

McKinsey Hackathon

Team «Supersonic Potato»

Shtanko Leonid

Zhukov Alexander

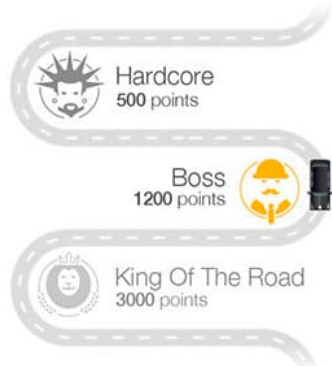
10/12/2017

Client Gamification

- **Gett** has a loyalty program for their clients
- Client travels by taxi, getting points and new statuses to have a chance to travel with best drivers
- And what about the **drivers**?

Loyalty Program


More rides mean more status points.
More status points mean you're more likely to ride with a top driver.



Be driven by the best

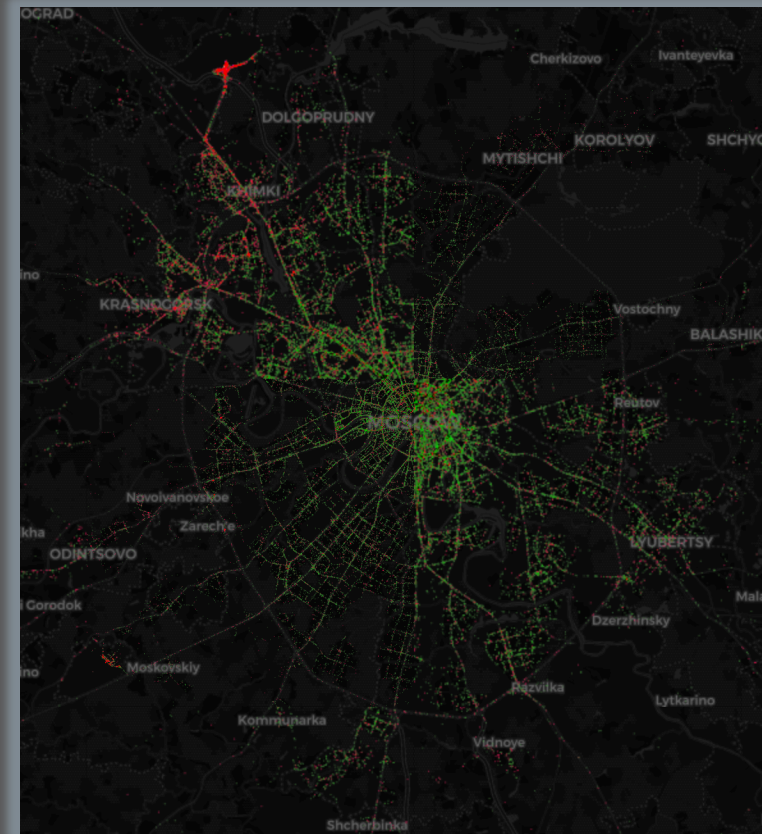
Free rides

Share the app with your friends and earn free rides



Driver Gamification

- Why do we need it?
 - Balancing of supply and demand between regions
 - Good balancing means customer satisfaction with the service
 - Motivation of drivers to work with **Gett**
 - Drivers' GH increases without business loss



Green – driver accepted the order

Red – driver refused the order

Solution

- Data:
 - Orders drivers
 - Destination text, origin text
 - Geostream
- Model:
 - Markov chain
- Goals:
 - Optimal strategy for driver (maximizing revenue)
 - Choosing such permutations of drivers between regions, that this balances supply and demand

Maximizing driver's revenue

Tree of scenarios traversal with estimating total income

$$\sum \gamma^i p(d_{i+1}|d_i, t) * cost(d_{i+1}, d_i) \rightarrow max$$

Maximizing common wealth

Brute-force (yet) way of getting optimal permutations

$$-cost_{empty\ ride} - p_1 * cost_1 + p_2 * cost_2 \rightarrow max$$

Driver Gamification strategy

- Motivate driver to accept “bad” order by promising to give him a “good” order for the next time (according to his points)
- Decreasing driver’s points because of declined orders (if declined for multiple times)
- Do nothing if order has high revenue for a driver
- Move driver to “good” region if he has high points

