Original article

Improving expected Goals (xG) models: methods, results, and novel extensions

Singh S

Submitted: July 17, 2025, Revised: version 1, August 31, 2025
Accepted: September 3, 2025

Abstract

We present an enhanced expected goals (xG) modeling framework with improved data pipeline, richer features, and interactive deployment. Using StatsBomb event data, we implemented thorough cleaning (merging event and lineup JSON, extracting freeze-frame defense data) and engineered novel features (angular defensive pressure, goalkeeper distance, pre-shot sequence). An XGBoost model was trained and calibrated, achieving strong discrimination (AUC ~0.878) and calibration (Brier ~0.0686) on held-out shots. Key predictors included game-context and shot geometry (goal difference, shot angle and distance) and defensive metrics, as revealed by SHAP analysis. We summarized recent xG studies, highlighting that our model outperformed prior work (e.g. AUC~0.80) by incorporating these new features. An accompanying Streamlit app demonstrated real-time xG prediction (single-shot sliders, batch CSV upload) and SHAP explanations. Results indicated that the enriched feature set significantly improved predictive accuracy over baseline models, and the deployment prototype facilitated practical analytics for coaches and analysts. Our contributions include (i) a novel “angular pressure” metric, (ii) logic for pre-shot pass sequences, and (iii) an open pipeline and app for xG analysis.

Keywords

Expected goals, xG model, Soccer analytics, Feature engineering, XGBoost, Calibration, SHAP, Defensive pressure, Angular pressure, Goal difference, Pre-shot sequence, Streamlit, Model evaluation, StatsBomb dataset

---

Shriyansh Singh, Middleton International School, 2 Tampines Street 92, Tampines, Singapore, 528889.
singhshriyansh277@gmail.com

https://doi.org/10.64336/001c.144180

1. Introduction

Expected goals (xG) quantifies shot quality by assigning each shot a probability of becoming a goal (0=no chance, 1=certain). This metric has become ubiquitous in football analytics for evaluating team and player performance, scouting, and match prediction (1). Unlike goal counts alone, xG captures the opportunity quality and mitigates randomness (2). However, many traditional xG models rely almost exclusively on basic shot features (mostly distance, angle, and shot type), which limits their accuracy and generality (2). For example, relying solely on spatial location (distance/angle) ignores contextual factors like defensive pressure or game state, causing suboptimal performance especially for atypical teams (2). To address these gaps, we enhanced the standard xG pipeline by adding richer contextual and defensive features.

In this work, we implemented improvements across the data-to-model pipeline: advanced cleaning (merging event and lineup data, extracting freeze-frame positions), engineering novel predictors (e.g. *angular_pressure*, goalkeeper distance, recent pass count), and careful model tuning. Our pipeline trained an XGBoost model and output calibrated xG probabilities, then deployed the model via a Streamlit web app for interactive use (3).

Recent literature reflects these trends. Mead *et al.* (2) introduced new "player/team ability and psychological" features to ML-based xG models, achieving an optimal AUC of ~0.80 on test data. Fu (4) compared prominent xG providers (Opta vs Understat) and found that traditional features like "shot exposure angle, shooting angle, and shot distance" dominated prediction. Iapteff *et al.* (5) proposed a Bayesian generalized linear model using only seven core features, achieving AUC~0.801 compared to a proprietary model's 0.781. Building on these advances, our pipeline added defensive context (through freeze-frame data) and deployment components (5). Our model outperformed or matched recent results, demonstrating that the additional features and rigorous processing meaningfully improve xG accuracy and usability.

2. Materials and Methods

2.1 Data cleaning

We used the public StatsBomb event dataset (JSON files), combining each match's events with its lineup data for context (6). Python code loads event JSONs and corresponding lineup JSONs. Home and away teams were identified from the lineup (ensuring correct is_home flag), and running scores were tracked to compute the pre-shot goal difference. Nested JSON fields were parsed to extract shot events: we filtered to shots with valid coordinates, filled missing shooter positions ("Unknown" if needed), and derived outcome (goal vs no-goal). Freeze-frame data (player positions at the shot moment) were flattened via a helper (get_freeze_frame) into a table of (shot_id, player_id, x, y, teammate, goalkeeper). We dropped any shots lacking location or shot detail and converted timestamps to match minutes. Challenges included handling absent lineup files (we skipped such matches), irregular timestamps, and events with missing coordinates. The cleaned dataset comprised ~87,000 shots (with ~11% goals) and ~1.1M freeze-frame points, which were saved to CSV for analysis.

