

1. Create or Open a Kaggle Notebook

1. Log in to your Kaggle account.
2. From the **Notebooks** page, click "**New Notebook**".
3. In the right-hand **Settings** panel:
 - Set **Accelerator** to **GPU P100**.
 - Under **Dataset**, click "**Add data**".

2. Attach the Chess Evaluations Dataset

1. In the **Add data** dialog, search for "**chess-evaluations**".
2. Select the dataset titled **Chess Evaluations** by **Ronak Badhe** (contains chessData.csv and random_evals.csv).
3. Click **Add** to mount it under: /kaggle/input/chess-evaluations/

3. Upload the Notebook File

1. In the notebook editor, click **File** → **Upload Notebook**.
2. Choose the **xgb-model.ipynb** file that contains the full feature-extraction, training, and benchmarking code.
3. Confirm to replace the current notebook contents if prompted.

4. Inspect Paths & Dependencies

The notebook expects input files at: /kaggle/input/chess-evaluations/chessData.csv

- /kaggle/input/chess-evaluations/random_evals.csv
- The working directory for outputs is set to: /kaggle/working/chess_cache/
- All required libraries (python-chess, stockfish, pandas, numpy, xgboost, etc.) are installed within the notebook itself—no additional setup is needed.

5. Run All Cells

1. From the notebook toolbar, click **Run** → **Run All**.
2. Watch the progress:
 - **Cell 1** (feature extraction) will take ~1–4 hours depending on your sample size and depth settings.
 - **Cell 2** (GPU training) will take ~10–20 minutes.
 - **Cell 3** (benchmarking) completes in a few minutes.

6. Download Outputs

- After execution, locate the following under the **Files** tab:
 - Extracted features: .parquet and .pkl files in /kaggle/working/chess_cache/.

- Trained model: quickEval_sf1_xgb_gpu_*.joblib in /kaggle/working/chess_cache/.
- Click **Download** next to each file to save locally.