

# **SKRIPSI**

## **PENERAPAN *LIGHT GRADIENT BOOSTING MACHINE* (LGBM) UNTUK PERHITUNGAN METRIK *EXPECTED GOALS* (xG) DALAM ANALISIS SEPAK BOLA**

Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Komputer

Fakultas Sains dan Teknologi

Universitas Islam Negeri Syarif Hidayatullah Jakarta



Disusun oleh:

**Fadhil Raihan Akbar**  
**NIM. 11210930000101**

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH JAKARTA  
2025 M/1447 H**

# **SKRIPSI**

## **PENERAPAN *LIGHT GRADIENT BOOSTING MACHINE* (LGBM) UNTUK PERHITUNGAN METRIK *EXPECTED GOALS* (xG) DALAM ANALISIS SEPAK BOLA**

Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Komputer

Fakultas Sains dan Teknologi

Universitas Islam Negeri Syarif Hidayatullah Jakarta



Disusun oleh:

**Fadhil Raihan Akbar**

**NIM. 11210930000101**

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH JAKARTA  
2025 M/1447 H**



## **LEMBAR PENGESAHAN SKRIPSI**



## **LEMBAR PENGESAHAN UJIAN**

## **LEMBAR PERNYATAAN**





## **ABSTRAK**



## **DAFTAR ISI**

<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
-------------------------------	----------



## **DAFTAR GAMBAR**



## **DAFTAR TABEL**

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

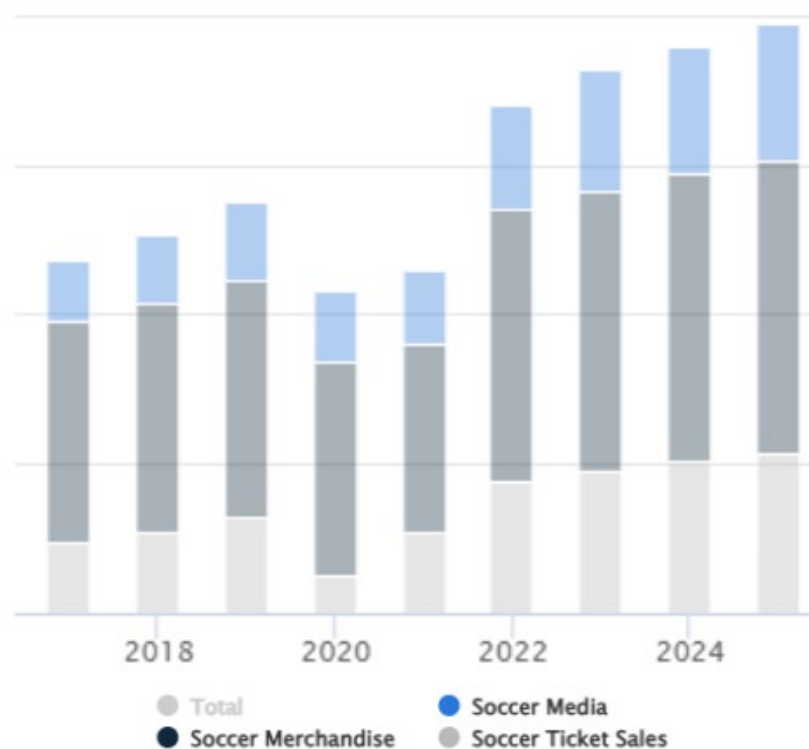
Ketidakpastian adalah elemen tak terpisahkan dalam setiap ranah olahraga, dan justru inilah yang membuatnya begitu menarik bagi banyak orang. Dalam sepak bola contohnya, perpaduan antara performa dan keberuntungan sering kali menjadi penentu kemenangan atau kekalahan, menjadikan olahraga ini semakin kompleks dan menarik untuk dianalisis secara mendalam. Sepak bola di era modern saat ini telah mengalami perubahan signifikan dengan adanya integrasi *data science* dan *machine learning* dalam berbagai aspek pengelolaan olahraga ini. Saat ini, sepak bola bukan hanya sekadar kompetisi olahraga, melainkan juga bagian dari industri yang lebih luas. Dalam era ini, data tidak lagi sekadar angka, tetapi menjadi landasan penting untuk mengukur performa dan membuat analisis berbasis fakta. Teknologi data *analytics* dan metrik spesifik semakin mendukung pengambilan keputusan dalam manajemen sepak bola. Penggunaan perangkat *wearable* dan sistem informasi mutakhir memungkinkan tim untuk mengumpulkan serta menganalisis data dalam jumlah besar, yang berperan penting dalam aspek taktik pertandingan, *scouting* pemain, hingga pencegahan cedera (Chatziparaskevas et al., 2024).

Di Indonesia sendiri, pasar sepak bola diproyeksikan menghasilkan pendapatan sebesar USD 158,30 juta pada tahun 2024. Pertumbuhan tahunan sebesar 2,80% (CAGR 2024-2029) diperkirakan akan meningkatkan volume pasar



hingga mencapai USD 181,70 juta pada tahun 2029. Secara global, pendapatan terbesar akan dihasilkan oleh Inggris, dengan nilai sebesar USD 9.696,00 juta pada 2024.

Selain itu, rata-rata pendapatan per pengguna (ARPU) di pasar sepak bola Indonesia diproyeksikan mencapai USD 28,27 pada 2024, dengan jumlah pengguna yang diperkirakan mencapai 6,2 juta pada tahun 2029. Tingkat penetrasi pengguna dalam pasar ini akan mencapai 2,0% pada tahun 2024 (Statista, 2024). Gambar 1.1 menampilkan proyeksi pendapatan tahunan pasar sepak bola di Indonesia.

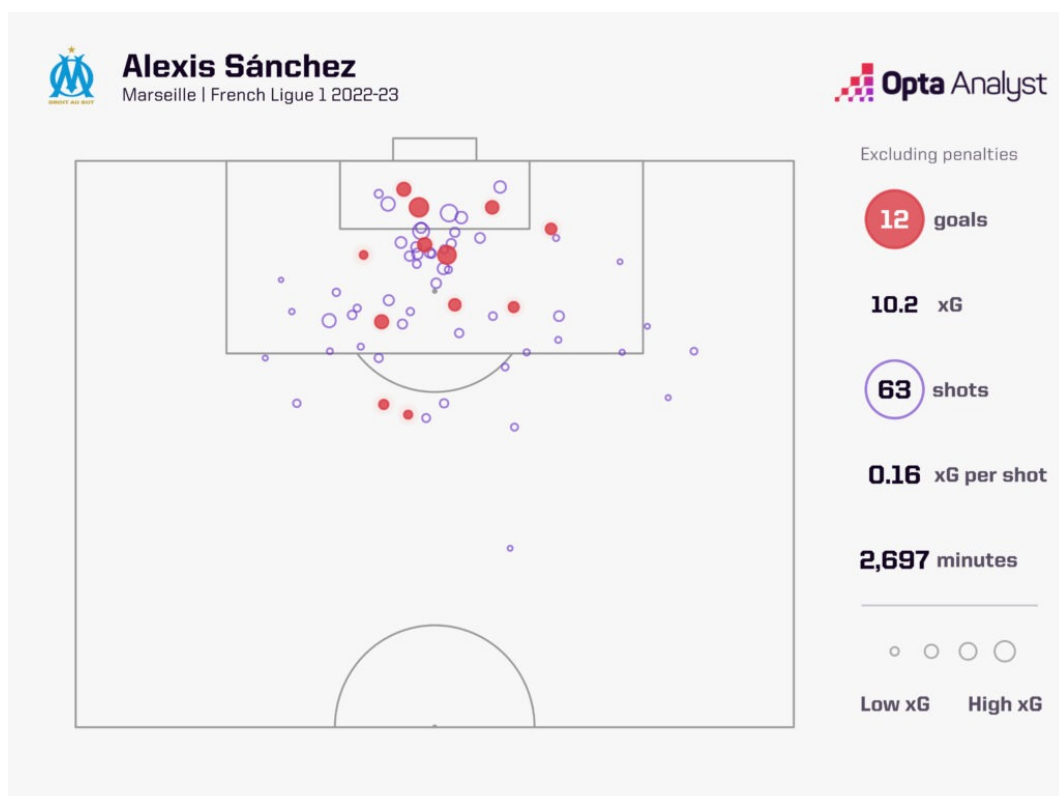


Gambar 1.1 Pendapatan Tahunan Pasar Sepak Bola di Indonesia

Dalam analisis sepak bola modern, metrik yang dikenal sebagai *Expected Goals* (xG) menjadi komponen penting dalam mengevaluasi kualitas peluang mencetak gol. Metrik ini menghitung probabilitas suatu peluang menghasilkan gol, yang pada akhirnya memberikan wawasan lebih akurat terkait hasil pertandingan.

Dengan mengakumulasi nilai xG dari setiap peluang dalam suatu pertandingan, dapat diperoleh hasil pertandingan yang diharapkan. Pemanfaatan xG terbukti memberikan wawasan berharga terkait performa tim dan pemain, terutama dalam menentukan efektivitas serangan serta pengambilan keputusan yang lebih baik di masa mendatang (Eggels, 2016).

Salah satu contoh penerapan xG dalam sepak bola adalah melalui analisis *shot-map*, yang memperlihatkan distribusi dan kualitas peluang yang dihasilkan oleh seorang pemain. Gambar 1.2 menampilkan *shot-map* Alexis Sanchez, di mana setiap titik tembakan dilengkapi dengan nilai xG yang menunjukkan peluang keberhasilan menjadi gol (Whitmore, 2023). Analisis semacam ini membantu pelatih dan manajemen tim dalam mengevaluasi kontribusi pemain serta mengidentifikasi area yang memerlukan perbaikan atau penguatan strategi.



Gambar 1.2 Visualisasi *Shot-map* xG

Perhitungan xG harus dilakukan dengan akurasi dan ketepatan tinggi untuk memberikan wawasan yang andal terkait kualitas peluang gol. Ketepatan ini sangat penting dalam mendukung keputusan strategis, seperti menentukan taktik permainan dan mengevaluasi kinerja pemain. Namun, perhitungan xG secara manual menghadapi banyak tantangan, terutama di era teknologi saat ini, karena data yang digunakan semakin kompleks dan bervolume besar. Penggunaan metode manual tidak hanya memakan waktu tetapi juga rentan terhadap kesalahan manusia. Oleh karena itu, metode *machine learning* menjadi solusi untuk mempercepat dan mempermudah perhitungan xG dengan memanfaatkan data historis, sehingga hasilnya lebih konsisten dan akurat.

Dengan menerapkan konsep ini, komputer dapat belajar dari data dan pengalaman untuk menghasilkan keputusan atau prediksi secara mandiri. Menurut Pratama et al. (2017), *machine learning* adalah bidang keilmuan yang memungkinkan komputer atau mesin untuk menjadi cerdas dengan cara belajar dari data yang diberikan. Metode ini sangat bermanfaat dalam berbagai aplikasi, termasuk dalam analisis sepak bola, karena mampu memproses data yang kompleks dan menghasilkan hasil yang lebih presisi.

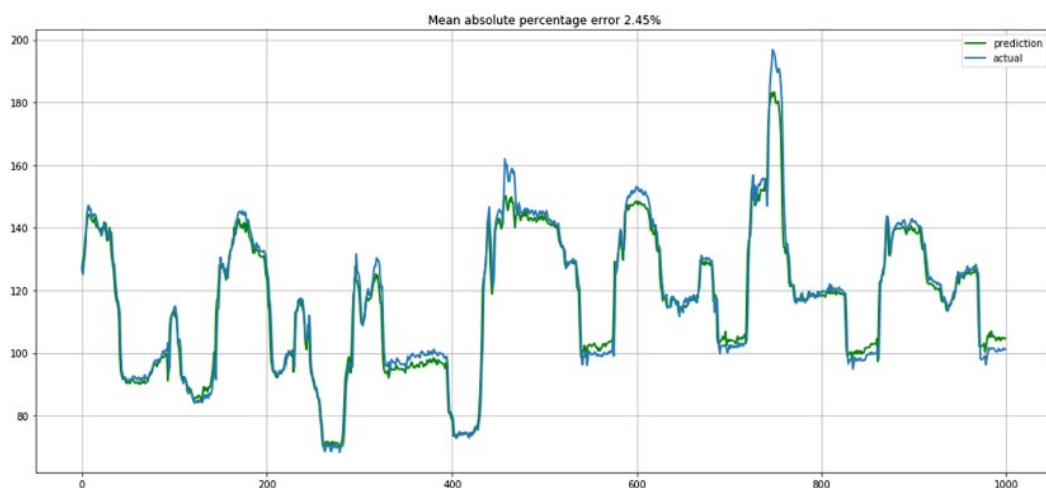
Dalam mengoptimalkan penerapan analisis xG yang telah dibahas sebelumnya, diperlukan model *machine learning* yang mampu menangani data historis dengan efisiensi tinggi. Pada tahap inilah *Light Gradient Boosting Machine* (LightGBM atau LGBM) memainkan peran penting. Secara umum, LightGBM

adalah metode *machine learning* berbasis *Gradient Boosting Decision Tree* (GBDT) yang digunakan untuk prediksi dan klasifikasi data (Ramadanti et al., 2024). Teknologi ini dirancang untuk mengatasi tantangan dalam pemrosesan data besar dan kompleks, menjadikannya unggul dalam berbagai aplikasi berbasis data.

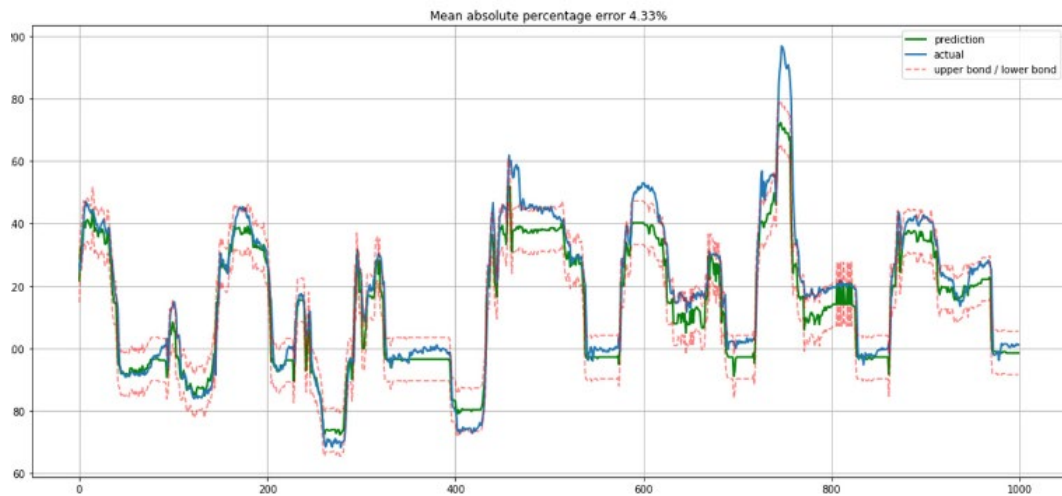
LightGBM memungkinkan pengembangan model xG secara efisien dengan mempelajari pola dari data historis. Dibandingkan dengan GBDT tradisional, LightGBM menawarkan proses pelatihan yang hingga 20 kali lebih cepat tanpa mengurangi akurasi yang dihasilkan (Hartanto et al., 2023). Kemampuan ini sangat penting dalam analisis sepak bola, terutama saat bekerja dengan data berukuran besar dan kompleks. Dengan efisiensinya, LightGBM memungkinkan evaluasi peluang gol dilakukan lebih cepat dan akurat, sehingga memberikan wawasan yang dapat diterapkan dalam waktu nyata untuk mendukung pengambilan keputusan strategis.

LightGBM juga memiliki beberapa keunggulan yang menjadikannya unggul dalam konteks pemodelan prediktif. Sebagai *framework gradient boosting* berbasis algoritma *decision tree*, LightGBM tidak hanya cepat tetapi juga mendukung distribusi data dalam skala besar dengan performa tinggi. Kemampuannya dalam klasifikasi yang unggul telah diaplikasikan secara sukses dalam diagnosis penyakit dan prediksi hasil klinis, memperkuat keandalannya dalam berbagai bidang analitik (Artzi et al., 2020). Dalam sepak bola, kemampuan klasifikasinya dapat digunakan untuk memprediksi hasil pertandingan dan mengidentifikasi peluang yang berpotensi menghasilkan gol.

Dibandingkan dengan model lainnya, seperti XGBoost, LightGBM menunjukkan kinerja prediktif yang lebih baik. Studi menunjukkan bahwa LightGBM menghasilkan nilai *Mean Absolute Percentage Error* (MAPE) sebesar 2,45%, yang lebih kecil daripada MAPE yang dihasilkan oleh XGBoost. Ini menandakan bahwa LightGBM lebih akurat dalam memprediksi variabel target, termasuk dalam konteks *output* daya termal maupun analisis sepak bola (Nemeth et al., 2019). Keunggulan dalam akurasi dan kecepatan ini membuat LightGBM menjadi pilihan utama dalam berbagai aplikasi berbasis data, termasuk pemodelan xG di sepak bola. Gambar 1.3 menunjukkan grafik yang membandingkan MAPE dari berbagai model, di mana LightGBM menonjol dengan nilai MAPE terbaik. Sementara itu, Gambar 1.4 menampilkan performa model lain, yang meskipun kompetitif, tidak mencapai tingkat akurasi yang sama seperti yang dicapai oleh LightGBM. Visualisasi ini memperkuat argumen tentang keunggulan LightGBM dalam analisis prediktif.



Gambar 1.3 Grafik Performa Model LightGBM



Gambar 1.4 Grafik Performa Model Lain

Dalam rangka mendukung analisis yang telah dilakukan sebelumnya dengan menggunakan model LightGBM untuk memprediksi nilai xG, penting untuk menggunakan dataset yang kredibel dan komprehensif. Oleh karena itu, penelitian ini memanfaatkan dataset *open-data* dari StatsBomb, yang dipublikasikan secara gratis oleh perusahaan tersebut untuk mendorong riset akademis dan analisis dalam bidang olahraga (StatsBomb, 2022). Data ini mencakup berbagai informasi rinci terkait pertandingan sepak bola, termasuk data peluang dan tembakan, yang relevan untuk pengembangan model xG. Dengan menggunakan dataset ini, penelitian dapat dilakukan secara lebih komprehensif, memungkinkan analisis mendalam berbasis data historis yang kredibel.



Gambar 1.5 Logo StatsBomb

StatsBomb merupakan perusahaan data olahraga yang didirikan oleh analis dan untuk para analis. Perusahaan ini memiliki tim yang berdedikasi tinggi dalam mengumpulkan serta menganalisis data olahraga terlengkap di dunia. Platform StatsBomb dirancang dari awal untuk memastikan pengumpulan dan analisis data lebih relevan serta fleksibel dibandingkan dengan penyedia lainnya. Kemampuan platform ini untuk merespons kebutuhan, peluang, dan tantangan baru secara cepat menjadikannya salah satu pemimpin dalam industri data olahraga (StatsBomb, 2024).

Penelitian awal mengenai xG dilakukan oleh Lucey et al. (2015), yang menggunakan dataset berisi 9.732 tembakan dan 10 detik cuplikan video sebelum setiap tembakan dalam pertandingan profesional. Dataset ini berasal dari Prozone, yang sekarang dikenal sebagai Stats Perform. Dalam penelitian tersebut, model yang dikembangkan dinamakan *Expected Goal Value* (EGV), dengan algoritma yang digunakan berupa *Conditional Random Models* berbasis model probabilistik. Penelitian ini menunjukkan bahwa pemanfaatan data spasial-temporal tidak hanya meningkatkan prediksi hasil tembakan tetapi juga memberikan wawasan lebih mendalam tentang strategi pemain selama pertandingan.

Penelitian sejenis dilakukan oleh Fairchild et al. (2018), yang memanfaatkan dataset berisi 1.115 tembakan non-penalti dari 99 pertandingan *Major League Soccer* (MLS) di Amerika Serikat. Berbeda dengan penelitian sebelumnya, dataset ini dikumpulkan secara manual. Model yang dibangun dalam penelitian ini adalah *Expected Goal Model*, dengan penggunaan algoritma *Logistic Regression* untuk memprediksi peluang gol berdasarkan posisi dan karakteristik

tembakan. Studi ini berfokus pada analisis spasial tembakan dan menyajikan dimensi fraktal yang menggambarkan pola distribusi peluang mencetak gol di MLS.

Tureen dan Olthoff (2022) memperluas kajian mengenai analisis individual dalam sepak bola melalui pengembangan model *Estimated Player Impact* (EPI). Penelitian ini menggunakan data dari 580 pertandingan *Premier League* dan 326 pertandingan *Women's Super League*, yang semuanya berasal dari penyedia data StatsBomb, sama seperti penelitian kali ini. Model EPI dibangun menggunakan algoritma *Generalised Linear Mixed Models* (GLMM), yang memungkinkan kuantifikasi dampak individu pemain terhadap berbagai aksi dalam pertandingan sepak bola. Studi ini menekankan pentingnya analisis hierarkis untuk mengidentifikasi peran spesifik pemain dalam memengaruhi hasil pertandingan.

Meskipun model xG telah diterapkan dalam berbagai penelitian, penggunaan algoritma LightGBM dalam pengembangan model ini masih terbatas. Cavus dan Biecek (2022) melakukan eksplorasi dengan menggunakan beberapa model, termasuk LightGBM, melalui Forester AutoML. Dataset yang digunakan terdiri dari 315.430 tembakan selama tujuh musim dari lima liga top Eropa, yang disediakan oleh Understat. Model yang dikembangkan dalam studi ini adalah *Explainable Expected Goals*, dengan memanfaatkan algoritma seperti XGBoost, *Random Forest*, LightGBM, dan CatBoost. Namun, hasil penelitian menunjukkan bahwa model terbaik dalam memprediksi peluang gol adalah *Random Forest*, yang mengindikasikan bahwa masih terdapat ruang untuk meningkatkan kinerja LightGBM dalam model xG di masa mendatang.



Berdasarkan penelitian sebelumnya, terdapat beberapa masalah dan tantangan yang mengindikasikan perlunya pengembangan lebih lanjut dalam penerapan model xG menggunakan algoritma LightGBM. Meskipun *Random Forest* telah menunjukkan kinerja terbaik dalam penelitian Cavus dan Biecek (2022), penggunaan LightGBM tetap menjanjikan karena memiliki keunggulan dalam efisiensi waktu dan kemampuan menangani data dalam skala besar. Selain itu, model xG sebelumnya sebagian besar menggunakan algoritma tradisional seperti *Logistic Regression* atau GLMM, yang mungkin kurang optimal dalam menganalisis data kompleks secara cepat dan akurat. LightGBM, dengan kemampuan *gradient boosting* yang canggih, dapat memberikan kombinasi ideal antara kecepatan pelatihan, skalabilitas, dan akurasi tinggi.

Dengan meningkatnya volume dan kompleksitas data sepak bola, khususnya data spasial dan temporal yang disediakan oleh StatsBomb, diperlukan model yang mampu memanfaatkan data tersebut secara maksimal. LightGBM diharapkan dapat mengatasi keterbatasan model-model sebelumnya dengan menghasilkan prediksi yang lebih akurat serta mendukung analisis yang lebih mendalam, misalnya melalui eksplorasi pola tembakan atau pengaruh pemain dalam waktu nyata. Implementasi algoritma ini dalam penelitian xG juga memungkinkan evaluasi baru dalam aspek kualitas peluang, yang tidak hanya berfokus pada jumlah tembakan, tetapi juga pada strategi dan konteks permainan. Oleh karena itu, penelitian ini mencoba membuktikan bahwa penerapan LightGBM dapat menjadi solusi yang lebih baik dalam pengembangan model xG untuk analisis sepak bola.

Berdasarkan latar belakang serta pedoman dari penelitian-penelitian sebelumnya, penulis menyimpulkan bahwa terdapat kebutuhan untuk mengembangkan model xG dengan algoritma yang lebih efisien dan akurat. LightGBM, dengan kemampuan dan keunggulannya dalam menangani *big data*, menawarkan peluang untuk menghasilkan model yang lebih baik dibandingkan model tradisional atau algoritma lain yang telah diterapkan. Oleh karena itu, penelitian ini dilakukan sebagai upaya inovatif dalam analisis sepak bola dengan mengimplementasikan LightGBM untuk xG. Dengan demikian, skripsi ini disusun dengan judul: **"PENERAPAN *LIGHT GRADIENT BOOSTING MACHINE* (LGBM) UNTUK PERHITUNGAN METRIK *EXPECTED GOALS* (xG) DALAM ANALISIS SEPAK BOLA."**

## **1.2 Identifikasi Masalah**

Berdasarkan latar belakang yang telah dipaparkan, berikut merupakan identifikasi masalah pada penelitian ini:

- a. Perhitungan manual xG tidak efisien memakan waktu dan rentan terhadap kesalahan.
- b. Keterbatasan model xG tradisional yang menggunakan algoritma tradisional (misalnya, *Logistic Regression* atau GLMM) belum optimal dalam menangani data spasial-temporal yang kompleks secara cepat dan akurat.
- c. Kurangnya optimalisasi lebih lanjut pada penerapan LightGBM dalam Analisis xG.

