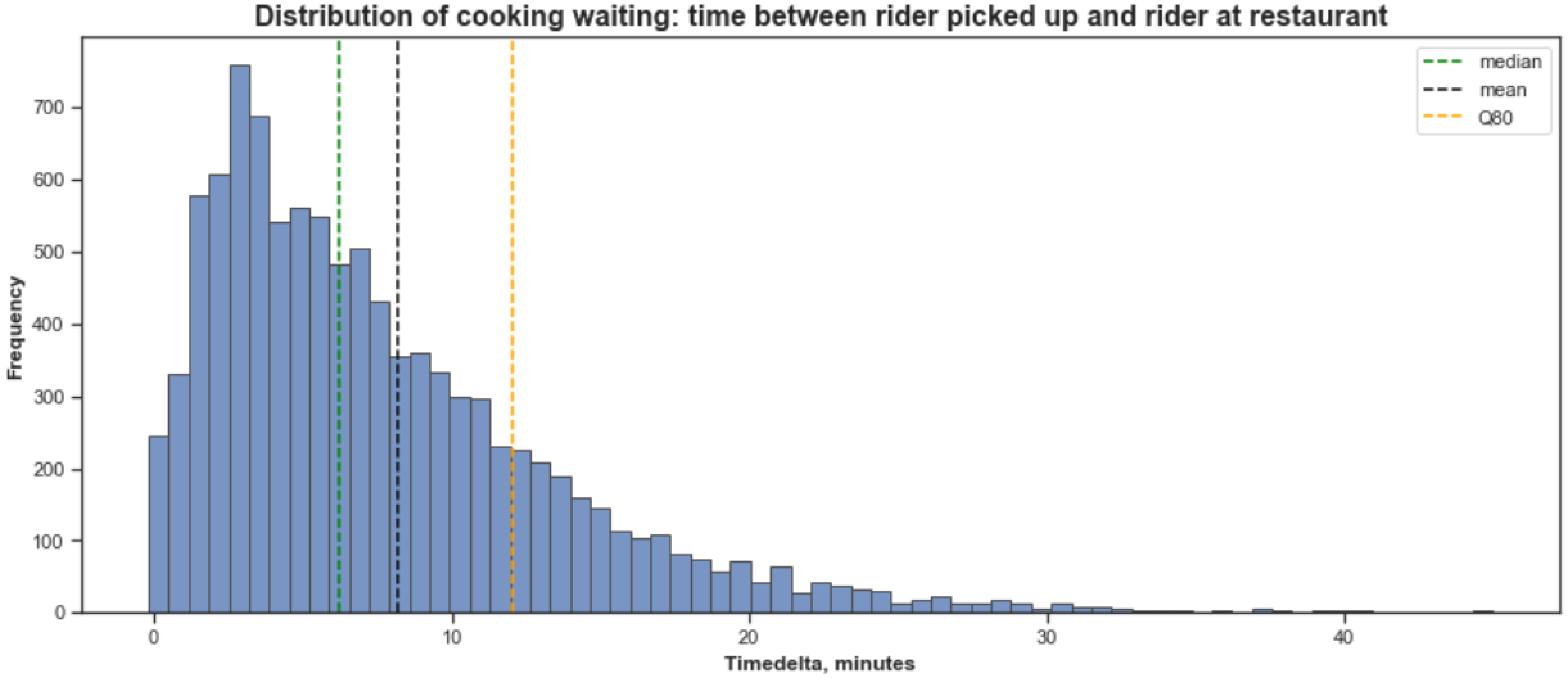
- System_delta in seconds at above chart is grouped by hours of orders created while the yellow line shows the number of orders. The chart below provides the timedelta distribution with average, median and Q80 values.

- The city of the issue belongs to UTC+8 timezone. We can see the large tail of the distribution, which corresponds to midnight. Due to the currently small number of night orders, it almost doesn't influence the median (84 sec) and Q80 (126 sec) values but increases the average value (200 sec). It would be an issue when the order number rises.

- That's the infrastructure problem, which adds an additional 2–3 minutes to delivery time.