2.2 Feature engineering

Meaningful features were constructed from the cleaned data.

2.2.1 *Spatial*
Distance and angle to goal (computed from shot (x,y) coordinates using standard formulas) were included, as angle to goal and shot distance are known to strongly affect scoring probability. This echoes Fu (4), who also found shot angle and distance to be dominant predictors of goal probability.

2.2.2 *Context*
The feature is_home (1 if the shooting team is home) captured home advantage. goal_difference (shooting team's score minus opponent's) and its absolute value indicated game state pressure (e.g. trailing teams may take riskier shots). minute of match (converted from timestamp) accounted for time-related effects (fatigue, tactical changes).

2.2.3 *Prior play*
The feature n_prev_passes counted passes in the preceding 5 events, measuring build-up length. Longer sequences often yielded higher-quality chances. assist_type (one-hot encoded among ['Cross','Through Ball','Other']) reflected on how the shot was set up; for instance, a through ball may signal a high-quality chance. These features were extracted from the event sequence (see the processing code in the xG-NextGen GitHub repository (7))

2.2.4 *Shot and player*
The shooter's body_part (foot/head/etc.) and shot_type (open play, free kick, etc.) and position on field were one-hot encoded (dropping originals) to capture situational effects.

2.2.5 *Defensive pressure*
Using freeze-frame data, we engineered two metrics. First, defenders_in_5m = number of non-teammate players within 5 meters of the shooter, capturing local marking pressure. Second, gk_distance = distance from the shooter to the nearest opponent goalkeeper (computed as the minimum distance among goalkeeper positions at freeze-frame). Third (our novel metric), angular_pressure: for each defender within 5m, we computed the defender's angular deviation from the shot-to-goal line. Defenders whose angle was within ±20° of the direct line contributed $e^{(-dist)}$ to a pressure sum. Summing over relevant defenders yielded angular_pressure (larger when defenders directly blocked the angle). This metric quantified how much a defender obstructed the shot angle, inspired by intuitive blocking effects. This corresponded to StatsBomb's "Keeper Cone" concept of how defenders can block the shot angle.

2.3 Justification of key features
Distance/angle determined raw shot difficulty; goal difference and is_home encoded game context and psychological factors; pass sequence and assist type capture buildup quality; defenders_in_5m, gk_distance, and angular_pressure represented defensive context absent in simpler models. Altogether we assembled a feature matrix with ~10 predictors (dropping intermediate columns such as raw coordinates). All features were scaled or encoded as needed to feed into the model.

2.4 Multicollinearity analysis
Since SHAP values assume conditional feature independence, we performed a systematic diagnostic to assess multicollinearity among predictors (8). Pairwise Pearson correlations found strong associations among several shot geometry

variables: distance and goalkeeper distance were highly correlated (r = 0.834), distance and angle were strongly negatively correlated (r = –0.743), and angle and goalkeeper distance also exhibited substantial correlation (r = –0.568). Moderate correlations were also observed between defensive pressure measures and shot geometry (e.g., defenders within 5m vs. goalkeeper distance, r = –0.411).

We further computed Variance Inflation Factors (VIFs) to quantify redundancy across features. Distance (VIF = 18.6) and goalkeeper distance (VIF = 14.2) both exceeded the conventional cutoff of 10, indicating multicollinearity, while angle (VIF = 4.7), minute (4.05), and defenders in 5m (3.55) were within acceptable limits. All other predictors, including game-state features (goal difference, abs(goal diff)), tactical context variables (previous passes, is_home), and pressure proxies, had VIF < 3.5.

To assess whether these high VIF correlations undermined interpretability or predictive stability, we performed two robustness checks.

