

## **Lichess Rating Prediction Models**

### **General Information**

- The Lichess rating prediction bot uses two models:
  - 1) A binary classification model to predict whether or not a user will reach a target rating for a specified time control within 2 years.
  - 2) A regression model to predict when a user is expected to reach the target rating, if they do reach it within 2 years.
- The models are both trained on a sample of ~10,500 lichess users from the [Lichess Swiss team](#). The users have had accounts since July 2019 or earlier. The model supports bullet, blitz, rapid, and classical time controls.
- The models predict what will happen to a user's rating given the information available 2 years ago, in August 2019. The observed outcomes occur between then and August 2021, when the data was collected and the model was built. An assumption of this approach is that users of the discord bot who have similar values of the model predictors as Lichess Swiss users in August 2019 will have similar rating growth patterns to them.
- The training data includes 5 randomly generated target ratings for each user and time control. The sampling distribution depends on the user's current rating, and is intended to give the greatest sample size to rating gains between 0-300 while including some training data for 300-700 point rating gains.
- The features used in the models are:
  - target rating gain
  - current rating
  - peak rating
  - rating values at various time lags
  - rating variance
  - frequency of rating updates
  - specified time control

- transformations and interactions of these features.

### **Classification model performance**

- The classifier is a binary [logistic regression](#) model
- In 74% of the training data the target rating is achieved.
- Distribution of modeled probabilities in the training data:

Minimum	25th Percentile	Mean	Median	75th Percentile	Maximum
0%	56%	74%	90%	97%	100%

- The model's [ROC AUC](#) is 90%. For rating gains that are harder to predict, in particular those between 50 and 200 points, the model's ROC AUC is 78%. (Baseline ROC AUC is always 50%)
- ROC AUC by time control:

Overall	Bullet	Blitz	Rapid	Classical
90%	89%	92%	88%	85%

### **Regression model performance**

- The regression model is [Ordinary Least Squares](#)
- Actual and Predicted Statistics in Days (\*predictions outside the range of 0-730 are overridden to be 0 or 730).

	Minimum	25th Percentile	Median	Mean	75th Percentile	Maximum
Actual	1	34	119	190	272	730
Predicted	0*	107	167	190	255	730*

- Error statistics

	Minimum	25th Percentile	Median	Mean	75th Percentile	Maximum
Raw error	-715	-58	36	0	98	651
Absolute error	0	42	89	116	154	715

- Mean absolute error by time control
  - Bullet (109), Blitz (111), Rapid (130), Classical (125)
- I compared the results of the OLS multiple regression model to the logarithmic fit extrapolation model from the original rating prediction bot. To ensure a fair comparison, I let the logarithmic fit model replace negative predictions with zeros just as the OLS multiple regression does. I also set the > 25 days of data requirement. It is clear from the following table that the OLS multiple regression greatly outperforms the logarithmic fit model. The mean absolute error is ~9X smaller and the median absolute error is ~2X smaller.

	<b>error</b>	<b>abs_error</b>	<b>log_model_error</b>	<b>log_model_abs_error</b>
<b>count</b>	51326	51326	51326	51326
<b>mean</b>	3	104	811	952
<b>std</b>	143	98	19807	19801
<b>min</b>	-715	0	-730	0
<b>25%</b>	-38	35	-76	40
<b>50%</b>	35	78	-1	133
<b>75%</b>	90	137	231	359
<b>max</b>	441	715	2355157	2355157

- However, this might be somewhat of an unfair comparison since the multiple regression model knows that the answer will never be greater than 2 years because of the scope of the training data, but the logarithmic fit model doesn't have that insight. So I tried another version where I

compared the two models on observations where the log model predicted less than 1 year - this way, the possibility of a > 2 year prediction by the logarithmic fit model is small and irrelevant. Here, the OLS multiple regression model is still more accurate, but by a smaller amount.

	<b>error</b>	<b>abs_error</b>	<b>log_model_error</b>	<b>log_model_abs_error</b>
<b>count</b>	37615	37615	37615	37615
<b>mean</b>	10	94	-57	131
<b>std</b>	132	93	187	146
<b>min</b>	-715	0	-730	0
<b>25%</b>	-20	31	-122	25
<b>50%</b>	36	70	-18	81
<b>75%</b>	86	122	41	183
<b>max</b>	431	715	360	730

- Besides the quantitative performance metrics, I believe the updated models have some additional advantages
  - The models acknowledge the possibility that the target rating gain won't be achieved and offer an estimate of that probability
  - The models follow a supervised learning approach so the predictions are connected to actual rating performance outcomes by real users. This means that the models can be improved in the future by including more features, a larger sample size, and more advanced classification and regression algorithms.
- Other future steps might include:
  - Using cross-validation to optimize models
  - Adding a prediction interval in addition to a point estimate for the predicted rating date
  - Expanding the model to include other variants / time controls