Hackathon MLB

Juan Felipe Orozco Cortes

Enero 2025

1. Introducción

English Version:

1.1. 1. Introduction

Presentation of the Challenge and Context:

> "Hello everyone! In this video, I will show you my solution for **Challenge 5: 'Prospect Prediction'** of the **Google Cloud x MLB™ Hackathon**. The idea is to predict the future potential of prospects using MLB™ historical data, AI tools from Google Cloud, and in this case, integrating **Gemini** to generate final summaries."

Versión en Español:

1.2. 1. Introducción

Presentación del Desafío y Contexto:

> "¡Hola a todos! En este video les mostraré mi solución para el Desafío 5: 'Predicción de Prospectos' de la **Hackathon Google Cloud x MLB™**. La idea es predecir el potencial futuro de los prospectos, utilizando datos históricos de la MLB™, herramientas de IA en Google Cloud y, en este caso, integrando también **Gemini** para generar resúmenes finales."

2. Generación de Datos Históricos (1901 - Presente)

English Version:

2.1. 2. Generation of Historical Data (1901 - Present)

2.1.1. Obtaining the Schedule (mlb_game_data.json)

"First, I created a script that calls the official MLB™ API from **1901 to the current year**, fetching all regular season game data (`gamePk`, date, teams, stadium, etc.). This data was saved into a **mlb_game_data.json** file."

2.1.2. Data Enrichment (GUMBO Feed)

"With each `gamePk`, I queried the **GUMBO feed** (`https://statsapi.mlb.com/api/v1.1/game/\{gamePk\}/feed/live`) to extract detailed statistics:

- **Game-level stats**: runs, hits, errors, winning team, game duration.
- **Player-level stats (boxscore)**: hits, atBats, HR, ERA, etc.

Finally, I generated two CSVs:

1. **mlb_games_stats.csv** (game-level stats).
2. **mlb_games_boxscore.csv** (player-level stats).

This provided over **51,000 records** in the boxscore section and **1,111 records** in the stats section."

Versión en Español:

2.2. 2. Generación de Datos Históricos (1901 - Presente)

2.2.1. Obtención del Calendario (mlb_game_data.json)

"Primero, creé un script que llama la API oficial de la MLB™ desde el año **1901 hasta el año actual**, tomando datos de todos los juegos de temporada regular (`gamePk`, fecha, equipos, estadio, etc.). Esto se guardó en un archivo **mlb_game_data.json**."

2.2.2. Enriquecimiento de Datos (GUMBO Feed)

"Con cada `gamePk` consulté el **GUMBO feed** (`https://statsapi.mlb.com/api/v1.1/game/\{gamePk\}/feed/live`) para extraer estadísticas detalladas:

- **A nivel de juego**: carreras, hits, errores, equipo ganador, duración del juego.
- **A nivel de jugador (boxscore)**: hits, atBats, HR, ERA, etc.

Finalmente, generé dos CSV:

1. **mlb_games_stats.csv** (a nivel de juego).
2. **mlb_games_boxscore.csv** (a nivel de jugador).

Esto me dio más de **51,000 registros** en la parte de boxscore y **1,111 registros** en la parte de stats."

3. Carga a BigQuery y Preparación

English Version:

3.1. 3. Upload to BigQuery and Preparation

3.1.1. CSV Upload to BigQuery

"To facilitate querying and transformations, I uploaded both CSVs to **BigQuery** using the `google.cloud.bigquery` library."

3.1.2. Reading from BigQuery

"Subsequently, another Python script handles:

- Connecting to BigQuery (with Google Cloud credentials configured).
- Downloading data from `games_boxscore` and `games_stats`.
- Combining them with `pandas`, creating a unified table that calculates `current_avg` (hits / atBats) and defines `next_avg` as the target metric to predict."

