

סיכום מלא - ModelTrainer מחלקת

מטרה כללית

שמחזה תוצאות משחקי שחמט עם דיוק של 60-70%, תוך למידה איטרטיבית Random Forest ליצור מודל ושיפור מתמיד.

תהליך הלמידה - 3 איטרציות

1 איטרציה: Baseline Model

אימון:

- עם 50 עצי החלטה Random Forest
- פרמטרים בסיסיים: max_depth=10, min_samples_split=5

בדיקה:

- Accuracy על test set
- Confusion matrix
- ראשוני Feature importance

accuracy תוצאה צפויה: ~55-60%

2 איטרציה: Hyperparameter Tuning

אימון מתוקן על בסיס תוצאות איטרציה 1

- ל-15 max_depth נמוך → הגדל accuracy אם
- max_depth=7-הקטן ל → overfitting אם
- (עצים 30, 50, 100) n_estimators כוונן
- (2, 5, 10) min_samples_split כוונן

בדיקה:

- השוואת ביצועים לאיטרציה 1
- הכי חשובים features איזה - feature importance ניתוח
- זיהוי דפוסי שגיאות

accuracy תוצאה צפויה: ~60-65%

3 איטרציה: Model Comparison & Final Optimization

אימון סופי:

- מאופטם (הפרמטרים הטובים מאיטרציה 2) Random Forest
- Logistic Regression השוואה עם
- אנסמבל אפשרי של שני המודלים

בדיקה סופית:

- ביצועים של כל מודל
- Feature importance מפורט
- ניתוח טעויות - איפה המודל נכשל

accuracy תוצאה צפויה: ~65-70%

(עצים 50 Random Forest מבנה)

עץ החלטה טיפוסי:



איך 50 עצים עובדים יחד:

1. **עץ 1-10**: rating_diff + win_rates מתמקדים ב.
2. **עץ 11-20**: avg_ratings + rating_diff מתמקדים ב.
3. **עץ 21-30**: features שילובים שונים של.
4. **עץ 31-40**: עצים עמוקים יותר עם פרטים דקים.
5. **עץ 41-50**: עצים רדודים לדפוסים כלליים.

הצבעה דמוקרטית:

- "עצים: "לבן מנצח 35"
- "עצים: "תיקו 10"
- "עצים: "שחור מנצח 5"
- **תוצאה סופית: לבן מנצח**

אחריות המחלקה

קלט:

- `X_train, X_test, y_train, y_test` מ-DataPreprocessor
- 5 features: rating_diff, win_rates, avg_ratings
- דוגמאות לאימון, ~900 לבדיקה ~3700

תהליך פנימי:

1. **אימון איטרטיבי** - 3 סיבובים של שיפור
2. **הערכת ביצועים** - accuracy, confusion matrix
3. **כוונון פרמטרים** - n_estimators, max_depth, min_samples_split
4. **השוואת מודלים** - Random Forest vs Logistic Regression
5. **ניתוח תוצאות** - feature importance, דפוסי טעויות

פלט:

- **המודל הטוב ביותר** (מאומן ומוכן לשימוש)
- **דוח ביצועים מפורט**:
 - Accuracy לכל איטרציה
 - Feature importance - הכי חשובים features איזה
 - Confusion matrix - איפה המודל טועה
 - המלצות לשיפור עתידי

מדדי הצלחה

טכניים:

- **Accuracy:** 65-70%
- **Feature importance:** rating_diff > 0.4 (הכי חשוב)
- **Overfitting control:** train/test accuracy הפרש קטן בין

למידה:

- והפרמטרים שלו Random Forest הבנת
- feature importance יכולת פירוש
- הבנת מתי ולמה מודל טועה

תובנות צפויות

Feature Importance (צפוי):

1. **rating_diff (40-50%)** - הפרש דירוגים הכי חשוב
2. **white_win_rate (20-25%)** - ביצועי לבן היסטוריים
3. **black_win_rate (15-20%)** - ביצועי שחור היסטוריים
4. **avg_ratings (10-15%)** - רמה כללית של השחקנים

דפוסים שגיאות צפויים:

- קושי לחזות תיקו (סט הכי קטן)
- (קטן rating_diff) טעויות במשחקים קרובים
- ביצועים טובים במשחקים עם הפרש דירוג גדול

אגב! accuracy המטרה: ללמוד מהתהליך, לא רק לקבל