

Informe de análisis de partidas de League of Legends

Brais Iglesias Otero

Índice

1. Introducción.
2. Objetivo del análisis.
3. Obtención del dataset.
4. Esquema de partidas.
5. Herramientas.
6. Diagrama en estrella.
7. Validación.
8. Creación de dimensiones.
9. Creación de la tabla de hechos y volcado de datos.
10. Consultas.
11. Conclusiones técnicas.
12. Conclusiones generales.

Índice de tablas

- Tabla 1: Descripción de dimensiones.
- Tabla 2: Verificación de registros.
- Tabla 3: Procesado de columnas.
- Tabla 4: Descripción de los Dashboards.
- Tabla 5: Código entregado en el proyecto.

Índice de figuras

- Figura 1: Mapa de la partida.
- Figura 2: Diagrama de estrella.
- Figura 3: Extracto de uno de los scripts creados *participantScript.R*
- Figura 4: Creación de la dimensión Game.
- Figura 5: Creación de la dimensión Campeón.
- Figura 6: Creación de la dimensión Item
- Figura 7: Procesado en Pentaho.
- Figura 8: Ejemplo de creación de una dimensión.
- Figura 9: Análisis del rol de soporte.
- Figura 10: Análisis de recursos.
- Figura 11: Análisis de KDA.

1. Introducción

La industria de los videojuegos es actualmente el sector de entretenimiento que mas masa de capital mueve con mucha diferencia al resto, esto se debe gracias a la evolución en la calidad de los mismos y el interés que muestran las nuevas generaciones con respecto a formas de entretenimiento más convencionales. Uno de los videojuegos que contribuye a engrandecer la industria es el conocido *League of Legends* [1]. Este juego es un *free-to-play* que tiene actualmente cerca de 30 Millones de jugadores en todo el mundo y un tejido competitivo a la altura de los deportes convencionales que mueve millones de euros al año.

La idea del juego es simple, dos equipos, formados por cinco jugadores cada uno, se enfrentan entre sí en un mapa que se reparte a la mitad (cada mitad pertenece inicialmente a un equipo) con el objetivo de destruir la base enemiga. Cada jugador deberá escoger un campeón, que viene siendo un personaje ficticio del juego, de los 130 existentes, teniendo cada uno unas habilidades y trucos distintos.

Este ecosistema deriva en una infinidad de alternativas y posibilidades que junto la estrategia de juego, la mecánica individual y la toma de decisiones lo convierten en uno de los juegos más interesantes del panorama, ya no solo actual, si no de toda la historia de los videojuegos. Claro ejemplo de esto es la industria de competición profesional que hay alrededor del mismo y que mueve cientos de miles de fans.

2. Objetivo del análisis

Se plantean varios objetivos en la realización de este trabajo. En primer lugar se pretende construir un *datawarehouse* que albergue información y métricas de partidas de League of Legends y en segundo lugar realizar un análisis de los datos disponibles para sacar conclusiones acerca del contenido de los mismos. Es importante destacar que la granularidad a la que se va a trabajar en este análisis es a nivel de **campeón**.

3. Obtención del dataset

El dataset seleccionado se recupera de la plataforma *Kaggle* mediante el siguiente enlace:

<https://www.kaggle.com/datasets/paololol/league-of-legends-ranked-matches>

Consta de 195MB de datos de partidas de League of Legends repartidos en los siguientes archivos:

champs.csv – datos de los campeones.

matches.csv – metadatos de la partida.

participants.csv – datos del jugador.

stats1.csv – métricas del jugador en la partida.

stats2.csv – continuación de stats1. No se utiliza

teamstats.csv – datos del equipo.

4. Esquema de partida

Antes de proceder con el proyecto conviene comentar previamente unos conceptos sobre el juego para que sea comprensible el trabajo realizado. Se explicará pues como es el funcionamiento habitual de una partida de *League Of Legends*.

Antes de comenzar a jugar, cada uno de los diez jugadores (5 de cada equipo) deberá escoger un campeón para jugar. Una vez empiece la partida, los jugadores se colocan en sus respectivas líneas y comenzarán a luchar entre ellos con el objetivo de matar al rival (cuando un jugador muere tiene que esperar un tiempo hasta revivir, este tiempo va aumentando conforme se alarga la partida) y destruir torres hasta llegar al nexo enemigo. El tiempo promedio de duración de una partida es de 25 minutos. A continuación se muestra una imagen del mapa para clarificar los conceptos, las partes azules corresponden al equipo azul así como las rojas al rojo.



Figura 1: Mapa de la partida

Recordando que la granularidad es a nivel de campeón, de estas partidas resultaría interesante estudiar métricas como cuál es el campeón que más mata, el que más muere, el que más *pentakills* (matar a los 5 enemigos en una misma pelea) consigue, el que más oro consigue ... etc. Posteriormente se profundizará en este estudio.

