Predicting Soccer Injuries with Machine Learning

Module 5: Improved Solution

Jennifer Lawless
DASC 9311: Data Science Project
November 5, 2025

Project Timeline

Week 1 (9/1-9/7): Dataset acquisition, initial exploration, environment setup

Week 2 (9/8-9/14): Review of existing Kaggle solutions, methodology research

Weeks 3-4 (9/15-9/28): EDA, feature analysis & engineering, data preprocessing

Weeks 5-7.5 (9/29-10/22): Baseline model implementation, initial performance benchmarking

Weeks 7.5-10.5 (10/23-11/5): Advanced model development, model fine-tuning

Today (11/5)

Weeks 10.5-13 (11/6-11/23): Ensemble model development, advanced model fine-tuning

Weeks 14-15 (11/24-12/7): Final model selection, results analysis, documentation

Baseline Models: Improvement Approach

Feature Set Simplification

Created a separate dataset (df_raw) without engineered features for Random Forest

Kept full feature set minus Balance_Test_Score for XGBoost

Model Persistence

Added code to save all three models using joblib

Hyperparameter Optimization

Limited the range for number of trees (100, 110)

More focused max_depth ranges

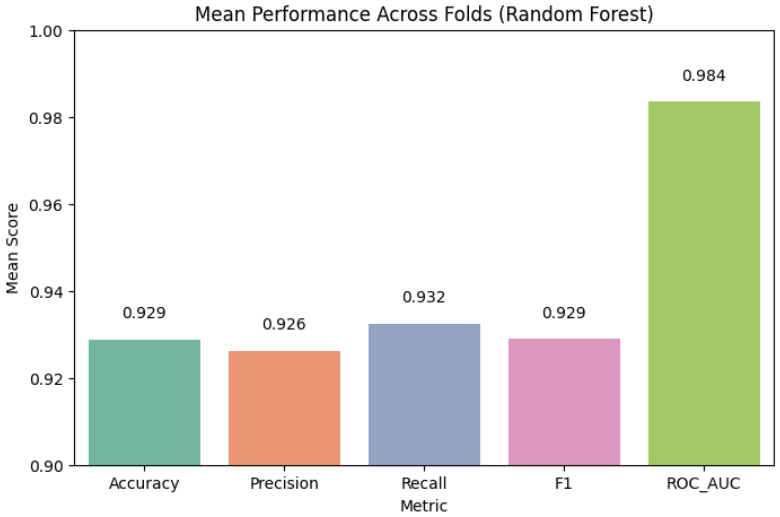
Simplified other parameters to reduce overfitting

Model 2: Random Forest

Original

Model Setup

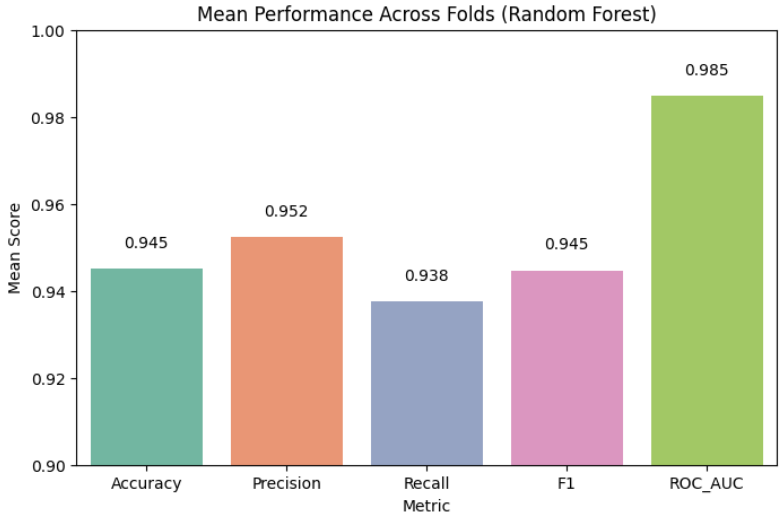
- **Full EDA dataset** (includes engineered features)
- **Hyperparameters Tuned:**
  - **Number of trees:** [50, 100, 200]
  - **Maximum depth:** [None, 10, 20, 30]
  - **Minimum samples per split:** [2, 5, 10]
  - **Minimum samples per leaf:** [1, 2, 4]
  - **Max features:** [‘sqrt’, ‘log2’]