3.1.3. Terminal Output

```
[INFO] Gemini configured with the provided API Key.
[INFO] Dataset 'mlb_dataset' already exists in the project 'maps-3d-439423'.
[INFO] Reading data from 'games_boxscore' from BigQuery...
[INFO] Data extracted from 'games_boxscore': 51151 rows, 14 columns.
[INFO] Reading data from 'games_stats' from BigQuery...
[INFO] Data extracted from 'games_stats': 1111 rows, 14 columns.
[INFO] Combined DataFrame: 50614 records after feature engineering.
[INFO] Final training size: 50614 rows, 6 features.
[INFO] Training RandomForest model...
[RESULT] MAE on test set: 0.1304
...
```

```
[INFO] Process completed. The model has been trained, the
    projection curve displayed, comparison performed, and
    Gemini text generated.
```

Listing 1: Terminal Output

"We observe that there are **50,614 final records** with **6 features**, and the model (**Random Forest**) produces an **MAE of 0.1304**."

Versión en Español:

3.2. 3. Carga a BigQuery y Preparación

3.2.1. Carga de CSV a BigQuery

"Para facilitar consultas y transformaciones, subí ambos CSV a **BigQuery** usando la biblioteca `google.cloud.bigquery`."

3.2.2. Lectura desde BigQuery

"Posteriormente, otro script en Python se encarga de:

- Conectarse a BigQuery (ya con las credenciales de Google Cloud configuradas).
- Descargar datos de `games_boxscore` y `games_stats`.
- Combinarlos con `pandas`, generando una tabla unificada que calcula `current_avg` (hits / atBats) y define `next_avg` como la métrica objetivo a predecir."

3.2.3. Resultado de la Terminal

```
[INFO] Gemini configurado con la API Key provista.
[INFO] El conjunto de datos 'mlb_dataset' ya existe en el
    proyecto 'maps-3d-439423'.
[INFO] Leyendo datos de 'games_boxscore' desde BigQuery...
[INFO] Datos extra dos de 'games_boxscore': 51151 filas, 14
     columnas.
[INFO] Leyendo datos de 'games_stats' desde BigQuery...
[INFO] Datos extra dos de 'games_stats': 1111 filas, 14
    columnas.
[INFO] DataFrame combinado: 50614 registros despu s de
    feature engineering.
[INFO] Tama o final para entrenamiento: 50614 filas, 6
    features.
[INFO] Entrenando modelo RandomForest...
[RESULT] MAE en conjunto de test: 0.1304
...
```

```
[INFO] Proceso completo. Se ha entrenado el modelo, mostrado
    la curva de proyecci n, realizado la comparaci n y
   generado un texto de Gemini.
```

Listing 2: Resultado de la Terminal

"Observamos que hay **50,614 registros finales** con **6 características**, y que el modelo (**Random Forest**) presenta un **MAE de 0.1304**."

4. Entrenamiento del Modelo & Despliegue de Escenarios

English Version:

4.1. 4. Training the Model & Scenario Deployment

4.1.1. RandomForestRegressor + Feature Engineering

"The model uses `current_avg`, `home_runs`, `away_runs`, `hits`, `homeRuns`, and `rbi`. After training (with an 80/20 train/test split), we obtain a Mean Absolute Error (MAE) of **0.13**, which, in the context of batting prediction, is a reasonable starting point."

