DATA ANALYSIS PROJECT

Team members:

- Veer Raj Singh - ET21BTHCS020
- Kuldeep Chamuah - ET21BTHCS071
- Sudarshan Biswakandali - ET21BTHCS056
- Debasish Phukan - ET21BTHCS022
- Nishant Prahar Saikia - ET21BTHCS078

LEAGUE OF LEGENDS (LOL)

1. **Two Teams, Five Players Each –** Each team competes to destroy the enemy's Nexus (main base).

2. **Over 160 Champions –** Players choose from a variety of characters, each with unique abilities.

3. **Different Roles – Players take on specific roles:**
   - Top (Tank/Duelist)
   - Jungle (Map Control)
   - Mid (Burst Damage)
   - ADC (Ranged Damage Dealer)
   - Support (Utility & Protection)

4. **Map: Summoner's Rift –** The main battlefield, divided into three lanes (Top, Mid, Bot) and a Jungle.

5. **Objectives Matter –** Teams fight for Dragon, Baron Nashor, and Turrets to gain an advantage.

IMPORTANCE OF PREDICTION MODELS IN LEAGUE OF LEGENDS

- **Strategic Decision-Making** – Helps players draft better and adapt strategies.
- **Performance Optimization** – Identifies key factors for winning (gold, objectives, vision).
- **Esports & Competitive Analysis** – Aids coaches and teams in counter-strategies.
- **Betting & Fantasy Leagues** – Improves accuracy in esports betting and fantasy leagues.
- **Game Development & Balance** – Assists in champion balancing and matchmaking.
- **AI Coaching & Training** – Provides insights for player improvement.
- **Viewer Engagement –** Enhances live streams with real-time predictions.
- **Machine Learning Applications** – A valuable case study for AI and data analytics.

OBJECTIVES OF THE STUDY

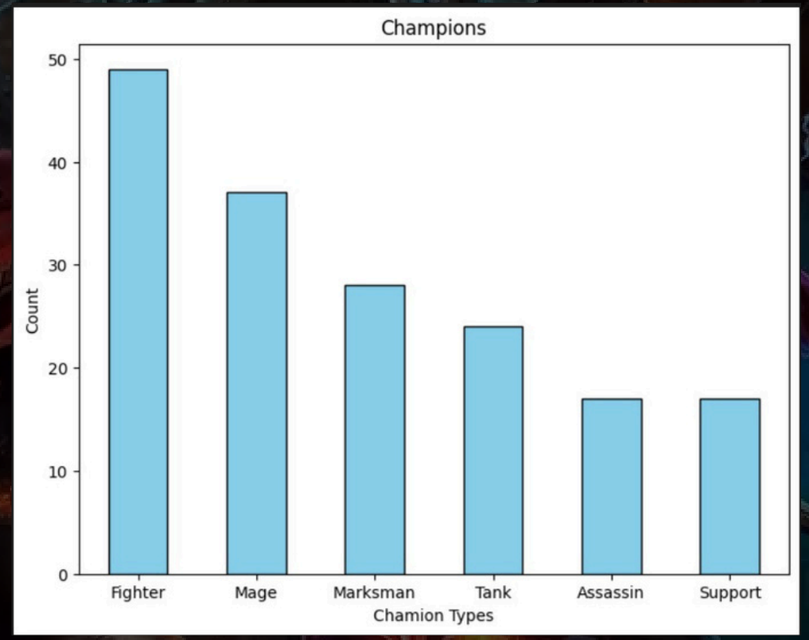
- **Objective:** Develop an AI model to predict Champion roles using in-game statistics.
- **Key Features:** Analyze parameters like difficulty, herotype, rangetype to identify Champion roles.
- **Model Optimization:** Test KNN, Decision Trees, Random Forest, and SVM for real-time accuracy.
- **Impact:**
  - **Matchmaking Balance:** Ensures fair team compositions by predicting player roles accurately.
  - **Gameplay Strategy:** Helps teams plan strategies based on expected role assignments.
  - **Champion Recommendations:** Suggests champions suited to a player's predicted role.