### 1.3 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, berikut merupakan rumusan masalah pada penelitian ini:

- a. Bagaimana penerapan algoritma LightGBM untuk meningkatkan akurasi dan efisiensi dalam perhitungan xG dalam analisis sepak bola?
- b. Bagaimana performa dari algoritma LightGBM dalam perhitungan xG dalam analisis sepak bola pada penilaian evaluasi nilai *Area Under Curve* (AUC) dan *Brier Score*?

### 1.4 Batasan Masalah

Batasan masalah yang terdapat pada penelitian ini yaitu:

- a. Penelitian ini hanya akan berfokus pada implementasi LGBM untuk perhitungan xG dalam analisis sepak bola.
- b. Data yang digunakan diambil dari Hudl StatsBomb *open-data* yang berlisensi resmi oleh StatsBomb Services Ltd yang berkantor pusat di University of Bath Innovation Centre, Carpenter House, Broad Quay, Bath, BA1 1UD.
- c. Data terbatas pada *event* data statistik pertandingan, termasuk posisi, jarak, teknik, sudut tembakan dan lainnya.
- d. Penelitian ini fokus pada perhitungan xG menggunakan LightGBM tanpa membandingkan dengan model lain.

- e. Model probabilitas dibangun menggunakan LightGBM, tanpa membahas algoritma lain.
- f. *Preprocessing* dilakukan menggunakan *Python*, fokus pada pembersihan dan transformasi data.
- g. Data dibagi untuk *training* dan *testing* tanpa validasi silang.
- h. Metrik evaluasi terbatas pada *Area Under Curve* (AUC) dan *Brier Score*.

### 1.5 Tujuan Penelitian

Tujuan dilakukannya penelitian ini adalah sebagai berikut:

- a. Penerapan algoritma LightGBM dalam upaya meningkatkan akurasi dan efisiensi perhitungan metrik xG pada analisis sepak bola.
- b. Evaluasi performa algoritma LightGBM dalam perhitungan metrik xG dengan menggunakan penilaian *Area Under Curve* (AUC) dan *Brier Score*.

### 1.6 Manfaat Penelitian

Manfaat dari penelitian ini yaitu sebagai berikut:

- a. Bagi peneliti, penelitian ini merupakan implementasi dari teori yang telah dipelajari dalam bidang analisis data dan *machine learning*, sehingga dapat lebih memahami penerapan algoritma LightGBM dalam perhitungan metrik xG. Selain itu, penelitian ini juga merupakan salah satu syarat kelulusan Strata Satu (S1) Sistem Informasi UIN Syarif Hidayatullah Jakarta.

- b. Bagi Universitas, penelitian ini dapat dijadikan sebagai tolak ukur pengetahuan mahasiswa terkait penerapan algoritma *machine learning* dalam analisis sepak bola, serta sebagai kontribusi dalam pengembangan penelitian di bidang ilmu komputer dan sistem informasi.
- c. Bagi pembaca, penelitian ini dapat memberikan informasi yang komprehensif mengenai algoritma LightGBM dan aplikasinya dalam perhitungan xG, serta dapat dijadikan sebagai referensi tambahan terkait penelitian dalam program studi Sistem Informasi UIN Syarif Hidayatullah Jakarta, khususnya dalam konteks analisis data olahraga. Penelitian ini juga dapat memberikan pemahaman tentang pentingnya analisis data dalam pengambilan keputusan dalam sepak bola.
- d. Bagi klub sepak bola, media sepak bola dan analis sepak bola, hasil dari penelitian ini dapat berfungsi sebagai referensi dalam mengadopsi metode analisis berbasis *machine learning*, serta dalam pengambilan keputusan strategis yang berkaitan dengan taktik permainan, rekrutmen pemain, dan evaluasi kinerja tim.

## 1.7 Metode Penelitian

Metode penelitian ini dibagi menjadi dua bagian, yaitu:

### a. Metode Pengumpulan Data

#### 1) Studi Literatur

Metode studi literatur dilakukan dengan mengumpulkan dan menganalisis berbagai sumber tertulis, seperti buku, artikel ilmiah, dan laporan penelitian yang relevan dengan topik penelitian.

## 2) *Data Extraction*

*Data extraction* adalah proses pengambilan data dari berbagai sumber untuk dianalisis lebih lanjut. Dalam penelitian ini, data yang digunakan diambil dari Hudl StatsBomb *open-data* yang tersedia di GitHub dengan lisensi resmi.

### b. Metode Analisis Data

Penelitian ini menggunakan metode data mining yang dikenal sebagai *Knowledge Discovery in Databases* (KDD). Metode KDD terdiri dari beberapa tahap yang saling berhubungan, sebagai berikut:

#### 1) *Data Selection*

*Data selection* adalah proses pemilihan sub set data yang relevan dari kumpulan data yang lebih besar untuk analisis lebih lanjut. Dalam penelitian ini, pemilihan data difokuskan pada informasi yang terkait dengan tembakan dan peluang gol, sehingga dapat digunakan dalam perhitungan metrik xG.

#### 2) *Preprocessing*

*Preprocessing* adalah langkah yang dilakukan untuk menyiapkan dan membersihkan data sebelum analisis. Ini melibatkan penghapusan data yang tidak relevan, pengisian nilai yang hilang,

dan pengubahan format data agar sesuai dengan kebutuhan analisis. Tahap ini penting untuk memastikan bahwa data yang digunakan dalam penelitian akurat dan dapat diandalkan.

### 3) Data Transformation

Data *transformation* adalah proses mengubah data ke dalam format yang lebih sesuai untuk analisis. Ini termasuk teknik seperti normalisasi, pengkodean variabel kategorial, dan agregasi data. Proses ini memungkinkan model *machine learning* untuk memproses data dengan lebih efisien dan efektif.

### 4) Data Mining

Pada tahap data mining, penelitian ini menggunakan algoritma LGBM untuk membangun model prediktif berdasarkan data yang telah diproses. LGBM dipilih karena kemampuannya dalam menangani data besar dengan efisiensi tinggi, serta akurasi yang dihasilkannya dalam perhitungan xG.

### 5) Evaluation

Setelah model dibangun, evaluasi dilakukan untuk mengukur performa model menggunakan metrik evaluasi seperti AUC dan *Brier Score*.

## 1.8 Sistematika Penulisan

Laporan pada penelitian ini terdiri atas lima bab, yaitu:

**BAB 1            PENDAHULUAN**

Bab ini membahas tentang latar belakang, identifikasi masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan dari penelitian ini.

**BAB 2            TINJAUAN PUSTAKA**

Bab ini membahas tentang teori-teori yang berkaitan dengan metrik xG dalam sepak bola, serta penerapan algoritma LightGBM dalam model prediksi, termasuk tinjauan mengenai penelitian-penelitian terdahulu yang relevan.

**BAB 3            METODOLOGI PENELITIAN**

Bab ini menjelaskan tentang tahapan metode yang digunakan dalam penelitian, meliputi metode pengumpulan data, proses *preprocessing*, analisis data, dan implementasi menggunakan algoritma LightGBM, serta tahapan evaluasi dengan metrik AUC dan *Brier Score*.

**BAB 4            HASIL DAN PEMBAHASAN**

Bab ini berisi hasil dari penerapan algoritma LightGBM dalam perhitungan metrik xG, serta analisis mendalam mengenai kinerja model berdasarkan evaluasi yang dilakukan. Hasil juga dibandingkan dengan model lain untuk menunjukkan efektivitas LightGBM.

**BAB V            PENUTUP**

Bab ini berisi kesimpulan dari hasil penelitian mengenai penerapan algoritma LightGBM dalam perhitungan metrik xG, serta saran-saran yang dapat digunakan untuk penelitian selanjutnya dalam bidang analisis sepak bola dan penerapan *machine learning*.



## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Analisis Sepak Bola**

Analisis sepak bola merupakan proses yang kompleks dan melibatkan berbagai aspek dari permainan yang saling terkait. Secara mendasar, analisis ini mencakup pengukuran komunikasi antar pemain, kemampuan adaptasi, tempo permainan, serta evaluasi taktik penyerangan dan pertahanan (Mclean et al., 2017). Analisis ini memperhitungkan dimensi sosial dan teknis dalam sepak bola, di mana pemahaman akan sistem permainan sangat penting dalam mengoptimalkan kinerja tim secara keseluruhan.

Lebih jauh, analisis dalam sepak bola tidak hanya fokus pada aspek teknis dan taktis, tetapi juga memperhatikan variabel fisik yang relevan dalam konteks permainan sepak bola pria dewasa. Di samping itu, terdapat variabel situasional yang perlu diperhatikan seperti lokasi pertandingan, kualitas lawan, dan status pertandingan yang berpengaruh pada performa tim (Sarmiento et al., 2014). Faktor-faktor ini menambah kompleksitas analisis dan menekankan pentingnya pendekatan menyeluruh yang mempertimbangkan kondisi dinamis permainan.

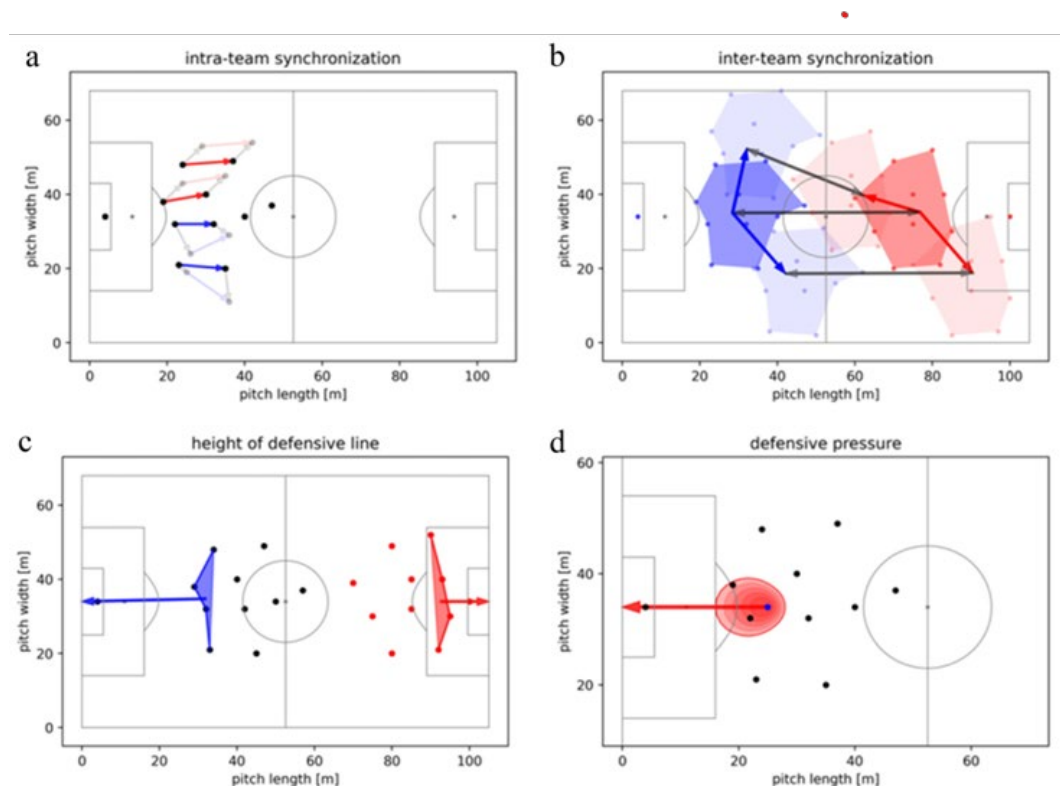
Dalam upaya meningkatkan performa pemain dan mengembangkan aktivitas pelatih, analisis sepak bola juga mengarah pada aspek-aspek mendetail seperti performa dalam situasi bola mati, perilaku sistem kolektif, komunikasi tim, dan profil aktivitas pemain. Fokus ini bertujuan untuk memberikan wawasan yang lebih dalam mengenai pola-pola permainan serta interaksi pemain di lapangan, yang

pada akhirnya membantu pelatih dalam menyesuaikan strategi berdasarkan analisis berbasis data yang komprehensif (Sarmiento et al., 2018).

Salah satu contoh penerapan analisis sepak bola yang semakin populer adalah penggunaan *data tracking* pemain. Data ini memungkinkan analisis yang lebih mendalam terhadap struktur permainan dengan memberikan wawasan mengenai performa tim, terutama dalam strategi bertahan. Implementasi analisis ini mengidentifikasi karakteristik permainan defensif yang berhasil, ditandai dengan tekanan tinggi, sinkronisasi gerakan antar pemain, keseimbangan pertahanan, serta organisasi pertahanan yang kompak dan terkoordinasi. Melalui *data tracking*, pelatih dan analis dapat memahami pola pertahanan yang efektif dan mengoptimalkan strategi tim berdasarkan perilaku lapangan yang terukur (Forcher et al., 2022).

Gambar 2.1 menunjukkan contoh visualisasi hasil analisis menggunakan *data tracking*, yang menggambarkan indikator-indikator kinerja utama dalam permainan bertahan. Pada bagian (a), visualisasi menunjukkan tingkat sinkronisasi intra-tim di mana perilaku gerakan yang sinkron digambarkan pada warna merah dan perilaku yang asinkron pada warna biru. Bagian (b) menggambarkan sinkronisasi gerakan antar tim melalui pusat massa (*centroid*) dari tim-tim yang berlawanan. Pada bagian (c), visualisasi ini menunjukkan tinggi garis pertahanan (biru) yang digunakan sebagai pengukur posisi bertahan. Terakhir, bagian (d) menunjukkan tekanan bertahan yang dilakukan oleh dua pemain bertahan (hitam) terhadap pemain penyerang (biru), dengan arah ancaman ke gawang yang

ditunjukkan oleh panah merah. Visualisasi ini memberikan gambaran yang jelas tentang dinamika taktik bertahan dalam sepak bola (Forcher et al., 2022).



Gambar 2.1 Contoh Visualisasi pada Analisis Sepak Bola (Forcher et al., 2022)

## 2.2 Expected Goals (xG)

*Expected Goals* atau xG adalah salah satu metrik yang semakin digunakan dalam analisis sepak bola modern untuk menilai peluang terjadinya gol berdasarkan kualitas dan lokasi tembakan yang dilakukan (Mead, O'Hare, & McMenemy, 2023). Metrik ini memberikan prediksi probabilitas yang lebih akurat dibandingkan statistik konvensional dalam memperkirakan keberhasilan suatu tim di masa mendatang. Dalam hal ini, xG membantu memberikan pandangan yang lebih

obyektif dan berbasis data mengenai kemungkinan pencapaian gol yang dihasilkan dari berbagai jenis tembakan selama pertandingan.

Metrik xG dirancang untuk memberikan skor probabilistik pada setiap tembakan, dengan nilai yang berkisar antara 0 dan 1, di mana 0 menunjukkan tidak ada peluang mencetak gol, dan 1 menunjukkan kepastian terjadinya gol. Penilaian ini memungkinkan xG untuk menangani unsur ketidakpastian dalam sepak bola dengan lebih baik dibandingkan metrik berbasis gol konvensional. Karena tembakan jauh lebih sering terjadi daripada gol, pendekatan ini memungkinkan analisis yang lebih stabil dan realistis dalam memahami efektivitas tim dan pemain di lapangan (Mead, O'Hare, & McMenemy, 2023).

Selain berguna untuk analisis taktis yang mendukung peningkatan performa di lapangan, xG juga memainkan peran penting dalam keputusan finansial klub. Metrik ini membantu dalam keputusan seperti perekrutan pemain dan negosiasi kontrak dengan memberikan wawasan yang lebih akurat mengenai kontribusi pemain. Dengan demikian, xG tidak hanya membantu klub dalam memaksimalkan performa di lapangan tetapi juga dalam mengelola sumber daya finansial secara lebih efisien (Mead, O'Hare, & McMenemy, 2023).

Penerapan xG memberikan keuntungan strategis bagi klub sepak bola dengan memperluas pemahaman terkait kualitas peluang yang dihasilkan. Hal ini memungkinkan klub untuk mengevaluasi kinerja pemain secara lebih mendalam dan membantu dalam pengembangan strategi permainan yang berbasis pada kualitas dan efektivitas peluang (Mead, O'Hare, & McMenemy, 2023). Dengan menerapkan xG, klub dapat membuat keputusan yang lebih baik dalam berbagai

aspek, termasuk analisis performa, perekrutan, dan perencanaan jangka panjang, yang menjadikan xG sebagai alat yang sangat berharga dalam manajemen modern sepak bola.

Di dalam konsepnya, perhitungan xG dapat dianggap sebagai permasalahan klasifikasi, karena melibatkan penentuan probabilitas tembakan menghasilkan gol berdasarkan berbagai faktor. Untuk menghitung probabilitas ini, metode *machine learning* dan statistika sering diterapkan, termasuk *logistic regression*, *gradient boosting*, *neural networks*, *support vector machines*, serta algoritma klasifikasi *tree-based*. Beragam pendekatan ini memungkinkan xG untuk memanfaatkan data historis dan pola dalam data tembakan untuk memodelkan kemungkinan gol secara lebih akurat, yang berguna dalam memberikan penilaian yang lebih detail tentang kualitas peluang tembakan (Herbinet, 2018).

Model xG dapat memiliki tingkat akurasi yang berbeda tergantung pada jumlah faktor yang dimasukkan ke dalam perhitungannya. Sebagai contoh, model xG standar biasanya memperhitungkan jarak tembakan ke gawang, sudut tembakan, bagian tubuh yang digunakan, dan jenis umpan yang mendahului tembakan.

Berdasarkan faktor-faktor tersebut, sebuah tembakan mungkin diberi nilai 0,30 xG. namun model yang lebih presisi, seperti Statsbomb xG, mempertimbangkan informasi tambahan seperti posisi kiper, status kiper, posisi pemain bertahan dan penyerang, serta tinggi dampak tembakan. Dalam kondisi kiper yang tidak berada di posisinya, model ini mungkin memberikan nilai yang lebih tinggi, misalnya 0,65 xG, untuk menggambarkan kualitas peluang yang lebih tinggi (Statsbomb, 2024).

Visualisasi dari model ini pada Gambar 2.2, yang merupakan Visualisasi xG pada Pertandingan Langsung, memperlihatkan bagaimana setiap faktor dihitung untuk menghasilkan prediksi xG yang mendalam dan akurat.



Gambar 2.2 Visualisasi xG pada Pertandingan Langsung (Statsbomb, 2024)

### 2.3 Machine Learning

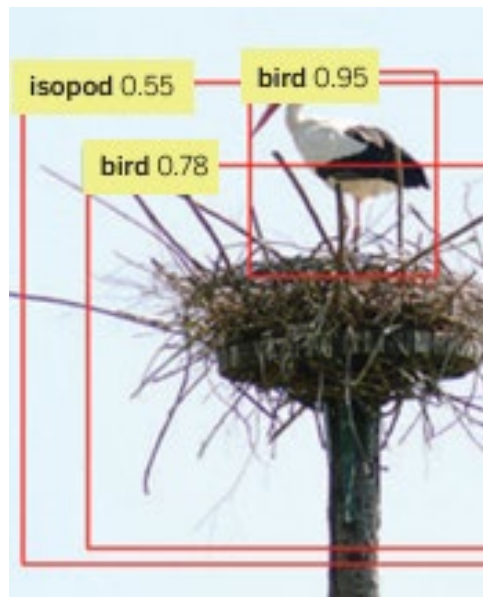
*Machine Learning* (ML) merupakan kemampuan suatu sistem untuk belajar dari data pelatihan yang spesifik terhadap masalah tertentu, dengan tujuan untuk mengotomatisasi proses pembangunan model analitik serta memecahkan tugas-tugas terkait. Dalam konteks ini, ML memungkinkan sistem komputer untuk mengidentifikasi pola dalam data tanpa campur tangan manual yang intensif, sehingga memungkinkan solusi otomatis terhadap berbagai masalah kompleks berbasis data (Janiesch et al., 2021).

Secara lebih mendalam, ML dapat dilihat sebagai bentuk kecerdasan buatan (AI) yang memanfaatkan data untuk melatih komputer dalam melakukan berbagai tugas tertentu, menggunakan algoritma untuk membangun serangkaian aturan

secara otomatis. Proses ini memungkinkan sistem untuk secara mandiri mengenali pola serta membuat keputusan berdasarkan data tanpa perlu diinstruksikan secara eksplisit, yang pada akhirnya meningkatkan ketepatan dan efisiensi sistem dalam memecahkan masalah kompleks (Schneider & Guo, 2018).

*Machine Learning* berbeda dari data *mining* dan statistik tradisional, baik dalam aspek filosofis maupun metodologis. Terdapat tiga pendekatan utama dalam ML yang membedakannya, yaitu statistika klasik, teori pembelajaran statistik Vapnik, serta teori pembelajaran komputasional (Kodama *et al.*, 2023). Ketiga pendekatan ini menyediakan dasar yang berbeda untuk pengembangan algoritma, dimana ML fokus pada kemampuan untuk terus memperbaiki kinerja model berdasarkan data pelatihan, dibandingkan hanya melakukan analisis data retrospektif sebagaimana dalam statistik tradisional.

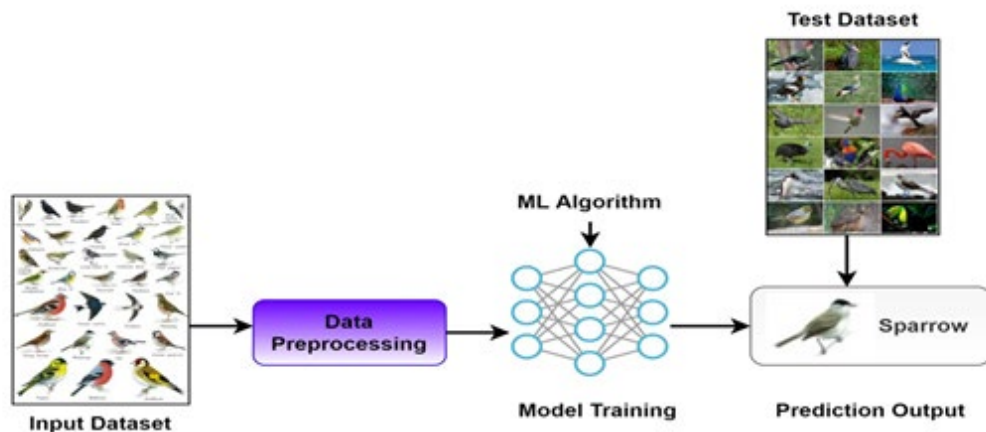
*Machine Learning* memiliki penerapan yang luas, salah satunya adalah *visual recognition*, yang memungkinkan pengenalan objek atau wajah dalam gambar secara otomatis. Dalam Gambar 2.3, ditampilkan implementasi ML pada aplikasi pengenalan visual, di mana teknologi ini mengenali dan mengklasifikasikan objek secara *real-time* berdasarkan data visual. Penerapan ini memanfaatkan kemampuan ML untuk belajar dari data visual guna membangun model yang mampu mendeteksi pola, bentuk, atau warna tertentu, serta mengenali objek-objek spesifik secara akurat dan efisien.



Gambar 2.3 Contoh Implementasi *Machine Learning* (Jordan & Mitchell, 2015)

Terdapat berbagai kategori dalam *Machine Learning*, meliputi *supervised learning*, *unsupervised learning*, dan *reinforcement learning*. Masing-masing pendekatan ini memiliki teknik-teknik unik, seperti *zero-shot learning*, *active learning*, *contrastive learning*, *self-supervised learning*, dan *semi-supervised learning* (Mahadevkar et al., 2022). Dalam Gambar 2.4, ditunjukkan contoh implementasi *supervised learning*, di mana model dilatih menggunakan data berlabel untuk dapat mengklasifikasikan atau memprediksi berdasarkan pola yang telah dikenali. Teknik-teknik ini memperkaya cara sistem mempelajari data visual, baik dengan data yang memiliki label atau tanpa label.





Gambar 2.4 Contoh Implementasi *Supervised Learning* (Mahadevkar et al., 2022)

Algoritma dasar dalam *Machine Learning* sangat beragam, mencakup *decision tree*, *random forest*, *artificial neural network*, *support vector machine* (SVM), serta algoritma *boosting* dan *bagging*, yang membantu dalam meningkatkan kinerja model dengan menggabungkan prediksi dari beberapa model. Selain itu, algoritma *backpropagation* (BP) berperan penting dalam *neural networks* untuk mengoptimalkan bobot model berdasarkan kesalahan yang dihasilkan pada prediksi awal, sehingga meningkatkan kemampuan sistem dalam memprediksi hasil dengan lebih akurat (Jin, 2020).