Improved

Model Setup

- **EDA dataset** (without engineered features)
- **Hyperparameters Tuned:**
  - **Number of trees:** [100, 110]
  - **Maximum depth:** [5, 10, 15]
  - **Minimum samples per split:** [5]
  - **Minimum samples per leaf:** [2]
  - **Max features:** [‘sqrt’]



Model 2: Comparison to Kaggle

Model	Original	Improved	Kaggle Solution 1	Kaggle Solution 2
Metrics	**Accuracy:** 0.929 **Precision:** 0.926 **Recall:** 0.932 **F1:** 0.929 **ROC-AUC:** 0.984	**Accuracy:** 0.945 **Precision:** 0.952 **Recall:** 0.938 **F1:** 0.945 **ROC-AUC:** 0.985	**Accuracy:** 0.906 **Precision:** N/A **Recall:** N/A **F1:** N/A **ROC-AUC:** N/A	**Accuracy:** **0.963 → 0.943** **Precision:** 0.951 **Recall:** 0.975 **F1:** 0.963 **ROC-AUC:** N/A

2nd Kaggle Solution 2

1st Improved

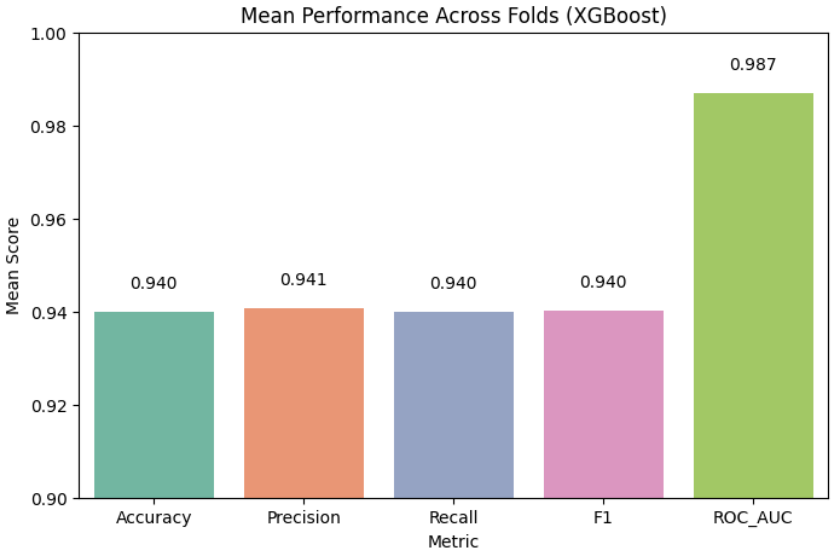
3rd Original

Model 3: XGBoost

Original

Model Setup

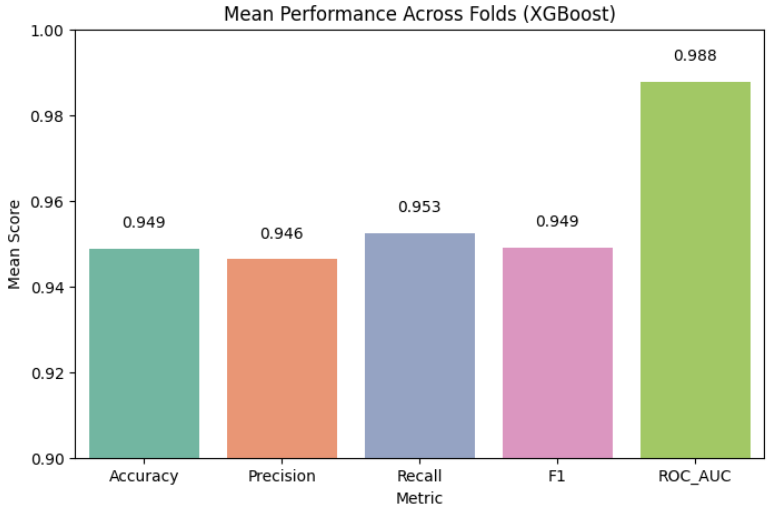
- **Full EDA dataset** (includes engineered features)
- **Max Iterations:** 1000 (ensures convergence)
- **Hyperparameters tuned:**
  Regularization strength = [0.01, 0.1, 1, 10]
  Penalty = L1 (Lasso) or L2 (Ridge)