Cooking_waiting: Automation

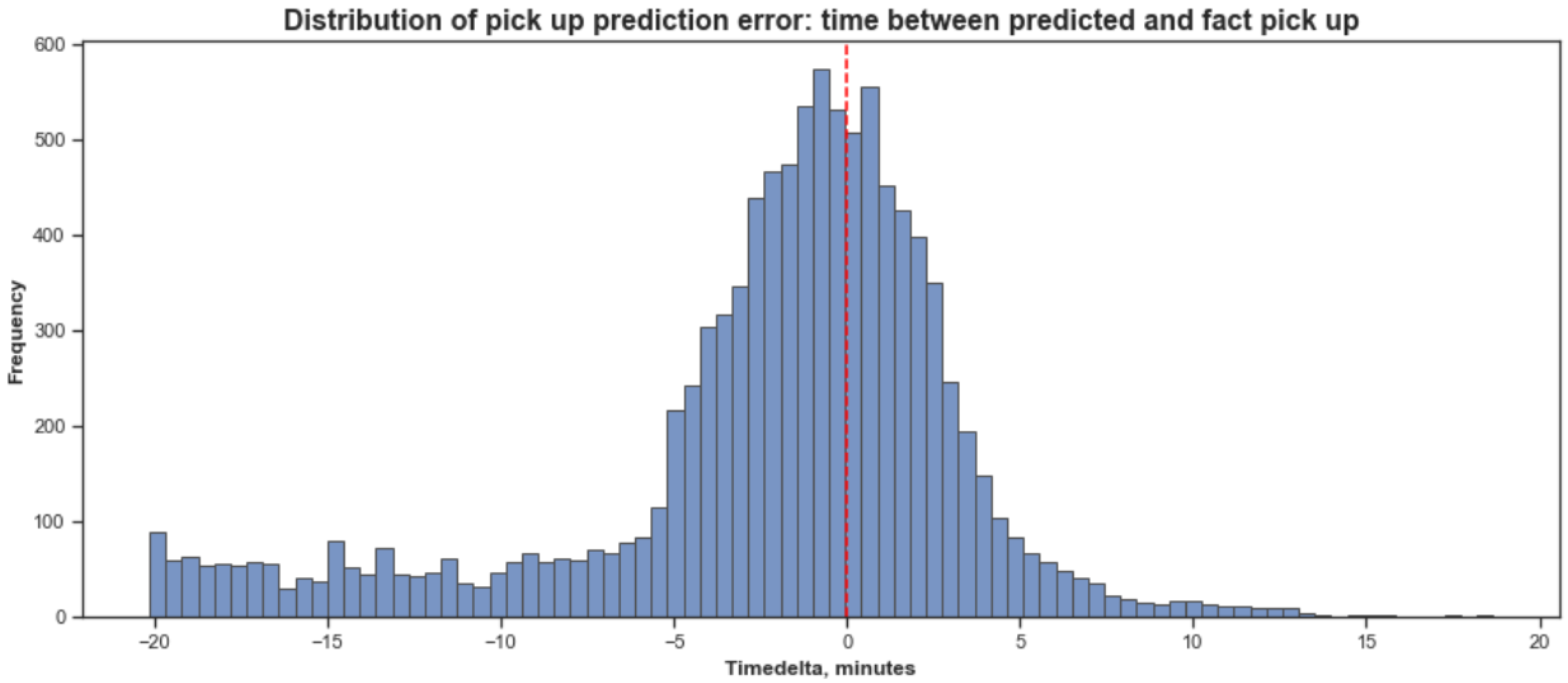


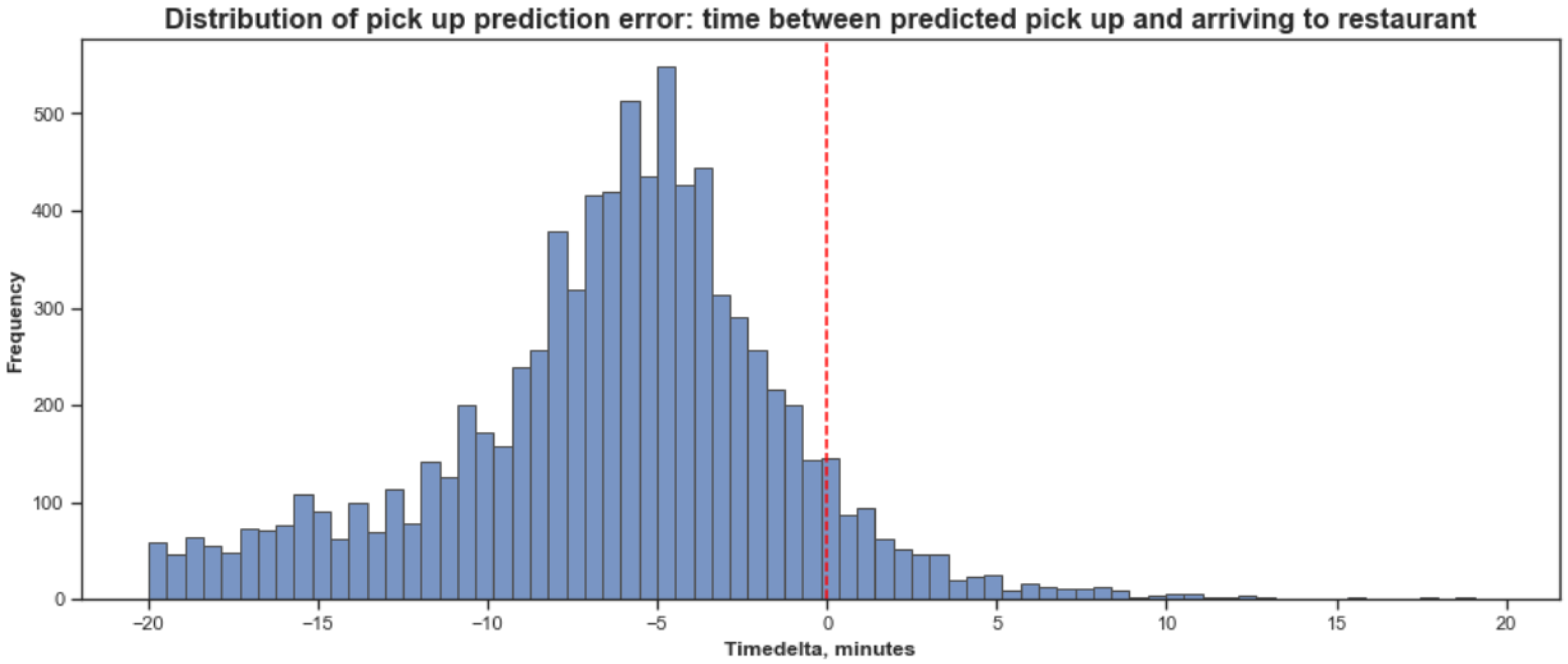