Dalam *Machine Learning*, metrik evaluasi adalah instrumen logis dan matematis yang digunakan untuk mengukur seberapa dekat hasil prediksi model terhadap nilai aktualnya. Metrik evaluasi memungkinkan analisis kinerja model secara mendalam, sehingga aspek seperti akurasi, kesalahan, dan ketepatan dalam memprediksi dapat diukur secara kuantitatif. Hal ini penting untuk memahami performa model dan menentukan langkah-langkah penyempurnaan lebih lanjut dalam pengembangan model (Plevris et al., 2022).

Beberapa metrik evaluasi yang paling sering digunakan dalam *Machine Learning* mencakup *Root Mean Squared Error* (RMSE), *Mean Absolute Error* (MAE), *Pearson Correlation Coefficient*, dan *Coefficient of Determination* ( $R^2$ ) (Plevris *et al.*, 2022). Metrik-metrik ini membantu dalam mengukur seberapa akurat dan presisi prediksi model terhadap data yang diujikan, sehingga para praktisi dapat memilih metrik evaluasi yang paling relevan dengan konteks data dan tujuan analisis mereka.

## 2.4 Data Preprocessing

Data *preprocessing* adalah tahap penting yang bertujuan untuk menghasilkan kumpulan data akhir yang akurat, bersih, dan siap digunakan dalam algoritma penambangan data berikutnya (García, Luengo & Herrera, 2016). Proses ini memastikan bahwa data yang akan dianalisis telah melalui serangkaian langkah perbaikan dan penyesuaian, seperti pembersihan dari kesalahan, penghapusan data duplikat, serta transformasi data ke format yang lebih sesuai. Dengan demikian, data yang dihasilkan akan memiliki kualitas yang lebih tinggi dan dapat mendukung proses penambangan data secara lebih efektif serta menghasilkan informasi yang lebih andal.

Teknik *preprocessing* data mencakup serangkaian langkah penting yang mencakup:

- a. Pembersihan data
- b. Integrasi data
- c. Reduksi data

d. Transformasi data.

Proses ini dirancang untuk memastikan bahwa data yang akan digunakan dalam penambangan memiliki kualitas yang tinggi dan struktur yang optimal (Sammut & Webb, 2017).

Data *preprocessing* merupakan tahap yang sangat krusial dalam *pipeline machine learning*, karena berperan langsung dalam menentukan kualitas data yang akan digunakan serta informasi yang dihasilkan dari proses tersebut (Bilal *et al.*, 2022). Tahap ini memastikan bahwa data mentah yang tersedia diubah menjadi data yang lebih terstruktur, bersih, dan relevan untuk analisis lebih lanjut. Kualitas data yang diproses dengan baik akan sangat mempengaruhi kinerja model *machine learning* yang dibangun, sehingga menghasilkan prediksi atau keputusan yang lebih akurat dan andal.

## 2.5 Feature Engineering

*Feature engineering* adalah proses rekayasa data secara cerdas untuk meningkatkan kinerja model *machine learning* dengan cara meningkatkan akurasi dan interpretabilitasnya (Verdonck *et al.*, 2024). Proses ini dilakukan melalui penyesuaian fitur yang telah ada atau dengan mengekstraksi fitur baru yang lebih bermakna dari berbagai sumber data. Teknik ini bertujuan untuk menciptakan representasi data yang lebih informatif, sehingga model dapat memahami hubungan yang lebih kompleks di dalam data. *Feature engineering* tidak hanya membantu dalam memperbaiki akurasi prediksi, tetapi juga memungkinkan pengguna untuk memahami bagaimana setiap fitur memengaruhi hasil akhir, menjadikannya

langkah penting dalam pengembangan model *machine learning* yang lebih efektif dan dapat diandalkan.

*Feature engineering* memungkinkan pengguna untuk membuat fitur-fitur baru secara mandiri yang lebih relevan dengan permasalahan yang sedang dianalisis (Das *et al.*, 2022). Fitur-fitur ini kemudian dapat digunakan untuk meningkatkan proses penerapan algoritma *machine learning* dalam membuat prediksi yang lebih akurat. Dengan menciptakan fitur yang disesuaikan dengan kebutuhan analisis, pengguna dapat membantu model *machine learning* mengenali pola-pola penting yang sebelumnya tidak terdeteksi, sehingga hasil prediksi menjadi lebih optimal dan bermakna.

Teknik-teknik esensial dalam *feature engineering* berperan penting dalam meningkatkan kinerja model prediksi di berbagai bidang. Teknik-teknik ini mencakup (Katya, 2023):

a. *Feature Selection*

*Feature Selection* merupakan proses memilih fitur-fitur yang paling relevan dan informatif dari kumpulan data yang tersedia. Dengan menyaring fitur yang tidak signifikan atau *redundant*, proses ini membantu mengurangi *noise* dan kompleksitas data. Hal tersebut sangat penting untuk mencegah *overfitting* dan memastikan bahwa model hanya menggunakan informasi yang benar-benar berkontribusi terhadap variabel target. Dengan demikian, model prediksi dapat bekerja lebih efisien dan menghasilkan akurasi yang lebih tinggi.

b. *Dimensionality Reduction*

*Dimensionality reduction* adalah teknik yang bertujuan untuk mengurangi jumlah fitur dalam *dataset* tanpa mengorbankan informasi penting yang terkandung di dalamnya. Teknik ini menyederhanakan struktur data, sehingga memudahkan proses analisis dan meningkatkan performa model. Metode seperti *Principal Component Analysis* (PCA) mengubah fitur asli menjadi komponen baru yang lebih ringkas, tetapi tetap merepresentasikan variasi data secara keseluruhan. Pendekatan ini tidak hanya mempercepat proses pelatihan model, tetapi juga meningkatkan interpretabilitas hasil.

c. *Interaction Term Creation*

*Interaction term creation* adalah proses menciptakan fitur baru dengan mengombinasikan dua atau lebih fitur yang ada. Teknik ini dirancang untuk menangkap interaksi atau hubungan sinergis antar fitur yang mungkin tidak terlihat saat dianalisis secara individual. Dengan menggabungkan fitur-fitur tersebut, model dapat lebih sensitif terhadap pola-pola kompleks yang berpengaruh terhadap hasil akhir, sehingga meningkatkan keakuratan prediksi.

Secara keseluruhan, penerapan teknik-teknik ini dalam *feature engineering* membantu mengoptimalkan data input sehingga algoritma *machine learning* dapat menghasilkan prediksi yang lebih akurat dan interpretasi yang lebih mendalam. Teknik-teknik tersebut berperan penting dalam menyederhanakan, menyoroti, dan memperkaya informasi yang terkandung dalam data, yang pada akhirnya berkontribusi terhadap peningkatan kinerja model di berbagai aplikasi.

## 2.6 Gradient Boosting

*Gradient boosting* merupakan teknik *machine learning* yang sangat efektif dan sering digunakan untuk menangani tugas dengan fitur heterogen serta data yang cenderung berisik. Teknik ini bekerja dengan menggabungkan prediksi dari sejumlah model sederhana atau *weak learners* untuk menghasilkan prediksi yang kuat. Dalam klasifikasi, *Gradient boosting* menghasilkan distribusi pada label kelas, sementara dalam regresi, model ini memberikan prediksi nilai tunggal atau *point prediction* untuk mendekati hasil yang diinginkan. Kemampuan *gradient boosting* dalam menghadapi variasi pada fitur dan ketidakpastian dalam data menjadikannya alat yang sangat kuat dalam berbagai aplikasi *machine learning* (Ustimenko, Prokhorenkova, & Malinin, 2021).

Proses *gradient boosting* dimulai dengan mengombinasikan *weak learners*, yaitu model yang performanya sedikit lebih baik dari prediksi acak, untuk membentuk *strong learner* secara iteratif. *gradient boosting* merupakan algoritma *boosting* yang dirancang khusus untuk masalah regresi.

Dalam algoritma ini, diberikan kumpulan data pelatihan  $D = \{x_i, y_i\}_1^N$ , dengan tujuan utama mencari aproksimasi  $\hat{F}(x)$  dari fungsi  $F^*(x)$ , yang memetakan instance  $x$  ke nilai output  $y$ , melalui minimisasi nilai ekspektasi dari fungsi loss tertentu  $L(y, F(x))$ . *Gradient boosting* membangun aproksimasi tambahan dari  $F^*(x)$  sebagai jumlah berbobot dari sejumlah fungsi, sehingga memungkinkan model meningkatkan akurasi prediksi melalui iterasi yang berfokus pada mengurangi kesalahan residu (Bentéjac, Csörgő, & Martínez-Muñoz, 2020).

Pada persamaan 2.1 menunjukkan bagaimana setiap model baru ( $x$ ) ditambahkan secara bertahap dengan bobot pada iterasi ke- $m$ , yang bertujuan untuk mengurangi kesalahan prediksi dari model sebelumnya.

$$F_m(x) = F_{m-1}(x) + \rho_m h_m(x) \quad (2.1)$$

Dalam proses iteratif *gradient boosting*,  $\rho_m$  adalah bobot yang diberikan pada fungsi ke- $m$ , yaitu  $h_m(x)$ . Fungsi-fungsi ini merupakan model-model dalam *ensemble*, seperti *decision tree*. Aproksimasi dari  $F^*(x)$  dibangun secara bertahap, dimulai dengan mendapatkan aproksimasi konstan untuk  $F^*(x)$  pada iterasi pertama. Hal ini dicapai dengan meminimalkan nilai *loss function*  $L(y_i, \alpha)$  untuk setiap data pelatihan, dengan  $\alpha$  adalah parameter konstanta yang mengoptimalkan fungsi tersebut. Pada iterasi pertama, aproksimasi ini diberikan oleh persamaan 2.2.

$$F_0(x) = \underset{\alpha}{\operatorname{argmin}} \sum_{i=1}^N L(y_i, \alpha) \quad (2.2)$$

Persamaan ini menunjukkan bahwa pada awalnya, model menghasilkan prediksi yang didasarkan pada nilai konstanta  $\alpha$  yang meminimalkan kesalahan prediksi keseluruhan,  $L(y_i, \alpha)$ , di seluruh dataset. Pendekatan ini digunakan untuk membangun dasar dari model *gradient boosting* sebelum melanjutkan ke iterasi selanjutnya, di mana model-model tambahan (seperti *decision tree*) akan berfungsi untuk memperbaiki prediksi dari model sebelumnya (Bentéjac, Csörgő, & Martínez-Muñoz, 2020).

Pada iterasi selanjutnya, model yang dibangun diharapkan dapat meminimalkan fungsi berikut:

$$(\rho_m, h_m(x)) = \underset{\rho, h}{\operatorname{argmin}} \sum_{i=1}^N L(y_i, F_m - 1(x_i) + \rho h(x_i)) \quad (2.3)$$

Namun, alih-alih menyelesaikan masalah optimisasi ini secara langsung, setiap model  $h_m$  dapat dipandang sebagai langkah *greedy* dalam optimisasi menggunakan metode *gradient descent* untuk  $F^*$ . Untuk itu, setiap model  $h_m$  dilatih menggunakan dataset baru  $D = \{x_i, r_{mi}\}_{i=1}^N$ , di mana residual palsu  $r_{mi}$  dihitung berdasarkan turunan dari fungsi *loss*  $L(y, F(x))$  terhadap  $F(x)$ , yang dievaluasi pada  $F(x) = F_{m-1}(x)$ , dengan rumus:

$$r_{mi} = \left[ \frac{\partial L(y_i, F(x))}{\partial L(x)} \right]_{F(x)=F_{m-1}(x)} \quad (2.4)$$

Nilai dari  $\rho_m$  kemudian dihitung dengan menyelesaikan masalah optimisasi pencarian garis. Proses ini, meskipun sangat efektif, dapat mengalami *overfitting* jika langkah-langkah iteratif tidak diatur dengan benar. Beberapa fungsi *loss* (misalnya *loss* kuadratik) dapat menyebabkan residual palsu menjadi nol pada iterasi berikutnya jika model  $h_m$  sangat cocok dengan residual palsu, yang akan menyebabkan proses tersebut berhenti terlalu cepat. Untuk mengatasi masalah ini dan mengontrol proses penambahan dalam *gradient boosting*, beberapa parameter regularisasi dipertimbangkan. Salah satu cara alami untuk meredakan *overfitting* adalah dengan menerapkan *shrinkage*, yang berfungsi untuk mengurangi setiap langkah *gradient descent* (Bentéjac, Csörgő, & Martínez-Muñoz, 2020).

*Gradient boosting* membedakan dirinya dari metode *boosting* lainnya dengan menggabungkan konsep-konsep dari teori klasifikasi untuk estimasi dan seleksi efek prediktor dalam model regresi. Dalam hal ini, *gradient boosting*



mempertimbangkan efek acak dan menawarkan pendekatan pemodelan yang lebih organik dan tidak bias. Berbeda dengan algoritma *boosting* lainnya yang mungkin mengasumsikan hubungan linier atau terlalu bergantung pada keputusan acak dalam tahap pemilihan model, *gradient boosting* memastikan bahwa estimasi prediktor disesuaikan secara cermat dengan data, meningkatkan akurasi model secara keseluruhan (Griesbach, Säfken, & Waldmann, 2020).

Selain itu, *gradient boosting* juga menawarkan kemampuan untuk menghasilkan perbaikan pada model non-konstan, dengan menggabungkan pengetahuan sebelumnya atau wawasan fisik terkait proses yang menghasilkan data (Wozniakowski, Thompson, Gu, & Binder, 2021). Ini menjadi keunggulan lain dari *gradient boosting*, karena ia tidak hanya mengandalkan data murni, tetapi juga dapat memanfaatkan pengetahuan domain atau pemahaman fisik tentang bagaimana data tersebut terbentuk. Dengan pendekatan ini, *gradient boosting* dapat meningkatkan prediksi dalam konteks yang lebih luas, termasuk dalam situasi di mana model yang lebih sederhana mungkin gagal.

Sebagai algoritma *Ensemble Learning* yang semakin berkembang, telah terbukti unggul dalam meningkatkan prediksi dibandingkan dengan model lain, seperti *artificial neural network*, terutama dalam konteks pemodelan dinamis *bioprocess* (Mowbray et al., 2020). Dalam penerapan ini, *gradient boosting* menggabungkan beberapa model pembelajaran yang lemah untuk menghasilkan prediksi yang lebih akurat, menunjukkan keunggulannya dalam memodelkan dan memprediksi proses yang dinamis dan kompleks, serta mampu mengatasi variasi yang ada dalam data yang digunakan.

Beberapa parameter dalam *gradient boosting*, seperti jumlah *node*, kedalaman maksimum, dan tingkat pembelajaran, dapat disesuaikan berdasarkan kinerja model pada *testing* set (Hu et al., 2023). Pengaturan parameter ini penting untuk memastikan model tidak hanya memberikan prediksi yang akurat, tetapi juga menghindari *overfitting*. Menyesuaikan parameter-parameter tersebut memungkinkan pemodel untuk mengoptimalkan performa model sesuai dengan karakteristik data yang digunakan, menjadikannya lebih fleksibel dan dapat diandalkan dalam berbagai jenis aplikasi.

## 2.7 Light Gradient Boosting Machine

*Light Gradient Boosting Machine* adalah kerangka kerja yang dirancang untuk mengimplementasikan algoritma *Gradient Boosting Decision Tree* (GBDT). LGBM memiliki beberapa keunggulan, termasuk kecepatan pelatihan yang lebih tinggi, penggunaan memori yang lebih rendah, akurasi yang lebih baik, serta dukungan untuk distribusi data dalam jumlah besar. *Framework* ini dikembangkan untuk mengatasi keterbatasan dalam GBDT tradisional, khususnya dalam hal kinerja dan efisiensi komputasi, sehingga memungkinkan pelatihan model pada dataset yang lebih besar dengan waktu yang lebih singkat (Huang & Chen, 2023).

LightGBM pertama kali dikembangkan pada tahun 2016 oleh tim peneliti di Microsoft sebagai peningkatan atas model GBDT yang populer, yaitu XGBoost. LightGBM diperkenalkan untuk meningkatkan efisiensi dan kecepatan yang lebih tinggi dari XGBoost, yang sering mengalami kendala kecepatan pada data berukuran besar. Dalam pengembangan LGBM, tim peneliti memperkenalkan dua

teknik baru: *Gradient-based One-Side Sampling* (GOSS) dan *Exclusive Feature Bundling* (EFB). Teknik ini dirancang untuk mengurangi jumlah sampel data dan fitur yang perlu diproses dalam pelatihan GBDT, sehingga mengatasi tantangan komputasi yang terkait dengan pemrosesan dataset besar (Kriuchkova, Toloknova, & Drin, 2024).

LightGBM menunjukkan kegunaan yang sangat luas dalam berbagai bidang dan masalah. Dalam masalah penugasan tugas multi-UAV (Unmanned Aerial Vehicle), model LightGBM memberikan solusi yang lebih baik dan cakupan solusi yang lebih luas dibandingkan algoritma lainnya. Hal ini menunjukkan kemampuannya dalam menangani masalah kompleks yang melibatkan banyak variabel dan pengambilan keputusan secara bersamaan (Wang & Zhang, 2023).

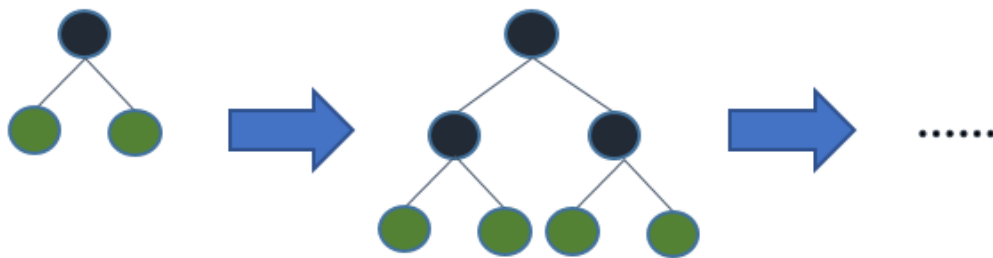
Dalam pemuliaan tanaman berbantuan genomik, LightGBM terbukti memberikan kinerja yang superior dalam hal presisi prediksi, kestabilan model, dan efisiensi komputasi (Yan et al., 2021). Hal ini membantu mempercepat proses pemuliaan tanaman dengan menggunakan data genomik untuk memilih sifat-sifat tanaman yang diinginkan.

Selain itu, LightGBM juga menunjukkan tingkat diskriminasi yang lebih tinggi dan kecepatan pelatihan yang lebih cepat dalam peningkatan efisiensi manajerial perusahaan pertanian yang terdaftar (Xi, 2023). Dalam hal ini, model LightGBM tidak hanya meningkatkan akurasi keputusan bisnis tetapi juga mempercepat proses pengambilan keputusan dengan efisiensi yang lebih baik.

Dalam prediksi beban termal bangunan, LightGBM lebih unggul dibandingkan dengan algoritma *Random Forest* (RF) dan *Long Short-Term Memory*

(LSTM) dalam hal akurasi prediksi dan biaya komputasi, menjadikannya pilihan yang sangat efisien untuk aplikasi di bidang konstruksi dan manajemen energi bangunan (Chen et al., 2023).

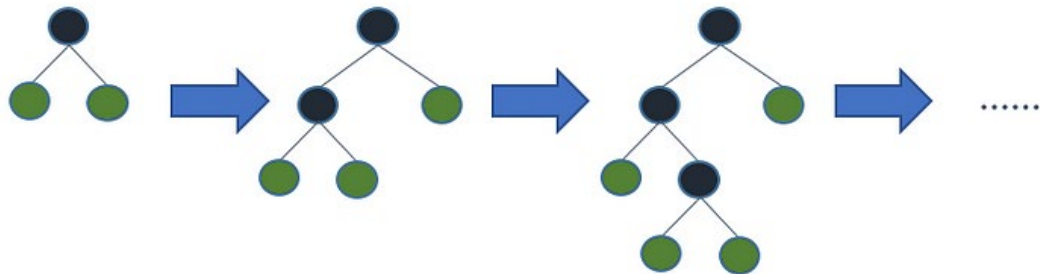
LightGBM menggunakan pendekatan yang berbeda dalam *decision tree learning* dibandingkan algoritma *decision tree* tradisional yang biasanya tumbuh berdasarkan tingkat atau kedalaman pohon (*depth-wise*). Dalam metode tradisional ini, semua *node* pada tingkat yang sama dianggap sama pentingnya, dan pohon bertumbuh secara berjenjang untuk mencakup setiap *node* pada tingkat tertentu, seperti yang ditunjukkan pada gambar 2.5 (LightGBM, 2024).



Gambar 2.5 Ilustrasi *Level-wise Tree Growth* (LightGBM, 2024)

Namun, LightGBM mengadopsi strategi pertumbuhan pohon berbasis daun atau *leaf-wise*, yang hanya membagi daun yang diharapkan memberikan peningkatan terbesar terhadap akurasi model, seperti pada gambar 2.6. Dengan fokus pada daun yang paling berpotensi untuk meningkatkan performa model, LightGBM membangun pohon secara lebih selektif dan efisien. Strategi *leaf-wise* ini bertujuan untuk memaksimalkan akurasi model dengan sumber daya yang lebih minimal, dibandingkan dengan metode tradisional yang sering kali menghasilkan

cabang-cabang pohon yang tidak diperlukan dan memperlambat proses pelatihan (LightGBM, 2024).



Gambar 2.6 Ilustrasi *Leaf-wise Tree Growth* (LightGBM, 2024)

Pendekatan *leaf-wise* dalam LightGBM sering disebut juga sebagai pertumbuhan "*greedy growth*," yang memungkinkan algoritma untuk menemukan dan membagi daun dengan dampak terbesar terhadap akurasi model tanpa harus mempertimbangkan semua cabang secara merata pada setiap tingkat (LightGBM, 2024). Hal ini dapat diibaratkan seperti memangkas cabang-cabang yang tidak perlu, dengan fokus pada jalur yang paling bermanfaat. Sebagai akibat dari pendekatan yang selektif ini, struktur pohon dalam LightGBM menjadi asimetris, di mana beberapa cabang tumbuh lebih dalam daripada cabang lainnya, karena tujuan utamanya bukan simetri, melainkan peningkatan akurasi model.

Manfaat dari strategi pertumbuhan berbasis daun ini adalah dalam hal kecepatan dan akurasi (LightGBM, 2024). Dari segi kecepatan, LightGBM menjadi sangat efisien karena metode *leaf-wise*-nya hanya membagi daun yang memberikan dampak signifikan pada model, sehingga menghindari pengembangan sub-pohon yang tidak berkontribusi banyak terhadap peningkatan akurasi. Selain itu, pertumbuhan *leaf-wise* ini cenderung menghasilkan model dengan tingkat

kesalahan (*loss*) yang lebih rendah dan akurasi yang lebih tinggi, karena algoritma dapat lebih terfokus pada bagian data yang paling informatif. Hal ini menjadikan LightGBM sebagai algoritma yang unggul dalam hal efisiensi dan ketepatan dalam menangani dataset yang besar dan kompleks.

Untuk mengimplementasikan LightGBM, *library* utama yang diperlukan adalah LightGBM itu sendiri, yang dapat diinstal melalui pengelola paket sesuai bahasa pemrograman yang digunakan, seperti *Python* atau R (LightGBM, 2024). Selain *library* utama tersebut, ada beberapa dependensi lain yang juga dibutuhkan, seperti CMake untuk membangun lingkungan pengembangan dan library CUDA jika ingin memanfaatkan akselerasi GPU untuk mempercepat proses komputasi. Dengan adanya dukungan GPU, LightGBM dapat menangani data dalam jumlah besar dengan lebih efisien, mempercepat pelatihan model secara signifikan.

Dalam pengembangan model LightGBM, interpretabilitas dan keterbukaan model merupakan aspek penting, terutama untuk memahami alasan di balik prediksi yang dihasilkan. Teknik interpretasi seperti *Permutation Feature Importance* (PFI) dan *Shapley additive explanations* (SHAP) menjadi metode yang sangat berguna untuk menjelaskan kontribusi setiap fitur dalam model terhadap prediksi akhir (Chaibi et al., 2021). PFI, misalnya, menilai pentingnya setiap fitur dengan mengevaluasi dampak perubahan nilai fitur terhadap akurasi model, sementara SHAP memberikan nilai yang menunjukkan pengaruh masing-masing fitur pada setiap prediksi. Dengan menggunakan teknik ini, pengguna dapat lebih memahami dan meningkatkan model yang mereka bangun.

