Entrega Final Del Proyecto

Entregado por:

Miguel Ángel Restrepo Patiño

Andrés Ortega Sierra

Asignatura:

Introducción A La Inteligencia Artificial

Profesor:

Raúl Ramos Pollán

UNIVERSIDAD DE ANTIOQUIA

Facultad de Ingeniería

UNIVERSIDAD DE ANTIOQUIA

FACULTAD DE INGENIERÍA

MEDELLÍN 2022

1). Introducción

Descripción del problema

En el modo de juego Battle Royale del juego PUBG (Player Unknown's Battle Grounds), los jugadores forman equipos y compiten entre sí hasta solo quedar un equipo al final; dadas las características del modo de juego, en cada partida la ubicación final es diferente, por lo que no siempre el juego termina en una zona específica, es por ello, que se desea desarrollar un modelo predictivo a partir de las estadísticas finales del juego y las calificaciones iniciales de los jugadores, para predecir en dónde estará la ubicación final de la zona al terminar la partida y así, lograr que tú y tu equipo lleguen hasta el final de la partida.

Datos de la competencia

El dataset que se va a implementar proviene de una competencia de Kaggle, en la cual se proporcionan más de 65.000 juegos de datos de jugadores anónimos, divididos en conjuntos de entrenamiento y prueba, estadísticas del juego y calificaciones de los jugadores. Este dataset viene compuesto por algunos archivos .csv que proporcionan toda esta información como, por ejemplo, los id de diversos jugadores.

Primero hay un archivo que se llama sample_submission_V2.csv, el cual tiene la siguiente información:

- Múltiples id de diversos jugadores.
- Datos de “ganar la persecución del lugar” para cada id.

Luego hay un archivo llamado test_V2.csv, el cual es el conjunto de prueba y tiene como información lo siguiente:

- Id de múltiples jugadores.
- Id de diversos grupos de jugadores.
- Id sobre la partida.
- Datos sobre asistencias médicas de un jugador a sus compañeros.
- Botas de los jugadores.
- Daño causado.
- Entre otros muchos datos sobre los jugadores.

Finalmente hay un archivo llamado train_V2.csv, el cual es el conjunto de entrenamiento e información sobre los mismos datos anteriores en el conjunto de prueba, pero con diferentes valores claramente.

Métricas

Las métricas de evaluación principal se evalúan según el “error absoluto medio (MAE)”, (https://en.wikipedia.org/wiki/Mean_absolute_error), el cuál se calcula mediante la siguiente expresión:

$$MAE = \frac{\sum_{i=1}^{n} |yi - xi|}{n} = \frac{\sum_{i=1}^{n} |ei|}{n}$$

Donde:

- yi es la predicción.
- xi es el verdadero valor.

Por otra parte, en las métricas de negocio se desea que el porcentaje de error sea lo más pequeño posible, garantizando así un acierto confiable y poder así, lograr saber dónde estará la ubicación final en el juego.

Criterio

En este modelo, se desea y espera poder predecir la ubicación final de la zona segura dados los diversos datos de jugadores en los conjuntos de prueba y entrenamiento mencionados anteriormente. Con la información resultante por el modelo, se desea tener mejores resultados a la jugar de jugar en el PUBG, dando a conocer cuál es la mejor estrategia, ser campero o arrasar con los jugadores que encuentres a tu paso, en ambas, debes llegar sí o sí a esa ubicación final de la zona segura.

2). Exploración descriptiva del dataset

En la amplia exploración del dataset, hemos empezado a trabajar con los datos que nos dan en la competición de Kaggle; primero que todo, hemos descargado los datos, los cuáles son 3 archivos, como inconveniente hemos tenido la descarga de los datos, dado que son muy pesados, al igual que para subirlos al Colab y hacer las pruebas, dado que el Colab los borra del almacenamiento pasadas veinticuatro horas desde que se subieron, por lo que no podemos estar subiendo los datos todos los días, dado esto, hicimos un código para cargar los datos desde el Drive, y así no tener que estarlos subiendo cada vez que vayamos a trabajar en el código, el problema con esta nueva estrategia es que a la hora de ejecutar los códigos, el Colab te pide usuario y contraseña para poder acceder a los datos que hay en el Drive y como están subidos en el Drive piden usuario y contraseña de la persona que subio los archivos por lo que va en contra de la política de privacidad.