Improved

Model Setup

- **EDA dataset** (without Balance_Test_Score)
- **Hyperparameters Tuned:**
  Number of trees: [100, 110]
  Maximum depth: [3]
  Learning rate: [0.01, 0.1]
  Subsample: [0.8]
  Minimum child weight: [3]



Model 3: Comparison to Kaggle

Model	Original	Improved	Kaggle Solution 1	Kaggle Solution 2
Metrics	**Accuracy:** 0.940 **Precision:** 0.941 **Recall:** 0.940 **F1:** 0.940 **ROC-AUC:** 0.987	**Accuracy:** 0.949 **Precision:** 0.946 **Recall:** 0.953 **F1:** 0.949 **ROC-AUC:** 0.988	**Accuracy:** 0.894 **Precision:** N/A **Recall:** N/A **F1:** N/A **ROC-AUC:** N/A	**Accuracy:** **0.969 → 0.945** **Precision:** 0.941 **Recall:** 1.000 **F1:** 0.970 **ROC-AUC:** N/A

2nd Kaggle Solution 2

1st Improved

3rd Original

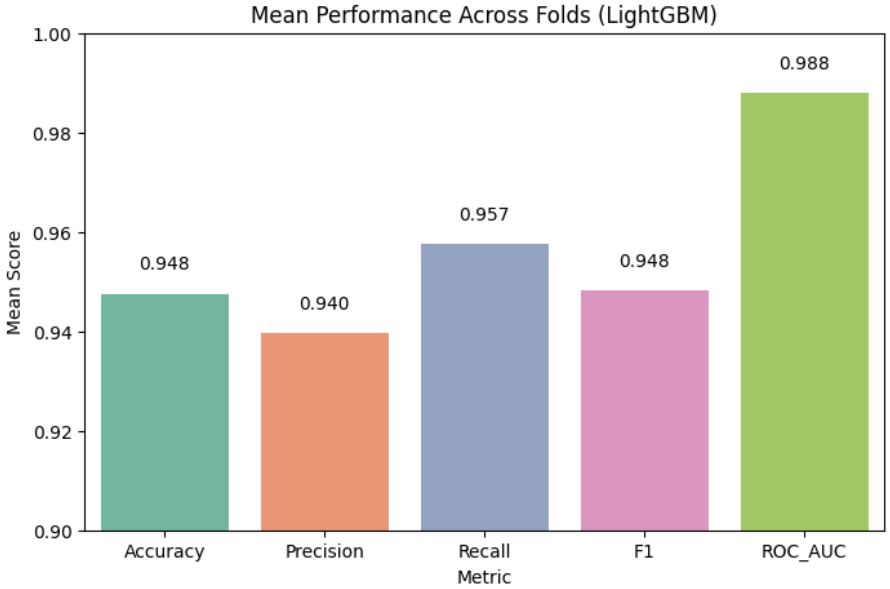
Advanced Model Development

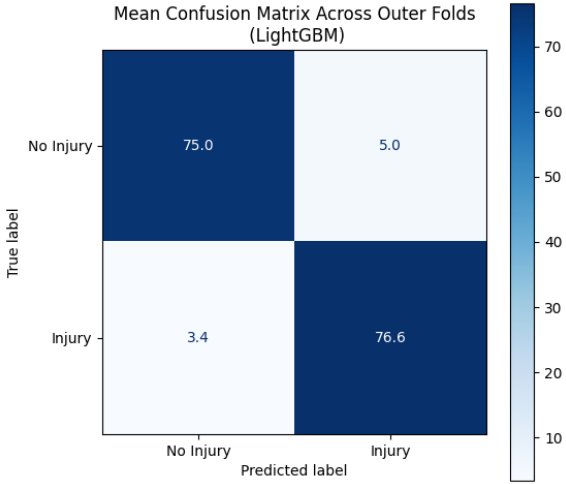
1. LightGBM
2. Support Vector Machine (SVM)