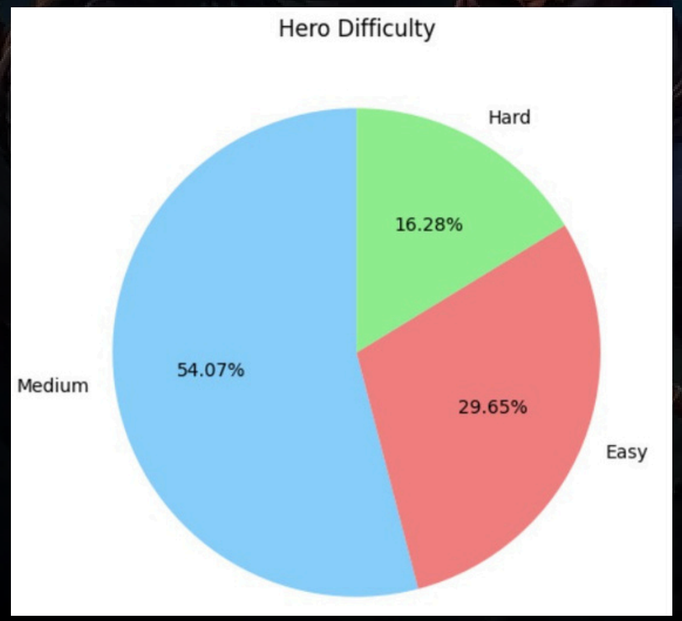
DATA PREPROCESSING FOR MODEL TRAINING

- **Handle Missing Values** – Fill missing data using mean, median, mode, or predictive methods to ensure consistency.
- **Data Cleaning** – Remove duplicates, fix errors, and standardize formats for reliable input.
- **Encoding Categorical Data** – Convert non-numerical features (e.g., champion names, roles) into numerical formats using one-hot or label encoding.

VISUALIZING DATA

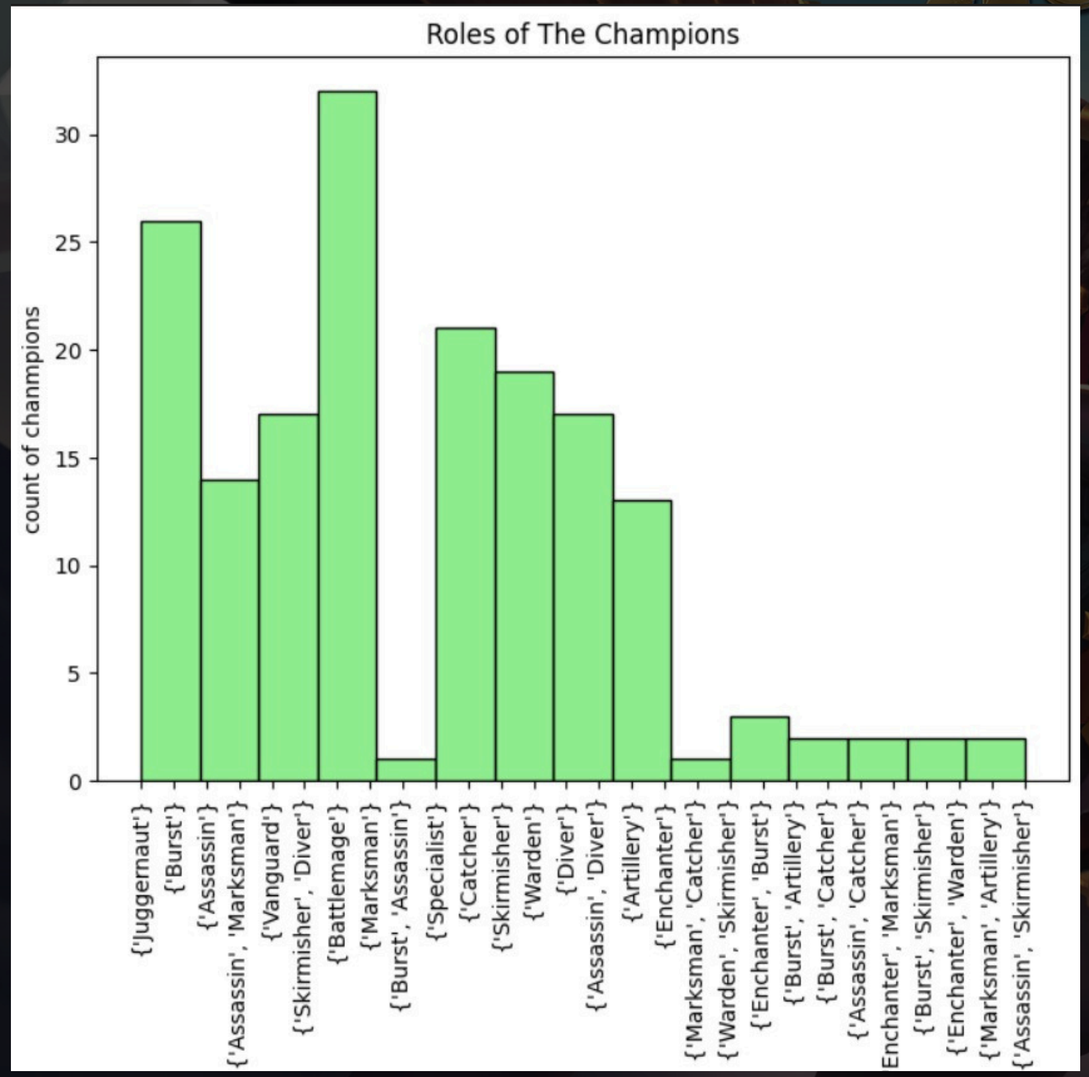
Bar Graph (Champion Types) – Show the count of different champion types (e.g., Tank, Assassin, Mage, Support).