Nilai SHAP, khususnya, dapat digunakan pada model LightGBM untuk memastikan interpretabilitas prediksi dengan tingkat keterbukaan yang lebih tinggi. Dengan mengaplikasikan nilai SHAP, model dapat meningkatkan performa inferensi serta mempercepat waktu pelatihan, terutama pada dataset yang kompleks. Selain itu, penggunaan SHAP dapat mengurangi kecenderungan model untuk “*fit-to-noise*” atau penyesuaian yang terlalu sensitif terhadap data acak, yang sering kali menjadi masalah dalam analisis data berukuran besar (Bugaj et al., 2021). Hal ini membuat SHAP menjadi alat interpretasi yang sangat efektif dalam membangun model LightGBM yang andal dan terbuka terhadap evaluasi.

Selain PFI dan SHAP, interpretabilitas dan keterbukaan dalam LightGBM dapat ditingkatkan melalui metode pembelajaran yang lebih adaptif seperti *personalized interpretability estimation* (ML-PIE). Dengan pendekatan ini, pengguna dapat mengarahkan proses sintesis model berdasarkan preferensi interpretabilitas yang dipersonalisasi, melalui algoritma evolusi *bi-objektif* yang mempertimbangkan interpretabilitas bersama dengan akurasi. Metode ML-PIE ini memungkinkan pengguna untuk menentukan prioritas interpretasi dalam pengembangan model, sehingga menghasilkan model LightGBM yang tidak hanya efisien tetapi juga mudah diinterpretasi, sesuai dengan kebutuhan spesifik dari pengguna atau lingkungan aplikasinya (Virgolin et al., 2021).

## **2.8 Brier Score**

Brier *Score* merupakan metrik evaluasi yang mengukur ketepatan dalam pemodelan prediksi, dengan cara membagi prediksi ke dalam beberapa kelompok

atau “bins” berdasarkan kesamaan nilai prediksi (Foster & Hart, 2022). Metrik ini memadukan skor kalibrasi dan skor penyempurnaan (*refinement*) untuk mengukur keahlian dalam pemodelan prediktif. Dengan menggabungkan aspek kalibrasi, yang menunjukkan seberapa baik prediksi sejalan dengan hasil aktual, dan aspek penyempurnaan, yang melihat kemampuan model dalam memisahkan atau membedakan hasil yang berbeda, *Brier Score* memberikan gambaran komprehensif mengenai performa model dalam memberikan prediksi probabilistik.

Penggunaan *Brier Score* dalam evaluasi model probabilitas penting karena metrik ini dapat mengukur kemampuan diskriminasi dan performa prediktif secara keseluruhan. Dengan kata lain, *Brier Score* tidak hanya melihat akurasi dari prediksi probabilitas tetapi juga sejauh mana model dapat membedakan antara kejadian yang mungkin terjadi dengan yang tidak (Dimitriadis et al., 2023). Hal ini membuat *Brier Score* menjadi pilihan yang baik untuk mengevaluasi performa model probabilistik, khususnya ketika diperlukan pemahaman yang lebih dalam mengenai kualitas prediksi yang bersifat probabilistik.

$$\mathbf{Brier\ Score} = (\mathbf{f_t} - \mathbf{o_t})^2 \quad (2.5)$$

*Brier Score* digunakan untuk menghitung selisih kuadrat antara nilai prediksi dan nilai aktual, sebagaimana terlihat pada Persamaan 2.5. Dalam konteks ini,  $\mathbf{f_t}$  merepresentasikan nilai probabilitas yang diprediksi untuk suatu peristiwa, sedangkan  $\mathbf{o_t}$  adalah nilai aktual dari peristiwa tersebut (biasanya 1 jika terjadi dan 0 jika tidak terjadi). *Brier Score* memiliki rentang nilai antara 0 hingga 1, di mana nilai yang lebih rendah menunjukkan prediksi yang lebih akurat, mendekati hasil aktual (BMJ Open, 2018).



*Brier Score* diperkenalkan oleh Glenn W. Brier pada tahun 1950 sebagai alat untuk menilai akurasi prediksi probabilitas (Foster & Hart, 2022). Skor ini menghitung selisih antara nilai prediksi dan realisasi aktual, di mana hasil perhitungan *Brier Score* memperlihatkan seberapa dekat prediksi tersebut dengan hasil aktual menggunakan formula *mean squared error* standar.

Sejak pertama kali diperkenalkan yaitu pada evaluasi ramalan cuaca, *Brier Score* telah berkembang menjadi metode yang diakui untuk mengukur akurasi model probabilitas dalam berbagai bidang, termasuk bisnis dan aplikasi lainnya (Petropoulos et al., 2022). Penerapan awalnya pada meteorologi menunjukkan bagaimana metode ini dapat memberikan wawasan yang lebih mendalam terhadap ketepatan perkiraan, yang kemudian menjadikan *Brier Score* sebagai standar dalam penilaian akurasi probabilitas di berbagai disiplin ilmu.

## **2.9 Receiver Operating Characteristic Area Under Curve (ROC AUC)**

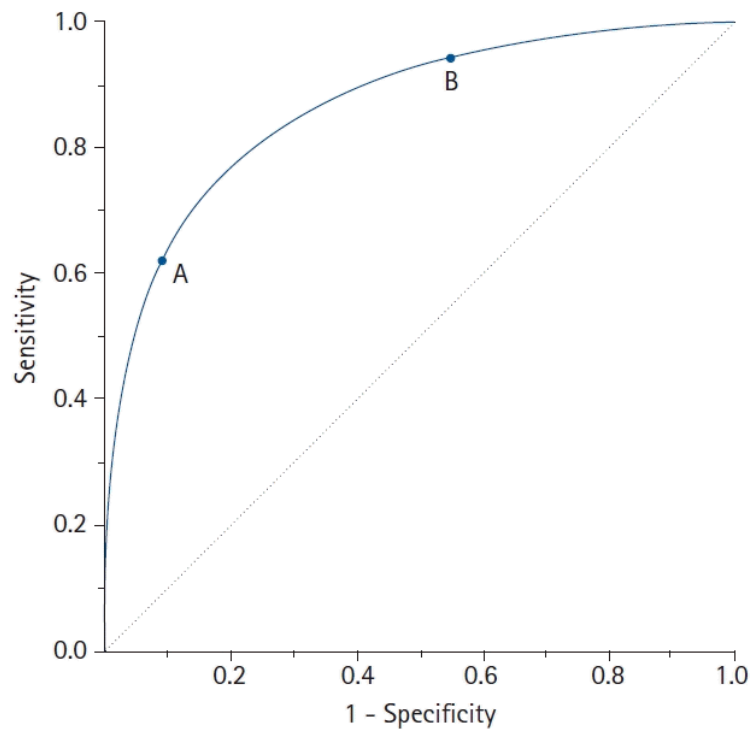
*Receiver Operating Characteristic* (ROC) adalah alat statistik yang digunakan untuk menilai kinerja model klasifikasi dengan menggambarkan hubungan antara dua parameter, yaitu *True Positive Rate* (TPR) dan *False Positive Rate* (FPR). Analisis ROC dapat dilakukan dengan memanfaatkan distribusi prior dan algoritma *elicitation* untuk memilih prior yang tepat, yang selanjutnya digunakan untuk menarik inferensi mengenai AUC (*Area Under the Curve*) dan karakteristik error model (Labadi et al., 2022).

ROC juga digunakan untuk mengevaluasi kinerja perangkat pengujian dan algoritma klasifikasi dalam menilai kepatuhan terhadap kriteria tertentu (Pendrill et

al., 2023). Dengan demikian, ROC menjadi alat yang penting untuk perbandingan dan evaluasi relatif dari berbagai sistem klasifikasi dalam konteks yang berbeda.

Kurva ROC menggambarkan kinerja model klasifikasi pada berbagai ambang batas klasifikasi dengan memplot dua parameter utama, yaitu TPR dan FPR. Salah satu kelemahan dari kurva ROC adalah kesulitan dalam menginterpretasi kinerja model jika terdapat banyak titik keputusan, karena setiap titik mewakili *trade-off* antara TPR dan FPR, yang dapat membuat sulit untuk menentukan titik terbaik yang mencerminkan kinerja keseluruhan model (Chen et al., 2023). AUC mengukur luas dua dimensi di bawah kurva ROC, dimulai dari titik (0,0) hingga (1,1). Semakin tinggi nilai AUC, semakin baik model dalam membedakan antara kelas positif dan negatif.

Pada Gambar 2.7, menampilkan kurva ROC AUC, di mana sumbu x menunjukkan nilai  $1 - \text{spesifisitas}$  (*False Positive Rate*) dan sumbu y menunjukkan sensitivitas pada semua nilai *cut-off* yang diukur dari hasil pengujian (Nahm, 2022). Ketika nilai *cut-off* yang lebih ketat diterapkan, titik pada kurva akan bergerak ke bawah dan ke kiri (Titik A). Sebaliknya, saat *cut-off* lebih longgar diterapkan, titik pada kurva bergerak ke atas dan ke kanan (Titik B). Garis diagonal  $45^\circ$  pada grafik ini berfungsi sebagai garis referensi, yang merepresentasikan kurva ROC dari klasifikasi acak.



Gambar 2.7 Contoh ROC AUC (Nahm, 2022)

ROC AUC memiliki peran penting dalam evaluasi model karena mampu mengukur kinerja model dalam berbagai kelompok risiko yang diprediksi (Carrington *et al.*, 2021). Ini memberikan informasi yang lebih mendalam yang dapat digunakan dalam pengambilan keputusan, memungkinkan pemahaman yang lebih komprehensif tentang bagaimana model berperforma di berbagai titik potong dan kelompok risiko.

Lebih lanjut, ROC AUC juga memungkinkan perbandingan yang wajar antar model dan membantu mengidentifikasi batas keputusan yang optimal serta potensi peningkatan AUC. Ini membuat AUC-ROC sangat bermanfaat dalam seleksi model yang lebih baik dan pemahaman tentang ruang yang dapat dioptimalkan untuk meningkatkan kinerja klasifikasi (Tafvizi *et al.*, 2022).

## **2.10 Knowledge Discovery in Databases (KDD)**

*Knowledge Discovery in Databases* atau KDD adalah proses yang bertujuan untuk mengekstraksi informasi yang dapat dipahami, menarik, dan bernilai dari data yang tidak terstruktur (Solanki & Sharma, 2021). Proses ini digunakan di berbagai bidang, seperti ilmu kehidupan, perdagangan, keuangan, dan kedokteran, untuk mengidentifikasi pola-pola yang tersembunyi dalam data yang besar dan kompleks (Solanki & Sharma, 2021). Proses ini mencakup berbagai teknik dan metode yang dapat digunakan untuk menggali wawasan dari data yang belum terorganisir.

KDD merupakan suatu bidang yang mengandalkan metode cerdas dalam data mining untuk menemukan pola-pola yang menjadi inti pengetahuan (Atloba, Balkir, & El-Mouadib, 2021). Pola-pola ini memungkinkan pengguna untuk memahami informasi yang terkandung dalam dataset besar, memberikan wawasan yang dapat diterapkan untuk pengambilan keputusan yang lebih baik dalam berbagai disiplin ilmu.

Proses KDD terdiri dari beberapa tahapan yang saling berinteraksi secara iteratif. Secara umum, tahapan tersebut mencakup (Chaudhary & Kishore, 2017):

- a. Pembersihan data (penghapusan data yang tidak konsisten),
- b. Integrasi data (penggabungan beberapa sumber data),
- c. Pemilihan data (pengambilan data yang relevan untuk tugas analisis),
- d. Transformasi data (pengolahan atau konsolidasi data untuk memudahkan mining),

- e. Data mining (aplikasi metode cerdas untuk mengekstraksi pola),
- f. Evaluasi pola (penilaian pola yang menarik berdasarkan ukuran keterminatan), dan
- g. Presentasi pengetahuan (penggunaan teknik visualisasi untuk menyajikan pengetahuan yang ditemukan kepada pengguna).

Data *mining* adalah bagian penting dalam proses KDD yang melibatkan berbagai alat dan teknik untuk menemukan pola yang berguna dalam basis data yang sangat besar (Chaudhary & Kishore, 2017). Salah satu bentuk terbaru dari data *mining* adalah ringkasan linguistik, yang bertujuan untuk memberikan deskripsi verbal yang dihasilkan oleh komputer mengenai pengetahuan yang tersembunyi dalam *database*, sering kali dalam bentuk aturan '*if-then*' yang menyerupai granula pengetahuan *fuzzy*. Selain itu, teknik *text mining* digunakan untuk mengekstraksi pola dari dokumen teks, yang melibatkan analisis teks untuk mengubah dokumen yang tidak terstruktur menjadi sekumpulan fitur yang sesuai, lalu menerapkan teknik data *mining* untuk ekstraksi pola (Chaudhary & Kishore, 2017).

Dalam KDD, *machine learning* berperan penting untuk menganalisis data, mengenali korelasi, dan memprediksi hasil yang akan terjadi (Kodati & Selvaraj, 2021). Teknik-teknik *machine learning* digunakan untuk melatih model dalam mengidentifikasi pola-pola yang ada dalam data, yang kemudian dapat digunakan untuk membuat prediksi yang lebih akurat dalam berbagai aplikasi, seperti analisis kesehatan atau analisis perilaku konsumen.

Aplikasi KDD sangat luas, salah satunya adalah dalam bidang kesehatan, di mana KDD digunakan untuk mengembangkan sistem medis yang dapat mendeteksi

dan memberikan saran pengobatan untuk penyakit dengan upaya minimal (Nwankwo, Ngene, & Onuora, 2023). Selain itu, KDD berbasis metode *gradient boosting machine* juga diterapkan dalam prediksi energi listrik, memberikan referensi praktis bagi aplikasi KDD pada sektor energi lainnya (Xie et al., 2022).

KDD juga memiliki keterkaitan yang erat dengan analisis olahraga, khususnya sepak bola, di mana pendekatan KDD yang komprehensif memungkinkan persiapan data yang tepat untuk prediksi hasil pertandingan olahraga, termasuk hasil pertandingan sepak bola (Głowania, Kozak, & Juszczuk, 2023). Dengan menggunakan teknik KDD, analisis yang lebih mendalam dapat dilakukan terhadap data pertandingan untuk mengidentifikasi faktor-faktor yang mempengaruhi hasil akhir pertandingan.

## **2.11 Python**

Python adalah bahasa pemrograman tingkat tinggi bersifat object-oriented, dikembangkan oleh Guido van Rossum, bahasa ini dirancang untuk menjadi mudah dipahami dan digunakan sehingga cocok baik untuk pemula yang sedang mempelajari dasar-dasar pemrograman maupun untuk para profesional yang mengerjakan proyek pemrograman di dunia nyata (Srinath, 2017). Python menawarkan sintaks yang sederhana dan intuitif, sehingga memungkinkan pengguna menulis kode dengan lebih cepat dan efisien. Selain itu, Python memiliki dukungan pustaka yang sangat luas serta komunitas yang aktif, menjadikannya pilihan populer untuk berbagai kebutuhan, mulai dari pengembangan web, analisis data, *machine learning*, hingga komputasi ilmiah dan otomatisasi sistem.

Python menawarkan keseimbangan antara kejelasan sintaks dan fleksibilitas dalam pengembangan alat-alat penelitian komputasi, sehingga sangat mendukung dalam menciptakan solusi untuk berbagai jenis permasalahan yang kompleks. Bahasa ini dirancang untuk menangani beragam tantangan yang melibatkan pengolahan dataset berukuran besar, penerapan algoritma yang rumit, serta pengembangan sistem komputasi (Pérez, Granger & Hunter, 2011). Kemampuan Python untuk berintegrasi dengan berbagai pustaka dan *framework* membuatnya menjadi pilihan utama dalam penelitian berbasis data dan pengembangan teknologi inovatif. Dengan ekosistem yang luas, Python memungkinkan peneliti dan pengembang untuk membangun, menguji, serta mengimplementasikan solusi secara efisien dan *scalable*.

## **2.12 Pandas**

Pandas adalah pustaka Python berperforma tinggi yang dirancang khusus untuk manipulasi, analisis, dan eksplorasi data. Pustaka ini banyak digunakan oleh peneliti data, analis, dan pengembang karena kemampuannya yang unggul dalam mengolah data secara efisien (Molin & Jee, 2021). Pandas menyediakan berbagai fungsi yang memudahkan proses pembersihan, transformasi, serta analisis data dalam berbagai format, seperti tabel, *file* CSV, dan *database*. Selain itu, Pandas juga mendukung integrasi dengan pustaka visualisasi seperti Matplotlib dan Seaborn, sehingga memungkinkan pengguna untuk membuat visualisasi data yang informatif dan menarik. Kemudahan penggunaan serta fleksibilitas Pandas menjadikannya

salah satu alat utama dalam analisis data modern dan pengembangan aplikasi berbasis data.

Salah satu kekuatan utama dari pustaka ini adalah penggunaan data *frame* dan *series*, yang menjadi inti dalam proses manipulasi, perhitungan, serta analisis data (Nelli, 2015). Data *frame* adalah struktur data berbentuk tabel dengan label pada baris dan kolom, mirip dengan tabel pada *database* atau *spreadsheet*, sehingga memudahkan pengolahan data dalam jumlah besar. Sementara itu, *series* merupakan struktur data satu dimensi yang berfungsi seperti *array*, tetapi dilengkapi dengan indeks yang memungkinkan akses data lebih fleksibel. Kombinasi dari dua struktur data ini memungkinkan pengguna untuk melakukan berbagai operasi analisis secara efisien, seperti pengolahan data numerik, transformasi data, serta agregasi hasil analisis dengan sintaks yang sederhana namun *powerful*.

### **2.13 Scikit-learn**

Scikit-learn merupakan pustaka Python yang menyediakan antarmuka standar untuk mengimplementasikan berbagai algoritma machine learning. Pustaka ini dirancang agar mudah digunakan, sehingga memudahkan pengguna dari berbagai latar belakang untuk mengembangkan model machine learning dengan lebih efisien. Selain mendukung algoritma untuk klasifikasi, regresi, dan clustering, Scikit-learn juga dilengkapi dengan berbagai fungsi penting lainnya, seperti data preprocessing, resampling, evaluasi model, serta pencarian hyperparameter. Fungsi-fungsi tersebut membantu memastikan bahwa proses pengolahan data,



pelatihan model, hingga evaluasi dapat dilakukan secara menyeluruh dan sistematis (Bisong, 2019).

## **2.14 Matplotlib**

Matplotlib adalah pustaka Python yang digunakan untuk pembuatan grafik dan visualisasi data. Pustaka ini menyediakan berbagai fitur yang memungkinkan pengguna untuk membuat beragam jenis grafik dan diagram, mulai dari grafik garis (*line plot*), grafik sebar (*scatter plot*), peta panas (*heatmap*), diagram batang (*bar chart*), diagram lingkaran (*pie chart*), hingga visualisasi data dalam bentuk tiga dimensi (3D plot) (Hunt, 2019). Kemampuan Matplotlib dalam menghasilkan visualisasi yang informatif dan berkualitas tinggi menjadikannya salah satu alat utama bagi peneliti dan analis data. Selain itu, pustaka ini mendukung kustomisasi penuh pada setiap elemen grafik, seperti warna, label, dan sumbu, sehingga memudahkan pengguna untuk menyajikan data secara lebih menarik dan sesuai dengan kebutuhan analisis.

## **2.15 Seaborn**

Seaborn adalah pustaka Python yang dirancang untuk membuat visualisasi grafik statistik dengan cara yang lebih mudah dan estetik. Pustaka ini menyediakan antarmuka tingkat tinggi untuk Matplotlib, sehingga memungkinkan pengguna membuat grafik kompleks dengan sedikit kode (Waskom, 2021). Seaborn juga terintegrasi erat dengan Pandas, sehingga pengguna dapat langsung memvisualisasikan data dari struktur data frame tanpa perlu konversi tambahan.

Dengan berbagai fitur bawaan, seperti pembuatan grafik hubungan antar variabel, distribusi data, serta anotasi statistik, Seaborn membantu dalam menyajikan visualisasi data yang informatif dan menarik. Kemudahan penggunaan serta desain visual yang lebih elegan membuat Seaborn menjadi pilihan utama bagi analis data dan ilmuwan data yang ingin meningkatkan kualitas visualisasi mereka.

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Pendekatan Penelitian

Penelitian ini melihat kinerja metode LGBM dalam memprediksi nilai xG dari data tembakan pada pertandingan sepak bola dengan menggunakan pendekatan kuantitatif. Penelitian ini menggunakan bahasa pemrograman Python dan platform Google Colaboratory untuk proses pengambilan, pembersihan, dan pemodelan data. Dataset diambil dari repositori terbuka StatsBomb yang tersedia di GitHub. Microsoft Word digunakan untuk penyusunan laporan penelitian, sedangkan Mendeley dimanfaatkan untuk manajemen referensi dan pembuatan daftar pustaka.

#### 3.2 Tempat dan Waktu Penelitian

##### 3.2.1 Tempat Penelitian

Penelitian ini dilakukan menggunakan *open-data* dari StatsBomb melalui repositori GitHub dengan proses analisis yang dilakukan menggunakan Python dan platform Google Colaboratory.

##### 3.2.2 Waktu Penelitian

Rencana waktu pelaksanaan penelitian ditunjukkan pada Tabel 3.1.

Tabel 3.1 Waktu Pelaksanaan Penelitian

No.	Tahapan	Februari 2025	Maret 2025	April 2025	Mei 2025	Juni 2025	Juli 2025
1	Landasan Teori						

2	Pengumpulan Data						
3	Analisis Data						
4	Interpretasi						
5	Pembuatan Laporan						

### 3.3 Metodologi Pengumpulan Data

#### 3.3.1 Studi Literatur

Data yang digunakan dalam penelitian ini terdiri atas data primer dan sekunder. Data primer diperoleh dari dataset terbuka yang disediakan oleh StatsBomb melalui repositori GitHub. Sementara itu, data sekunder diperoleh dari berbagai jurnal ilmiah, buku, dan sumber internet yang relevan dengan topik penelitian, khususnya yang berkaitan dengan analisis xG, pemodelan prediktif, dan algoritma LightGBM.

#### 3.3.2 Pengambilan Data

Pengambilan data dilakukan dengan mengunduh dataset event pertandingan sepak bola dari repositori open-source StatsBomb di GitHub. Proses ini dilakukan menggunakan skrip Python di platform Google Colaboratory. Dataset yang digunakan mencakup data tembakan dalam pertandingan, termasuk informasi seperti lokasi, jarak, sudut tembakan, serta atribut kontekstual lainnya yang mendukung perhitungan nilai xG.

### 3.4 Pengembangan Model

#### 3.4.1 Metode KDD

Penelitian ini menggunakan pendekatan Knowledge Discovery in Databases (KDD) dalam proses pengembangan model. Metode KDD memiliki keunggulan dalam membantu mengidentifikasi pola tersembunyi dari kumpulan data yang kompleks sehingga dapat menghasilkan informasi yang lebih mudah dipahami. Proses KDD terdiri dari beberapa tahapan, yaitu: *preprocessing* data, pemilihan data (*data selection*), transformasi data, proses data *mining*, dan evaluasi pengetahuan yang diperoleh (*knowledge evaluation*) (Ramos et al., 2021).

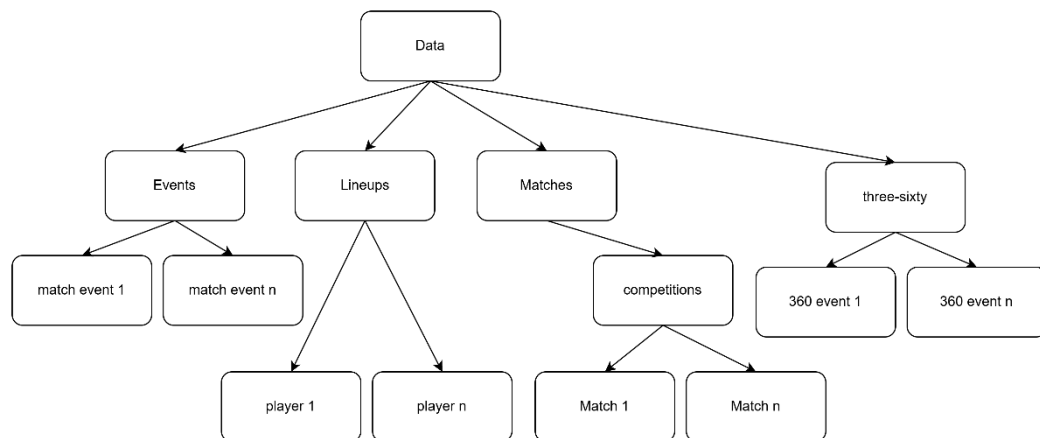
a. *Data Selection*

Data dari StatsBomb *open-data* diambil dengan mengakses repositori resmi di GitHub. Pertama, kita perlu mengidentifikasi kompetisi apa saja yang tersedia dalam dataset. Setiap kompetisi kemudian terdiri dari beberapa musim (edisi), dan masing-masing musim ini mewakili rentang waktu berlangsungnya pertandingan yang terdokumentasi. Di dalam setiap musim terdapat fase-fase pertandingan: untuk kompetisi sistem gugur biasanya meliputi babak perempat final, semi final, final, dan seterusnya, sedangkan untuk liga reguler umumnya hanya ada satu fase liga utama, dengan beberapa kompetisi seperti, Piala FA yang juga memiliki babak *play-off*. Setelah fase-fase ditentukan, barulah kita mengakses data pertandingan. Dalam konteks StatsBomb, satu pertandingan terdiri dari serangkaian *event*, dan masing-masing *event* ini dapat memiliki *event* terkait.