Churn identification & prediction for drivers

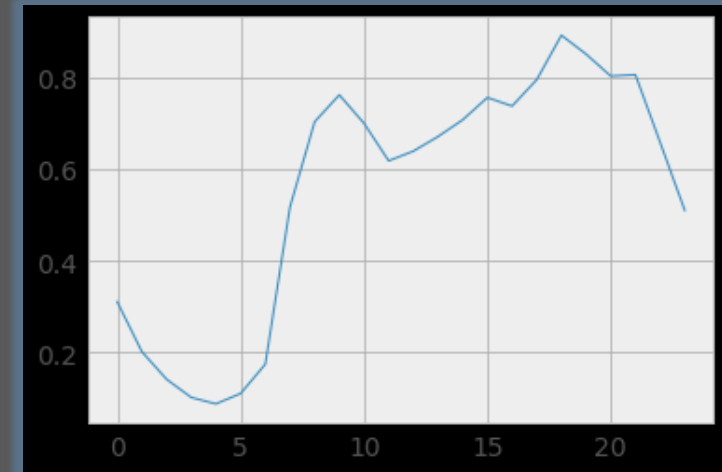
The drivers may migrate around Gett, Yandex and Uber. Most often it looks like dynamic change of Gett's share in separate drivers

- How we can identify and measure it?
- How we can predict and work with it?
- What the current situation and is it enough flexible parameter for identification during 4 weeks only?

Churn identification

- Consider standard demand for drivers during work day as probability to take order with **Gett** for different hours
- For each driver identify the border of his workday and predict expected value for this specific time period
- Calculate real number of orders with **Gett**
- Sum expected value during the week, repeat the same for real orders number
- Define **Gett** share for driver as real number of orders / expected number*

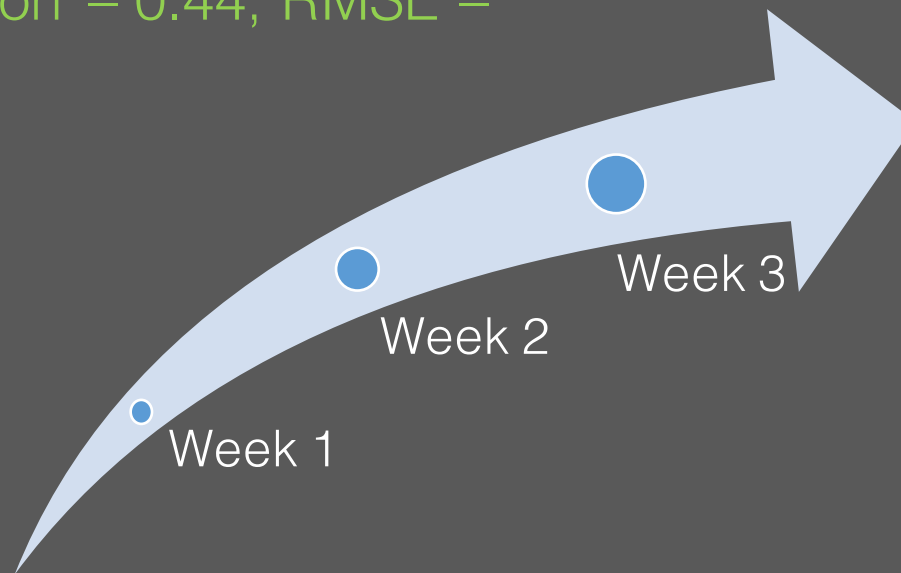
*rarely might be more then 100%



driver_gk	week1	week2	week3	week4	share trend
0	59%	69%	64%	65%	
2	85%	80%	51%	51%	
5	95%	60%	76%	63%	
7	77%	62%	73%	65%	
9	100%	74%	72%	46%	
10	29%	62%	74%	79%	

Churn prediction

- Create separate linear regression for each driver based on the first three weeks of **Gett's** share
- Predict the fourth week share on the driver level
- Check the quality (**correlation – 0.44, RMSE – 0.38**)
- Then → analyze



Churn prediction: conclusion

- The Gett's drivers share is enough flexible and predictable value
- The general picture is about high variation, but generally Gett's share is slightly decreased

Share changes	-0,2	-0,1	-0,05	0	0,05	0,1	0,2
% of drivers	12%	14%	11%	27%	12%	14%	10%
% Drivers revenue	9%	14%	12%	29%	13%	14%	9%

- The recommendations:
 - Be ready in advance to propose competitive proposal for your drivers under the risk
 - Find the optimal solution to prevent churn

Conclusion

- Gett should detect possible churning drivers in time
- That's where the Driver Gamification strategy can help
- The company revenue will increase with new great drivers
- The quality of service will always be great.

Some fun



Real customers insights

Average rated	Driver Android	Driver iOS
Rider Android	4.910	4.923
Rider iOS	4.884	4.913

The matching of mobile platform between riders and drivers are really influence on final rating. As we see Apple drivers are rated higher in average and by Android and by Apple riders, but the difference between Android and Apple drivers is three time higher for Apple riders.

- Typically Apple riders do lower rating
- Apple riders likes Android drivers much lower than Apple drivers

Thanks.

Supersonic Potato team