1. Pavel Kuzmin, Moscow Russia. Graduated from Moscow University of Physics and Technology (MIPT) in 2013. Worked in the field of statistical analysis, automatic control, dynamic and statistical modeling. Now data scientist in Severstal Digital (subsidiary company of Severstal - Russian steel and mining company). Specialization now is predictive maintenance of equipment.

2. A lot of problems of anomaly detection in the field of physical processes can be solved this way. Chosen several parameters, which mostly influence the energy consumption, created the model, that predicts consumption by these parameters and then labeled data, that is almost unpredictable as anomalies. Please refer to parts I-III of the report.

3. Three pieces of code:

a. part of feature engineering:

```
{df['month_sin'] = np.sin(df['month']*2*pi/12)
 df['month_cos'] = np.cos(df['month']*2*pi/12)
 df['day_sin'] = np.sin(df['day']*2*pi/30)
 df['day_cos'] = np.cos(df['day']*2*pi/30)
 df['weekofyear_sin'] = np.sin(df['weekofyear']*2*pi/52)
 df['weekofyear_cos'] = np.cos(df['weekofyear']*2*pi/52)
 df['dayofyear_sin'] = np.sin(df['dayofyear']*2*pi/365)
 df['dayofyear_cos'] = np.cos(df['dayofyear']*2*pi/365)
 df['hour_sin'] = np.sin(df['hour']*2*pi/24)
 df['hour_cos'] = np.cos(df['hour']*2*pi/24)}
```

Unusual encoding for time data as sin and cos of the periode. This helped to increase the quality of KNN regressor, because shows the distance between objects correctly.

b.

```
{train['high_temp'] =train ['Temperature_mean_date']>17
train.loc[((train['woking_day']==True)&(train['high_temp']==True)&(train['is_holiday']==False)),'model_num'] = 1
train.loc[((train['woking_day']==True)&(train['high_temp']==False)&(train['is_holiday']==False)),'model_num'] = 2
train.loc[((train['woking_day']==False)&(train['high_temp']==False)&(train['is_holiday']==False)),'model_num'] = 3
train.loc[((train['woking_day']==False)&(train['high_temp']==True)&(train['is_holiday']==False)),'model_num'] = 4}
```

Finding the right threshold of temperature when machines of the building with high consumption start their work helped to build fine models for consumption.

c. {train.is_abnormal = (train.is_abnormal_1&train.is_abnormal_2)}
This short line increased the model performance greatly (according to the public leader-board). It show that we label as anomalies only points labeled as anomalies by both models.

4. Before using MLP as one of models tried to use huber regressor. Changed it to MLP because could not find nice way to engineer features for linear model.
   Tried write the form of the finction for consumption over datetime and temperature in the exact form and searched parameters with scipy.optimize. Not helped.

Tried RNNs to predict consumption like for time series data. I guess it can help though I could not find nice solution.
Also tried to build a big model for all buildings or at least one for each building. Did not manage to do it.

5. No

6. I used MAE to fir models. And tried not to overfit with extreme high regularisation.

7. I used a laptop with 4 cores and 16 Gb of RAM. This is enough to run all four notebooks one by one. To get the same submission as my final one just see REEDME.md and follow instructions.

8. See useful graphs in reports/figures , in notebooks (there are more in notebook 1.0-pk-…), or in my report.

9. See part IV of report:

   - More precise information about holidays can improve the models quality extremely.
   - The information about the equipment operation modes can also help to improve the model. For example the temperature of turning the central conditioning system or heating systems on and off.
   - Exploring the possible usage and the goals can help in feature engineering. For example understanding which type of abnormal behavior is really abnormal for our needs.
   - One more model can be added to catch low and high frequency processes as long term and shot-term non-stationarity.
   - Looking on the problem as time-series task and search not anomalies for all the dataset, but search outliers based on previous seen data only – marking only previously unseen data.