Como estrategia final decidimos utilizar el API Kaggle la cual está basada en cargar los datos directamente desde la competición de kaggle, lo único que debíamos hacer es subir al Colab un archivo JSON cada vez que vayamos a ejecutar el código este archivo JSON se descarga desde Kaggle. Para simplificar el paso de cargar el archivo JSON al Colab de Google, hemos puesto las credenciales de unos de los integrantes en el notebook, aunque esto implique ser un poco arriesgado a la hora de las políticas de privacidad, ya que se vulnera el usuario y la contraseña al mostrarlas explícitamente en el código. Dado esto, ya no tenemos que cargar ni un solo archivo, simplemente es correr el notebook y todo funciona con normalidad. Para más información les dejo el link para encontrar los pasos a seguir: https://www.youtube.com/watch?v=gwDOUuBH7ws&t=234s

Una vez hecho esto, continuamos con importar las librerías que vemos necesarias para la correcta ejecución de los códigos, segundamente utilizamos el código para cargar los datos y asignamos nombres a cada uno de los archivos cargados, y luego los corrimos para ver si funcionaban bien los códigos y los datos y poderlos visualizar bien.

3). Interacción de desarrollo

NoteBook

1) El primer paso en nuestro notebook es importar las librerías que vamos a utilizar, en el desarrollo de los códigos vemos necesaria la implementación de otras librerías y funciones, como por ejemplos las funciones de machine learning las cuales utilizamos para las predicciones, de igual manera estas bibliotecas las añadiremos en esta primera parte del código.

2) En el segundo paso, instalamos Kaggle para poder acceder al contenido, luego lo que hacemos es cargar las credenciales de Kaggle, es decir, para cada perfil, existe un usuario y una clave, esto es llamado como el API de Kaggle, el cual es un archivo JSON, que al descargarlo y abrirlo con el bloc de notas, nos muestra nuestro usuario y clave. Para descargar nuestro archivo JSON, lo primero que debemos hacer es abrir la página oficial de Kaggle, ir a nuestra foto de perfil y clickear, luego le damos en Account y posteriormente bajamos hasta donde está la información del API, por último, le damos en Create New API Token y se nos descargará automáticamente el archivo. Al hacer esto, podremos cargar el dataset de la competencia de Kaggle directamente a nuestro Goggle Colab y poder trabajar con los archivos csv.

3) En el tercer paso, lo que hacemos es descargar los datos de la competencia de Kaggle directamente al Colab, dado que en el paso anterior pusimos las credenciales, no tendremos problemas para descargar el dataset, además, le daremos permiso a Colab de administrar los archivos descargados, dado que ellos se descargan en .zip.

4) En el paso cuatro, lo que hacemos es que una vez descargado el daset, definimos cada uno de los tres archivos descargados y los imprimimos, para comprobar que todo el proceso de descarga y carga del dataset, funciona correctamente.

5) En el quinto paso, lo que hacemos es mirar el tipo de coincidencia entre el archivo test y el archivo train, luego fusionamos ambos archivos y luego eliminamos 3 IDs, finalmente mostramos la información luego de haber hecho los pasos anteriores y observamos como queda.

6) En el sexto paso lo que hacemos es eliminar una fila que no contiene valores y luego confirmamos si efectivamente se elimina.

7) A partir de este punto, empezamos a trabajar con los modelos de procesamiento de datos, pero primero empezamos con el pre-procesamiento. En el punto siete hacemos primero que todo la selección de características, elegimos todas las variables excepto el resultado; luego hacemos la comprobación, seguidamente creamos una división de prueba de tren dividido y finalmente hacemos el escalado de características.

8) En el punto número ocho, empezamos a hacer la regresión lineal, primero instanciamos, luego adaptamos el modelo, luego predecimos y finalmente evaluamos en el error medio absoluto MAE. La regresión lineal es un método estadístico con el cual buscamos predecir el comportamiento de una variable a partir de los datos dados con anterioridad. Luego de evaluar, lo que hacemos es coger las puntuaciones y hacer una gráfica con los resultados en un intervalo de 1 a 70.

9) En el paso número nueve, lo que hacemos es utilizar la regresión de árbol de decisión, decidimos utilizar este tipo de regresión dado que una de las principales ventajas de los árboles de decisión es que pueden captar interacciones no lineales entre variables en los datos que la regresión lineal no puede. Los árboles de decisión son modelos predictivos formados por reglas binarias (si/no) con las que se consigue repartir las observaciones en función de sus atributos y predecir así el valor de la variable respuesta.. Los métodos basados en árboles se han convertido en uno de los referentes dentro del ámbito predictivo debido a los buenos resultados que generan en problemas muy diversos.