5. Herramientas

Para el desarrollo del proyecto se han empleado las siguientes herramientas:

- **Rstudio** [2]: para la gestión de columnas y joins de los csv proporcionados.
- **Pentaho** [3]: para el mapeo y el volcado de los datos al datawarehouse.
- **Tableau** [4]: para la obtención de la representación visual de los datos.

6. Diagrama de estrella

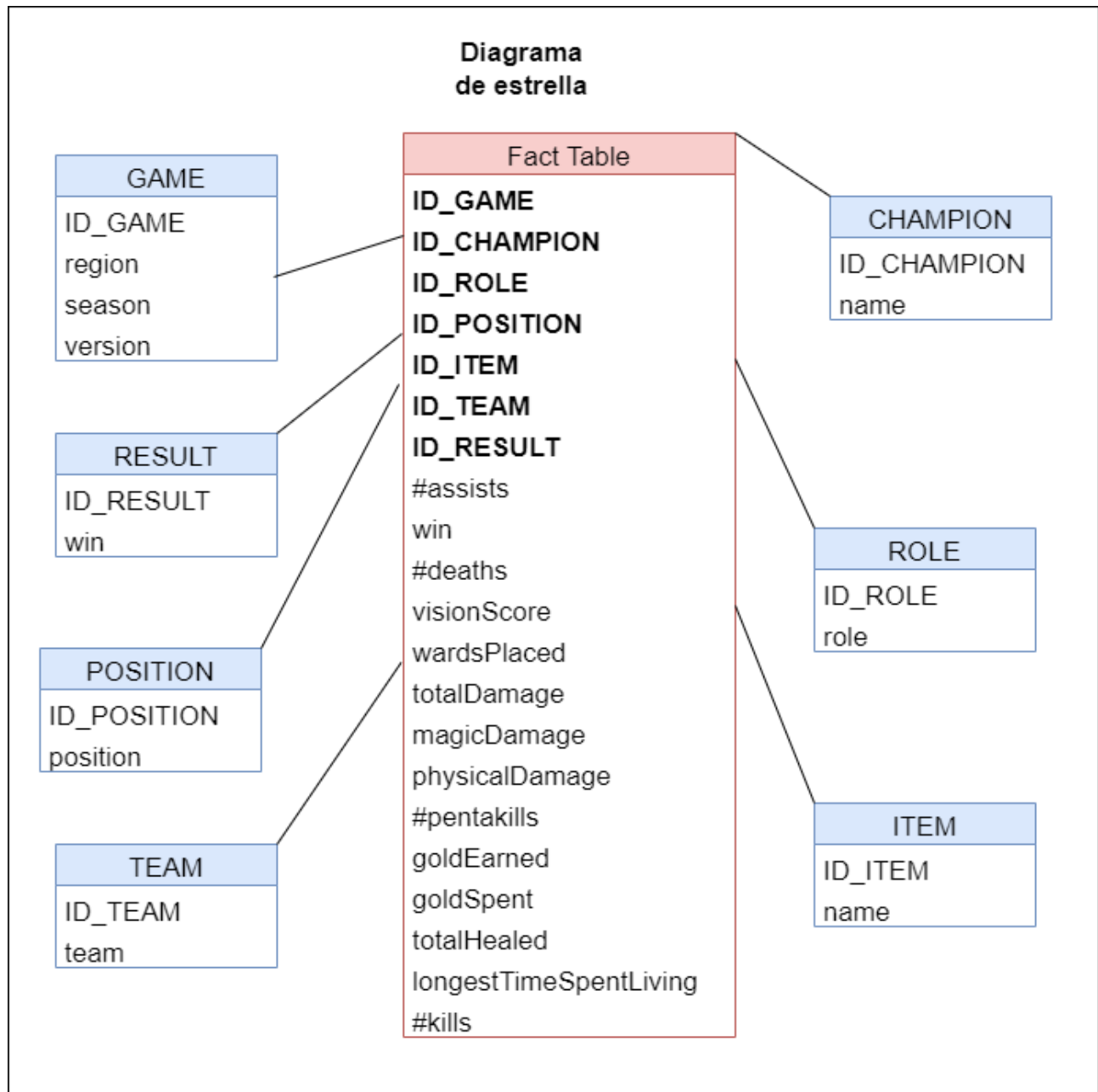


Figura 2: Diagrama de estrella

A continuación se realizará una explicación acerca tanto de las dimensiones como de las métricas escogidas para este trabajo.

Dimensiones		
Nombre	Descripción	Valores
Champion	Esta dimensión es clave debido a la granularidad de campeón de nuestro análisis y representa el nombre de un campeón concreto escogido en la partida.	Nombre del campeón. Ejemplos: Akali, Aatrox, Thresh... Hay 138 campeones.
Role	Representa el rol que el jugador físico tiene para la partida.	NONE SOLO

		DUO DUO_CARRY DUO_SUPPORT
Position	Representa la línea del mapa en la que jugará el jugador.	TOP JUNGLE MID BOT
Team	Representa el equipo al que pertenece un jugador en una partida concreta.	RED BLUE
Item	Representa unos objetos que pueden ser comprados por el jugador y otorgan habilidades concretas y que se identifican por su nombre.	Hay 108 items.
Game	Representa metadatos relacionados con una partida concreta.	Región: EUW,NA,TR,EUN Season: season a la que pertenece la partida. Versión: número de versión de la partida.
Result	Dimesión que representa el resultado de la partida para un jugador	WIN {0,1}

Tabla 1: Descripción de dimensiones.