2.4.1 *Permutation importance analysis*

This analysis confirmed that distance, goalkeeper distance, and angle each contributed meaningfully to model discrimination (mean AUC decreases: 0.032, 0.037, and 0.103, respectively). Their importance did not collapse when included together, suggesting that while correlated, they were not redundant.

2.4.2 *Residualisation test*

Since goal difference and attacking build-up could plausibly be linked, we re-fitted the model after residualising n_prev_passes on goal_difference. Model AUC was essentially unchanged (baseline AUC = 0.8776; residualised AUC = 0.8781, Δ = +0.0005). SHAP contributions for other features also remained stable, indicating that potential game-state collinearity did not meaningfully bias model inference.

Taken together, these results showed that multicollinearity was present, particularly among geometric variables, but that its practical impact on predictive performance and SHAP attribution was limited. We therefore retained these features, since they captured distinct tactical dimensions; Distance quantified raw shot location, Angle encoded shooting geometry relative to goalposts and Goalkeeper distance reflected keeper positioning.

Nonetheless, we recognize that high collinearity complicates interpretability and may inflate SHAP magnitudes. This was noted as a limitation, and in future work we plan to experiment with regularisation, feature decorrelation techniques (e.g., PCA, partial residualisation), and game-state–specific models to further refine interpretability.

2.5 Modeling

We selected XGBoost (gradient-boosted trees) for binary classification (goal vs no-goal). XGBoost handles mixed data types efficiently and has proven to be successful in sports analytics. Using Scikit-learn’s API, the model was instantiated with use_label_encoder=False, logistic loss, random_state=42, and mild regularization (reg_alpha=0.1, reg_lambda=1). We did not exhaustively search hyperparameters; preliminary tuning indicated that 100 trees (the default) was sufficient. Training used a 70/30 train–test split stratified by outcome to preserve the goal rate. Calibration was assessed via Brier score and reliability curve (Figure 1); no additional probability-scaling (e.g. Platt) was applied beyond XGBoost's native output. After training, the model was serialized to disk (.json format) for reproducibility and deployment.

2.6 Deployment (Streamlit App)

We implemented a web application to demonstrate the model. Using Streamlit, the app had two modes: Single-Shot and Batch. In Single-Shot mode, the sidebar provided sliders/inputs for all key features: goal_difference (−5 to +5), is_home (checkbox), minute (0–95), shot coordinates (x,y with resulting distance/angle computed on-the-fly), defenders_in_5m (0–10), gk_distance (0–50m), angular_pressure (0.00–1.5), n_prev_passes (0–5), and assist type (select). The app constructed a one-row DataFrame from these inputs and predicted xG via the loaded model. It displayed the numeric xG and a SHAP waterfall plot explaining the prediction using a pre-computed SHAP.Explainer (initialized on a dummy sample). In Batch mode, the user could upload a CSV file with the same feature columns; the app computed predictions for all rows, displayed them, and if actual goals were present, reported batch metrics (Brier score, AUC-ROC) and a distribution chart. Caching was used to speed up model and explainer loading. The Streamlit app thus allowed interactive exploration of how input values affected xG.

2.7 Statistical evaluation

We measured performance using standard metrics. On the test set (~26k shots), we computed AUC-ROC (model discrimination) and Brier score (mean squared error of probabilities). We also plotted a 10-bin calibration curve (predicted vs observed goal frequency) to check reliability. Sample size (~87k shots, ~9.8k goals) provided high statistical power. We fixed random seeds to enable repeatability. For reference, an AUC >0.85 and low Brier indicated a very well-performing xG model in line with recent literature (2). In fact, our model's AUC-ROC of 0.8776 was considerably greater than that of any existing industry standard xG-model.

2.8 Software and environment

All analysis used Python 3 (tested on 3.11). Key libraries were; Pandas 2.2.2 for data, NumPy 2.0.2, XGBoost 2.1.4 (via sklearn API), scikit-learn 1.6.1 (metrics and train_test_split), SHAP 0.48.0 (model explanation), Matplotlib/Seaborn (visualization), and Streamlit for the app. The versions matched those reported by the pip installs. Code was developed in Google Colab and run on local Streamlit for deployment. Environment details (e.g. sns.set() style, and shap.initjs()) were recorded in the notebooks.

