*Konstantin Sozykin, Bachelor 3rd course, group BS3-5*
*gogolgrind@gmail.com*
*k.sozykin@innopolis.ru*
***Supervisor: Rauf Yagfarov, r.yagfarov@innopolis.ru***

**Kobe Bryant Shot Selection**

*Challenge description*

Kobe Bryant marked his retirement from the NBA by scoring 60 points in his final game as a Los Angeles Laker on Wednesday, April 12, 2016. Drafted into the NBA at the age of 17, Kobe earned the sports highest accolades throughout his long career.
Using 20 years of data on Kobe's swishes and misses, can you predict which shots will find the bottom of the net? This competition is well suited for practicing classification basics, feature engineering, and time series analysis. Practice got Kobe an eight-figure contract and 5 championship rings.
It is binary classification task, but main goal of this challenge it's predict probabilities of shot's and miss made by Kobe Bryant.

*Plots and visualizations*

Figure 1 consist distribution of miss and shots in Cartesian Plane. Figure 2 preset dependency between shot distance and distance calculated in euclidean metric. We can use on of them, I decided use second case. It is interesting normal in baseball there are 4 periods, but in dataset maximal periods is seven. And it can be transform to good binary feature. See Figure 3.

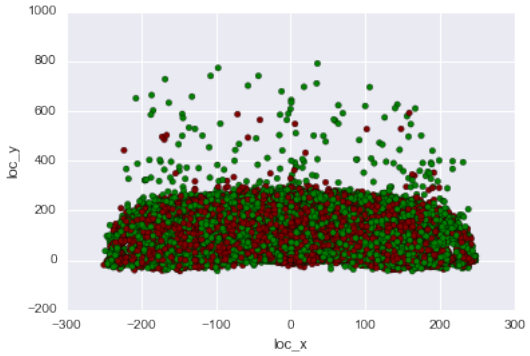



Figure 1: Visualization of target variable on Cartesian Plane, red - miss, green - shots

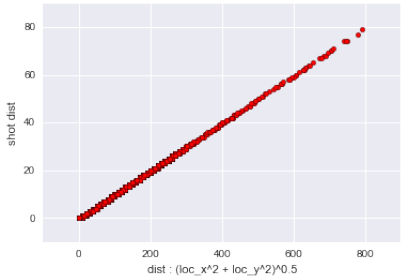



Figure 2: Correlation of shot_distance and self computed distance

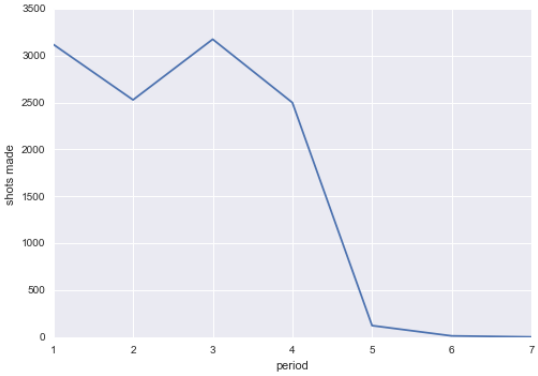



Figure 3: Frequency of shots in every period

*Feature engineering*

| Featute name | Using in the final model |
|---|---|
| action_type | Megred with combined_shot_type |
| combined_shot_type | Megred with action_type |
| game_event_id | Deleted |
| minutes_remaining | Megred with seconds |
| seconds_remaining | Megred with minutes |
| period | It is used as is |
| playoffs | It is used as is |
| season | It is used as is |
| shot_type | Deleted |
| shot_zone_area | Deleted |
| shot_zone_basic | Deleted |
| shot_zone_range | Deleted |
| team_name | Deleted Kobe had one team |
| game_date | Deleted |
| matchup | Used only create away feature by find @ symbol. See comments |
| opponent | It is used as is |
| shot_id | Delted |

This table present features witch I use to make final prediction. And there are some additional comments.
Design witch feature will drop was heuristically, based on detail reading of data in Excel.
In addiction following binary features was created:

$$not_home$$

- if machup string consist @ symbol it means, that Kobe play not at home

*last*

- According to figure 3, it period ¿ 4, there small change make shot.

*time*

- Computed as 60 * minutes + seconds

*angle*

- Part of polar coordinate plane, idea was got from one public scripts

Important, that the most features was interpreted as categorical. Another hypothesis was use **shot_zone** features instead of coordinates. It's not change performance

*The best and alternate models*

| Estimator name | CV Log Loss Score | Parametrs |
|---|---|---|
| Logistic Regression | 0.6114 | C:0.7 |
| Random Forest | 1.1016 | default |
| Graditent Boosting Tress | 0.6036 | learning_rate:0.1 max_depth:5 max_features:log2 estimators:200 |
| XGBoost | 0.6030 | learning_rate: 0.05 max_depth:4 estimators: 200 subsample:0.8 |
| Neural Network | 0.61366 | See Figure 4 |