7. Validación

La validación de los datos consiste en la verificación de su calidad y el descarte de registros en caso de no ser válidos. En este caso, no se encontraron problemas de datos mal formados o nulos, sin embargo se recortaron los datos disponibles para la tabla de hechos.

Registros de métricas inicial	Registros de métricas recortado
99 999 registros	10 000 registros

Tabla 2: Verificación de registros.

Con respecto a las columnas se realiza un procesado simple con la herramienta R que consiste en la eliminación de las columnas que no son relevantes para este análisis, se eliminan un total de 34 columnas del archivo *stats1.csv*. Después de esta eliminación se realiza un *join* con un archivo de *participantesV2.csv* previamente procesado, para generar el archivo final que se volcará al *data warehouse* con pentaho.

Archivo original	Archivo procesado	Cambios
<i>stats1.csv</i>	<i>stats.csv</i>	Se eliminan las columnas innecesarias (34), se hace un <i>join</i> con <i>participantesV2.csv</i> y se extraen los 10000 primeros registros.
<i>participantes.csv</i>	<i>participantesV2.csv</i>	Se hace un <i>join</i> con <i>teamstats.csv</i> para obtener para cada participante la información del equipo y

		se eliminan columnas innecesarias (11).
--	--	---

Tabla 3: Procesado de columnas.

Todos los scripts creados para estos procesos se adjuntarán con la documentación final.

```
setwd("C:/Users/USUARIO/Desktop/Master/primer/primer cuatrimestre/SI/Proyecto/DataBase")

file<-read.csv(file = "teamstats.csv",sep=",")

fileParticipant<-read.csv(file = "participants.csv",sep=",")
#Elimino columnas sobrantes
file<-dplyr::select(file,c(-5,-6,-7,-8,-10,-11,-12))

#Extraigo los valores únicos de cada columna para cada dimensión
particiPantsComplete<-merge(file,fileParticipant,by.x =1,by.y=2)

#Elimino los que no pertenecen a cada equipo (1-5 equipo 100,6-10 equipo 200)
particiPantsComplete<-particiPantsComplete[(particiPantsComplete$teamid==100&particiPantsComplete$player<=5|
particiPantsComplete$teamid==200&particiPantsComplete$player>5),]

towrite<-dplyr::select(particiPantsComplete,c(-6,-8,-10,-11))
write.csv(towrite,file = "participantsV2.csv")
```

Figura 3: Extracto de uno de los scripts creados *participantScript.R*

8. Creación de las dimensiones

En este apartado se describirá el proceso de creación de las tablas de dimensiones.

Game

Los datos de esta dimensión están contenidos en el archivo *matchV2.csv*, que es una versión generada del archivo original *match.csv* en la que se balancea más las regiones ya que en el archivo original mayoritariamente hay registros de la región EUW. Este balanceo se realizó en un script en R aparte. Una vez obtenido esto se crea una transformación en la herramienta pentaho que lee los datos del csv, filtra las columnas que interesan, las ordena por el ID y las vuelca en la base de datos en la tabla *game*.

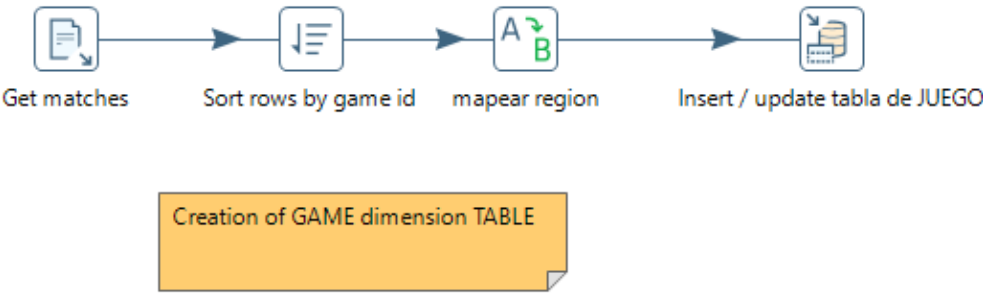


Figura 4: Creación de la dimensión Game

Champion

Esta dimensión se vuelca de forma directa en el datawarehouse ya que el archivo *champs.csv* contiene los datos que nos interesan (id y nombre) por lo que únicamente se ha de crear una transformación en Pentaho que lea el archivo y lo vuelque en la tabla *champion* del *data warehouse*.