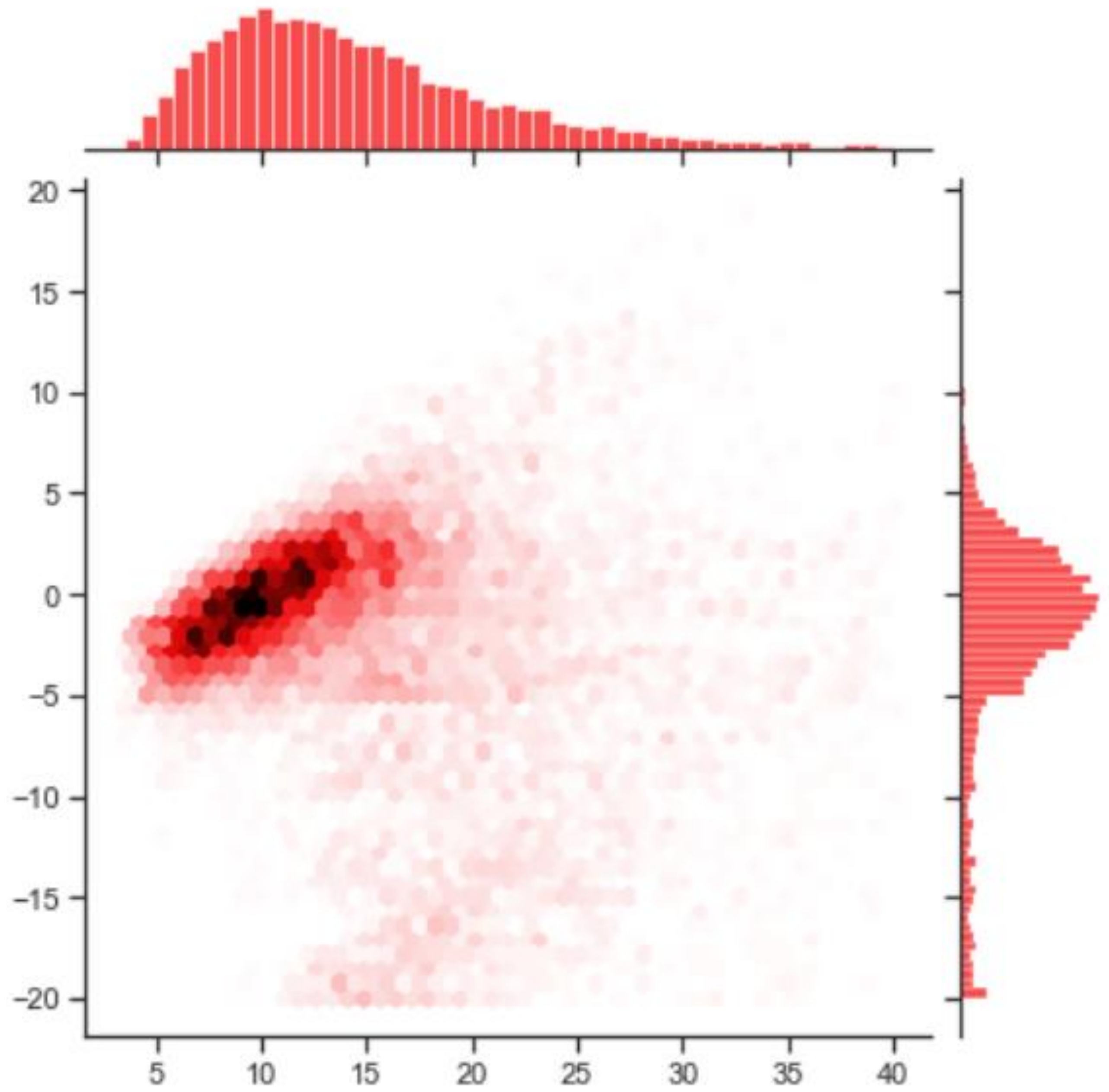
Timedelta of waiting the order at restaurant could be reduced by ~7 minutes

