Predicting rare decay phenomenon $\tau^- \rightarrow \mu^-\mu^-\mu^+$

achm

12 Oct, 2015

Abstract

The LHCb collaboration provides simulated and real data used in the search for lepton flavor violating decay phenomenon $\tau^- \rightarrow \mu^-\mu^-\mu^+$. This document describes the details of an ensemble model using three model differ in predictivity power, ensemble weight based on the underlying distribution of features. The methodology makes no use of agreement data, mass data, or any local evaluation of agreement test and correlation test.

1 Introduction

The document will present the motivation behind the model design, details of each model, how to ensemble them, results of local score and public leaderboard final score.

1.1 Motivation

Since the classifier is trained on mix of simulated signal and real data background, it is required to pass the agreement test such that the classifier does not have a large discrepancy on real and simulated data. This is one of the most challenging tasks in this competition since the rule required winner of the competition must not make any usage of agreement data, the prediction task is now subjected to an unknown constraint.

In order to understand the constraint, lots of trials were made in order to understand what kind of model or features will in general fail to pass the agreement test. Mass, Production, min_ANNmuon are removed during the prediction task. It's found that number of hits in the SPD detector (SPDhits) provides high predicitivy power based on local evaluation but once it's included the model will have a high KS score - How to make use of this feature will definitely be the key part on winning the competition.

As the competition goes by, the modeling framework was separated into three categories, namely, Weak, Semi-Strong and Strong. As its name suggest, they are model having weak-to-strong predictivity and low-to-high KS score in general. The prediction is based on the ensemble of three models.

1.2 Hardware & Software

The task was run under a windows 7 machine, Intel i7-4770 with 16 GB ram and NVIDIA GTX750 Ti. Python 2.7 is the only language with the following libraries:

xgboost: https://github.com/dmlc/xgboost

keras: https://github.com/fchollet/keras

scikit-learn: https://github.com/scikit-learn/scikit-learn

hep_ml: https://github.com/arogozhnikov/hep_ml

NumPy: https://github.com/numpy/numpy

Pandas: https://github.com/pydata/pandas

SciPy: https://github.com/scipy/scipy

1.3 Weak model

The weak model is an ensemble of two models:

1. NN (keras) without the use of SPDhits
2. An ensemble model inspired by Kaggle community

$$Pred_{Weak} = \frac{a_1 * Pred_1 + a_2 * Pred_2}{10}, where\ a_1 + a_2 = 10$$

The first model gives 0.9835~0.9840 score and the second model gives 0.9835~0.9840 score on public leaderboard. The average prediction $(a_1, a_2 = 5)$of two models gives 0.9845~0.9850 score on public leaderboard.

1.4 Semi-Strong model

The semi-strong model is an ensemble of two models:

1. Gradient boosting (xgboost) without the use of SPDhits
2. NN (keras) without the use of SPDhits without the use of SPDhits

The data set was split into 5 sets and cross-validation is performed, where the evaluation set contains data with min_ANNmuon>0.4 only. Hyper-parameters optimization is performed via random search and seed changing based on the score obtained on weighted area under the ROC curve on the leave out evaluation data set. Each method will generate 5 models and therefore 10 models are generated. The prediction is the averaging of 10 models.

The first model gives an average score of 0.9875 ~ 0.9885 and the second model gives an average score of 0.9867~0.9880 on local evaluation. The ensemble of two models is based on the formula as follow

$$Pred_{Semi-Strong} = \frac{b_1 * Pred_1 + b_2 * Pred_2}{1000}, where\ b_1 + b_2 = 1000$$

The best model gives average score of 0.9880~0.9890 on local evaluation. Since the KS score gives 0.090 ~ 0.1050, the model cannot be used alone but very likely can be used if agreement data is available via grid search on the KS score.

1.5 Strong model

The strong model is an ensemble of two models:

1. Gradient boosting (xgboost)
2. NN (keras)

The data set was split into 5 sets and cross-validation is performed, where the evaluation set contains data with min_ANNmuon>0.4 only. Hyper-parameters optimization is performed via grid search and seed changing based on the score obtained on weighted area under the ROC curve on the leave out evaluation data set. Each method will generate 5 models and therefore 10 models are generated. The prediction is the averaging of 10 models.

The first model gives an average score of 0.9923 and the second model gives an average score of 0.9918 on local evaluation. The ensemble of two models is based on the formula as follow

$$Pred_{Strong} = \frac{c_1 * Pred_1 + c_2 * Pred_2}{1000}, where\ c_1 + c_2 = 1000$$

The best model gives average score of 0.9927 on local evaluation. Since it cannot pass the agreement test, the model cannot be used alone.

1.6 Ensemble

Originally different stacking parameters are used as the following simple formula:

$$d_1 * Pred_{Weak} + d_2 * Pred_{Semi-Strong} + d_3 * Pred_{Strong}, where\ d_1 + d_2 + d_3 = 1$$

Later on, a better approach on weighting the ensemble model based on the similarity measure on the distribution of features, using clustering or classification likes K-Nearest Neighbors.

In laymen terms, treat test set as a new observation. If the observation happened before, use strong model; if the observation is uncertain, use semi-strong model; if the observation is new observation with no previous similar result, use weak model.

In order to learn the similarity of data θ, train data are all denoted as 1 and test data all denoted as 0, followed by a classification model (NN is used here). Let the prediction of the similarity be $\theta, where\ 0 \leq\ \theta \leq 1$, the ensemble can be

$$Pred = \big(1 - f(\theta)\big) * Pred_{Weak} +\ \rho * f(\theta) * Pred_{Semi-Strong} + (1 - \rho) * f(\theta) * Pred_{Strong},$$

$$where\ 0 \leq\ \rho \leq 1\ and\ f\ is\ the\ user - defined\ function\ s.t. 0 \leq\ f(\theta) \leq 1$$

Different f were tried, such as linear, hyperbolic tangent, direct cutoff, etc. My final decision, is a sigmoid function with two parameters target α and slope control β, with following formula

$$f(\theta) = \frac{1}{1 + e^{-(\theta-\alpha)*\beta}}$$

2 Final Result

Final submission parameters are as followed:

Parameters	Submission 1	Submission 2
a1, a2	0, 10	5, 5
b1, b2	Optimized	Optimized
c1, c2	Optimized	Optimized
d1, d2, d3	-	-
ρ	0	0
α	0.1435	
β	25	50

Results on leaderboard:

Submission	Public	Private
1		
2		

3 Acknowledgement

4 References

[1]: http://arxiv.org/pdf/1409.8548.pdf Roel Aaij et al., Search for the lepton flavour violating decay $\tau^- \to \mu^-\mu^-\mu^+$, 2015, JHEP, 1502:121, 2015

[2]: https://kaggle2.blob.core.windows.net/competitions/kaggle/4488/media/lhcb_description_official.pdf Flavour of Physics, Research Documentation

[3]: https://xgboost.readthedocs.org/en/latest/ XGboost documentation

[4]: http://keras.io/ Keras documentation