Misalnya, sebuah tusukan (*dribble*) bisa jadi dipicu oleh operan rekan tim yang sebelumnya dieksekusi operan tersebut, kemudian tercatat sebagai *event* terkait. Namun, karena operan juga tercatat sebagai *event* utama, jika kita menarik

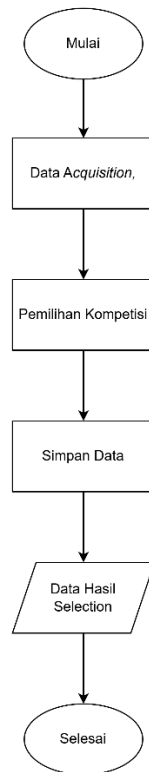
semua *event* terkait tanpa seleksi, kita akan mendapati banyak duplikasi operan tercatat dua kali, sekali sebagai *event* utama dan sekali lagi sebagai *event* terkait. Sebaliknya, jika kita sama sekali mengabaikan *event* terkait, kita bisa kehilangan jejak kronologi aksi yang sebenarnya terjadi di lapangan.

Untuk mengatasi masalah ini, saat ini hanya situasi gol dan kartu (kuning/merah) yang diikuti sebagai *event* terkait dalam pemrosesan data StatsBomb. Dengan begitu, kita tetap menjaga konteks penting seperti *assist* sebelum gol atau pelanggaran yang berujung kartu tanpa menumpuk terlalu banyak duplikasi. Pada Gambar 3.1 dijelaskan struktur data yang dimiliki oleh StatsBomb *open-data*.



Gambar 3.1 Struktur Data StatsBomb *open-data*.

Data *event* dari StatsBomb disediakan dalam format JSON pada repositori GitHub mereka. Karena pengambilan data langsung dari GitHub juga memakan waktu, biasanya file-file JSON tersebut diunduh sekali saja lalu dikonversi dan disimpan dalam format *Parquet* untuk penggunaan selanjutnya. Dengan cara ini, analisis bisa dilakukan lebih cepat tanpa perlu terus-menerus mengunduh data mentah. Gambar 3.2 menunjukkan *flowchart* dari *data selection*.



Gambar 3.2 *Flowchart Selection*

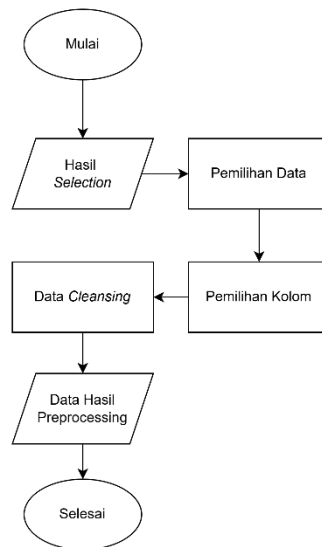
b. *Data Preprocessing*

Tahap data *preprocessing* bertujuan untuk menyiapkan data hasil seleksi agar dapat dianalisis secara optimal dan digunakan dalam proses pelatihan model. Proses ini disesuaikan dengan karakteristik data *event* sepak bola yang diperoleh dari StatsBomb, yang memiliki struktur sangat baik dan konsisten sehingga mempermudah proses pembersihan dan pengolahan data.

Langkah pertama adalah pemilihan data, yaitu dengan mengambil hanya *event* yang bertipe *Shot* dan berasal dari situasi permainan terbuka (*open play*), karena jenis tembakan ini paling relevan dalam konteks prediksi *expected goals*. Setelah itu, dilakukan pemilihan kolom dengan memilih fitur-fitur yang berpotensi mendukung prediksi, seperti posisi tembakan, bagian tubuh yang

digunakan, tekanan lawan, serta pola permainan. Kolom-kolom yang bersifat administratif atau tidak relevan terhadap tujuan model, seperti nama pemain dan identifikasi pertandingan, tidak disertakan.

Langkah terakhir adalah data *cleansing*, yang meliputi pemeriksaan nilai kosong dan duplikat. Namun, karena data StatsBomb memiliki format yang sangat terstruktur dan tiap peristiwa dalam pertandingan bersifat unik, data yang diperoleh relatif bersih dan tidak memerlukan proses pembersihan lanjutan. Tahapan *preprocessing* ini menghasilkan *dataset* yang konsisten, bebas duplikasi, dan siap untuk dianalisis lebih lanjut pada tahap transformasi dan pemodelan. Pada Gambar 3.2 dijelaskan *flowchart* dari tahap *preprocessing*.



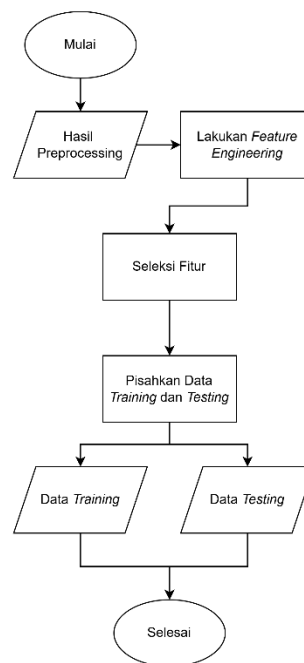
Gambar 3.3 *Flowchart Preprocessing*

### c. *Data Transformation*

Tahap ini bertujuan untuk memperkaya representasi data agar dapat meningkatkan performa model pada tahapan data mining. Pertama, dilakukan proses feature engineering untuk menciptakan fitur-fitur baru yang merepresentasikan dinamika



permainan secara lebih mendalam. Transformasi ini memungkinkan data mentah memberikan wawasan yang lebih bermakna dan relevan dalam konteks prediksi performa tembakan. Fitur-fitur seperti jarak dan sudut tembakan ke gawang serta segmentasi waktu pertandingan ditambahkan untuk memperkaya informasi spasial dan temporal. Setelah fitur baru ditambahkan, data kemudian dibagi menjadi data latih dan data uji agar proses pelatihan dan evaluasi model dapat dilakukan secara terpisah. Alur tahapan transformation ditunjukkan pada Gambar 3.3.

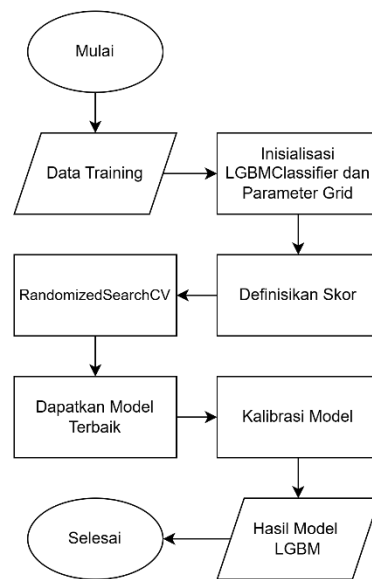


Gambar 3.4 *Flowchart Transformation*

#### d. *Data Mining*

Pada tahapan ini pemodelan xG dilakukan menggunakan algoritma LightGBM. Namun sebelum model dilatih, terdapat beberapa proses penting yang harus dilakukan, yaitu pencarian *hyperparameter* terbaik dan kalibrasi probabilitas. Pencarian *hyperparameter* dilakukan dengan menggunakan *RandomizedSearchCV* sebanyak 100 iterasi, yang mengevaluasi berbagai

kombinasi parameter dengan *5-fold cross-validation*. Proses ini menggunakan metrik skor *roc\_auc* sebagai acuan untuk menentukan kombinasi parameter terbaik dan secara otomatis melakukan *refit* pada model dengan skor tersebut. Setelah memperoleh model dengan konfigurasi terbaik, dilakukan kalibrasi probabilitas menggunakan *CalibratedClassifierCV* untuk memastikan bahwa prediksi probabilitas dari model merefleksikan tingkat kepercayaannya secara akurat (Davis & Robberechts, 2024). Selain pelatihan dan kalibrasi, tahap ini juga mencakup analisis fitur untuk memahami kontribusi tiap variabel dalam proses prediksi. Gambar 3.4 menunjukkan alur dari tahapan data *mining* dalam penelitian ini.

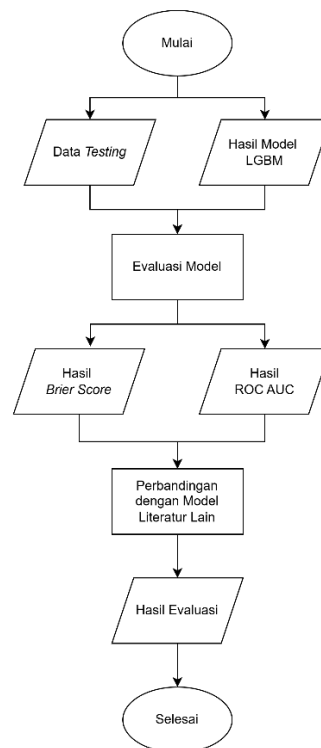


Gambar 3.5 *Flowchart Data Mining*

e. *Evaluation*

Setelah proses data *mining* selesai, tahap selanjutnya adalah evaluasi terhadap model yang telah dibuat. Evaluasi dilakukan dengan mengukur sejauh mana model mampu memberikan prediksi yang akurat dan probabilitas yang baik

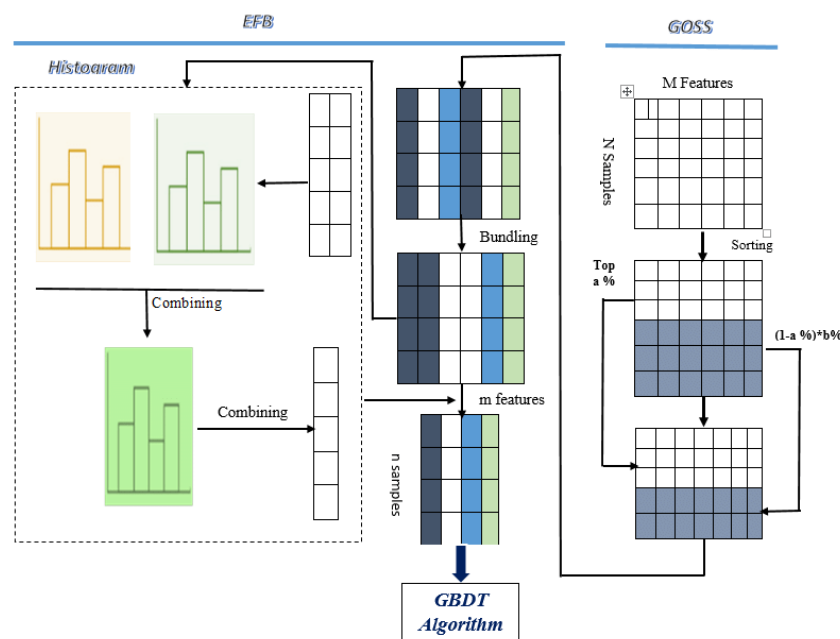
terhadap data uji. Dua metrik utama yang digunakan adalah *Brier Score* dan ROC AUC. *Brier Score* digunakan untuk menghitung kesalahan prediksi probabilistik model terhadap kelas sebenarnya, yang secara konsep mirip dengan *mean squared error* dan memberikan gambaran seberapa baik kalibrasi model dalam memprediksi probabilitas kelas (Eggels *et al.*, 2016). Sementara itu, ROC AUC digunakan untuk mengevaluasi kemampuan model dalam membedakan antara kelas positif dan negatif, dengan menilai seberapa besar kemungkinan model memberikan skor lebih tinggi untuk contoh positif dibandingkan negatif, tanpa memperhatikan nilai probabilitas aktualnya (Decroos & Davis, 2019). *Flowchart* dari tahapan evaluasi model ditunjukkan pada Gambar 3.5.



Gambar 3.6 *Flowchart Evaluation*

### 3.4.2 Permodelan LightGBM

Pada penelitian ini, metode yang digunakan adalah LightGBM (Light Gradient Boosting Machine) untuk membangun model prediksi. LightGBM dirancang untuk menangani data berukuran besar dengan efisiensi tinggi melalui dua teknik utama: *Gradient-based One-Side Sampling* (GOSS) dan *Exclusive Feature Bundling* (EFB) seperti ditunjukkan pada Gambar 3.6.



Gambar 3.7 GOSS dengan EFB (Mohammed *et al.*, 2021)

Teknik GOSS berfokus pada efisiensi pelatihan model dengan mempertahankan seluruh data yang memiliki nilai gradien besar yang mengandung lebih banyak informasi dan secara acak mengambil sebagian dari data dengan gradien kecil (Ke et al., 2017). Namun, karena proses ini dapat mengubah distribusi data asli, LightGBM memperkenalkan pengali konstan saat menghitung *information gain* untuk data dengan gradien kecil guna menyeimbangkan kontribusi antara dua kelompok data tersebut. Pendekatan ini memungkinkan model untuk

tetap fokus pada sampel yang paling berpengaruh terhadap pembaruan model tanpa kehilangan akurasi secara signifikan.

Sementara itu, teknik EFB dirancang untuk mengatasi tantangan ketika terdapat banyak fitur yang bersifat saling eksklusif, yaitu fitur-fitur yang tidak pernah aktif secara bersamaan. Algoritma ini menggabungkan fitur-fitur eksklusif tersebut ke dalam fitur padat (*dense feature*) dalam jumlah yang jauh lebih sedikit, sehingga mengurangi dimensi data dan beban komputasi (Ke et al., 2017). Selain itu, LightGBM juga mengoptimalkan algoritma histogram dasar dengan cara mengabaikan nilai nol pada fitur, yakni dengan mencatat hanya nilai-nilai non-nol menggunakan struktur data khusus. Kombinasi dari GOSS dan EFB menjadikan LightGBM sangat efisien dan *scalable* dalam membangun model prediksi dari dataset dengan jumlah *instance* dan fitur yang sangat besar.

### **3.5 Analisis Data dan Interpretasi Hasil**

Analisis data dalam penelitian ini dilakukan berdasarkan pendekatan KDD (*Knowledge Discovery in Database*) yang mencakup lima tahapan utama: *data selection*, *preprocessing*, *transformation*, *data mining*, dan *evaluation*. Proses analisis dimulai dari tahap *data selection*, yaitu dengan menyiapkan dataset yang relevan untuk membangun model prediksi. Tahap selanjutnya adalah *preprocessing* yang meliputi pembersihan data, penanganan *missing value*, penghapusan duplikasi.

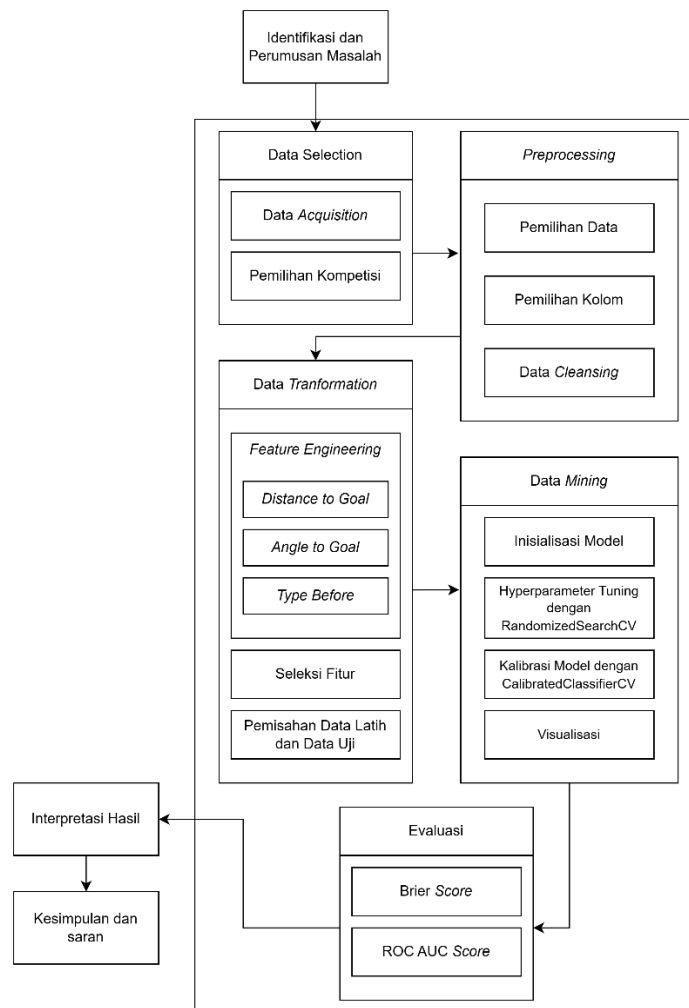
Pada tahap *transformation*, dilakukan pembagian data menjadi data latih dan data uji, serta dilakukan transformasi fitur agar sesuai dengan kebutuhan

algoritma yang digunakan. Tahap data *mining* dilakukan dengan membangun model prediksi menggunakan algoritma LightGBM, serta melakukan *hyperparameter tuning* menggunakan *RandomizedSearchCV* untuk memperoleh kombinasi parameter terbaik berdasarkan nilai skor ROC AUC.

Kemudian, pada tahap *evaluation*, performa model diukur menggunakan dua metrik utama, yaitu ROC AUC dan *Brier Score*. ROC AUC digunakan untuk mengevaluasi kemampuan model dalam membedakan antara kelas positif dan negatif, sementara *Brier Score* digunakan untuk mengukur akurasi probabilistik dari prediksi model. Hasil dari seluruh tahapan analisis ini serta interpretasi terhadap performa model akan dijelaskan secara rinci pada Bab 4.

### **3.6 Tahapan Penelitian**

Tahapan penelitian yang dilakukan dapat dilihat pada Gambar 3.7.



Gambar 3.8 Tahapan Penelitian

Tahapan pertama dalam penelitian ini adalah identifikasi dan perumusan masalah yang dilakukan dengan mengkaji literatur yang relevan untuk memahami permasalahan aktual dan peluang riset di bidang analisis pertandingan sepak bola menggunakan teknik pembelajaran mesin. Literatur yang digunakan berasal dari jurnal ilmiah, artikel konferensi, serta laporan penelitian terkini. Setelah permasalahan dirumuskan, penelitian dilanjutkan dengan mengikuti alur metode KDD yang terdiri dari tahapan *data selection*, *preprocessing*, *transformation*, *data mining*, dan *evaluation*.

Pada tahap data *selection*, data dikumpulkan dengan mengunduh dataset publik dari GitHub yang berisi catatan pertandingan sepak bola, lalu difokuskan pada data *event* yang relevan untuk keperluan prediksi. Selanjutnya, dilakukan *preprocessing* berupa pemilihan variabel yang akan digunakan, pembersihan data dari nilai yang hilang dan duplikat, proses *encoding* untuk fitur kategorial, serta normalisasi data numerik.

Tahap berikutnya adalah *transformation* yang mencakup rekayasa fitur (*feature engineering*), analisis korelasi antar variabel, seleksi fitur penting, dan pembagian data menjadi set pelatihan dan pengujian. Pada tahap data *mining*, digunakan algoritma LightGBM dengan proses *tuning hyperparameter* melalui *RandomizedSearchCV*, serta dilanjutkan dengan proses kalibrasi model agar hasil prediksi probabilistik menjadi lebih akurat.

Tahap terakhir adalah *evaluation*, yaitu evaluasi performa model menggunakan metrik ROC AUC untuk mengukur kemampuan klasifikasi dan *Brier Score* untuk menilai ketepatan probabilitas prediksi model. Setelah seluruh tahapan metode KDD selesai, penelitian diakhiri dengan analisis hasil dan penarikan kesimpulan serta saran untuk penelitian selanjutnya.

### **3.7 Perangkat Penelitian**

Penelitian ini menggunakan perangkat keras (*hardware*) dan perangkat lunak (*software*) dengan spesifikasi yang dijelaskan pada Tabel 3.2.



Tabel 3.2 Spesifikasi Hardware dan Software

<i>Hardware</i>	Laptop Lenovo ADA 11	AMD Athlon Gold 3150U with Radeon Graphics 2.40 GHz
		12.0 GB
		256 GB SSD
		Monitor 15 Inch
<i>Software</i>	Sistem Operasi	Windows 11 Home
	<i>Tools</i>	Google Colaboratory
	Bahasa Pemrograman	Python

## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **4.1     *Data Selection***

Tahap data *selection* merupakan langkah awal dalam proses analisis di mana data yang relevan dipilih dari sumber yang tersedia untuk digunakan dalam penelitian. Pada penelitian ini, data diperoleh dari repositori terbuka StatsBomb melalui GitHub, yang menyediakan data *event* pertandingan sepak bola dalam format JSON. Dari seluruh data yang tersedia, hanya data dengan tipe *event Shot* yang dipilih karena fokus penelitian adalah memprediksi kemungkinan terciptanya gol dari sebuah tembakan. Selain itu, hanya kolom-kolom tertentu yang dipilih, seperti informasi tentang teknik tembakan, bagian tubuh yang digunakan, pola permainan, serta posisi awal tembakan, karena kolom-kolom tersebut dianggap memiliki relevansi langsung terhadap peluang mencetak gol. Proses ini bertujuan untuk menyederhanakan dataset dan memastikan bahwa hanya fitur yang bermakna yang digunakan dalam tahap analisis selanjutnya.

##### **4.1.1   *Data Acquisition***

Penelitian ini menggunakan data dari repositori GitHub StatsBomb open-data yang diambil melalui proses unduhan menggunakan *tool* aria2 di Google Colab dengan bahasa pemrograman Python. Untuk mengakses data tersebut, pertama-tama dilakukan pengunduhan *file master.zip* yang berisi data pertandingan sepak bola. Berikut adalah tahapan yang dilakukan dalam proses pengumpulan data:

- a. Mengunduh *file* master.zip dari repositori GitHub StatsBomb open-data dengan menggunakan perintah di Google Colab dan *tool* aria2 untuk mempercepat proses unduhan.
- b. Menyusun skrip Python di Google Colab untuk mengekstrak semua *event* data yang ada pada file yang diunduh.
- c. Mengkonversi *event* data menjadi format *dataframe* menggunakan *pandas* untuk mempermudah pengolahan data lebih lanjut.
- d. Menyimpan *dataframe* yang telah diproses dalam format *parquet* untuk memudahkan analisis data selanjutnya. *Dataframe* yang dihasilkan mencakup berbagai kolom terkait informasi pertandingan, yang akan diseleksi dan diproses lebih lanjut untuk analisis yang lebih mendalam.

#### 4.1.2 Pemilihan Kompetisi

Langkah ini bertujuan untuk memastikan bahwa hanya data pertandingan yang relevan dan sesuai dengan fokus penelitian yang digunakan dalam proses analisis. Data mentah yang tersedia di repositori open-data StatsBomb mencakup berbagai jenis kompetisi, termasuk pertandingan pria, wanita, dan kelompok usia muda. Oleh karena itu, proses seleksi dilakukan secara sistematis untuk menyaring data berdasarkan dua kriteria utama: jenis kelamin peserta dan tingkat kompetisi.

Data kompetisi difilter untuk hanya menyertakan pertandingan pria dengan memeriksa atribut *competition\_gender* yang bernilai '*male*'. Selanjutnya, untuk memastikan bahwa hanya kompetisi tingkat senior yang disertakan, dilakukan pengecualian terhadap kompetisi yang mengandung kata kunci seperti 'U21', 'U23', 'U18', dan lainnya dalam nama kompetisi, yang menunjukkan kelompok usia muda.

Setelah mendapatkan daftar kompetisi yang valid, data pertandingan (*matches*) dari kompetisi tersebut dimuat dan difilter lebih lanjut untuk hanya menyertakan pertandingan antara dua tim pria. Proses ini menghasilkan kumpulan data pertandingan yang sesuai dengan fokus penelitian, yaitu analisis pertandingan sepak bola pria tingkat senior. Tabel 4.1 Menunjukkan Daftar Kompetisi yang akan digunakan.

Tabel 4.1 Daftar Kompetisi

<i>Competition ID</i>	<i>Season ID</i>	<i>Competition Name</i>
11	4	FIFA World Cup
2	44	Premier League
37	90	La Liga
72	30	UEFA Champions League
43	106	Bundesliga
49	3	Serie A
4	1	Ligue 1
55	27	Copa America
9	42	African Cup of Nations
16	1	Eredivisie

## 4.2 Data Preprocessing

Tahap *preprocessing* adalah tahapan yang berisi serangkaian proses untuk membersihkan dan menyiapkan data agar siap digunakan dalam analisis dan pemodelan pada tahapan selanjutnya. Dengan *preprocessing* yang tepat, kualitas data meningkat dan hasil pemodelan di tahap berikutnya menjadi lebih akurat dan andal.

