MEMORIA

Curso: The Bridge – Data Science Bootcamp

Proyecto: Exploratory Data Analysis

Alumno: Toby Nicholas KEMPE

Fecha entrega: 2022-11-06

Conseguir Datos:

Para este proyecto se tuvo que conseguir datos de dos formas (webscraping y API/librería), pero en tres pasos distintos y consecutivos. Primero, se tuvo que usar Selenium para crear un Beautiful Soup de los datos del ranking y su asociado game_id de la lista de mejores juegos del foro Board Game Geek. Segundo, se pudo pedir unos parámetros de cada juego usando el game_id conseguido en el primer paso. Estos parámetros se pidieron a través una librería compilado por los usuarios de GitHub, lcosmin y philsstein, que aprovecha el xmlapi y xmlapi2 de Board Game Geek. El BGGClient módulo de la librería boardgamegeek facilitó este paso de recopilación de datos. Por último, había unos datos que quería incluir en el estudio que estaban fuera del ámbito del API y la librería así tuve que apoyarme de nuevo en Selenium para crear un webscraping de las 1500 páginas individuales, una por cada juego de la lista de ranking, para conseguir los precios actuales de mercado además de la valoración de los usuarios del foro sobre el nivel de dependencia de idioma de cada juego.

Tratar Datos:

El 80% de los variables a incluir en análisis preliminar se pudo mapear directamente del soup resultante del Selenium/Beautiful Soup o del diccionario resultante de las peticiones por BGGClient. El problema siempre yace en el otro 20% donde se tuve que hacer, entre otras cosas:

- limpieza de espacios y signos de puntuación sobrantes,
- ajuste de tipo (string/int/dict, etc.) para que fuera utilizable,
- introducir protecciones en el código para que valores no existentes, no utilizables u erróneos no causaron fallos de bucles o corrupción de los datos deseados, y
- convertir datos cualitativos a cuantitativos para que su subsecuente análisis fuera más fácil.

Aunque, el análisis no se apoyó de una manera que los valores *None* afectaran promedios, sí fue importante hacer varias comprobaciones visuales de los datos para detectar duplicaciones. A pesar, de usar el método *getdummies* con las categorías y mecánicas asociadas a cada juego había varios elementos que resultaron repetidos y no pudo averiguar la causa de este fallo a nivel de Python. Los elementos repetidos estaban libres de espacios sobrantes, etc. y se los aplicaron controles de igualdad que inexplicablemente se

devolvieron como *True*. Debido a restricciones de tiempo, se tuvo que limitar algunos análisis debido a este problema de tratamiento.

Analizar Datos:

El análisis se guio por la búsqueda de correlaciones entre variables y valoración de juego y por siguiente en la posición en el ranking. Se analizó a manera de descartar para la presentación muchos elementos que, aunque tentaron conclusiones interesantes, no seguían la narrativa de la presentación final y se quedaron por el camino. Entre otras cosas, la incidencia de cada tipo de categoría y mecánica de juego además de sus respectivas estadísticas descriptivas a nivel global.

Lo que sí, hubiera sido interesante estudiar en el futuro más a fondo sería las diferencias de valoración, precio, weight, etc. entre juegos de pertenecientes a distintas categorías o que utilizaban distintas mecánicas de juego.

Representar Datos:

Dado que el análisis se efectuó sobre datos estáticos sin información geográfica, la gran mayoría de los gráficos se hicieron con Seaborn. Se usó Plotly principalmente para idenificar los *outliers* del dataset. La presentación se hizo en Microsoft PowerPoint.