- Riders have to wait for the order for over 8 minutes on average: over 6 minutes in half of all orders and over 12 minutes in 20% of cases.
- We can notice it's a regular problem throughout the day.
- It could be solved by later assignment to an order. Currently, riders confirm orders almost at the same time as restaurants (sometimes even earlier).
- Also, we have a prediction model for the time when the order should be completed by the restaurant. Let's see how close it is.

Cooking_waiting: Automation



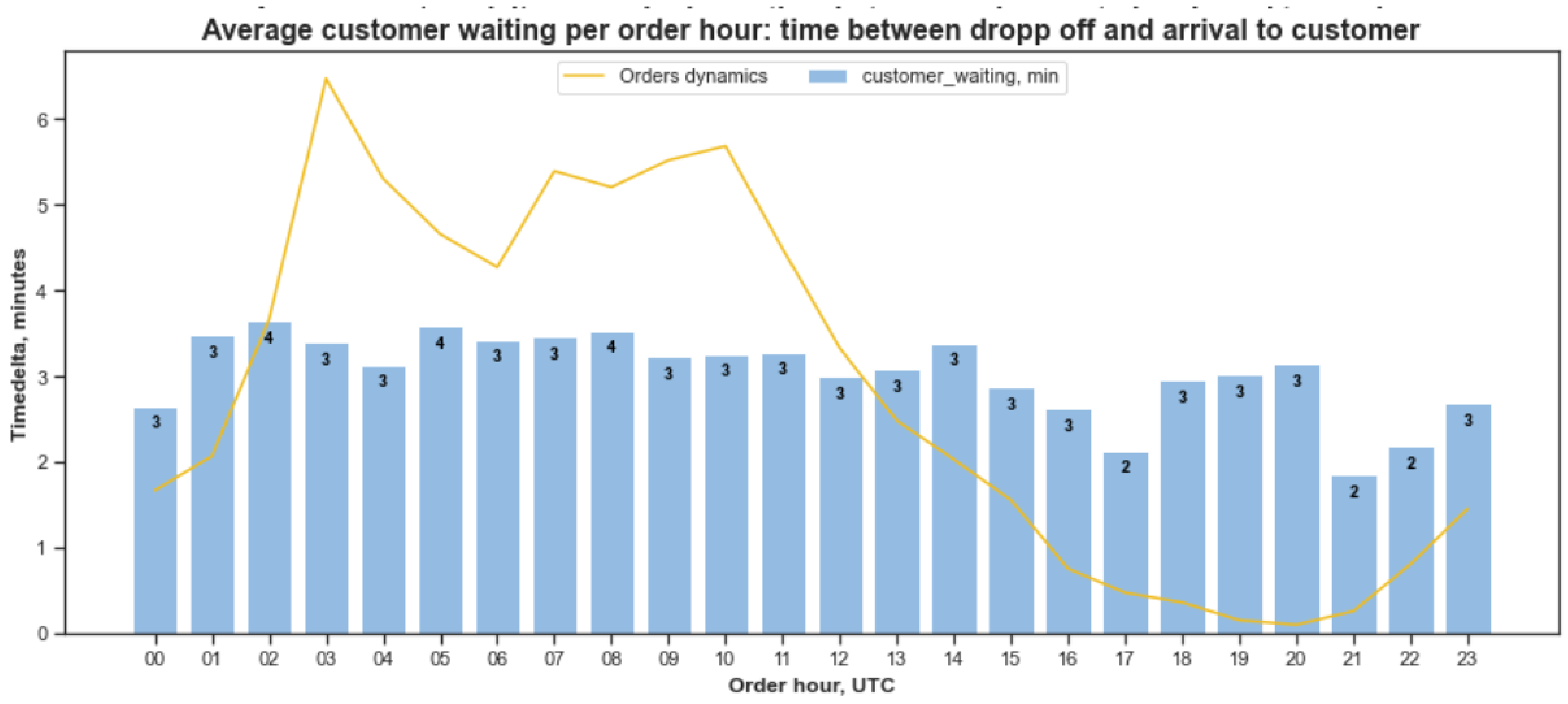


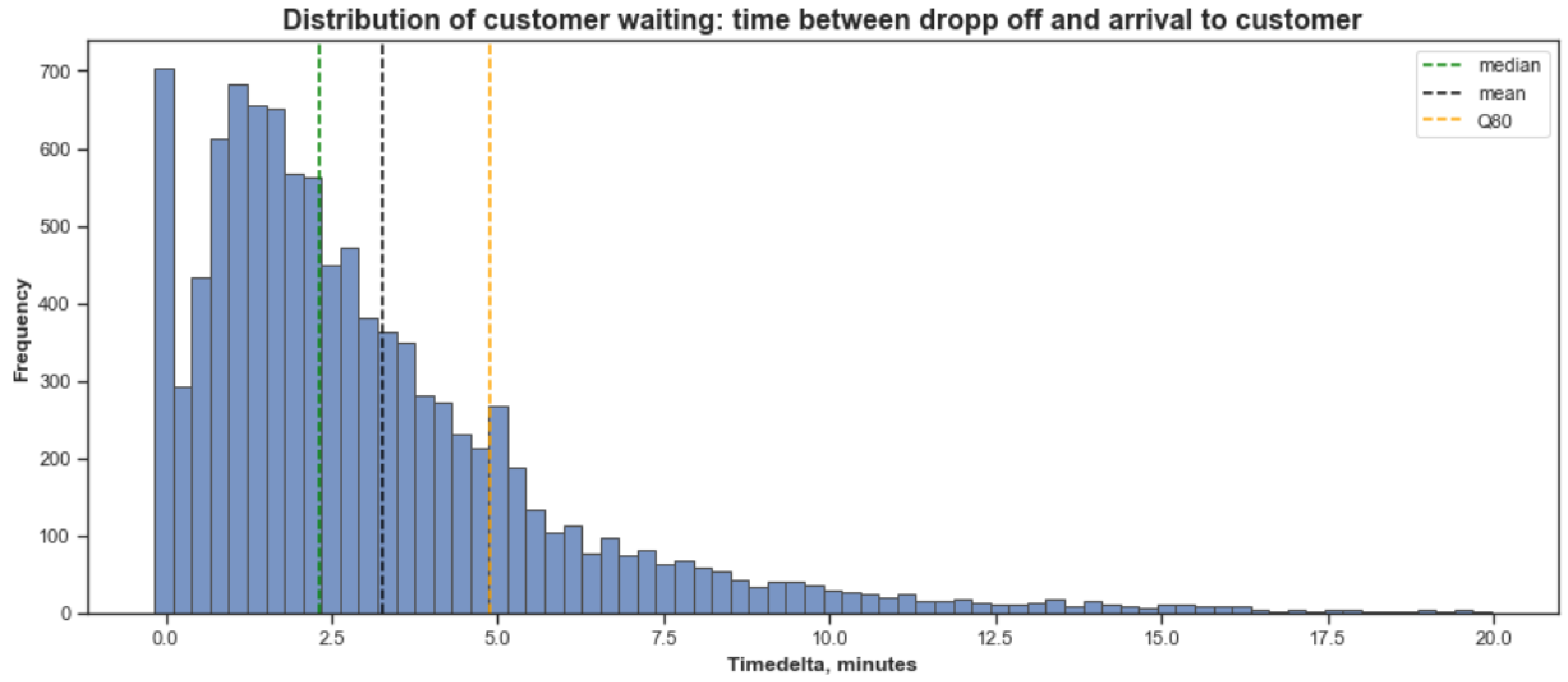


The timedelta between predicted and fact pick up and between predicted pick up and the rider's arrival shows the same bias of 6 min.

- Above is the correlation between cooking time (h) and prediction error (v). The prediction model could be improved, especially for long-cooked orders, but later rider assignment seems to be a more interesting solution.
- The right zone in the histogram means that the rider picked the order later than predicted; the left zone means earlier.
- From lower chart we can notice that riders arrive before prediction threshold in the biggest part of cases and have to wait the order at restaurant.