   En este paso, al utilizar la regresión de árbol de decisión, el error medio absoluto nos dió un poco menor que al utilizar la regresión lineal, lo que indica que la regresión de árbol de decisión es un poco mejor.

10) Este es el último paso en nuestro notebook, lo que hacemos finalmente es predecir para cada jugador, crear un dataframe y luego lo que hacemos es tratar de enviar los resultados obtenidos

a la competencia de Kaggle, para ello creamos un documento de envío y luego predecimos las probabilidades de clase para los datos de prueba reales. Aquí termina el notebook.

4). Retos y consideraciones

Durante la ejecución del proyecto, tuvimos como primer inconveniente que uno de los integrantes del proyecto cancelara la materia por motivos que aún desconocemos; dado esto nos vimos en la obligación de continuar con el proyecto con los dos integrantes restantes aunque esto significaba tener problemas a la hora de realizar los codigos ya que cada uno de los integrantes se encargaba de una tarea específica entorno al proyecto. Otros integrantes de otros proyectos han querido unirse a nosotros por la razón de que sus compañeros también han cancelado la materia, a pesar de nuestra aceptación hasta el momento ninguno confirmó hacer parte del equipo.

Otra consideración importante es que el tratamiento de los dataset son de suma importancia y de alta complejidad, ya que al principio intentamos diversos métodos para descargar el dataset desde la competencia de Kaggle y seguidamente subirlos a Google Colab, pero todos estos primeros intentos fueron fallidos debido a que los archivos son muy pesados, luego de intentarlo arto, dimos con la solución de utilizar el API de Kaggle, lo cual nos resultó bastante viable y así poder terminar el proyecto.

Un último reto que tuvimos fue en el notebook, al utilizar diversos métodos de regresión tuvimos problemas con algunos, tales como el algoritmo KNN, el algoritmo de bosque aleatorio y el algoritmo de regresión vectorial. Dado que los archivos del dataset son muy grandes, estos algoritmos con los que tuvimos problemas se quedaban mucho tiempo haciendo las iteraciones y al reducir el tamaño del dataset, los resultados obtenidos eran muy malos, por lo que optamos no poderlos

en nuestro notebook y trabajar con aquellos que sean más ágiles y eficientes.

5). Conclusiones

Quiero acabar nuestro informe diciendo la siguiente frase, “En la práctica es que se aprende”, creo que a pesar de hacer todos los laboratorios y los task de cada uno, siento que aprendí más en la ejecución del proyecto, porque en este recopilamos y utilizamos casi todo lo que usamos en los laboratorios. Concluimos que hay competencias de kaggle muy buenas y con muchos retos por delante, que hay cosas muy difíciles de hacer, de interpretar, que estos problemas grandes los debemos desglosar primero e ir entendiendo y resolviendo cada una de las partes desglosadas. En nuestro proyecto pudimos predecir finalmente lo que se requería, nos funcionaron algunos métodos, otros no, unos muy demorados, otros no tanto, consideramos que para resolver las regresión con otros métodos más demorados sería genial aprender a poner a trabajar varios computadores de manera sincronizada. Por último dar gracias a nuestro profe y tutor por acompañarnos y guiarnos durante la ejecución tanto de la asignatura como del proyecto, me siento agradecido…

Videos De Las Entregas

- https://www.youtube.com/watch?v=cPnnSyjZZ5s

Bibliografía

- *Kaggle*. (04 de 10 de 2018). Obtenido de https://www.kaggle.com/competitions/pubg-finish-placement-prediction/overview

- https://www.youtube.com/watch?v=gwDOUuBH7ws&t=234s

- https://www.kaggle.com/competitions/pubg-finish-placement-prediction

- https://github.com/

- Colaboradores de Wikipedia. (2022, 13 de mayo). Error absoluto medio. En *Wikipedia, la enciclopedia libre* . Recuperado el 24 de junio de 2022 a las 14:40, de https://en.wikipedia.org/w/index.php?title=Mean_absolute_error&oldid=1087554218

- https://www.cienciadedatos.net/documentos/py07_arboles_decision_python.html