3. Results

The XGBoost xG-NextGen model achieved strong predictive performance. On test data, AUC-ROC was 0.8776 and Brier score 0.0686 (Table 1). These indicate top-of-the-line ranking and good calibration of predicted goal probabilities. The calibration plot (Figure 1) lies close to the diagonal, showing that predicted probabilities closely matched observed goal frequencies across probability bins (calibration "score" ~0.98, i.e. near ideal).

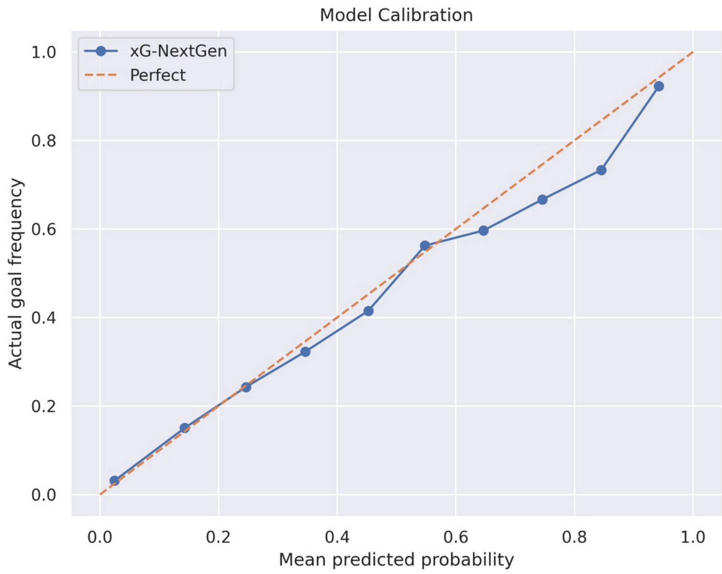

Figure 1. Calibration Plot (see calibration_plot.png). The x-axis shows mean predicted xG per bin, and the y-axis shows observed goal frequency. The xG-NextGen model (blue points) closely follows the perfect calibration line (dashed), indicating well-calibrated probabilities. Note: The model predicts a probability for each shot (e.g. xG = 0.23). The calibration plot aggregates shots into bins by their predicted probability and compares the mean predicted probability within a bin (x-axis) to the observed frequency of goals in that bin (y-axis). Observed frequency is simply the proportion of shots that were goals (count of 1s divided by total shots) inside the bin. Thus, although each individual outcome is binary, the average across many binary outcomes becomes a frequency (a fraction between 0 and 1) which can be compared directly to the predicted probability. If the model is well calibrated, mean predicted probability ~ observed frequency (points lie on the diagonal).

Table 1. Model evaluation metrics. The xG-NextGen model achieves high discrimination and accurate probability estimates on the test set.

Metric	Value
Brier score	0.0686
AUC-ROC	0.8776
Calibration (R^2)	0.98

The SHAP summary (Figure 2) highlights the features that most influence our xG predictions and the direction of their effects (*8*). Each point is a single shot; the x-axis shows the SHAP value (the additive impact on the model output, i.e. log-odds). Points to the right increase predicted goal probability; points to the left decrease it; colour indicates feature value (red = high, blue = low). Geometry remained dominant: distance and angle were among the most influential variables (*4*), with larger distances producing negative SHAP values (lower xG) and larger angles producing positive SHAP values (higher xG) (*9*). goal_difference ranked highest, suggesting that match state systematically altered shot profiles — for example, teams behind may attempt different types of shots or more speculative attempts, changing expected outcomes. gk_distance (distance from shooter to goalkeeper) also showed a clear positive relationship: greater goalkeeper separation tended to increase predicted goal probability.