Customer_waiting: Automation

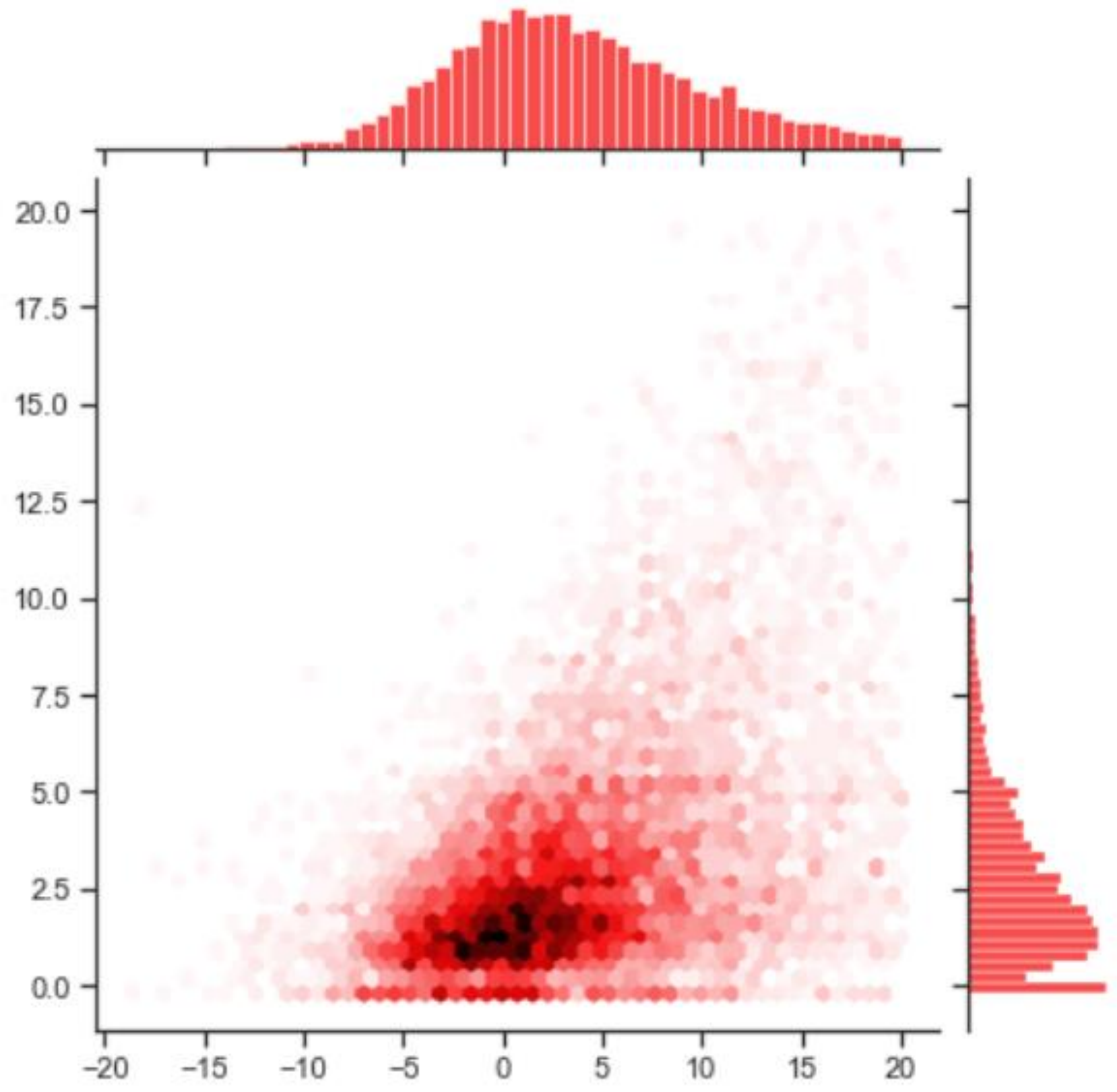


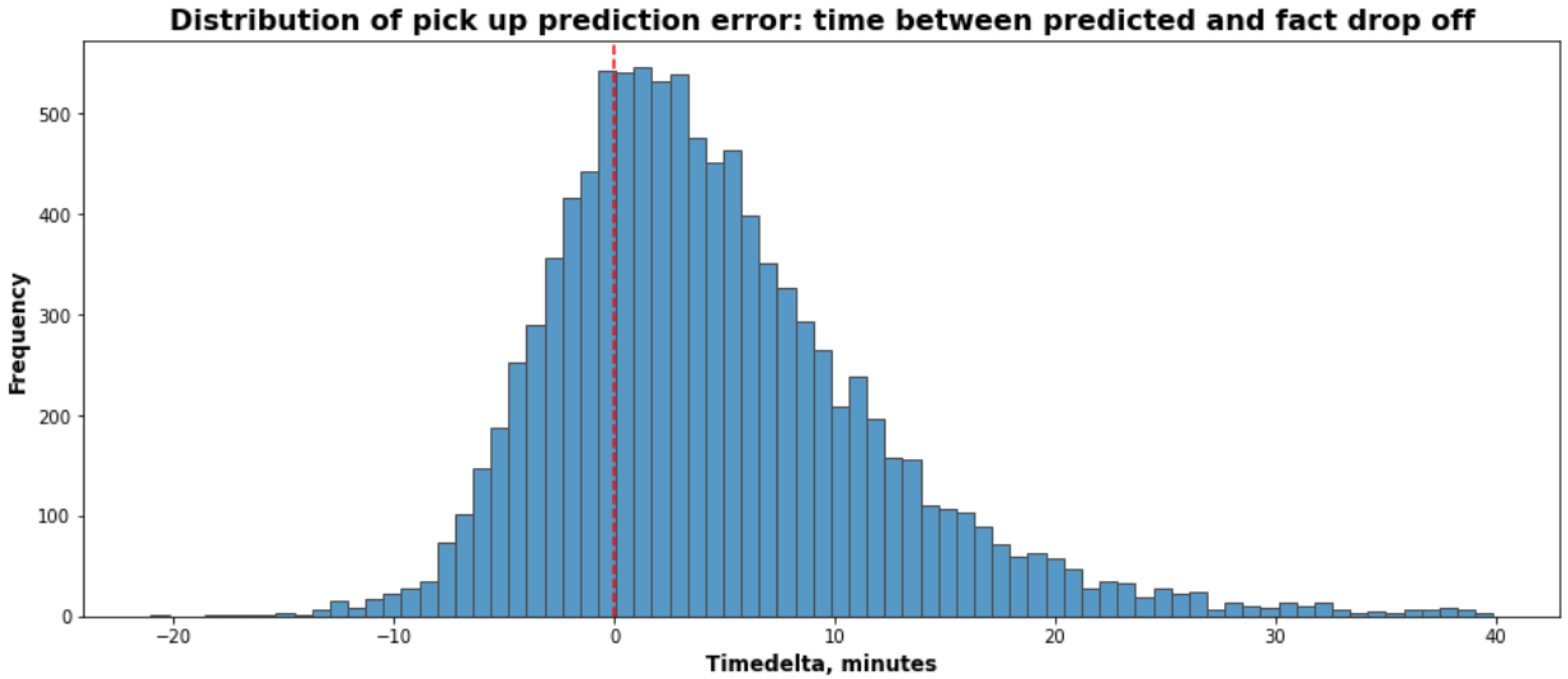


The timedelta between dropping off the order and arriving to customer could be reduced by ~3 minutes.

- Regardless the time of order, riders have to wait customer over 2,5 minutes in half of orders, over 3,3 minutes in average and over 5 minutes in 20% of orders.
- It could be solved by notification of customer about soon delivery.
- It could be solved by more detailed information about the delivery point: probably the rider doesn’t wait for the customer but calls and tries to find the entrance.
- We also have some prediction model for the entire delivery time, which is shown to the customer, let's see the correlation of its quality with this waiting time.

Customer_waiting: Automation





Timedelta of customer waiting correlates with error of predicted delivery – the more delay, the more minutes are spent for dropping.

- promised_delivery_time - initial promise we provide customer when order is placed/created using our prediction models.
- Seems that hungry and angry customers are not able for fast communication:)
- It could be reduced by the development of the delivery prediction model.
- It could be reduced also by fixing the firstly mentioned metric – system_delta, because it's a part of delivery time. It's not a part of UTR but can influence on customer waiting time.