4.1.2. Visualization of Evolution

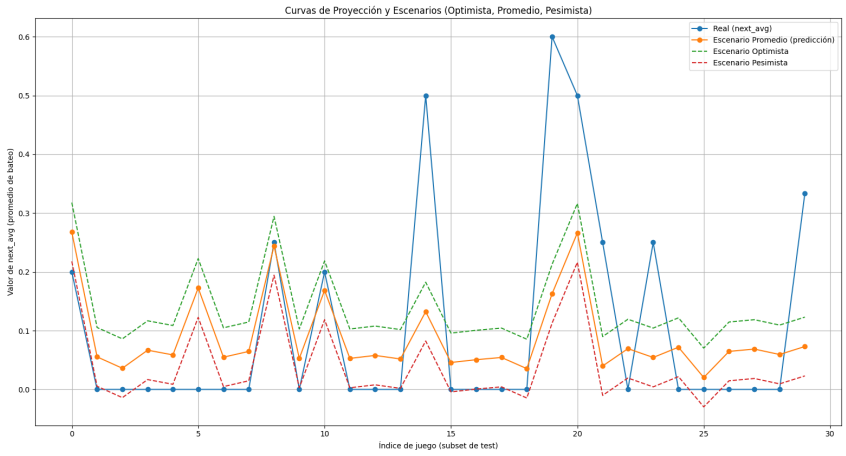

Figura 1: Projection Curve Comparison

.As you can see in the graph (**blue line**), this is the real value of `next_avg`. The **orange line** represents the baseline prediction of the model. Additionally, we generated:

- **Optimistic Scenario (green dashed)**: prediction + 0.05
- **Pessimistic Scenario (red dashed)**: prediction - 0.05.

This helps us illustrate a probable range of future performance, taking into account some uncertainty."

Versión en Español:

4.2. 4. Entrenamiento del Modelo & Despliegue de Escenarios

4.2.1. RandomForestRegressor + Feature Engineering

"El modelo utiliza `current_avg`, `home_runs`, `away_runs`, `hits`, `homeRuns` y `rbi`. Tras el entrenamiento (con una división train/test 80/20), obtenemos un error absoluto medio de **0.13**, lo cual, en el contexto de la predicción de bateo, es un punto de partida razonable."

4.2.2. Visualización de la Evolución

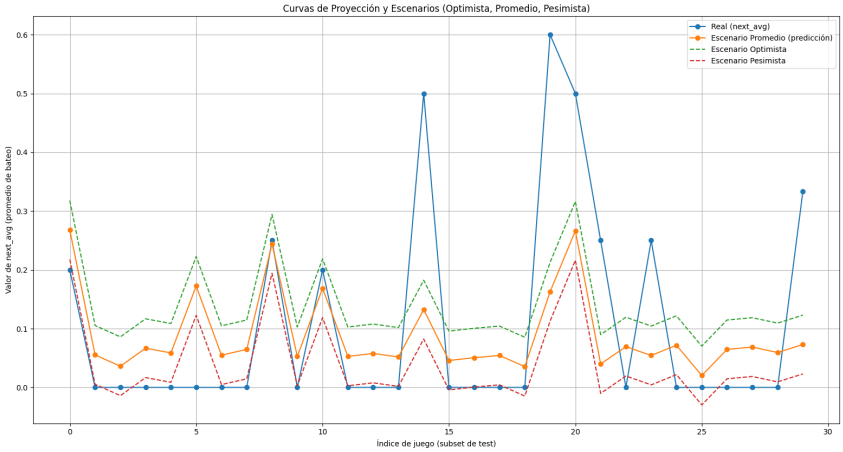

Figura 2: Comparación de la Curva de Proyección

"Como pueden ver en la gráfica (**línea azul**), ese es el valor real de `next_avg`. La **línea naranja** representa la predicción base del modelo. Aparte, generamos:

- **Escenario Optimista (verde punteado)**: la predicción + 0.05
- **Escenario Pesimista (rojo punteado)**: la predicción - 0.05.

Esto nos ayuda a ilustrar un rango probable del rendimiento futuro, teniendo en cuenta cierta incertidumbre."

5. Comparación de Prospectos con Jugadores Históricos

English Version:

5.1. 5. Comparison of Prospects with Historical Players

5.1.1. Prospect Similarity

Çhallenge #5 also asked for a comparison of prospects with historical players to see who they resemble. Our script generated the following output for prospect **114879** in the year **1901**:"

```
[INFO] Players most similar to prospect 114879 in 1901:
 player_id   player_name   year_of_game  current_avg
    similarity
      117563   Bob Lawson          1901         0.0
         0.315772
      114052  Harry Felix          1901         0.0
         0.315772
      112836 Bill Cristall        1901         0.0
         0.315772
      124541    Bob Wood          1901         0.0
         0.315772
      116042 Harry Hogan          1901         0.0
         0.315772
```

Listing 3: Prospect Similarity

"This suggests that these players share similar statistical characteristics with prospect ID **114879** based on our model. We used a Euclidean distance on the same features as the RandomForest."