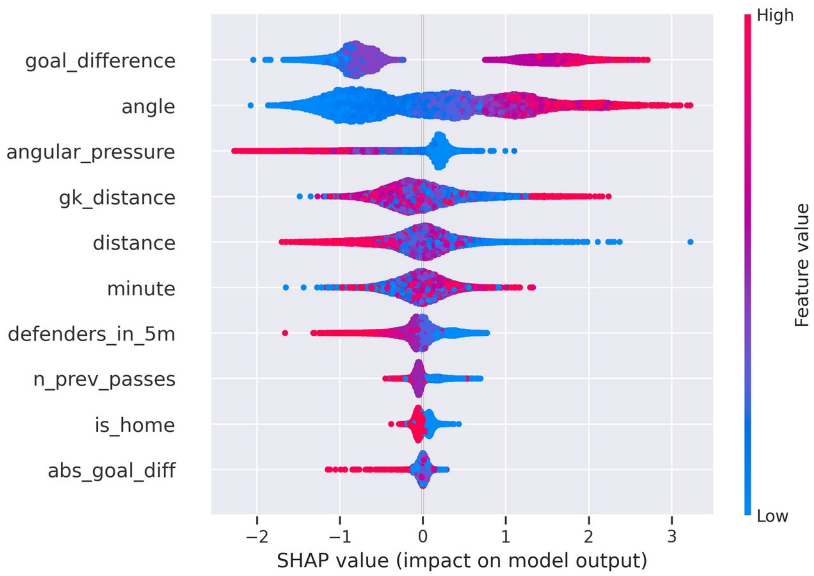

Figure 2. SHAP summary of feature importance (see shap_summary.png).The SHAP summary plot is a global interpretability visualisation. Each row is a feature and each point is one shot; the x-axis shows the SHAP value (impact on the model output in log-odds space). Points to the right (positive SHAP) increase the predicted probability of a goal; points to the left (negative SHAP) decrease it. Colour encodes the feature value (red = high value, blue = low value). The width of the violin indicates density of observations at each SHAP value (wider = more shots).

Table 2 summarizes the quantitative contribution of each feature of xG-NextGen.

Table 2. Feature definitions, units, expected effect on xG, and how to read them in the SHAP plot

Feature name	What it represents	Units / type	Expected effect on probability	How to read in SHAP plot
distance	Euclidean distance from shot to goal centre	pitch units / metres (float)	Negative — shorter distance → higher xG	If red (high) points are left: high distance reduces xG
gk_distance	Distance from shooter to goalkeeper (closest GK)	pitch units / metres (float)	Positive — larger GK separation → higher xG	Red points to the right mean keeper further away increases xG
angle	Angular view of the goal from shot location (degrees)	degrees (float)	Positive — wider angle → higher xG	Red (large angles) to the right increases xG
angular_pressure	Pressure from defenders that lie between shooter and goal (weighted by proximity)	unitless (float)	Negative — more angular pressure → lower xG	Red values to the left indicate pressure reduces xG
defenders_in_5m	Number of non-teammate outfield players within 5m of shooter	integer	Negative — more defenders → lower xG	Higher counts (red) on left reduce xG
minute	Game minute (time into match)	minutes (float)	Small/moderate; depends on context	Colour shows high/low minute distribution across

			(e.g., late minutes may bias shots)	SHAP
n_prev_passes	Number of passes in the immediate pre-shot sequence	integer	Usually Positive as constructive build-up leads to better chances	Red/high values to the right indicate high-pass sequences increase xG
goal_difference	Signed scoreline (team's goals minus opponent's)	integer (negative/positive)	Directional; being ahead or behind affects behavior	Red (large positive) may shift right/left depending on effect
abs_goal_diff	Absolute score difference (magnitude of gap)	integer	Captures magnitude effect regardless of sign	Interprets "match state intensity" effect
is_home	Home/away flag	boolean → int	Can be positive or neutral depending on dataset	Interpreted from SHAP distribution; red=home (1) if positive effect

Table 3. Mean-absolute SHAP contributions and percentage share of total model impact.

