Evaluating Prediction Models: Performance, Error Analysis, and Distribution Shifts

Teodora Taleska

Introduction

This report evaluates models for predicting basketball shot types. We compare a *Baseline Classifier*, *Logistic Regression*, and *Random Forest* optimized through standard and nested cross-validation. Random Forest achieved the highest accuracy and lowest log-loss.

We also examined the relationship between shot distance and prediction errors, finding a weak negative correlation using the *Spearman rank coefficient*, indicating errors slightly decrease with distance but are not strongly dependent on it.

Finally, adjusting for the true competition type distribution increased log-loss significantly while accuracy remained unchanged, suggesting that while predictions were correct at the same rate, their confidence degraded. This underscores the importance of evaluating models under real-world data distributions.

Part 1: Model Evaluation and Comparison

Methodology

Dataset and Problem Statement

The dataset consists of basketball shot data with the goal of predicting the *ShotType* (6 categories) using all other variables. The dataset is assumed to be a representative sample of the data-generating process.

Chosen Models

Three models were compared: a *Baseline Classifier* predicting the most frequent class, *Logistic Regression* for multi-class classification, and *Random Forest*, an ensemble model sensitive to hyperparameters such as the number of trees. Random Forest was chosen for its strong performance and sensitivity to hyperparameter tuning.

Evaluation Method

The models were evaluated using *5-fold cross-validation* with a 70-30 train-test split. The dataset contains 5024 instances, resulting in approximately 3500 training instances. With 5 folds, each fold contained around 700 instances, ensuring a balance between robust performance estimation and computational efficiency. For Random Forest, two hyperparameter tuning strategies were used: *Standard Cross-Validation*, where hyperparameters were tuned on the training folds, and *Nested Cross-Validation*, where an outer loop evaluated performance and an inner loop tuned hyperparameters [1].

Metrics

The models were evaluated using *Accuracy*, the proportion of correctly predicted instances, and *Log-Loss*, which measures the confidence of predicted probabilities for multi-class classification. Accuracy is defined as:

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(y_i = \hat{y}_i),$$

where N is the total number of instances, y_i is the true label, and $\hat{y}_i$ is the predicted label. Log-Loss is defined as:

$$\text{Log-Loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log(p_{i,c}),$$

where $y_{i,c}$ is 1 if the true label of instance i is class c, and $p_{i,c}$ is the predicted probability of class c.

Results

The performance of the models is summarized in Table 1, which reports the accuracy and log-loss of the models tested on the unseen test data, with the uncertainties coming from the cross-validation folds.

Table 1. Model Performance: Accuracy and Log-Loss (Mean ± Standard Error)

Model	Accuracy	Log-Loss
Baseline	0.6001 ± 0.0218	1.1684 ± 0.0102
Logistic Regression	0.7427 ± 0.0198	0.6514 ± 0.0076
RF (Standard CV)	0.7732 ± 0.0193	0.6032 ± 0.0079
RF (Nested CV)	$\mathbf{0.7752 \pm 0.0193}$	$\mathbf{0.6010 \pm 0.0089}$

The results indicate that the Random Forest model with nested cross-validation (CV) outperforms both the baseline and logistic regression models, achieving the highest accuracy and lowest log-loss. Nested CV is beneficial as it performs an additional layer of cross-validation for hyperparameter tuning, leading to a more robust model and better generalization. This extra tuning likely explains why the Random Forest model

trained with nested CV achieves superior performance compared to the standard cross-validation version. The logistic regression model, while improving upon the baseline, still lags behind the Random Forest models, highlighting the strength of ensemble methods in this context.

Part 2: Error Analysis and Adjusting for True Distribution

Assessing the Influence of Shot Distance on Prediction Errors

Methodology

To investigate whether prediction errors depend on shot distance, we used the *Spearman rank correlation coefficient*. This non-parametric measure assesses the strength and direction of the monotonic relationship between two variables, making it suitable for this analysis because it does not assume a linear relationship between variables and it does not require normally distributed data.

The Spearman correlation coefficient ρ is defined as:

$$\rho = 1 - \frac{6\sum d_i^2}{n(n^2-1)},$$

where d_i is the difference between the ranks of corresponding values of the two variables and n is the number of observations.

We calculated the Spearman correlation between the prediction errors (for each model) and the shot distance. A negative correlation indicates that errors tend to decrease as distance increases, while a positive correlation suggests the opposite. The significance of the correlation was assessed using the p-value, with a threshold of $\alpha = 0.05$.

Results

The Spearman correlation coefficients and their corresponding p-values for each model are summarized in Table 2.

Table 2. Spearman Correlation Between Prediction Errors and Shot Distance

Model	Spearman CC	p-value
Baseline	-0.143	2.45×10^{-8}
Logistic Regression	-0.081	0.0016
RF (Standard CV)	-0.067	0.0089
RF (Nested CV)	-0.075	0.0038

Conclusion

The results in Table 2 show weak negative correlations between prediction errors and shot distance for all models, with p-values below $\alpha = 0.05$, indicating statistical significance. This suggests that errors tend to decrease slightly as shot distance increases, but the effect is weak. The strongest correlation was observed for the *Baseline* model ($\rho = -0.143$), while the *Random Forest* models showed the weakest correlations ($\rho = -0.067$ and $\rho = -0.075$). These findings indicate that shot distance has a minor influence on prediction errors, but it is not a strong determinant of model performance.

Estimating Model Performance on the True Distribution

Methodology

To estimate how the models would perform on data with the true relative frequencies of competition types (NBA: 0.6, EURO: 0.1, SLO1: 0.1, U14: 0.1, U16: 0.1), we re-weighted the evaluation metrics based on the true distribution. This approach accounts for the mismatch between the dataset's competition type distribution and the true distribution. The steps are as follows:

- For each competition type, compute the model's accuracy and log-loss on the corresponding test samples.
- Weight these metrics by the true relative frequencies of the competition types.
- Aggregate the weighted metrics to obtain the overall performance estimate.

This re-weighting ensures that the evaluation reflects the model's performance on the true data distribution, where NBA games are more prevalent.

Results

The re-weighted accuracy and log-loss for each model are summarized in Table 3.

Table 3. Model Performance on True Distribution: Accuracy and Log-Loss

Model	Accuracy	Log-Loss
Baseline	0.6121	1.1953
Logistic Regression	0.7405	0.8949
RF (Standard CV)	0.7720	1.0308
RF (Nested CV)	**0.7721**	**1.0077**

Conclusion

The results in Table 3 show that the *Random Forest (Nested CV)* model achieved the highest accuracy (0.7721) and the lowest log-loss (1.0077) on the true distribution, outperforming the other models. However, compared to the results on the original dataset (where log-loss was significantly lower), the log-loss increased for all models, indicating that the models' confidence in their predictions degraded when evaluated on the true distribution. This suggests that the models are less calibrated for the true distribution, particularly for the more prevalent NBA competition type. Despite this, the accuracy remained relatively stable, showing that the models still make correct predictions at a similar rate.

References

[1] Trevor Hastie, Robert Tibshirani, Jerome Friedma. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*.