#### 4.2.1 Pemilihan Data

Pemilihan jenis *event* yang tepat sangat krusial untuk memastikan relevansi dan kualitas analisis. Berdasarkan dokumentasi resmi dari StatsBomb, setiap peristiwa dalam pertandingan dikategorikan dengan identifier unik. *Event* dengan *type.id* 16 merepresentasikan aksi "*Shot*" atau tembakan, yang menjadi fokus utama dalam model xG karena langsung berkaitan dengan upaya mencetak gol.

Setelah data kompetisi kita dapatkan, maka selanjutnya kita akan mengambil data *event* yang sesuai dengan kompetisi yang kita dapatkan dengan mendapatkan *match\_id* yang akan digunakan nantinya. Kemudian kita akan menyimpan semua *event* yang sesuai dengan *match\_id* pada *file parquet*. Selanjutnya kita dapat memilih *event shot* yang ada pada setiap pertandingan untuk dijadikan *dataset* untuk pelatihan dan pengujian model nantinya. Hasil *dataframe* ditunjukkan pada Gambar 4.1.

period	minute	second	start_x	start_y	team_name	player_name	end_x	end_y	type	...
1	7	15	115.4	29.4	England	Harry Maguire	120.0	34.9	16	...
1	26	58	101.1	55.3	England	Bukayo Saka	117.5	41.9	16	...
1	29	8	113.4	49.1	England	Mason Mount	120.0	45.0	16	...
1	31	47	110.5	40.7	England	Harry Maguire	120.0	36.8	16	...
1	34	9	112.0	38.0	England	Jude Bellingham	120.0	43.0	16	...

Gambar 4.1 Hasil Pemilihan Data

#### 4.2.2 Pemilihan Kolom

Pada tahapan ini, proses awal yang dilakukan adalah memilih sub set data yang sesuai dengan tujuan penelitian. Dalam konteks prediksi xG, hanya data

dengan tipe *event Shot* yang diambil karena data tersebut merepresentasikan momen-momen tembakan yang menjadi fokus analisis. Setelah *event Shot* teridentifikasi, dilakukan proses seleksi kolom atau fitur yang dianggap memiliki nilai prediktif terhadap hasil tembakan (*shot outcome*). Fitur-fitur yang dipilih mencakup atribut spasial (seperti posisi awal dan akhir tembakan), temporal (menit dan detik), teknis (teknik tembakan, bagian tubuh yang digunakan), serta konteks permainan (tekanan lawan, pola permainan, dan tipe *event* sebelumnya). Fitur-fitur ini ditujukan untuk menangkap berbagai aspek yang dapat memengaruhi kemungkinan sebuah tembakan menjadi gol. Hasil seleksi kolom ditampilkan pada Tabel 4.2.

Tabel 4.2 Nama dan Deskripsi Kolom

Nama Kolom	Deskripsi
<i>period</i>	Periode pertandingan saat tembakan terjadi
<i>minute</i>	Menit pertandingan saat tembakan dilakukan
<i>second</i>	Detik pertandingan saat tembakan dilakukan
<i>start_x, start_y</i>	Koordinat awal tembakan (lokasi pemain saat menembak)
<i>position</i>	Posisi pemain di dalam tim (Bek, Gelandang, Penyerang)
<i>shot_outcome</i>	Hasil tembakan (0 = tidak gol, 1 = gol)
<i>shot_body_part</i>	Bagian tubuh yang digunakan dalam menembak
<i>shot_first_time</i>	Apakah tembakan dilakukan secara langsung tanpa kontrol bola
<i>shot_one_on_one</i>	Apakah tembakan dilakukan dalam situasi satu lawan satu dengan kiper
<i>shot_open_goal</i>	Apakah tembakan dilakukan ke gawang yang kosong
<i>shot_aerial_won</i>	Apakah pemain memenangkan duel udara sebelum tembakan
<i>shot_key_pass</i>	Apakah tembakan didahului oleh umpan kunci
<i>possession</i>	Nomor penguasaan bola dari tim

<i>play_pattern</i>	Pola permainan yang terjadi sebelum tembakan
<i>under_pressure</i>	Apakah pemain berada dalam tekanan saat melakukan tembakan
<i>shot_technique</i>	Teknik tembakan yang digunakan

Pemilihan variabel ini bertujuan untuk menyederhanakan kompleksitas data serta meningkatkan fokus pada fitur-fitur yang relevan dalam konteks perhitungan xG. Langkah ini dilakukan untuk mengurangi redundansi informasi dan meminimalkan risiko *overfitting* akibat penggunaan variabel yang tidak informatif. Selain itu, untuk fitur-fitur yang bersifat kategorial, dilakukan pendekatan dengan mengambil langsung nilai ID atau representasi numerik yang sudah tersedia dari masing-masing kategori. Dengan demikian, proses ini menghindari kebutuhan akan transformasi tambahan seperti *one-hot encoding* atau *label encoding*, yang dapat menambah dimensi data secara signifikan tanpa memberikan kontribusi informatif yang sepadan. Pendekatan ini tidak hanya menjaga efisiensi pemrosesan data, tetapi juga mempertahankan struktur semantik dari variabel kategorial dalam bentuk yang lebih ringkas dan langsung digunakan oleh model. Gambar 4.2 menunjukkan contoh data setelah proses pemilihan variabel dilakukan.

	period	minute	second	location_x	...	technique
0	1	1	42	111.0	...	93
1	1	4	47	96.0	...	93
2	1	8	37	107.0	...	93
3	1	17	26	111.0	...	93
4	1	21	16	105.0	...	93
...	...	...	...	...	...	...
68863	2	65	39	106.1	...	93
68864	2	69	0	114.9	...	93
68865	2	82	41	103.6	...	93
68866	2	85	10	108.5	...	93
68867	2	85	58	117.0	...	91

Gambar 4.2 Contoh Data Sesudah Pemilihan Variabel

### 4.2.3 Data Cleansing

Tahap data *cleansing* dilakukan untuk memeriksa kelengkapan dan keunikan data dengan tujuan memastikan bahwa tidak terdapat nilai kosong (*missing values*) maupun data duplikat yang dapat memengaruhi proses analisis. Pada penelitian ini, proses pembersihan data menunjukkan bahwa data yang digunakan telah bersih secara struktural. Hal ini disebabkan oleh karakteristik data sepak bola yang cenderung unik di mana setiap peristiwa dalam pertandingan memiliki identitas dan konteks yang berbeda serta karena data yang disediakan oleh StatsBomb telah tersusun secara rapi dan konsisten. Struktur data yang baik ini sangat membantu dalam mempercepat proses *preprocessing* dan meningkatkan kualitas hasil analisis, karena tidak memerlukan upaya koreksi data secara signifikan.

## 4.3 Data Transformation

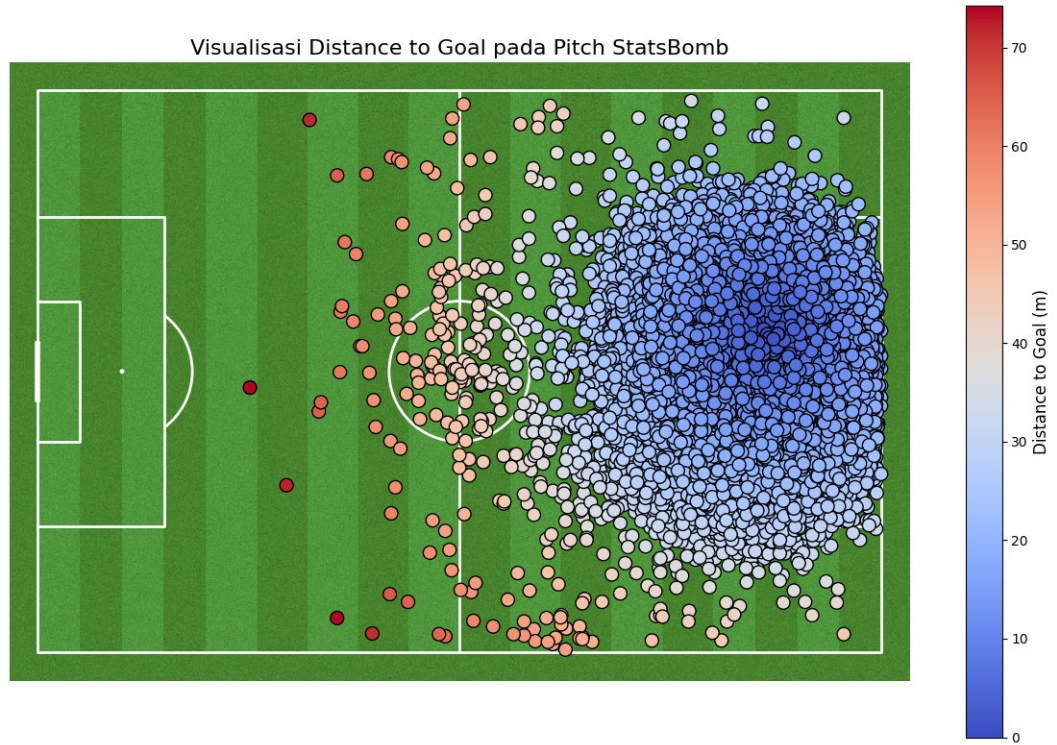
### 4.3.1 Feature Engineering



Tahapan pertama dalam proses *transformation* pada penelitian ini adalah melakukan *feature engineering* dengan menambahkan tiga fitur baru, yaitu jarak dan sudut tembakan terhadap gawang serta kejadian sebelum terjadinya *shot*. Fitur ini ditambahkan untuk memberikan informasi spasial yang lebih kaya kepada model, mengingat lokasi dan sudut tembakan serta momentum sangat berpengaruh terhadap kemungkinan terciptanya gol.

*a. Distance to Goal*

Fitur pertama dalam proses *feature engineering* adalah menghitung jarak antara posisi tembakan dan pusat gawang. Informasi spasial ini penting karena jarak tembakan merupakan salah satu faktor utama yang memengaruhi kemungkinan terciptanya gol. Semakin dekat jarak tembakan ke gawang, secara umum peluang untuk mencetak gol menjadi lebih besar. Gambar 4.4 menunjukkan visualisasi fitur *distance to goal* pada lapangan pertandingan berdasarkan koordinat StatsBomb. Titik-titik pada visualisasi merepresentasikan lokasi awal tembakan, dengan warna yang menunjukkan jaraknya terhadap gawang, semakin biru berarti semakin dekat dan semakin merah berarti semakin jauh.



Gambar 4.3 Visualisasi *Distance to Goal*

Dalam *dataset* ini, koordinat pusat gawang StatsBomb berada pada titik ( $x = 104.0$ ,  $y = 34.0$ ), yang merepresentasikan titik tengah di antara dua tiang gawang. Jarak dihitung menggunakan rumus *Euclidean distance*, yaitu akar kuadrat dari jumlah kuadrat selisih antara koordinat tembakan dan koordinat pusat gawang. Secara matematis, perhitungan ini dinyatakan sebagai berikut:

$$Distance = \sqrt{(x_{goal} - x_{start})^2 + (y_{goal} - y_{start})^2} \quad (4.1)$$

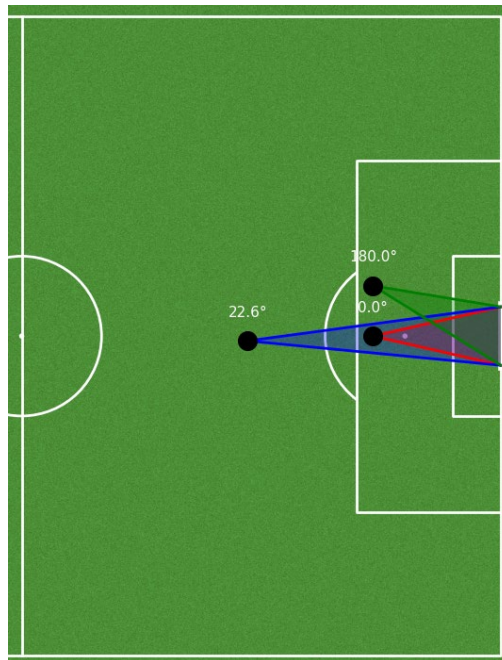
Fungsi ini diimplementasikan dalam kode Python yang akan mengembalikan nilai jarak dalam satuan relatif terhadap sistem koordinat StatsBomb. Hasil perhitungan disimpan dalam kolom baru bernama *distance\_to\_goal* dan digunakan sebagai salah satu fitur masukan dalam model prediksi. Gambar 4.7 menunjukkan contoh hasil dari proses penambahan fitur ini.

distance_to_goal
15.448625
14.905368
26.828716
10.837435
19.568598

Gambar 4.4 *Distance to Goal*

b. *Angle to Goal*

Selain jarak, fitur penting lainnya yang ditambahkan dalam proses *feature engineering* adalah sudut tembakan terhadap gawang, atau dikenal sebagai *open play angle*. Fitur ini merepresentasikan seberapa besar ruang terbuka yang tersedia bagi penembak untuk mengarahkan bola ke area di antara kedua tiang gawang. Semakin lebar sudut yang terbuka, semakin besar peluang tembakan untuk menghasilkan gol.



Gambar 4.5 Visualisasi Sudut Tembakan

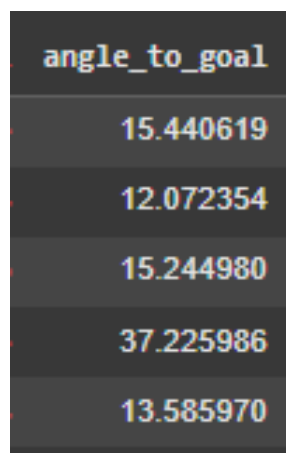
Perhitungan sudut dilakukan dengan mengacu pada tiga titik, posisi tembakan dan dua titik tiang gawang (kanan dan kiri). Dalam sistem koordinat StatsBomb, gawang terletak pada posisi horizontal tetap yaitu  $x = 120$ , dengan tiang bawah (kanan) berada pada  $y = 43,66$  dan tiang atas (kiri) pada  $y = 36,34$ . Fungsi python digunakan untuk menghitung besar sudut terbuka menggunakan hukum cosinus. Langkah-langkahnya meliputi:

- i) Hitung jarak dari posisi tembakan ke masing-masing tiang gawang (A dan B).
- ii) Hitung panjang sisi antara kedua tiang (C).
- iii) Gunakan hukum cosinus untuk mencari sudut di antara kedua sisi tersebut.

Sudut dalam radian dikonversi ke derajat menggunakan fungsi `np.degrees()`.

Jika tembakan dilakukan tepat dari titik tengah gawang ( $x = 120$  dan  $y$  berada

antara dua tiang), maka sudut maksimal akan diberikan sebesar 180 derajat. Sebaliknya, jika posisi tembakan berada sejajar secara horizontal dengan gawang tetapi tidak dalam rentang vertikal antara tiang, maka sudut dianggap 0 derajat. Hasil perhitungan ini disimpan dalam kolom *angle\_to\_goal*, yang menjadi input penting dalam proses pemodelan. Gambar 4.5 menunjukkan hasil dari *feature engineering* tersebut.



angle_to_goal
15.440619
12.072354
15.244980
37.225986
13.585970

Gambar 4.6 *Angle to Goal*

c. *Type Before*

Fitur *type\_before* ditambahkan sebagai bagian dari proses *feature engineering* untuk memberikan konteks temporal terhadap peristiwa tembakan yang dianalisis. Fitur ini merepresentasikan jenis *event* yang terjadi tepat sebelum tembakan dilakukan, dengan mengambil nilai *type.id* dari *event* sebelumnya dalam urutan kronologis pertandingan. Informasi ini bertujuan untuk menangkap dinamika permainan yang mendahului tembakan, seperti apakah tembakan tersebut terjadi setelah dribel, operan, intersepsi, atau aksi defensif lawan. Tabel 4.3 beberapa *type.id* umum dalam data, yang digunakan untuk mengidentifikasi jenis peristiwa dalam pertandingan sepak bola.

Tabel 4.3 Deskripsi Jenis *type* dalam Pertandingan Sepak Bola.

Event Type	Type ID	Deskripsi Singkat
<i>50/50</i>	33	Dua pemain dari tim berbeda berebut bola lepas.
<i>Bad Behaviour</i>	24	Pelanggaran di luar permainan yang berujung kartu.
<i>Ball Receipt*</i>	42	Momen penerimaan atau usaha menerima operan.
<i>Ball Recovery</i>	2	Usaha merebut kembali bola lepas.
<i>Block</i>	6	Pemain menghalangi bola dengan tubuhnya.
<i>Carry</i>	43	Pemain menguasai bola saat bergerak atau diam.
<i>Clearance</i>	9	Menghalau bola dari area bahaya tanpa niat mengoper ke rekan.
<i>Dispossessed</i>	3	Pemain kehilangan bola karena ditekel tanpa mencoba dribel.
<i>Dribble</i>	14	Usaha pemain melewati lawan dengan menggiring bola.
<i>Dribbled Past</i>	39	Pemain dilewati oleh lawan saat dribel.
<i>Duel</i>	4	Duel 1v1 antara pemain dari tim berbeda.
<i>Error</i>	37	Kesalahan pemain yang mengarah pada tembakan lawan.
<i>Foul Committed</i>	22	Pelanggaran yang dilakukan terhadap lawan (tidak termasuk <i>offside</i> ).
<i>Foul Won</i>	21	Pelanggaran yang diterima dan menghasilkan tendangan bebas atau penalti.
<i>Goal Keeper</i>	23	Segala aksi penjaga gawang (penyelamatan, <i>smother</i> , <i>punch</i> , dll).
<i>Half End</i>	34	Peluit akhir babak pertandingan oleh wasit.
<i>Half Start</i>	18	Peluit awal babak pertandingan oleh wasit.
<i>Injury Stoppage</i>	40	Penghentian permainan karena cedera.
<i>Interception</i>	10	Pemain memotong jalur operan lawan untuk mencegah bola sampai ke target.
<i>Miscontrol</i>	38	Kehilangan kontrol bola karena sentuhan yang buruk.
<i>Offside</i>	8	Pelanggaran posisi <i>offside</i> .

<i>Own Goal Against</i>	20	Gol bunuh diri oleh tim sendiri.
<i>Own Goal For</i>	25	Gol bunuh diri yang menguntungkan tim.
<i>Pass</i>	30	Umpan dari satu pemain ke pemain lain.
<i>Player Off</i>	27	Pemain keluar lapangan tanpa pergantian (misalnya karena cedera).
<i>Player On</i>	26	Pemain kembali masuk ke lapangan setelah <i>Player Off</i> .
<i>Pressure</i>	17	Aksi menekan pemain lawan di area tertentu, direkam bersama durasi tekanan.
<i>Referee Ball-Drop</i>	41	Wasit menjatuhkan bola untuk melanjutkan pertandingan setelah jeda (misalnya cedera).
<i>Shield</i>	28	Pemain melindungi bola agar keluar lapangan tanpa dikejar lawan.
<i>Shot</i>	16	Upaya mencetak gol dengan bagian tubuh legal.
<i>Starting XI</i>	35	Informasi awal pemain yang bermain dan formasi tim.
<i>Substitution</i>	19	Pergantian pemain saat pertandingan berlangsung.
<i>Tactical Shift</i>	36	Perubahan posisi pemain atau formasi taktik dalam pertandingan.

Dengan menambahkan konteks ini, model dapat memahami alur permainan yang berujung pada tembakan dan mengenali pola peristiwa yang secara statistik lebih mungkin menghasilkan gol. Fitur *type\_before* diisi hanya jika terdapat *event* sebelumnya, jika tembakan merupakan *event* pertama dalam urutan, maka fitur ini dikosongkan. Gambar 4.7 menunjukkan hasil dari *feature engineering* tersebut.

type_before
2
43
42
4
4

Gambar 4.7 Type Before

#### 4.3.2 Seleksi Fitur

Tahap ini bertujuan untuk memilih fitur-fitur yang paling relevan dan berpengaruh terhadap prediksi model, sehingga dapat meningkatkan efisiensi dan akurasi pemodelan. Seleksi fitur dilakukan setelah proses *feature engineering* selesai, dengan mempertimbangkan konteks domain serta performa masing-masing fitur dalam mendukung prediksi *shot\_outcome*. Fitur yang memiliki kontribusi kecil atau *redundan* dapat dihilangkan untuk menghindari kompleksitas berlebih dan mengurangi risiko *overfitting*. Proses ini membantu model fokus pada informasi yang benar-benar penting. Tabel 4.1 menunjukkan fitur-fitur yang dipertahankan setelah melalui tahap seleksi.

Tabel 4.4 Fitur-Fitur Pada Tahap Seleksi

No.	Fitur
1	<i>minute</i>
2	<i>second</i>
3	<i>play_pattern</i>
4	<i>position</i>
5	<i>shot_technique</i>



6	<i>shot_body_part</i>
7	<i>shot_type</i>
8	<i>shot_first_time</i>
9	<i>shot_open_goal</i>
10	<i>shot_one_on_one</i>
11	<i>shot_aerial_won</i>
12	<i>under_pressure</i>
13	<i>distance_to_goal</i>
14	<i>angle_to_goal</i>
15	<i>shot_key_pass</i>
16	<i>start_x</i>
17	<i>start_y</i>
18	<i>possession</i>

#### 4.3.3 Pemisahan Data Uji dan Data Latih

Salah satu tahapan penting dalam proses *transformation* adalah pemisahan data menjadi data latih dan data uji. Tujuan dari proses ini adalah untuk mengevaluasi kinerja model secara objektif terhadap data yang belum pernah digunakan dalam proses pelatihan. Pemisahan data dilakukan menggunakan fungsi *train\_test\_split* dari *library scikit-learn*, dengan proporsi 90% data sebagai data latih dan 10% sebagai data uji. Parameter *random\_state* disetel ke angka 42 untuk menjamin konsistensi hasil pemisahan saat kode dijalankan ulang. Setelah proses ini dilakukan, diperoleh 61.981 baris data untuk pelatihan dan 6.887 baris data untuk pengujian. Gambar 4.6 menunjukkan jumlah baris dan kolom data latih dan uji.

	Train	Test
Rows	61981	6887
Columns	17	17

Gambar 4.8 Jumlah Baris dan Kolom Data Latih dan Uji.

## 4.4 Data Mining

### 4.4.1 Perancangan Model

Pada penelitian ini, algoritma yang digunakan untuk membangun model adalah LightGBM, yang dikenal memiliki efisiensi tinggi dan performa unggul pada data dengan dimensi besar.

#### a. Inisiasi Model

Langkah pertama dalam perancangan model adalah melakukan inisialisasi algoritma yang akan digunakan. Proses inisialisasi dilakukan dengan membuat objek *LGBMClassifier* dari *library scikit-learn*, dan parameter *random\_state* diatur ke nilai 42 untuk memastikan hasil yang *reproduksibel*. Selanjutnya, ditentukan ruang pencarian *hyperparameter* yang akan digunakan dalam proses *tuning*, yang meliputi parameter *min\_child\_samples*, *num\_leaves*, *reg\_lambda*, *reg\_alpha*, dan *max\_depth*. Parameter-parameter ini diatur dalam bentuk distribusi acak menggunakan fungsi *sp\_randint* dan *sp\_uniform* dari *scipy.stats*, yang akan menjadi acuan dalam proses pemilihan kombinasi terbaik pada tahap pencarian *hyperparameter* berikutnya.

#### b. Definisi Fungsi Scoring

Setelah inisiasi model dilakukan, langkah selanjutnya adalah mendefinisikan fungsi penilaian yang digunakan untuk mengevaluasi performa model selama

proses *hyperparameter tuning*. Pada penelitian ini digunakan dua metrik evaluasi, yaitu ROC AUC dan *Brier Score*. Fungsi penilaian ini didefinisikan dalam bentuk *dictionary* dengan nama *scoring*, di mana ROC AUC dipanggil secara langsung dan *Brier Score* didefinisikan menggunakan *make\_scorer* dari *scikit-learn* dengan argumen *greater\_is\_better=False* karena nilai *Brier Score* yang lebih kecil menunjukkan performa yang lebih baik.

#### c. Pelatihan Model

Setelah fungsi penilaian ditentukan, tahap berikutnya adalah menginisialisasi proses *hyperparameter tuning* menggunakan *RandomizedSearchCV*. Teknik ini digunakan untuk mencari kombinasi parameter terbaik dari model LightGBM berdasarkan evaluasi dengan *5-fold cross-validation* dan 100 iterasi pencarian. Tidak seperti proses *tuning* standar yang hanya mempertimbangkan satu metrik, dalam penelitian ini digunakan pendekatan *refit kustom* untuk menyeimbangkan antara akurasi klasifikasi dan kualitas kalibrasi probabilitas.

