Introduction

There are two main metrics which everyone would like to maximize: average payout and payout sharpe. Since the last MMC update we have an opportunity to choose between pure CORR payout and CORR+MMC payout. An optimal payout scheme and variance between training, validation and live metrics are main topics of this post.

Methods

All training data were predicted using 2-fold CV with dividing data into two ranges of 1-60 and 61-120 eras. Validation data were generated using model fitted on 1-120 eras range. Example prediction model, LinearModel and 80 random XGBoost and NN-based models were provided by the Numerai team. LightGBM models were generated by myself using 15 different hyperparameters. Every LGB model has 4 variations: raw model (_3 marks in the text below), 100% feature neutralized (_0) model, and 2 special feature neutralization procedures which will not be revealed in the text (_2 and _6). These marks are related to my accounts jackerparkerX, where X can be replaced with 3, 2 or 6. These accounts contain models generated in the similar way for the live rounds 214-227. Keep in mind that live data estimations in the text below were done two weeks ago in the middle of August.

Multiple MetaModels were generated for groups of predictions (MM of XGBoost predictions, MM of LGB_2 models, etc). Ensemble was done using averaging of prediction ranks for every single model. This formula was chosen based on my previous experience with combining different models and it is not the topic of the current study. However, there are a lot of other ways to combine models: simple averaging, weighting averaging based on mean or sharpe of single models and much more.

Additionally, full MetaModel (MM_full) was generated using all predictions used in this analysis. This MetaModel was used for MMC calculation.

Results

The first important question raised under this analysis was which predictions should be used as MM for MMC calculation. Example predictions are often used as a proxy for the MM. On the other hand, “real” MM is generated using all staked predictions where most of them are worse than Example predictions in terms of live performance. MMC calculation on MM_full seems to be a more accurate proxy of “real” MM. Difference between MMC calculated in both ways is shown in Figure 1. MMC values calculated using Example Prediction as “real” MM are higher than ones calculated using MM_full, but values have a strong correlation. The MMC_full is used for MMC calculation for the rest of this study.

[

Fig1

1288×665 76.3 KB

](https://forum.numer.ai/uploads/default/original/1X/44761f43fc9b6755fa5316002a687f59f6770e41.png)

Figure 1.

Difference in training_mmc_mean, training_mmc_sharpe, val_mmc_mean and val_mmc_sharpe for two methods of MMC calculation: using ExamplePredictions or MM_full as a proxy for real MetaModel. Yellow dots (base label) are models provided by the Numerai team; labels 0, 2, 3, and 6 represent LightGBM models described in the method section.

Another important question is how well validation or training statistics correlate with each other. The correlation between training and validation data is shown in Figure 2. Function y=x was plotted to simplify analysis of relative performance between training and validation data. All dots placed below this line represent models where training performance is higher than validation one. While training and validation data highly correlate for CORR performance, MMC performance has ultimately low correlation here. One of the possible explanations is market regime difference between training and validation eras. For example, the second half of validation eras were chosen artificially by the Numerai team to extend validation data with some “hard” eras. That can lead to selection bias which also increases regime difference. All models with feature exposure close to Example predictions (~0.075) fit the same linear distribution between training and validation MMC, CORR and MMC+CORR values. Models with low feature exposure (_2 and _0) have higher validation metrics than expected by common trend. Example prediction model has a significant CORR drop in validation eras which increase MMC performance of low feature exposure in validation data. However, such models have a very low MMC in training data. It should be noted that such behaviour can be also explained by overfitting in training eras, what is additionally confirmed in the live data section below.

[

Fig2

1287×948 103 KB

](https://forum.numer.ai/uploads/default/original/1X/a64ca3d2fc4b263872913f6176028a064ad4a110.png)

Figure 2.

Scatter plots of training vs validation metrics for COR, MMC and MMC+CORR.

Standard deviations of MMC, CORR and MMC+CORR mean values for validation and training data are shown in Figure 3. MMC std values are much higher than MMC mean values which leads to low MMC sharpe in general and high sharpe difference between validation and training data. So, one should be aware of the high probability that MMC validation sharpe could be not representative.

[

Fig3

1288×320 42.6 KB

](https://forum.numer.ai/uploads/default/original/1X/c4961415a9f9fd3a0c534d44917d9dedd06c1f5a.png)

Figure 3.

Standard deviation of metrics between training and validation eras.

MMC+CORR mean values are higher for most models than pure CORR mean performance for both training and validation data (Figure 4). That means that in general most models will get an extra payout in average by choosing MMC+CORR scheme vs pure CORR payout. However, sharpe difference between these payout schemes shows that only one group of models (_6) will increase sharpe, while the rest of models will get decreased sharpe for MMC+CORR.

[

Fig4

1288×645 79.9 KB

](https://forum.numer.ai/uploads/default/original/1X/12a5f3d6c1df66de7e847505ff1319da1409e8e7.png)

Figure 4.

Scatter plots for pure CORR vs MMC+CORR metrics for validation and training data.

The analysis of relative change in performance was done for 4 meta-models for groups (_0, _2, _3, _6) of LGB models. The percentage change in MMC_mean, MMC_sharpe, CORR_mean and CORR_sharpe was calculated between the “best” LGB model and MM based on 15 LGB similar models generated using different hyperparameters (Figure 5). “Best” model was chosen by training_CORR_sharpe maximization in the RandomCV approach. Training metrics in most cases are better in MM compared to single best model. However, all validation metrics decreased in case of MM. Interpretation of these results is not clear, but in my opinion that could be a signal of overfitting in training eras.

Figure 5.

Percentage difference in MMC_mean, MMC_sharpe, CORR_mean and CORR_sharpe between group MM and best model in the group.

Another interesting metric which can be studied is correlation between CORR and MMC scores per era (Figure 6). Most of the models have a strong correlation between this metric in validation and training data, but some groups of models are outliers here and show a negative correlation between CORR and MMC in the validation eras. A negative correlation leads to high sharpe for MMC+CORR payout scheme. There is also a trend that models with lower correlation between CORR and MMC show a higher validation CORR and MMC mean values.

[

Fig6

1287×1142 76.2 KB

](https://forum.numer.ai/uploads/default/original/1X/3318ac5d36a39fb05fad25db5a19d3d854ea9097.png)

Figure 6.

Scatter plots for correlation between CORR and MMC scores per era in training and validation data.

Live data performance. Example predictions live performance was estimated using 56 live rounds data (168-224) and 10 rounds (214-224) from integration_test account. Jackerparker models were estimated using only 10 rounds (214-224). The difference between training, validation and live performance are shown in Figure 7. The example predictions on the full range of live data shows that live COR performance is much lower than training performance and close to the validation. At the same time, live MMC performance is close to the training one. The rest of models show a similar trend on the last 10 live

rounds, except the jackerparker2. The latter model has low feature exposure (~0.02) and that could be an indirect proof that feature neutralized models tend to have more stable performance in all training, validation and live eras.

[

Fig7

1288×704 43.1 KB

](https://forum.numer.ai/uploads/default/original/1X/800e425ad919ec813cb691e7d8fbc9dbdc92dad0.png)

Figure 7.

Model performance for live, validation and training data.

Conclusions.

Using MMC+CORR payout seems to be an optimal method for most of the models which shows positive MMC on validation data. It increases mean payout value in most cases, but payout sharpe values benefits should be inspected by validation data.

Current live data is closer to validation performance. Question about which metrics (validation or training) we should trust more is still open.