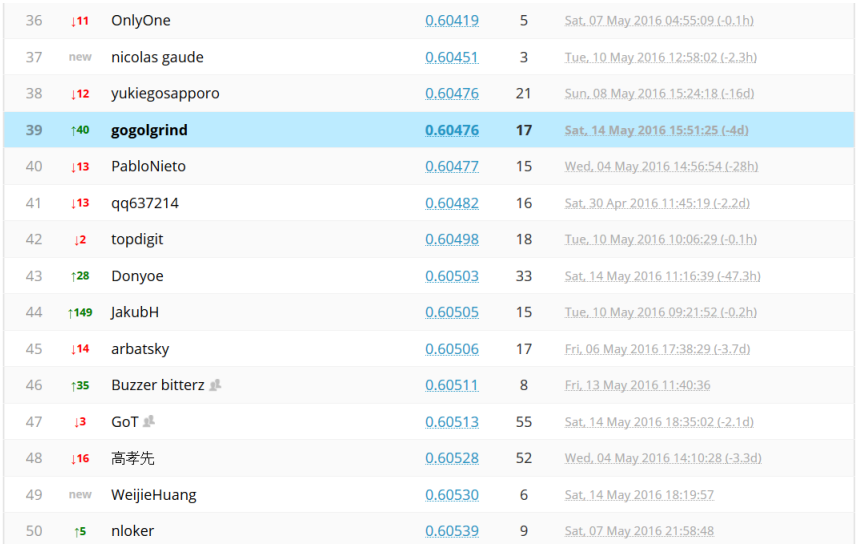
```
layers0 = [('input', L.InputLayer),
           ('hidden1', L.DenseLayer),
           ('dropout1', L.DropoutLayer),
           ('hidden2', L.DenseLayer),
           ('output', L.DenseLayer)]
clf3 = NeuralNet(layers=layers0,
                 objective_loss_function  = obj.categorical_crossentropy,
                 input_shape=(None, X_train.shape[1]),
                 hidden1_num_units=16,
                 dropout1_p=0.5,
                 hidden2_num_units=16,
                 output_num_units=2,
                 output_nonlinearity=act.softmax,
                 update=upd.sgd,
                 update_learning_rate=0.02,
                 eval_size=0.2,
                 verbose=0,
                 max_epochs=50)
```

Figure 4: Topology and parameters of NN

Metric for this competition is log loss. And according to table gradient boosting tree has best result. Also I try to implement stacking, use this topic as tutorial :

$$https://vk.com/wall-72870626\_4612$$

But it not got more better result than single model yet. I need to try different configurations. Some comments about Neural Network. It was implemented it Lasagne Framefork (with Theano Backed). It consist 3 layers, layer two is dropout (used for dimensionality reduction). Unfortunately it has performance similar to simple logistic regression.



| | | | | | |
|---|---|---|---|---|---|
| 36 | ↓11 | OnlyOne | 0.60419 | 5 | Sat, 07 May 2016 04:55:09 (-0.1h) |
| 37 | new | nicolas gaude | 0.60451 | 3 | Tue, 10 May 2016 12:58:02 (-2.3h) |
| 38 | ↓12 | yukiegosapporo | 0.60476 | 21 | Sun, 08 May 2016 15:24:18 (-16d) |
| **39** | **↑40** | **gogolgrind** | **0.60476** | **17** | **Sat, 14 May 2016 15:51:25 (-4d)** |
| 40 | ↓13 | PabloNieto | 0.60477 | 15 | Wed, 04 May 2016 14:56:54 (-28h) |
| 41 | ↓13 | qq637214 | 0.60482 | 16 | Sat, 30 Apr 2016 11:45:19 (-2.2d) |
| 42 | ↓2 | topdigit | 0.60498 | 18 | Tue, 10 May 2016 10:06:29 (-0.1h) |
| 43 | ↑28 | Donyoe | 0.60503 | 33 | Sat, 14 May 2016 11:16:39 (-47.3h) |
| 44 | ↑149 | JakubH | 0.60505 | 15 | Tue, 10 May 2016 09:21:52 (-0.2h) |
| 45 | ↓14 | arbatsky | 0.60506 | 17 | Fri, 06 May 2016 17:38:29 (-3.7d) |
| 46 | ↑35 | Buzzer bitterz | 0.60511 | 8 | Fri, 13 May 2016 11:40:36 |
| 47 | ↓3 | GoT | 0.60513 | 55 | Sat, 14 May 2016 18:35:02 (-2.1d) |
| 48 | ↓16 | 高孝先 | 0.60528 | 52 | Wed, 04 May 2016 14:10:28 (-3.3d) |
| 49 | new | WeijieHuang | 0.60530 | 6 | Sat, 14 May 2016 18:19:57 |
| 50 | ↑5 | nloker | 0.60539 | 9 | Sat, 07 May 2016 21:58:48 |

Figure 5: My current rank

My nickname at kaggle is gogolgrind. Solution described here get rank around top 50 from 366 participants. There are one mount till end of challenge. And I will try to improve my result. I think, there are two ways to do it. First, to find new features by more detail analysis. Second is understand how to use in right way ensemble technique like stacking and etc.