## **Used model:**

I have chosen to use a gradient tree boosted regression model from XGBoost library, because it is quick to setup and quickly fits to provide good predictions.

If I have more time I would:

* compare current model with Tensorflow DNN Regression model (for better modeling of dependencies between features dimensions),
* try a different target - quantile range instead single number (similar to this [link](https://towardsdatascience.com/regression-prediction-intervals-with-xgboost-428e0a018b)).

## **Model evaluation:**

I used GridSearchCV from Scikit-Learn library for tuning hyperparameters. I measured model performance by target (duration in seconds) errors improvement: Root Mean Square Error (RMSE) and Mean Average Error (MAE).

Best results for duration prediction errors on 20% of dataset (unseen during training):

* mae: 611.56867
* rmse: 860.86352

## **Data processing:**

Based on Exploratory Data Analysis (EDA), I have noticed that some records had empty values. I had to remove some rows if any important column was empty (e.g. actual delivery time). Other less important columns I have filled with its average value (e.g. num\_distinct\_items). Some values were clearly en errors (or outliers like record from date 2014-10-19), so I have also removed them from the training dataset. Since delivery time longer than 6 hours is very rare, I removed records with such a long delivery time for better accuracy on shorter times.

## **Features engineering:**

Due to my experience and nature of human demands, I have chosen:

* Numerical features with linear values (e.g. total\_items, num\_distinct\_items, total\_onshift\_dashers, total\_busy\_dashers etc.),
* Categorical features (one hot encoded) for dimensionality distinction between features like market\_id or store\_primary\_category,
* Calendar features (harmonic values extracted by trigonometric functions) for detecting correlation with day of week, day of month or happenings like holidays.
* Time features (harmonic values of minutes in day) for detecting rush hour or social time.

## Other:

If I would have a wider dataset (in perspective of created\_at column), calendar features would detect more dependencies between orders.

I would like to also check weather phenomenas correlation and maybe some geographical features (like locations and distance) if I have deliveries locations.