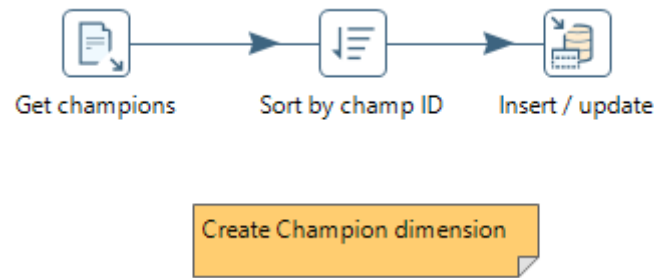


Figura 5: Creación de la dimensión Campeon

Role

En este caso, en la tabla *role* del *data warehouse* se ha creado una entrada para cada rol asignándole un id. Posteriormente en el archivo correspondiente a la fact table se ha mapeado el antiguo valor con el nuevo ID.

Postion

Se sigue el mismo procedimiento que para *role*.

Team, Result

En este caso, se utiliza el id que traen los datasets y se le asigna un nombre en la tabla del data warehouse.

Item

Como la información de estas entidades no se disponía en el dataset original de Kaggle, se realiza la descarga del API de desarrolladores de League of Legends de un archivo que contenía esta información y se procesa mediante un script de R cuyo resultado de salida es un archivo *items.csv* con el id, el nombre y el costo del item. Estos datos se vuelcan posteriormente mediante una transformación en Pentaho a la tabla *item*.

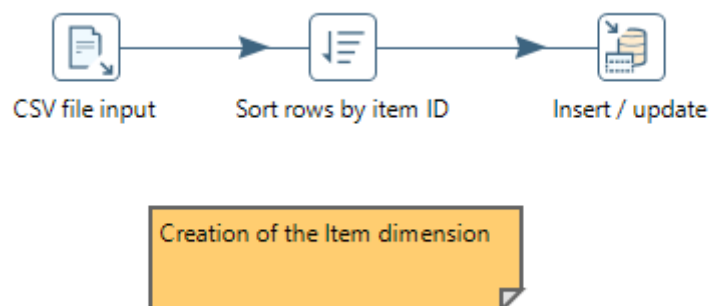


Figura 6: Creación de la dimensión Item

9. Creación de la tabla de hechos y volcado de datos

Para la creación de la tabla de hechos, al no disponer de métricas calculadas si no que son todas directas del dataset disponible la única intervención requerida fue mapear las columnas de dimensiones su id correspondiente.

Descripción del proceso ETL

El proceso de volcado, como se puede ver en la siguiente imagen se realizó con comprobaciones a cada paso y sus respectivas alertas. La idea detrás de esto es que cada vez que se ejecute realizará los volcados que falten y de forma independiente entre dimensiones, de tal forma que si se ejecuta por primera vez y no hay ningún problema creará todas las dimensiones y la tabla de hechos, y si hay algún error en alguna dimensión o en la tabla de hechos, se informa del error mientras que se vuelcan el resto de tablas y en la siguiente ejecución se intentará volcar únicamente la que falta.