UTR: Conclusions

UTR – orders based value:

According to the previously mentioned data, we can define the average UTR as 1 hour divided by the average delivery time. Excluding the system_delta (3.35 min mean) and starting delivery time for rider from rider_accepted_at till rider_drop_off, consider the **avg_delivery_time as 23,25 min and UTR as 2.57.**

The following improvements could increase the rider's performance:

Later rider’s assignment to order => decrease time of waiting at restaurant (~8 min);

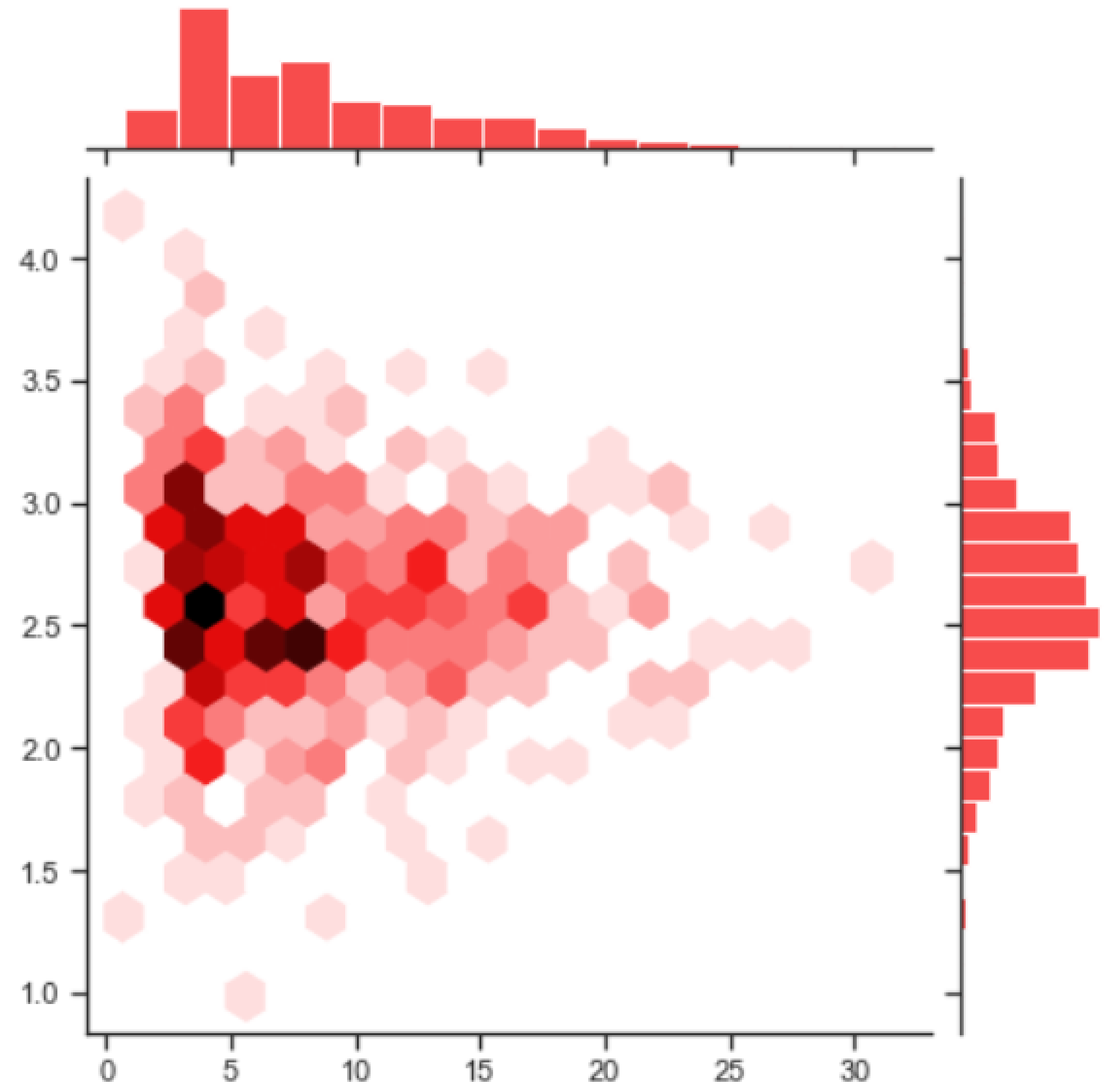
- Notifications close before delivery and fixing infrastructure cases of sending orders to restaurants => save time for waiting customer (~2 min).
- Improvement of both prediction models increase chances for success of two points above

In the most optimistic scenario, we will save **about 10 minutes** on average and **increase UTR to 4.5.**

UTR – shifts based value:

- 468 riders planned shifts for 2 days, 459 of them completed at least one shift, and all of them delivered at least one order.
- Average UTR based on worked shifts: 2.59
- Average UTR based on planned shifts: 2.43

I suggest it’s not a big difference, as well as for those riders who planned the shift but didn’t start the job. Moreover, the correlation between worked hours (h) and UTR (v) shows some distribution of riders by their quality, but two days of data seem to be insufficient for any conclusions.



The same I can say regarding supply-demand balance. So, I don’t think, that scheduling is the issue here. Proposals mentioned above should influence on all riders.