**Pipeline:**  
I transformed the date variable to day of week, week of year.  
I kept the year out of the catboost because tree models do not work well with unseen values like future years (even though the test set is just from 2017).  
I added a linear regression which receive the year and promotion as inputs and predicts a sale, this should accommodate for the linear increase in sales.  
From this prediction the catboost make a correction.  
I navigated through the data and made plots available in the PowerPoint, where I found predictive power in the store.csv meta data, for example type column.  
I transformed the store\_id and sku\_category columns into one-hot-encoded vectors and did nothing for the store.csv parameters – will use ordered boosting while training.  
**Model:**  
I chose Catboost as my model, which is known for performing well on tabular data with categories, doesn't require standardization and I used It's ordered boosting for the store.csv columns. I tried different kinds of hyper parameter: depth (5, 10, 15), lr (0.05, 0.1, 0.2)  
The training starts with learning rate of lr and then another batch of training with lr/2. Plots in power-point. I did the hyper-parameter search with the whole 2017 (annotated) data as validation set. Also I tried to give a bit more weight for later years 2013-1,2014-1.1,2015-1.2,2016-1.3,2017-1.4. The plots are at the end of the power point. Eventually I chose for the final model to use lr=0.1, depth=15 and no weighting. This hyper parameter achieved R^2 of 0.929 on validation set of the entire labeled 2017 data.  **Horizon:**To check my models' horizon, I generated a mean error vs day (plot in the power point). The errors are relatively uniform across time, so the final model should be able to handle long time distance (~1 year) with good accuracy (~0.93 R^2).  
**Performance:**  
I used the last 1000 samples as validation set.  
The r^2 value on them was a bit over 0.97. **Plotting:**  
I generated the graph of the validation real vs prediction and saved the test predictions.