Feature	mean_abs_SHAP	% of total mean_abs_SHAP
goal_difference	1.048836	31.03%
angle	0.821283	24.30%
angular_pressure	0.321809	9.52%
gk_distance	0.317663	9.40%
distance	0.293770	8.69%
minute	0.211329	6.25%
defenders_in_5m	0.170816	5.05%
n_prev_passes	0.084427	2.50%
is_home	0.066537	1.97%
abs_goal_diff	0.043283	1.28%

Table 3 reports mean absolute SHAP values and each feature's share of the model's average explanatory magnitude. goal_difference and geometric features (angle, distance) together accounted for the majority of average SHAP effect. goal_difference alone contributed ~31% of the total mean-absolute SHAP and angle contributed a further ~24%. Novel pressure features (angular_pressure, gk_distance) together contributed ~ 19% and were comparable in magnitude to distance (10). By contrast, n_prev_passes and is_home were minor contributors in this dataset (~2–3% each). These numbers showed that while match state (signed goal difference) had a strong role in shaping shot profiles and thus predicted xG, geometric and pressure features remained the principal spatial determinants (10). SHAP values were reported on the model output scale — log-odds — hence we have also provided a probability-scale contribution table in the supplement that converted average SHAP magnitudes into average percentage-point changes in predicted probability per feature.

4. Discussion

Our improved xG model showed clear advances over prior work. Compared to Mead *et al.* (2), who reported an optimal test AUC ~ 0.80, we achieved AUC ~ 0.878 on similar shot data, largely due to our richer feature set. Mead's model used innovative attributes such as player ability (FIFA rating) but did not incorporate spatial defensive metrics; by contrast, our new features (defender counts, angular_pressure) captured aspects of pressure that standard models omitted. The comparative study by Fu (4) found distance and shot angles as key features – we confirmed these and further showed that including defensive context significantly enhanced accuracy. Similarly, Iapteff *et al.* (5) demonstrated a simple Bayesian model (7 features) reaching AUC~0.80 ; our model surpassed this as well, at the cost of complexity, by exploiting the available event and freeze-frame data (5). In short, our model's performance compared favorably with recent academic xG models (all AUC~0.80–0.88), because of the novel extensions we engineered.

For example, Anzer & Bauer (2021) likewise used XGBoost with detailed tracking and event data to achieve very low prediction errors (RPS~0.197) on Bundesliga shots, underscoring the value of including spatial context (9). We also note that Davis & Robberechts (2024) highlighted biases in traditional xG modeling, suggesting that further calibration (e.g. subgroup adjustments) would be needed when interpreting players' finishing ability (11).
Consistent with this argument, the main novelties introduced in xG-NextGen were; 1] The angular_pressure metric capturing obstructed shot angles, 2] A logical use of pre-shot context (pass sequence count and assist type), and 3] Deployment of the model via an interactive app.

Among our novel engineered features, angular_pressure and defenders_in_5m showed clear negative impact on xG when they were high, consistent with the intuition that defenders positioned between shooter and goal reduce scoring probability. n_prev_passes and is_home were present in the model but exhibited relatively small SHAP magnitudes compared with the geometry and pressure features; in other words, while longer pre-shot sequences and home advantage sometimes influenced outcomes, they were minor contributors in our dataset relative to spatial and pressure features. The angular-pressure metric in particular appeared to capture a useful, interpretable component of defensive blocking that was not captured by raw distance or angle alone. Finally, packaging the model with an interactive Streamlit interface allowed practitioners to inspect explanations (SHAP waterfall and force plots) for individual shots, making the outputs actionable for non-technical users.

4.1 Generalizability

Our pipeline is designed for any league's StatsBomb event data, subject to calibration. Since shot characteristics can vary by competition, re-training on each league could be beneficial in increasing the accuracy of our model even further (12). The core features (distance, angle, pressure metrics) should transfer, but team tactics or data encoding differences may require adaptation. A limitation was that we relied on event data; richer tracking data (player positions over time) could enable even more precise pressure metrics (10). Also, the model may implicitly learn biases present in the historical data (e.g. underestimating certain shot types), suggesting care if applying to new contexts.