Pie Chart (Hero Difficulty Levels) – Display the distribution of champions based on difficulty (e.g., Easy, Medium, Hard).

VISUALIZING DATA



Histogram (Champion Roles) – Represent the frequency of different roles (e.g., Top, Jungle, Mid, ADC, Support).

METHODOLOGY

1. **Data Loading & Preprocessing**
   - Loaded the dataset using Pandas and created a backup copy.
   - Handled missing values by filling with mode or dropping irrelevant columns.
   - Encoded categorical variables using Label Encoding.
   - Standardized numerical features for consistency.
2. **Exploratory Data Analysis (EDA)**
   - Analyzed data structure using .info() and .describe().
   - Visualized key attributes like game duration and gold earned using histograms and box plots.

METHODOLOGY

3. **Model Implementation**
    - Split data into 80% training and 20% testing.
    - Trained multiple models:
        - K-Nearest Neighbors (KNN)
        - Support Vector Machine (SVM)
        - Decision Tree
        - Random Forest
    - Evaluated models using accuracy, confusion matrix, and classification reports.
4. **Results & Insights**
    - SVM performed best due to its ability to handle non-linear decision boundaries, while KNN excelled by leveraging feature scaling and local pattern recognition

MODEL IMPLEMENTATION

Machine Learning Models Used:

- **K-Nearest Neighbors (KNN) –** Predicts a champion's role by finding the most similar champions based on attributes.
- **Support Vector Machine (SVM) –** Identifies decision boundaries between different champion roles using optimized hyperplanes.
- **Decision Tree –** Learns decision rules from champion data to classify their roles.
- **Random Forest –** Improves prediction accuracy by combining multiple decision trees.

MODEL EVALUATION

- **80-20% Train-Test Split** – Trained models on 80% of data, tested on 20%
- **Performance Metrics** – Evaluated models using accuracy, precision, and recall
- **Model Comparison** – SVM performed best due to its ability to handle non-linear decision boundaries, while KNN excelled by leveraging feature scaling and local pattern recognition

CHALLENGES

- **Handling Missing Values:** Some features had missing data that required imputation.
- **Feature Selection Complexity:** Identifying key factors affecting match outcomes was tricky.
- **Model Performance Variance:** Some models, like Logistic Regression, performed poorly due to complex data relationships.
- **Imbalanced Features:** Certain game attributes might have been underrepresented, affecting prediction accuracy.

CONCLUSION

- The analysis provided insights into key factors affecting match outcomes.
- SVM performed best due to its ability to handle non-linear decision boundaries, while KNN excelled by leveraging feature scaling and local pattern recognition.
- The project demonstrated the importance of data preprocessing and feature engineering in improving accuracy.

THANK YOU