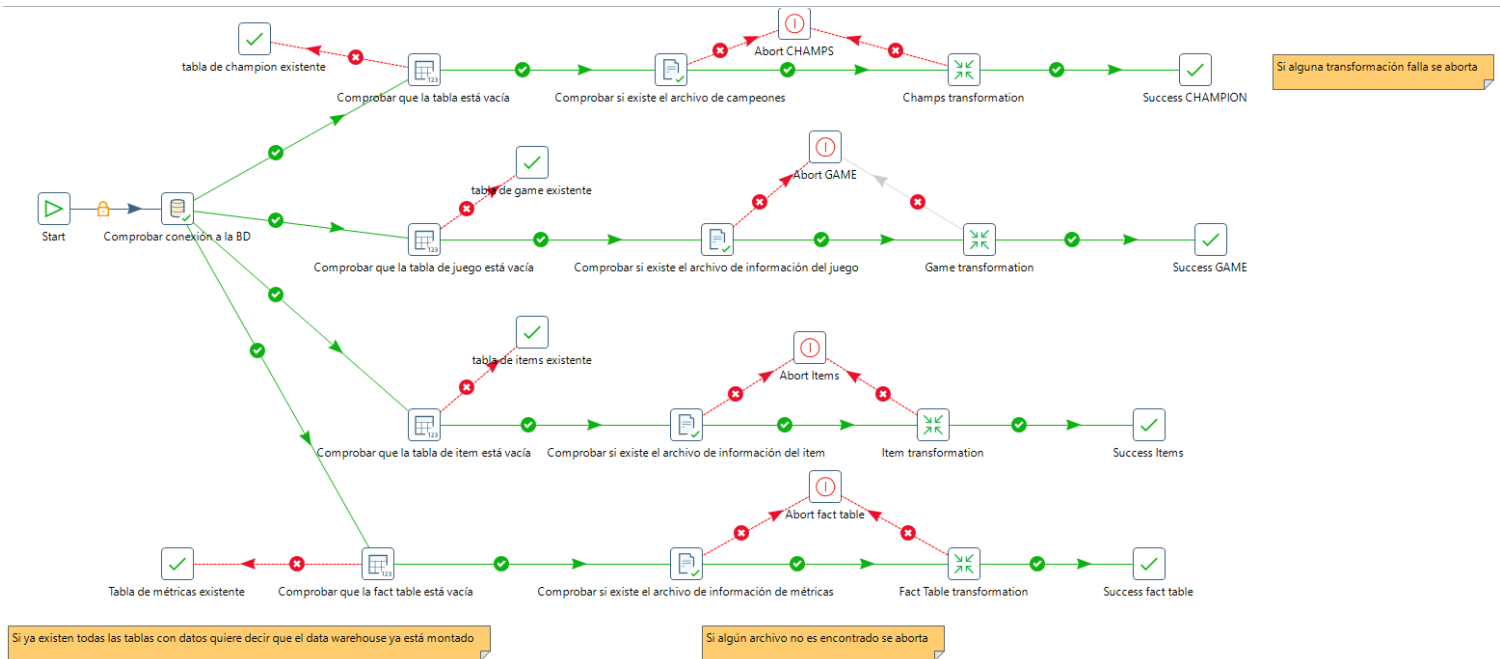


Figura 7: Procesado en Pentaho.

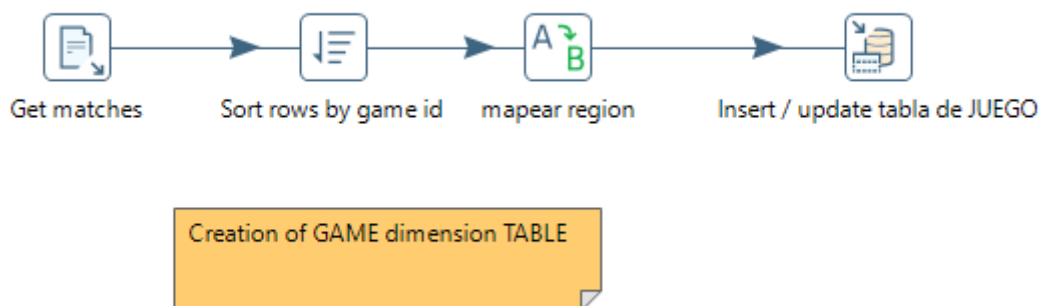


Figura 8: Ejemplo de creación de una dimensión.

10. Consultas

En este apartado se mostrarán las conclusiones y se describirán las consultas realizadas sobre los datos. Este tipo de información resultaría útil desde la perspectiva del jugador para ver aquellos campeones que son más fuertes según sus estadísticas y aprovecharlos, así como también desde la perspectiva de los desarrolladores para mantener balanceado el juego. Las consultas realizadas se agrupan en secciones del juego que se consideran relevantes. Los apartados son *KDA*, *Recursos* y *Soporte*. Resulta interesante destacar que las gráficas mostradas en este documento pierden la

interactividad, sin embargo, llegan a ofrecer más información de la que se puede apreciar en este documento:

Dashboard	Descripción
KDA	Hace referencia a la performance de un campeón en relación a kills, muertes, rachas conseguidas y daño realizado.
Recursos	Hace referencia a los recursos obtenidos por un campeón.
Soporte	Hace referencia a un análisis al rol de support

Tabla 4: Descripción de los Dashboards.