Fungsi *custom\_refit* yang digunakan bertujuan untuk memilih model dengan nilai ROC AUC tertinggi, tetapi juga mempertimbangkan *Brier Score* agar model yang terpilih tidak hanya mampu mengklasifikasi dengan baik, tetapi juga memberikan estimasi probabilitas yang kalibrasinya baik. Jika model dengan ROC AUC tertinggi memiliki *Brier Score* lebih kecil dari ambang batas tertentu (misalnya -0.1), maka model tersebut akan dipilih. Jika tidak, sistem akan memilih model dengan *Brier Score* terbaik. Setelah objek *RandomizedSearchCV* dikonfigurasi dengan parameter, metrik penilaian, dan

fungsi *refit*, proses pelatihan model dilakukan dengan memanggil fungsi *fit* pada data latih (*X\_train* dan *y\_train*).

Parameter-parameter ini dipilih secara otomatis oleh algoritma pencarian (*RandomizedSearchCV*) berdasarkan kinerja terbaik dalam pelatihan. Setiap parameter memainkan peran penting dalam mengontrol struktur pohon, regularisasi, pembobotan, serta teknik pembelajaran untuk meningkatkan akurasi dan mencegah *overfitting*. Tabel 4.5 menunjukkan konfigurasi akhir model LightGBM setelah *tuning*.

Tabel 4.5 Konfigurasi Akhir Model LightGBM Setelah *Tuning*

Parameter	Nilai
cv	3
method	isotonic
boosting_type	gbdt
num_leaves	15
max_depth	84
min_child_samples	146
min_child_weight	0.001
min_split_gain	0.0
colsample_bytree	1.0
subsample	1.0
subsample_for_bin	200000
subsample_freq	0
learning_rate	0.1
n_estimators	100
reg_alpha	0.513

reg_lambda	0.971
random_state	42
importance_type	split

d. Pemilihan Model Terbaik dan Kalibrasi Probabilitas

Setelah proses *RandomizedSearchCV* selesai, langkah selanjutnya adalah mengambil model terbaik yang diperoleh dari hasil pencarian *hyperparameter*. Model terbaik ini diakses melalui atribut *best\_estimator\_* dan merupakan konfigurasi LightGBM dengan performa optimal berdasarkan kriteria *custom refit* yang telah ditentukan sebelumnya. Untuk meningkatkan akurasi estimasi probabilitas, dilakukan tahap kalibrasi menggunakan *CalibratedClassifierCV* dengan metode *isotonic regression* dan *cross-validation* sebanyak tiga lipatan. Kalibrasi ini bertujuan agar *output* probabilitas dari model lebih merefleksikan tingkat kepercayaan yang sebenarnya, terutama dalam konteks prediksi tembakan yang menghasilkan gol atau tidak. Proses pelatihan ulang dilakukan terhadap model yang telah dikalibrasi menggunakan data latih sebelum model digunakan dalam tahap evaluasi akhir.

#### 4.4.2 Permodelan LGBM

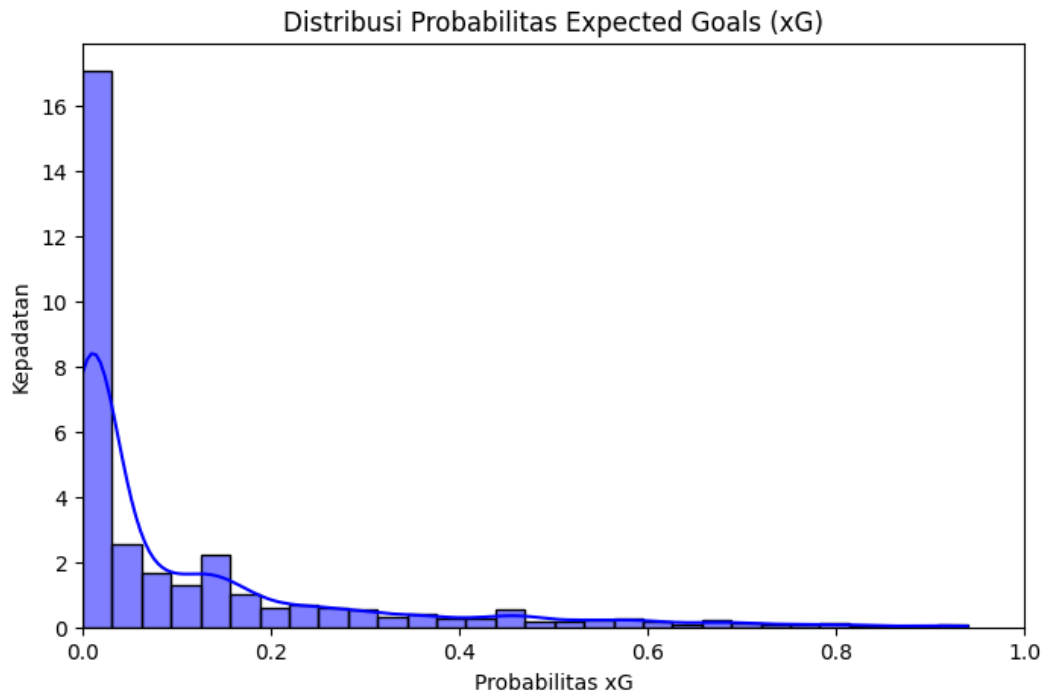
Setelah melakukan pencarian nilai *hyperparameter* terbaik melalui proses *RandomizedSearchCV*, nilai-nilai tersebut digunakan pada tahap permodelan akhir. Nilai-nilai parameter pada *hyperparameter* mencerminkan peran penting dari dua teknik inti dalam LightGBM, yaitu *Gradient-based One-Side Sampling* (GOSS) dan *Exclusive Feature Bundling* (EFB).

Teknik GOSS memungkinkan model untuk fokus pada *instance* dengan gradien tinggi, yang biasanya lebih informatif untuk proses pembelajaran, sehingga mempercepat pelatihan tanpa mengorbankan akurasi. Hal ini sangat relevan dengan parameter *min\_child\_samples* yang cukup besar (146), karena membantu menjaga kestabilan pembagian *node* meskipun jumlah data yang di sampling dikurangi oleh GOSS. Sementara itu, teknik EFB menggabungkan fitur-fitur eksklusif yang tidak aktif bersamaan, sehingga memungkinkan penggunaan jumlah *num\_leaves* yang besar (15) tanpa meningkatkan kompleksitas model secara drastis.

#### **4.4.3 Visualisasi**

Salah satu langkah awal dalam proses evaluasi model prediktif adalah dengan memahami pola distribusi dari nilai xG yang dihasilkan. Dalam konteks ini, dilakukan visualisasi berupa histogram dan kurva distribusi (KDE plot) terhadap nilai-nilai prediksi dari model LGBM. Tujuan utama dari visualisasi ini adalah untuk mengidentifikasi karakteristik sebaran data, termasuk kecenderungan *skewness*, serta untuk memahami apakah model cenderung menghasilkan prediksi yang konservatif (nilai xG rendah) atau agresif (nilai xG tinggi). Gambar 4.9 menyajikan histogram distribusi nilai xG yang diprediksi oleh model LGBM, dilengkapi dengan estimasi *kernel density estimation* (KDE) untuk memberikan

representasi yang lebih halus terhadap bentuk sebaran tersebut.



Gambar 4.9 Histogram Distribusi Nilai xG

Distribusi nilai xG yang dihasilkan oleh model LGBM menunjukkan karakteristik *positively skewed* yang sangat kuat, dengan sebagian besar nilai terkonsentrasi mendekati 0.0, sebagaimana ditunjukkan oleh frekuensi tertinggi pada bin pertama (sekitar 0.0–0.02) dan puncak tajam kurva KDE di titik tersebut. Hal ini mencerminkan realitas bahwa mayoritas tembakan dalam sepak bola merupakan peluang berkualitas rendah misalnya, tembakan dari jarak jauh atau sudut sempit dengan probabilitas gol yang sangat kecil. Modus distribusi yang berada sangat dekat dengan nol memperkuat interpretasi ini. Di sisi lain, distribusi juga memiliki *long tail* ke kanan, yang membentang hingga nilai xG tinggi (seperti 0.4, 0.6, hingga mendekati 1.0), merepresentasikan peluang emas seperti penalti, tap-in, atau situasi satu lawan satu yang meskipun jarang terjadi, menyumbang

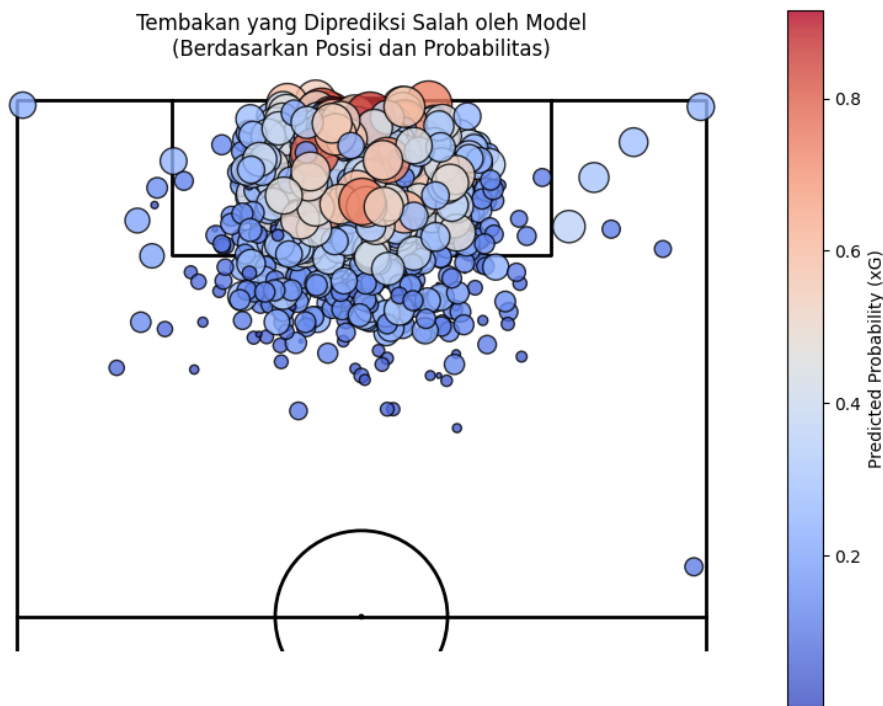
proporsi signifikan terhadap total peluang gol. Pola ini mencerminkan distribusi *heavy-tailed* atau bahkan menyerupai *power law*, di mana sebagian kecil peristiwa ekstrem (tembakan berkualitas sangat tinggi) memberikan dampak besar terhadap akumulasi nilai xG total. Kurva KDE berperan penting dalam menghaluskan bentuk distribusi dan memperjelas pola yang tidak tampak secara eksplisit dalam histogram, termasuk punuk kecil di kisaran xG menengah (0.15–0.4) yang dapat menunjukkan keberadaan sub-kategori peluang dengan tingkat kesulitan sedang.

Area di bawah kurva KDE antara dua titik (misalnya 0.05–0.10) dapat diinterpretasikan sebagai probabilitas kumulatif kemunculan tembakan dalam rentang tersebut, dengan luas area yang sangat besar di dekat 0.0 menunjukkan bahwa probabilitas tembakan memiliki xG rendah ( $P(0 \leq xG \leq 0.05)$ ) sangat tinggi. Secara keseluruhan, distribusi ini mencerminkan sifat permainan sepak bola yang berkarakter skor rendah, dengan dominasi tembakan risiko rendah dan hanya sebagian kecil peluang berkualitas tinggi, sehingga memperkuat validitas statistik serta kesesuaian konteks model dengan realitas permainan.

Melanjutkan dari analisis distribusi numerik, pendekatan visual berbasis spasial juga digunakan untuk memperdalam interpretasi terhadap perilaku model dalam konteks pertandingan sebenarnya. Peta tembakan (shot map) disusun untuk memvisualisasikan lokasi-lokasi di lapangan tempat tembakan dilakukan, dengan setiap titik pada peta dilengkapi oleh nilai xG yang diprediksi oleh model. Representasi visual ini menggunakan variasi warna atau ukuran titik sebagai indikator kuantitatif nilai xG, sehingga memungkinkan identifikasi cepat terhadap tembakan-tembakan berisiko tinggi maupun rendah berdasarkan posisi



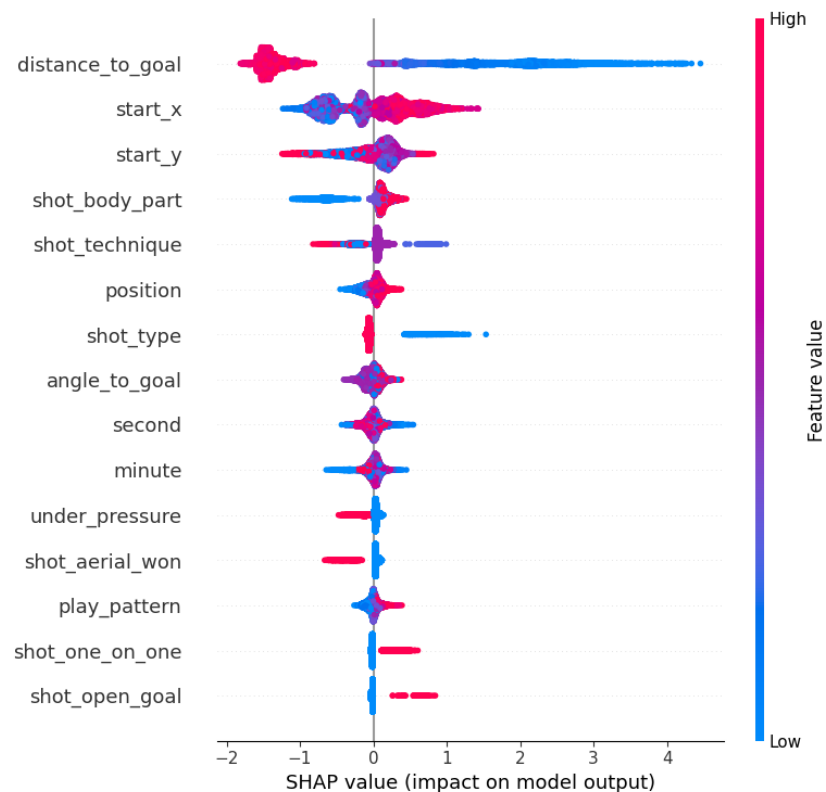
geografisnya di dalam area permainan. Gambar 4.10 menyajikan peta sebaran tembakan tersebut, yang menjadi alat bantu penting dalam memahami dinamika spasial peluang dalam suatu pertandingan secara intuitif dan informatif.



Gambar 4.10 Peta Sebaran dengan Nilai xG

Selain visualisasi spasial melalui peta tembakan, interpretasi lebih mendalam terhadap perilaku internal model dilakukan dengan menggunakan pendekatan explainable AI, yaitu SHAP (*SHapley Additive exPlanations*). Metode ini memungkinkan analisis kontribusi masing-masing fitur terhadap prediksi nilai xG, baik secara global (menjelaskan pengaruh fitur terhadap keseluruhan prediksi model) maupun secara lokal (menjelaskan kontribusi fitur terhadap prediksi spesifik pada satu observasi/tembakan). Dengan memanfaatkan SHAP, dapat diidentifikasi fitur mana yang paling berperan dalam membentuk nilai prediksi, seperti jarak tembakan, sudut terhadap gawang, jenis aksi sebelum tembakan, atau

posisi pemain bertahan terdekat. Visualisasi pada Gambar 4.11 menunjukkan bagaimana nilai SHAP terdistribusi untuk berbagai fitur penting dalam model, serta bagaimana masing-masing nilai input memengaruhi naik atau turunnya prediksi xG.



Gambar 4.11 Visualisasi SHAP untuk Interpretasi Model xG

Analisis SHAP global menunjukkan *distance\_to\_goal* sebagai fitur paling berpengaruh, dengan nilai rendah (titik biru) memberikan SHAP positif tinggi yang mendorong prediksi gol, dan nilai tinggi (titik merah) memberikan SHAP negatif yang menurunkan peluang gol, mencerminkan hubungan invers yang sangat kuat dan intuitif. Fitur kedua terpenting, *start\_x*, memperlihatkan nilai tinggi (posisi lebih maju di lapangan) terkait SHAP positif, menunjukkan tendangan dari posisi lebih dekat gawang lawan meningkatkan peluang gol, sedangkan nilai rendah

terkait SHAP negatif. Sedangkan *start\_y* memiliki pengaruh signifikan, nilai ekstrem di sisi lapangan (titik merah dan biru) memberikan SHAP negatif atau mendekati nol, sementara nilai tengah lapangan menghasilkan SHAP positif, mengindikasikan bahwa tendangan dari posisi sentral lebih berpeluang gol karena sudut tembak yang lebih menguntungkan dibandingkan posisi samping yang sempit.

Fitur *shot\_body\_part* menunjukkan dampak cukup besar, dengan nilai tinggi (mewakili kaki dominan atau sundulan) berkorelasi positif terhadap prediksi gol, sedangkan nilai rendah (kaki non-dominan atau bagian tubuh lain) cenderung negatif, menandakan efektivitas bagian tubuh tertentu dalam mencetak gol. *shot\_technique* juga berpengaruh signifikan, teknik tendangan tertentu (seperti *volleys* atau tendangan melengkung) memiliki SHAP positif, sedangkan teknik lain mengurangi peluang gol. Pada *position*, nilai SHAP positif terkait posisi pemain menyerang (*striker*, *midfielder*), sementara posisi bertahan (bek, kiper) cenderung negatif, mengindikasikan posisi sangat memengaruhi probabilitas gol. *shot\_type* memperlihatkan tipe tendangan efektif (tendangan kuat, penalti) dengan nilai SHAP positif, berbeda dengan tendangan spekulatif yang negatif. Sedangkan *angle\_to\_goal* memiliki pengaruh moderat, sudut tembak lebar (nilai tinggi) menghasilkan SHAP positif, sedangkan sudut sempit negatif, sesuai dengan logika bahwa sudut tembak lebih terbuka meningkatkan peluang gol.

Fitur waktu seperti *second* dan *minute* berpengaruh kecil, dengan distribusi SHAP yang tersebar dan pola halus, menunjukkan waktu dalam detik atau menit saat tendangan diambil hanya memiliki efek minimal atau spesifik terhadap

probabilitas gol. *under\_pressure* menunjukkan bahwa tendangan dalam tekanan (titik merah) memiliki SHAP negatif, sedangkan tanpa tekanan positif, konsisten dengan penurunan peluang gol saat ada pengawalan ketat. *shot\_aerial\_won* memiliki dampak kecil, duel udara yang dimenangkan sebelum tendangan (titik merah) memberikan sedikit peningkatan probabilitas gol. *play\_pattern* juga berdampak kecil, dengan pola tertentu (serangan balik cepat, *set piece*) memberikan SHAP positif, dan pola lain (*open play* stagnan) negatif, mengindikasikan peran pola serangan terhadap efektivitas tembakan.

Fitur situasional seperti *shot\_one\_on\_one* memiliki pengaruh kecil namun konsisten, tendangan dalam situasi satu lawan satu dengan kiper (titik merah) meningkatkan probabilitas gol, sedangkan bukan situasi tersebut cenderung netral atau negatif. Terakhir, *shot\_open\_goal* memiliki dampak paling kecil tetapi sangat positif, menunjukkan bahwa tendangan ke gawang kosong secara jelas meningkatkan probabilitas gol walaupun frekuensinya rendah, dan model secara efektif menangkap efek signifikan ini.

#### **4.5 Evaluation**

Tahap evaluasi dilakukan terhadap data uji yang telah dipisahkan pada proses data *transformation*. Evaluasi bertujuan untuk mengukur sejauh mana model LightGBM mampu memberikan prediksi probabilistik yang akurat terhadap kemungkinan terciptanya gol (*expected goals*). Metrik evaluasi yang digunakan pada penelitian ini terdiri dari dua metrik utama, yaitu *Brier Score* dan ROC AUC. Kedua metrik ini dipilih karena sesuai dengan tujuan dari model xG, yaitu

menghasilkan prediksi dalam bentuk probabilitas, bukan klasifikasi biner semata. Tabel 4.6 berikut menunjukkan hasil evaluasi model LightGBM berdasarkan kedua metrik tersebut.

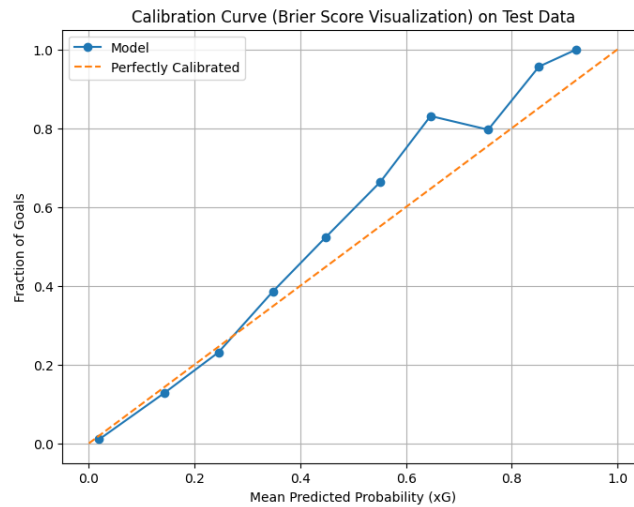
Tabel 4.6 Hasil Evaluasi Model LightGBM

Metrik Evaluasi	Nilai
<i>Brier Score</i>	0.0663
ROC AUC	0.9134

#### 4.5.2 *Brier Score*

Nilai *Brier Score* sebesar 0.0663 mengindikasikan bahwa prediksi probabilitas yang dihasilkan oleh model memiliki tingkat kesalahan kuadrat yang sangat rendah. Hal ini menandakan bahwa model mampu mengestimasi peluang terciptanya gol secara akurat dan konsisten terhadap data aktual pada data uji. Dalam konteks model *expected goals* (xG), nilai *Brier Score* yang rendah sangat penting karena model ini tidak hanya bertujuan untuk mengklasifikasi hasil tembakan, tetapi juga memberikan probabilitas yang merepresentasikan peluang realistis terjadinya gol.

Visualisasi hasil kalibrasi pada data uji ditampilkan pada Gambar 4.12, yang menunjukkan hubungan antara rata-rata probabilitas prediksi (xG) dan proporsi aktual terjadinya gol (*fraction of goals*). Kurva yang mendekati garis diagonal membuktikan bahwa model memiliki kalibrasi yang baik dan dapat diandalkan dalam memberikan prediksi probabilitas.



Gambar 4.12 Calibration Curve (Brier Score Visualization) on Test Data

Lebih lanjut, nilai ini menunjukkan bahwa prediksi probabilitas yang dikeluarkan oleh model memiliki kalibrasi yang baik, artinya semakin tinggi nilai probabilitas prediksi suatu tembakan, maka semakin besar pula kecenderungannya untuk benar-benar menjadi gol dalam data aktual. Ini dibuktikan melalui kurva kalibrasi, di mana garis biru (hasil model) mengikuti dengan cukup dekat garis oranye putus-putus yang merepresentasikan kondisi kalibrasi sempurna. Meskipun terdapat sedikit deviasi pada beberapa titik terutama di rentang probabilitas menengah ke atas namun secara keseluruhan, distribusi prediksi tetap mencerminkan tren empiris yang logis dan stabil.

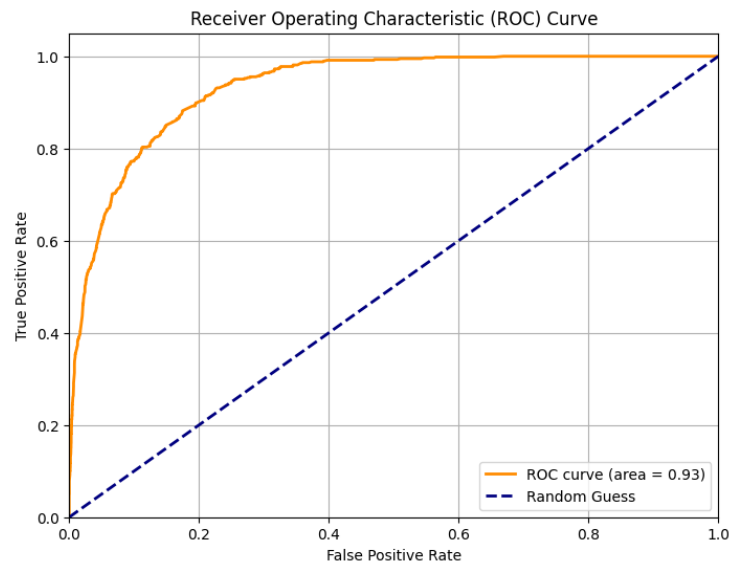
Kinerja ini menjadi indikator bahwa proses kalibrasi menggunakan metode *isotonic regression* berhasil menyesuaikan *output* model sehingga prediksi probabilitas tidak bersifat *overconfident* maupun *underconfident*. Hal ini sangat krusial pada skenario data *imbalanced* seperti prediksi xG, di mana sebagian besar

tembakan memang tidak berujung pada gol, dan model cenderung rawan bias terhadap mayoritas kelas.