5.1.2. Integration with Gemini

.ªdditionally, at the end of the process, we used Gemini to generate a summary text interpreting the results: explaining the MAE, the `next_avg` metric, and how the historical player comparisons were established."

Versión en Español:

5.2. 5. Comparación de Prospectos con Jugadores Históricos

5.2.1. Similitud de Prospectos

"El desafío #5 pedía también la comparación de prospectos con jugadores históricos para ver a quién se parece más. Nuestro script generó la siguiente salida para el prospecto **114879** en el año **1901**:"

```
[INFO] Jugadores m s parecidos al prospecto 114879 en el
    a o 1901:
 player_id   player_name   year_of_game  current_avg
    similarity
      117563   Bob Lawson          1901         0.0
          0.315772
      114052  Harry Felix          1901         0.0
          0.315772
      112836 Bill Cristall        1901         0.0
          0.315772
      124541    Bob Wood          1901         0.0
          0.315772
      116042 Harry Hogan          1901         0.0
          0.315772
```

Listing 4: Similitud de Prospectos

"Esto sugiere que estos jugadores tienen características estadísticas similares al prospecto con ID **114879** según nuestro modelo. Usamos una distancia euclidiana en las mismas **features** que el RandomForest."

5.2.2. Integración con Gemini

"Adicionalmente, al final del proceso, usamos Gemini para generar un texto resumen interpretando los resultados: explicando el MAE, la métrica `next_avg` y cómo se establecen las similitudes con los jugadores antiguos de la MLB™."

6. Conclusiones y Próximos Pasos

English Version:

6.1. 6. Conclusions and Next Steps

6.1.1. Conclusions

.Our solution demonstrates how to build a complete pipeline:

- Data collection from 1901 using the MLB™ APIs.
- Transformation and storage in BigQuery.
- Training a model to predict the 'next batting average'.
- Visualization of scenarios (optimistic, average, pessimistic).
- Historical comparison for prospects."

6.1.2. Future Improvements

"The model could be refined with more features, integrating historical WAR, or using more advanced methodologies (e.g., an RNN that captures the temporal order of games). Confidence intervals could also be used instead of a fixed delta for optimism/pessimism."

6.1.3. Closing

.And with that, we conclude our presentation for **Challenge #5: 'Prospect Prediction'** in the **Google Cloud x MLB™ Hackathon**. Thank you so much for watching, and I hope this is inspiring for your own baseball and AI projects!"

Versión en Español:

6.2. 6. Conclusiones y Próximos Pasos

6.2.1. Conclusiones

"Nuestra solución demuestra cómo construir un pipeline completo:

- Recolección de datos históricos desde 1901, con las APIs de MLB™.
- Transformación y almacenaje en BigQuery.
- Entrenamiento de un modelo para predecir el 'próximo promedio de bateo'.
- Visualización de escenarios (optimista, promedio, pesimista).
- Comparación histórica para prospectos.

"

6.2.2. Futuras Mejoras

"Se podría refinar el modelo con más características, integrar WAR histórico, o emplear metodologías más avanzadas (p. ej., un RNN que capte el orden temporal de los juegos). También se podrían usar intervalos de confianza estadísticos más precisos en lugar de sumar/restar un delta fijo."

6.2.3. Cierre

> "Y con esto concluimos nuestra presentación para el Desafío #5: 'Prospect Prediction' en la **Google Cloud x MLB™ Hackathon**. ¡Muchas gracias por ver este video y espero que les resulte inspirador para sus propios proyectos de béisbol e inteligencia artificial!"