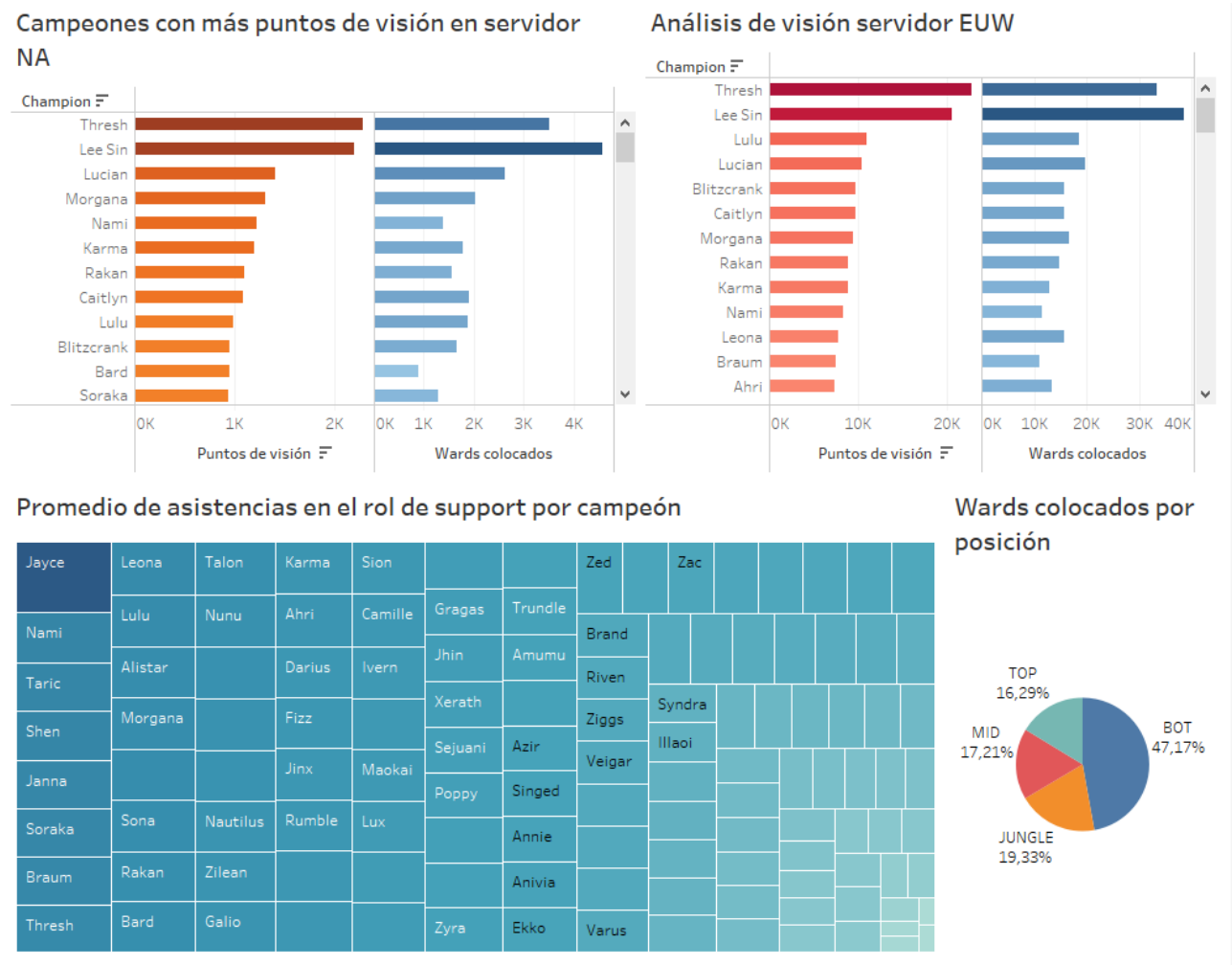
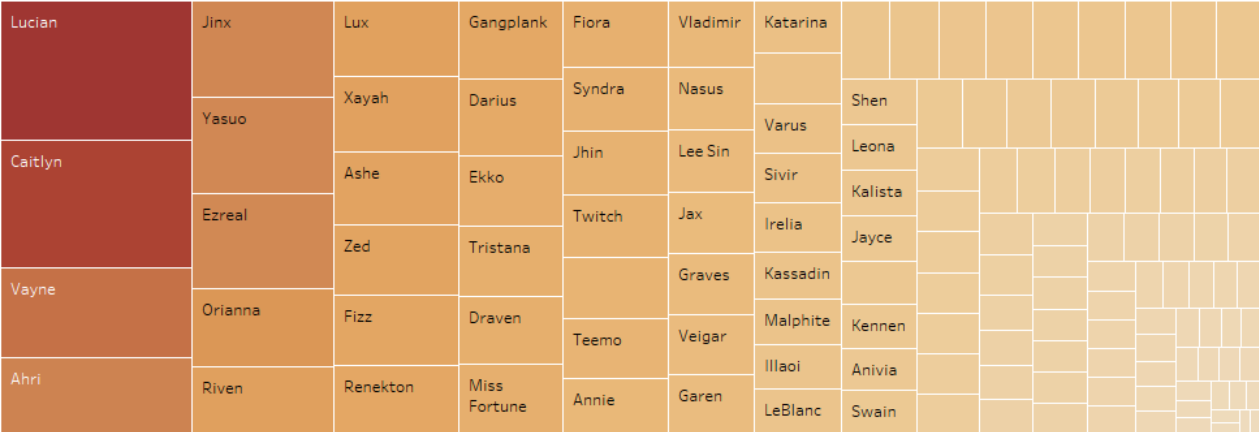


Figura 9: Análisis del rol de soporte.