#### 4.5.3 ROC AUC

Model LightGBM menunjukkan performa diskriminatif yang sangat baik dengan nilai ROC AUC sebesar 0.93. Nilai ini menunjukkan bahwa model memiliki kemampuan yang sangat tinggi dalam membedakan antara tembakan yang menghasilkan gol dan tembakan yang tidak. ROC *curve* yang dihasilkan memiliki kurva yang menjauh signifikan dari garis diagonal (*random guess*), mengindikasikan bahwa model mampu mengidentifikasi *true positive* dengan tingkat *false positive* yang rendah pada sebagian besar *threshold*.

Performa ini juga memperlihatkan bahwa fitur-fitur yang digunakan dalam pelatihan, termasuk informasi spasial, teknik tembakan, serta konteks permainan (seperti tekanan dan posisi lawan), berhasil dikombinasikan secara efektif oleh model untuk mengenali pola-pola yang berkontribusi terhadap terciptanya gol. ROC *curve* yang stabil dan area kurva yang luas merupakan indikator bahwa model dapat digunakan dalam skenario nyata yang memerlukan prediksi *ranking* (misalnya untuk mengurutkan kualitas peluang tembakan). Visualisasi kinerja diskriminatif model ditampilkan pada Gambar 4.13, yang memperlihatkan perbandingan antara ROC *curve* model dengan *baseline random guess*.



Gambar 4.13 Receiver Operating Characteristic (ROC) Curve

#### 4.5.4 Perbandingan dengan Model Pada Literatur Lain

Untuk menilai kinerja model yang dikembangkan dalam penelitian ini secara lebih komprehensif, dilakukan perbandingan terhadap hasil evaluasi dari beberapa model xG yang telah dikembangkan pada studi sebelumnya. Hasil dari model LightGBM yang dibangun dalam penelitian ini menunjukkan kinerja yang sangat kompetitif, dengan *Brier Score* sebesar 0.0663 dan ROC AUC sebesar 0.9134, mengungguli sebagian besar model yang ada dalam literatur terdahulu. Rangkuman perbandingan hasil evaluasi model pada beberapa studi terdahulu dapat dilihat pada Tabel 4.7.

Tabel 4.7 Perbandingan Hasil Evaluasi Model xG pada Berbagai Literatur

Penulis & Tahun	Model	<i>Brier Score</i>	ROC AUC
Scholtes & Karakuş (2024)	<i>Bayesian Hierarchical</i>	0.075	–
ElHabr (2023)	XGBoost (Opta npxG)	0.0715	–
Cavus & Biecek (2022)	LightGBM	0.173	0.818
Haaren (2021)	<i>Boosting Machine</i>	0.082	0.793



Eggels et al. (2016)	<i>Random Forest</i>	–	0.814
Anzer & Bauer (2021)	GBM	–	0.822
Mead et al. (2023)	XGBoost	0.0799	0.800
<b>Model Penelitian ini</b>	LightGBM + Calibration	<b>0.0663</b>	<b>0.913</b>

Berdasarkan Tabel 4.7, model LightGBM yang dikembangkan dalam penelitian ini menunjukkan kinerja superior dibandingkan model-model xG yang telah dikembangkan sebelumnya. Dengan Brier Score sebesar 0.0663 dan ROC AUC 0.9134, model ini mencatatkan hasil terbaik di antara model pembanding.

Model ini mengungguli pendekatan Bayesian hierarchical dari Scholtes & Karakuş (2024) dengan Brier Score 0.075, meskipun ROC AUC tidak dilaporkan. Demikian pula, model ini lebih unggul dibandingkan XGBoost dari Opta npxG perusahaan sports analytics asal Inggris menurut ElHabr (2023) yang mencatat Brier Score 0.0715, meskipun tanpa nilai ROC AUC. Sementara itu, model LightGBM Cavus & Biecek (2022) menghasilkan Brier Score 0.173 dan ROC AUC 0.818, jauh di bawah model ini meskipun menggunakan algoritma yang sama.

Model ini juga lebih baik dibandingkan XGBoost Mead et al. (2023) dengan Brier Score 0.0799 dan ROC AUC 0.800, serta model Boosting Machine dari Haaren (2021) yang mencatat Brier 0.082 dan ROC AUC 0.793. Model klasik seperti Random Forest (Eggels et al., 2016) dan GBM (Anzer & Bauer, 2021) masing-masing mencatat ROC AUC 0.814 dan 0.822, namun tanpa pelaporan Brier Score.

#### 4.6 Interpretasi Hasil

Model expected goals (xG) yang dikembangkan dalam penelitian ini menunjukkan performa yang cukup baik dalam mengestimasi probabilitas terjadinya gol berdasarkan variabel-variabel yang relevan dalam situasi tembakan. Evaluasi performa model melalui metrik-metrik seperti Brier score, log loss, dan ROC-AUC mengindikasikan bahwa model memiliki tingkat akurasi prediksi yang memadai serta kestabilan yang layak untuk digunakan dalam konteks analisis performa sepak bola. Hal ini menunjukkan bahwa model mampu menangkap pola-pola signifikan dalam data historis dan menerjemahkannya menjadi prediksi probabilistik yang representatif terhadap kemungkinan terciptanya gol.

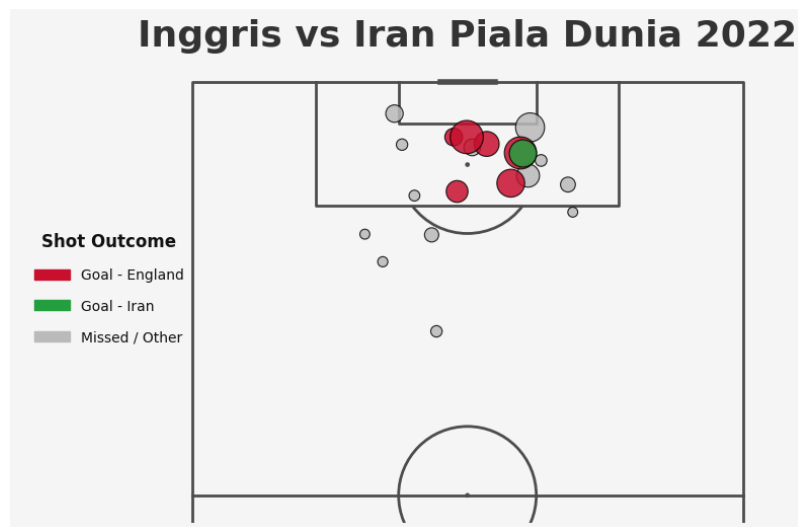
Untuk menginterpretasikan lebih lanjut kemampuan model dalam konteks dunia nyata, dilakukan penerapan perhitungan xG terhadap data spesifik dari suatu pertandingan sebagai studi kasus. Sebagai contoh, Tabel 4.8 menyajikan visualisasi distribusi nilai xG dari masing-masing peluang yang tercipta dalam pertandingan antara tim nasional Inggris melawan Iran pada Piala Dunia FIFA 2022. Hasil ini memberikan gambaran empiris mengenai kualitas peluang yang diciptakan oleh masing-masing tim selama pertandingan berlangsung, serta menunjukkan bagaimana model dapat digunakan untuk menganalisis efisiensi konversi peluang menjadi gol secara lebih objektif dan terukur.

Tabel 4.8 Statistik Efektivitas Penyerangan Berdasarkan *Expected Goals*

Tim Nasional	Total <i>Expected Goals</i> (xG)	Jumlah Tembakan	Jumlah Gol Aktual	Rata-rata xG per Tembakan	Diferensial Gol (Gol – xG)
Inggris	3.09	13	6	0.238	+2.91
Iran	0.87	7	1	0.125	+0.13

Selain dari sisi akurasi prediktif, model xG yang dikembangkan juga memberikan kontribusi signifikan dalam konteks visualisasi dan analisis pertandingan sepak bola. Dengan kemampuannya mengkuantifikasi kualitas peluang secara numerik, model ini dapat menghasilkan metrik xG yang informatif dan aplikatif, sehingga dapat menjadi alat bantu strategis bagi tim analis dalam mengevaluasi performa tim maupun pemain secara lebih objektif. Analisis berbasis xG juga membuka ruang untuk interpretasi yang lebih dalam terhadap dinamika pertandingan, tidak hanya berdasarkan skor akhir, tetapi juga berdasarkan kualitas peluang yang diciptakan dan dihadapi.

Lebih lanjut, model ini juga dapat diintegrasikan sebagai alat ukur performa yang tervisualisasi secara intuitif dan informatif. Gambar 4.14 menyajikan visualisasi sebaran tembakan (shot map) beserta nilai xG masing-masing peluang yang terjadi pada pertandingan antara Inggris melawan Iran di Piala Dunia 2022. Visualisasi ini memungkinkan pemahaman spasial yang lebih baik terhadap lokasi dan kualitas peluang, serta membantu mengidentifikasi area-area strategis yang menjadi sumber utama ancaman serangan selama pertandingan. Dengan demikian, model xG ini tidak hanya berfungsi sebagai alat prediksi, tetapi juga sebagai media analisis taktis yang kaya informasi.



Gambar 4.14 *Shot Map* Inggris vs Iran

Untuk mengevaluasi sejauh mana performa model xG ini dalam konteks prediksi pertandingan secara langsung, dilakukan perbandingan hasil prediksi dengan data xG yang disediakan oleh beberapa penyedia statistik sepak bola ternama seperti Opta, Pro Football Focus (PFF), FBref, dan xGScore. Perbandingan ini dilakukan pada tiga pertandingan kunci di ajang Piala Dunia 2022, yaitu Inggris vs Iran, Inggris vs Prancis, dan Argentina vs Kroasia, serta satu pertandingan final UEFA Euro 2024. Tabel 4.9 menyajikan nilai xG yang dihasilkan oleh model ini pada masing-masing pertandingan tersebut, beserta perbandingannya dengan estimasi dari penyedia statistik lainnya. Analisis ini bertujuan untuk menilai konsistensi dan validitas model dalam konteks aplikatif, serta menakar sejauh mana model yang dikembangkan mampu menghasilkan estimasi yang kompetitif dibandingkan dengan standar industri dalam bidang analisis sepak bola berbasis data.

Tabel 4.9 Perbandingan Model dengan Penyedia Statistik Sepak Bola

<i>Pertandingan</i>	<i>Skor</i>	<i>Sumber</i>	<i>xG Tim A</i>	<i>xG Tim B</i>
---------------------	-------------	---------------	-----------------	-----------------

<i>England vs Iran – WC 2022</i>	6 – 2	<b>LGBM</b>	England: <b>3.09</b>	Iran: <b>0.87</b>
		Opta	England: 2.109	Iran: 1.751
		xGScore.io	England: 2.14	Iran: 1.42
		FBref	England: 2.1	Iran: 1.4
		PFF	England: 2.14	Iran: 1.62
<i>England vs France – WC 2022</i>	1 – 2	<b>LGBM</b>	England: <b>1.98</b>	France: <b>0.64</b>
		PFF	England: 2.4	France: 0.73
		xGScore.io	England: 2.55	France: 1.21
		FBref	England: 2.4	France: 0.9
		Opta	England: 2.407	France: 1.012
<i>Argentina vs Croatia – WC 2022</i>	3 – 0	<b>LGBM</b>	Argentina: <b>2.01</b>	Croatia: <b>0.95</b>
		PFF	Argentina: 2.12	Croatia: 0.30
		xGScore.io	Argentina: 2.76	Croatia: 0.57
		Opta	Argentina: 2.33	Croatia: 0.52
		FBref	Argentina: 2.3	Croatia: 0.5
<i>Spain vs England – Final EURO 2024</i>	2 – 1	<b>LGBM</b>	England: <b>0.67</b>	Spain: <b>1.63</b>
		xGScore.io	England: 0.63	Spain: 1.9
		FBref	England: 0.5	Spain: 1.9
		Opta	England: 0.527	Spain: 1.953
		PFF	–	–

Pada pertandingan antara Inggris melawan Iran di fase grup Piala Dunia 2022 yang berakhir dengan skor 6–2, model yang dikembangkan dalam penelitian ini menghasilkan estimasi nilai xG sebesar 3,09 untuk Inggris dan 0,87 untuk Iran. Jika dibandingkan dengan data dari penyedia statistik lainnya, terdapat perbedaan yang cukup signifikan. Opta mencatat xG sebesar 2,109 (Inggris) dan 1,751 (Iran), sementara xgscore.io melaporkan nilai 2,14 (Inggris) dan 1,42 (Iran). FBref

memberikan estimasi serupa yaitu 2,1 untuk Inggris dan 1,4 untuk Iran, sedangkan PFF mencatat 2,14 untuk Inggris dan 1,62 untuk Iran. Meskipun terdapat variasi antar penyedia, model ini menunjukkan kecenderungan yang lebih tinggi dalam memperkirakan dominasi Inggris, dengan nilai xG yang mencerminkan secara lebih jelas disparitas kualitas peluang yang tercipta di antara kedua tim.

Pada pertandingan perempat final antara Inggris dan Prancis (1–2), model ini memprediksi xG sebesar 1,98 untuk Inggris dan 0,64 untuk Prancis. Angka ini mengindikasikan bahwa Inggris menciptakan peluang dengan kualitas lebih tinggi dibanding Prancis, meskipun hasil akhir menunjukkan sebaliknya. Bila dibandingkan dengan penyedia data lainnya, PFF mencatat 2,4 (Inggris) dan 0,73 (Prancis), xgscore.io memberikan 2,55 dan 1,21, sementara FBref dan Opta masing-masing memperkirakan 2,4 dan 0,9 serta 2,407 dan 1,012. Secara umum, model ini memberikan estimasi yang lebih konservatif untuk Prancis, namun tetap sejalan dengan kesimpulan bahwa Inggris memiliki dominasi peluang dalam pertandingan tersebut.

Selanjutnya, pada laga semifinal antara Argentina dan Kroasia yang berakhir dengan kemenangan Argentina 3–0, model ini memperkirakan nilai xG sebesar 2,01 untuk Argentina dan 0,95 untuk Kroasia. Estimasi ini relatif sejalan dengan hasil observasi dan mendekati beberapa penyedia data resmi. PFF mencatat 2,12 untuk Argentina dan hanya 0,30 untuk Kroasia. Sementara itu, xgscore.io dan FBref memberikan nilai yang sedikit lebih tinggi untuk Argentina, yakni masing-masing 2,76 dan 2,3, dan nilai lebih rendah untuk Kroasia, yaitu 0,57 dan 0,5. Opta memberikan estimasi sebesar 2,336 (Argentina) dan 0,520 (Kroasia). Secara umum,

model ini menampilkan prediksi yang stabil dan mencerminkan keseimbangan realistis antara dominasi Argentina dan ketidakmampuan Kroasia menciptakan peluang berkualitas.

Terakhir, pada pertandingan final Euro 2024 antara Spanyol dan Inggris yang berakhir dengan kemenangan Spanyol 2–1, model ini menghasilkan nilai xG sebesar 1,63 untuk Spanyol dan 0,67 untuk Inggris. Estimasi ini mendekati angka dari beberapa penyedia statistik. Xgscore.io mencatat nilai sebesar 1,90 (Spanyol) dan 0,63 (Inggris), sementara FBref dan Opta memberikan hasil serupa, yakni 1,9 dan 0,5 (FBref), serta 1,953 dan 0,527 (Opta). Sayangnya, data dari PFF untuk pertandingan ini tidak tersedia. Perbandingan ini menunjukkan bahwa model yang dikembangkan mampu memberikan prediksi yang sejalan dengan tren umum yang tercermin dalam data statistik publik, sehingga memperkuat validitas model sebagai alat analisis performa pertandingan sepak bola tingkat tinggi.

#### **4.7 Keterbatasan Penelitian**

Penelitian ini memiliki beberapa keterbatasan yang perlu diperhatikan. Pertama, keterbatasan utama terletak pada kelengkapan dan cakupan data. Dataset yang digunakan berasal dari sumber open-data StatsBomb yang hanya mencakup liga dan turnamen tertentu, seperti Liga Inggris, La Liga, dan Piala Dunia. Hal ini membatasi kemampuan generalisasi model terhadap kompetisi lain yang memiliki karakteristik permainan berbeda, baik dari segi level kompetisi, gaya bermain, maupun kualitas pemain. Selain itu, meskipun data StatsBomb dikenal kaya akan detail teknis, sejumlah fitur krusial dalam analisis xG seperti posisi kiper atau

intensitas tekanan dari pemain bertahan tidak selalu tersedia atau hanya tersedia dalam jumlah terbatas (misalnya *freeze frame* data). Kekosongan ini dapat mengurangi kemampuan model dalam merepresentasikan konteks situasional dari suatu tembakan secara menyeluruh.

Kedua, keberlakuan model yang dibangun secara spesifik pada data dari satu liga atau turnamen tertentu dapat membatasi performanya ketika diterapkan pada kompetisi lain. Perbedaan gaya bermain antar liga, taktik dominan, tingkat kemampuan teknis pemain, serta kondisi permainan yang kontekstual dapat memengaruhi performa model secara signifikan. Dengan demikian, validitas eksternal dari model ini masih perlu diuji secara lebih luas sebelum dapat digunakan secara general.

Ketiga, keterbatasan dalam pemahaman domain atau domain knowledge turut menjadi tantangan dalam eksplorasi fitur. Tanpa pemahaman mendalam mengenai peran spesifik pemain, strategi taktis, dan pola permainan, terdapat kemungkinan bahwa beberapa fitur bersifat terlalu dangkal atau bahkan mengarah pada interpretasi yang menyesatkan. Hal ini menunjukkan pentingnya kolaborasi antara peneliti data dan praktisi atau analis sepak bola untuk memperkaya proses *feature engineering* dan interpretasi model.



## BAB V

### PENUTUP

#### 5.1 Kesimpulan

Penelitian ini melakukan penerapan model Light Gradient Boosting Machine (LGBM) untuk melakukan perhitungan metrik Expected Goals (xG) dalam konteks analisis performa tembakan pada pertandingan sepak bola. Data yang digunakan merupakan data tembakan yang telah melalui proses pembersihan dan praproses, serta dilakukan rekayasa fitur berbasis konteks spasial, temporal, dan teknis. Selanjutnya, dilakukan pelatihan model menggunakan algoritma LGBM, evaluasi performa dengan metrik seperti *Brier Score* dan ROC AUC, serta analisis interpretasi model menggunakan visualisasi SHAP dan distribusi nilai prediksi xG. Berdasarkan hasil pembahasan penerapan model LGBM untuk prediksi xG dalam pertandingan sepak bola, dapat ditarik kesimpulan:

- a. Penelitian ini menerapkan algoritma LightGBM untuk meningkatkan akurasi dan efisiensi dalam perhitungan xG dalam analisis sepak bola menggunakan *open-data* dari StatsBomb. Proses dimulai dengan tahapan *feature engineering* yang mencakup variabel-variabel penting seperti *distance\_to\_goal*, *angle\_to\_goal*, dan *type\_before* yang berperan signifikan dalam menentukan probabilitas terjadinya gol. Untuk meningkatkan kalibrasi prediksi probabilistik model, digunakan metode *CalibratedClassifierCV* dengan teknik *isotonic regression* dan *3-fold cross-validation*. Parameter model disetel secara spesifik guna mengoptimalkan performa, antara lain *boosting\_type = gbdt*, *num\_leaves*

= 15, *max\_depth* = 84, *learning\_rate* = 0.1, *n\_estimators* = 100, serta regulasi melalui *reg\_alpha* = 0.513 dan *reg\_lambda* = 0.971. Model juga dirancang dengan kontrol terhadap *overfitting* melalui *min\_child\_samples* = 146, *subsample* = 1.0, dan *colsample\_bytree* = 1.0. Hasil konfigurasi ini menunjukkan bahwa LightGBM dapat digunakan secara efisien dan akurat untuk memodelkan metrik xG dalam domain sepak bola, dengan mempertimbangkan kontribusi fitur-fitur relevan dan teknik kalibrasi prediktif.

- b. Performa algoritma LightGBM dalam perhitungan xG dievaluasi menggunakan dua metrik utama, yaitu *Area Under Curve* (AUC) dan Brier Score. Berdasarkan hasil evaluasi, model LightGBM menunjukkan nilai Brier Score sebesar 0.0663, yang mengindikasikan tingkat kalibrasi probabilistik yang sangat baik dan kesalahan prediksi yang rendah. Selain itu, nilai ROC AUC mencapai 0.9134, yang menunjukkan bahwa model memiliki kemampuan diskriminatif yang sangat tinggi dalam membedakan antara peluang yang berujung pada gol dan yang tidak. Jika dibandingkan dengan model-model lain yang digunakan dalam studi ini, LightGBM menunjukkan performa yang relatif unggul berdasarkan kedua metrik evaluasi tersebut. Validasi tambahan terhadap hasil prediksi juga telah dilakukan pada data pertandingan nyata sebagaimana dijelaskan pada bagian interpretasi hasil. Model ini mampu menghasilkan estimasi xG yang selaras dengan konteks situasi pertandingan, sehingga menunjukkan potensi yang baik dalam penerapan nyata untuk analisis sepak bola.

## 5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, berikut beberapa saran yang dapat dijadikan dan dipertimbangkan untuk penelitian selanjutnya:

- a. Pada penelitian ini hanya menggunakan satu algoritma pembelajaran mesin, yaitu LightGBM, karena mempertimbangkan efisiensi dan kompleksitas model. Untuk penelitian selanjutnya disarankan untuk mengeksplorasi dan membandingkan beberapa algoritma lain, seperti XGBoost, CatBoost, atau model berbasis neural network, guna memperoleh perspektif yang lebih komprehensif terkait performa dalam perhitungan xG.
- b. Fitur-fitur yang digunakan dalam model ini masih terbatas pada variabel yang tersedia dari open-data StatsBomb. Penelitian selanjutnya disarankan untuk melakukan pengayaan fitur, seperti memasukkan variabel taktis, posisi pemain bertahan lawan, atau kondisi pertandingan (misalnya skor sementara atau menit ke berapa dalam pertandingan), guna meningkatkan konteks spasial dan temporal dalam prediksi xG.
- c. Proses kalibrasi dilakukan menggunakan metode *isotonic* melalui *CalibratedClassifierCV*, namun belum dilakukan evaluasi terhadap metode kalibrasi alternatif. Penelitian selanjutnya dapat mempertimbangkan untuk membandingkan beberapa pendekatan kalibrasi, seperti *Platt scaling* atau *beta calibration*, untuk melihat dampaknya terhadap probabilitas prediktif model.

## DAFTAR PUSTAKA

- 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.
- Cavus, M., & Biecek, P. (2022). Explainable expected goal models for performance analysis in football analytics. 2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA), 1–9. doi:10.1109/DSAA54385.2022.10032440
- Davis, J., & Robberechts, P. (2024). Biases in expected goals models confound finishing ability. *arXiv preprint arXiv:2401.09940*.
- Decroos, T., & Davis, J. (2019). Interpretable prediction of goals in soccer. In *Proceedings of the AAAI-20 workshop on artificial intelligence in team sports*.
- Eggels, H., Van Elk, R., & Pechenizkiy, M. (2016). Explaining soccer match outcomes with goal scoring opportunities predictive analytics. *3rd Workshop on Machine Learning and Data Mining for Sports Analytics (MLSA 2016)*. CEUR-WS.org
- ElHabr, T. (2023). *xG model calibration*. Tony ElHabr – Tony’s Blog. Retrieved May 11, 2025, from <https://tonyelhabr.rbind.io/posts/opta-xg-model-calibration/>
- Malikov, D., & Kim, J. (2024). Beyond xG: A Dual Prediction Model for Analyzing Player Performance Through Expected and Actual Goals in European Soccer Leagues. *Applied Sciences*, 14(22), 10390.
- Méndez, M., Montero, C., & Núñez, M. (2023). Improving the expected goal value in football using multilayer perceptron networks. *Asian Conference on Intelligent Information and Database Systems* (pp. 352-363). Cham: Springer Nature Switzerland.

- 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>
- Mohammed, M. A., Kadhem, S. M., & Maisa'a, A. A. (2021). Insider attacker detection using light gradient boosting machine. *Tech-Knowledge*, 1(1), 67-76.
- 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>
- Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30.
- Ramos, S., Soares, J., Cembranel, S. S., Tavares, I., Foroozandeh, Z., Vale, Z., & Fernandes, R. (2021). Data mining techniques for electricity customer characterization. *Procedia Computer Science*, 186(3), 475–488. <https://doi.org/10.1016/j.procs.2021.04.168>