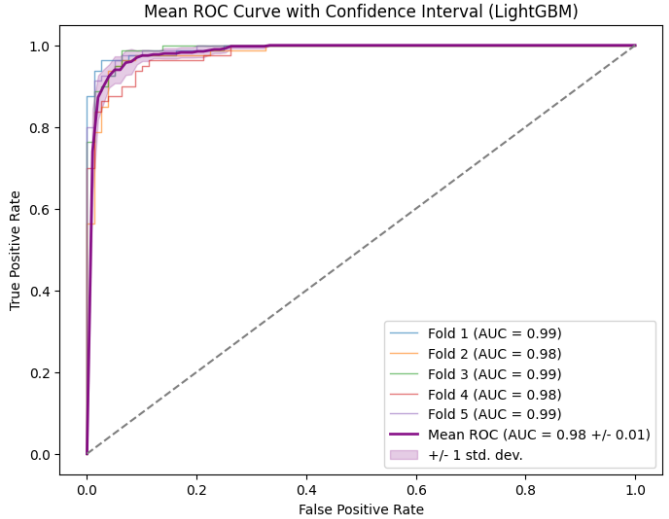
Model 4: LightGBM

Model Setup

- **Algorithm:** LightGBM (Gradient Boosting Machine)
- **Device:** GPU-accelerated
- **Early Stopping:** 30 rounds on validation set
- **Hyperparameters tuned:** RandomizedSearchCV