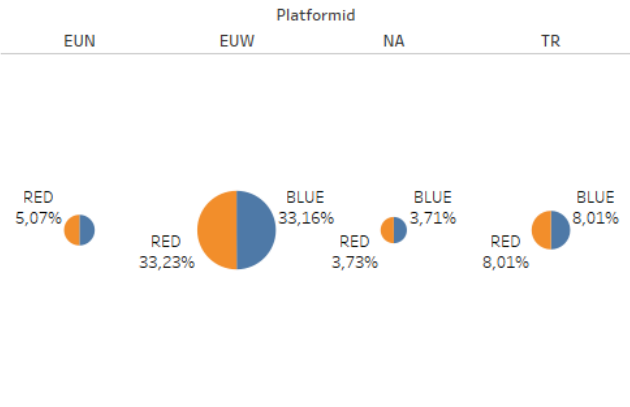
En este análisis podemos extraer conclusiones lógicas que indican que este aspecto del juego se desarrolla de forma correcta. Podemos ver como los campeones que más puntos de visión obtienen, tanto en europa como en norte américa son campeones que ocupan la posición de support, lo cuál es lógico ya que su misión es apoyar al equipo y debido a objetos específicos estos tienen la capacidad de colocar más *wards* (elementos del juego que dan puntos de visión) que el resto. Además vemos como la posición en la que más posiciones se colocan es la *Botlane*, lo cuál es lógico también ya que en esa línea juegan 4 jugadores mientras que en las demás sólo 2, seguida de la posición de

jungla. Quizás sería interesante proponer algún cambio para que la visión en el rol de la jungla se desmarqué un poco más de otras posiciones (MID y TOP) ya que es un elemento clave y podría dinamizar el juego. Por otro lado, con respecto al promedio de asistencias por campeón nos encontramos con una sorpresa en el top 1 ya que no es un campeón diseñado para jugar de support, mientras que los otros 20 campeones que le siguen en el orden en esa lista sí lo son. Esto podría servir para concluir que el campeón Jayce podría necesitar una mejora.

Mapa de farmeo



Oro conseguido en función del lado del mapa y región



Oro ganado por posición

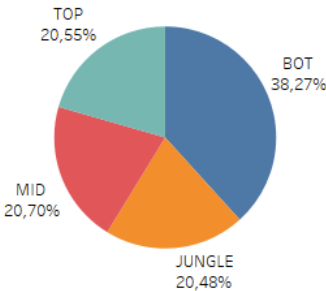


Figura 10: Análisis de recursos.

Con respecto a este análisis, en el mapa de farmeo podemos ver como los campeones mostrados en el top son campeones que suelen adoptar el trabajo de *carry*, por lo que es coherente que presenten más farmeo con el objetivo de obtener más oro y recursos. Por otro lado, observando la comparación de oro conseguido en función del lado del mapa podemos ver como en todas las regiones esta balanceado casi a la perfección, esto indica que está bien desarrollado el sistema y que un lado no ofrece excesivas ventajas con respecto a otro, sin embargo, con cada cambio aplicado al juego es conveniente revisar este tipo de gráficos para observar que ambos lados del mapa son igual de jugables y uno no presenta ventajas sobre el otro.