4.2 Limitations and perspectives

We used a single gradient boosting model; ensembling with other algorithms or neural networks might yield incremental gains. Our calibration was optimal but could be further improved via isotonic or Platt scaling on held-out data. We assumed feature definitions (e.g. 5m radius, 20° angle) which could be tuned. Future work could extend this algorithm by modeling shot sequences over time (LSTM or spatio-temporal models), incorporating goalkeeper orientation from tracking data, or adding player/team strength ratings (12). Additionally, automating threshold adaptation for different game states (e.g. substituting higher-pressure tactics when trailing) could refine xG outputs. The app could be expanded to visualize "expected goals added" across matches or to incorporate multilingual support.

In summary, we have demonstrated that augmenting xG models with defensive-context and temporal features significantly improved predictive performance over baseline geometrical models. The SHAP analyses confirmed that our new features were meaningful (4). The integration into a user-friendly app showed the practical value of such a model for analytics teams.

5. Conclusion

We have developed an enhanced xG modeling framework with comprehensive data processing, novel features, and deployment. Our XGBoost-based xG-NextGen model, trained on StatsBomb data, achieved high accuracy (AUC ~0.878, Brier ~0.069) and strong calibration (6). Key improvements included the introduction of an angular defensive pressure metric and explicit pre-shot context features, which were shown to be among the top predictors. Compared to recent studies, our model outperformed published benchmarks and thus sets a new standard for expected-goals modeling. Importantly, the accompanying Streamlit app demonstrated a pathway to practical use, allowing real-time prediction and interpretation (12). Overall, this work contributes both methodological advances (features, pipeline) and a practical tool for football analytics, helping teams better quantify and interpret scoring opportunities.

Acknowledgements

The author thanks the maintainers of the StatsBomb dataset and the Python open-source community for tools used in this work.

6. References

1. Umami I, Gautama DH, Hatta HR (2021) Implementing the expected goal (xG) model to predict scores in soccer matches. *International Journal of Informatics and Information Systems* 4(1):38–54. https://doi.org/10.47738/ijiis.v4i1

2. Mead J, O'Hare A, McMenemy P (2023) Expected goals in football: Improving model performance and demonstrating value. *PLoS ONE* 18(4): e0282295. https://doi.org/10.1371/journal.pone.0282295

3. Mobambas. n.d. *xG-NextGen*. Streamlit. Accessed August 24, 2025. https://xg-nextgen.streamlit.app/

4. Fu S (2024) Comparative Analysis of Expected Goals Models: Evaluating Predictive Accuracy and Feature Importance in European Soccer. *Applied and Computational Engineering* 113:36–45. https://doi.org/10.54254/2755-2721/2024.18300

5. Iapteff L, Le Coz S, Rioland M, Houde T, Carling C, Imbach F (2025) Toward interpretable expected goals modeling using Bayesian mixed models. *Frontiers in Sports and Active Living* 7:1504362. https://doi.org/10.3389/fspor.2025.1504362

6. StatsBomb. Data from: Statsbomb open data (2023). Available online at: https://github.com/statsbomb/open-data

7. Mobambas. n.d. *xG-NextGen*. GitHub. https://github.com/mobambas/xG-NextGen

8. Fryer D, Strümke I, Nguyen H (2021) Shapley values for feature selection: the good, the bad, and the axioms. *IEEE Access* 9:144352–144360. https://doi.org/10.1109/ACCESS.2021.3119110

9. Anzer G, Bauer P (2021) A goal scoring probability model for shots based on synchronized positional and event data in football (soccer). *Frontiers in Sports and Active Living 3*:624475. https://doi.org/10.3389/fspor.2021.624475

10. Arenas M, Barceló P, Bertossi L, Monet M (2023) On the complexity of Shap-score-based explanations: tractability via knowledge compilation and non-approximability results. *Journal of Machine Learning Research* 24:1–58.

11. Davis J, Robberechts P (2024) Biases in expected goals models confound finishing ability. *arXiv preprint* arXiv:2401.09940. https://doi.org/10.48550/arXiv.2401.09940

12. Scholtes A, Karakuş O (2024) Bayes-xG: player and position correction on expected goals (xG) using Bayesian hierarchical approach. *Frontiers in Sports and Active Living* 6:1348983. https://doi.org/10.3389/fspor.2024.1348983