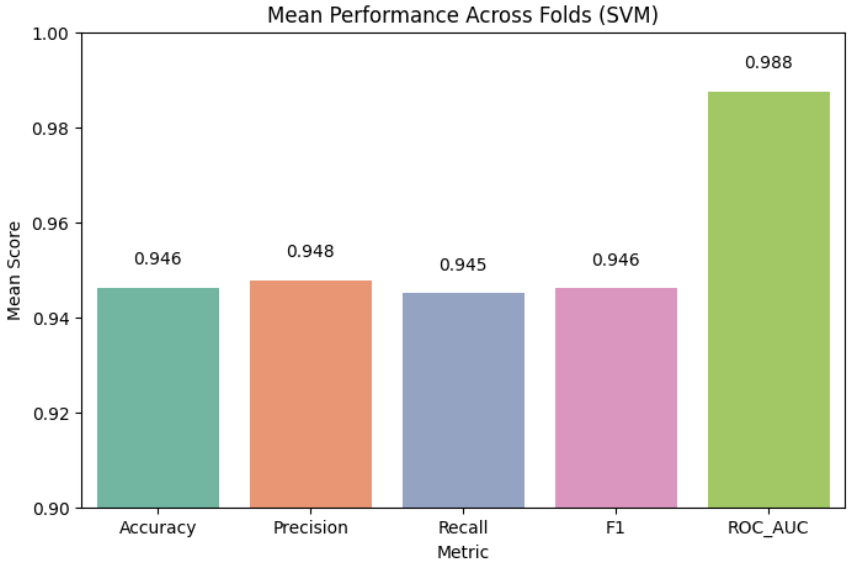


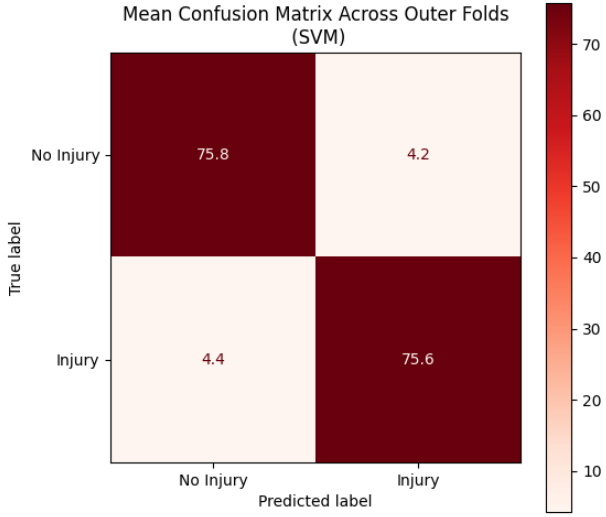


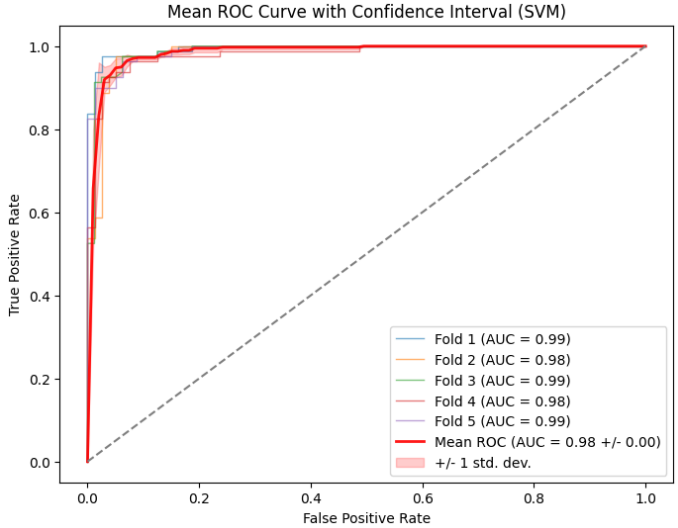
Model 5: SVM

Model Setup

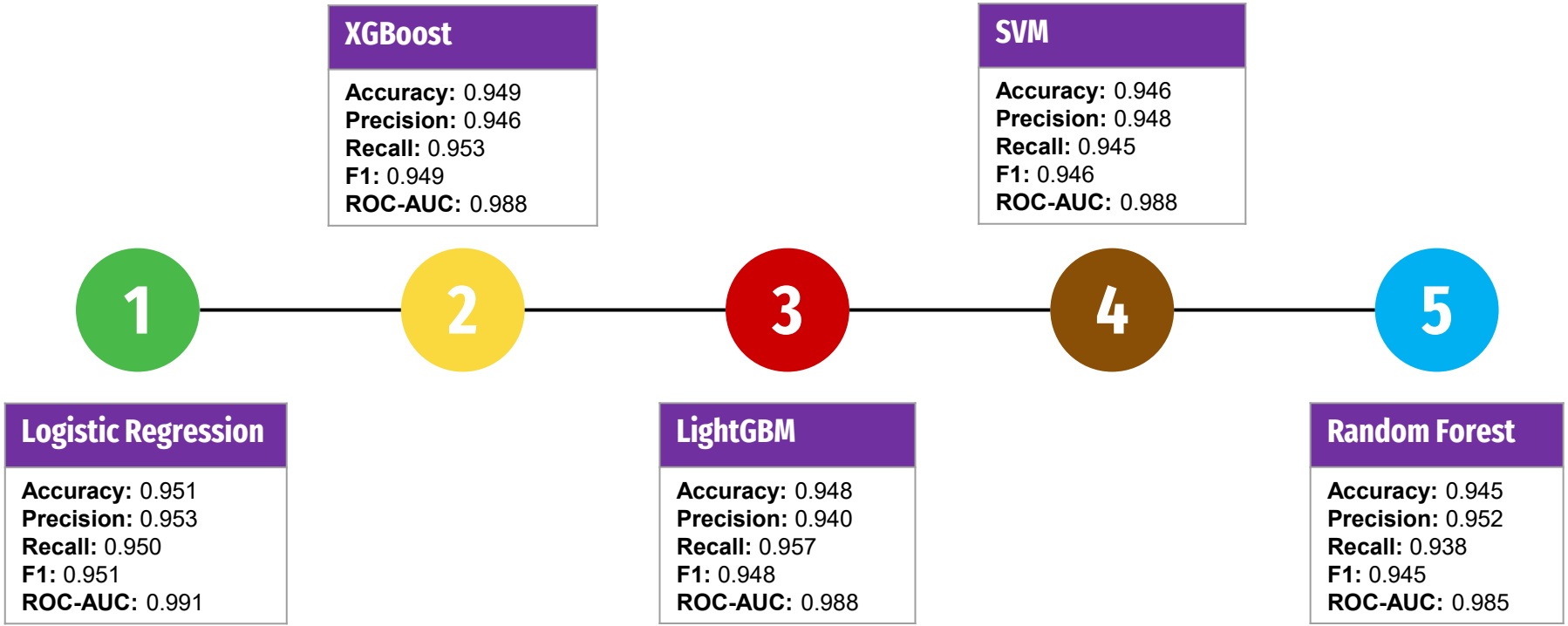
- **Algorithm:** Support Vector Machine (binary classification)
- **Kernel:** RBF (Radial Basis Function)
- **Hyperparameters tuned:**
  C (regularization) = 1.0
  Gamma = ‘scale’







Model Comparison



Next Steps

Ensemble model development

- Build Stacking Ensemble, Voting Classifier, Bayesian Model Averaging models

Model Fine-Tuning

- Fine-tune the models with hyperparameter tuning, feature selection, and other methods

Final Evaluation

- Evaluate and analyze each model
- Determine the best-performing model

Documentation

- Document all steps taken
- Write the final report and prepare for presentations

Thank You!