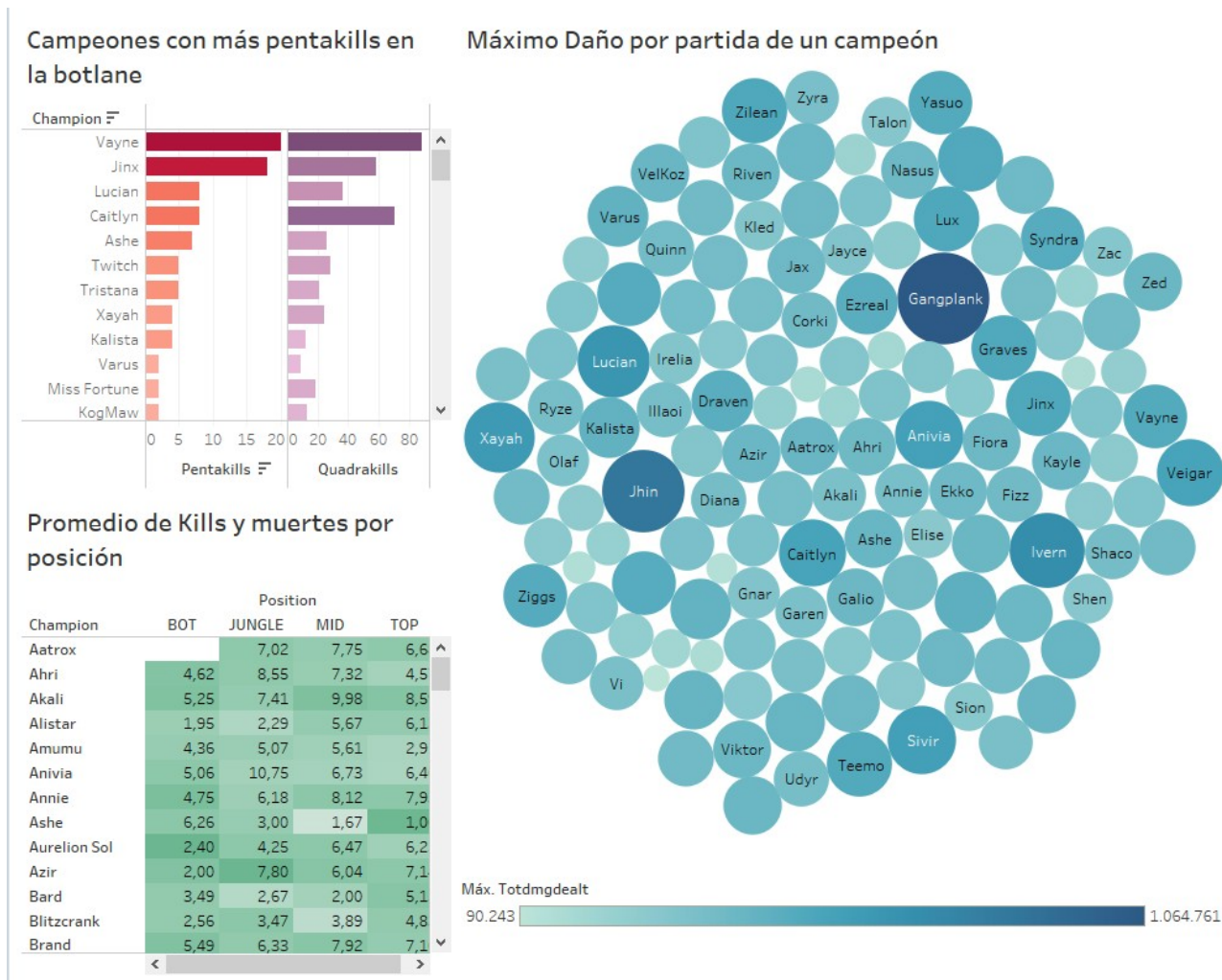


Figura 11: Análisis de KDA.

El último análisis, relacionado con la *performance* de cada campeón en la partida con respecto a las *Kills*, *Deaths* y *Assits* (KDA), nos ofrece resultados interesantes. Con respecto al máximo daño por partida podemos ver que el campeón *Gangplank* se desmarca con una diferencia con respecto a los demás bastante considerable, esto puede llevar a pensar que un ajuste en las estadísticas del campeón (daño de sus habilidades) puede ser llevada a cabo por los desarrolladores para reducir la brecha que hay con el resto de los campeones y los jugadores no abusen de él.

11. Conclusiones técnicas

Una vez realizado todo el proceso, desde el diseño y planteamiento del problema hasta la ejecución del mismo, se puede obtener una visión más concreta del esfuerzo que implica realizar este tipo de proyectos. Desde el punto de vista técnico no se consideró un trabajo de gran dificultad en el sentido de comprensión y dificultad de las herramientas o procesos realizados. Sin embargo, si cabe destacar que pese a ser procesos de sencilla comprensión requieren una delicadeza extrema a la hora de ser diseñados y ejecutados, la impresión obtenida en la realización de este proyecto es que la dificultad reside en realizar de forma correcta el diseño del análisis y tomar las decisiones adecuadas con respecto a ciertos procesados de los datos, para en el futuro poder construir consultas que sean útiles para resolver problemas de verdad.

12. Conclusiones generales

De forma más genérica, se considera extremadamente útil este tipo de procesos para obtener información acerca de de inteligencia de negocio. En este caso, con la información obtenida en las consultas se pueden extraer conclusiones como, qué campeón es el más fuerte, cuál es la diferencia con el resto, qué posición interesa menos a los jugadores, qué campeón no gusta a la comunidad Todas estas preguntas, y una infinidad más de ellas, sirven tanto a desarrolladores como a jugadores, a los primeros les puede ayudar a mantener el juego balanceado y a realizar *patches* con mejoras a campeones que se usen poco o al revés, y a los jugadores les puede ayudar a aprovecharse de los campeones que los datos indican como más fuertes entre otras muchas cosas.

Anexo

Código entregado:

Archivo	Descripción	Herramienta
participanteScript.R	Script para unir los archivos relativos al participante.	Rstudio
scriptFactTable.R	Script para el preprocesado de el archivo que contiene las métricas.	
scriptItems.R	Script para obtener la información de los Items.	
LeagueOfLegends_job.kjb	Job de Pentaho para la construcción del <i>data warehouse</i> .	Pentaho
T_Champs.ktr	Transformación en Pentaho para la dimesión de campeón.	
T_FactTable.ktr	Transformación en Pentaho para la tabla de métricas de campeón.	
T_Items.ktr	Transformación en Pentaho para la dimesión de Item.	
T_Juego.ktr	Transformación en Pentaho para la dimesión de Juego.	
T_ROLE.ktr	Transformación en Pentaho para la dimesión de Role.	

Tabla 5: Código entregado en el proyecto.

Nota: Para ejecutar los archivos .R se debe cambiar el directorio de trabajo actual.

Referencias

- [1] League of Legends, <https://www.leagueoflegends.com/es-es/> (último acceso 21/12/2022)
- [2] Rstudio, <https://posit.co/> (último acceso 21/12/2022)
- [3] Pentaho, <https://www.hitachivantara.com/> (último acceso 21/12/2022)
- [4] Tableau, <https://www.tableau.com/es-es> (último acceso 21/12/2022)