Proyecto Final Machine Learning

INDICE

1-. Objetivo:

El objetivo de este proyecto es procesar y analizar un conjunto de datos obtenido en la plataforma Kaggle, el cual consta de 23 características y 721 ejemplos, y que está relacionado con los diferentes atributos de los Pokemons liberados desde la primera hasta la sexta generación. Para ello, les aplicaré diferentes técnicas de limpieza y preparación como manejo de valores nulos, codificación de variables categóricas y escalado. Tras el primer análisis, usaré gráficas y formulas estadísticas para analizar e identificar las posibles relaciones existentes entre las características y mi variable objetivo, para así poder preparar mejor el conjunto de datos y utilizarlo para entrenar un modelo de clasificación y hacer predicciones con él. Tras la limpieza, preparación y analisis de los datos, desarrollaré y entrenaré con ellos un modelo de Machine Learning, el cual voy a usar para intentar predecir la Tasa de captura de cada Pokémon en base a los atributos de cada uno de ellos y proporcionar así una herramienta útil en aplicaciones prácticas relacionadas con el juego de Pokemons.

Para esta tarea, utilizare las siguientes librerías con los siguientes fines: numpy, pandas, matplotlib, y seaborn para representación gráfica y manejo de los datos; scipy, statistic, Collection, itertools e imblearn, para análisis y reequilibrio de los datos; sklearn para entrenar, evaluar y analizar nuestro modelo de Machine Learning y predecir con él.

2-. Obtención De Los Datos:

Como ya he mencionado antes, este conjunto de datos lo he obtenido de la plataforma Kaggle y consta de 23 columnas y 721 filas. Tras importar el dataset, hago un primer análisis de las distintas características y observo la siguiente información de cada una de ellas:

- Number: Es el Id del Pokémon dentro de la pokedex. Variable de tipo categórica ordinal. El tipo de dato es int64 y la columna contiene 721 no nulos.

- Name: Nombre del Pokémon dentro de la pokedex. Variable de tipo Categórica nominal. El tipo de dato es str, contiene un valor único para cada Pokemon y 721 no nulos.

- Type_1: Variable que indica el primer tipo del Pokémon. Variable de tipo categórica nominal. El tipo de dato es str, contiene 18 clases distintas y 721 no nulos.

- Type_2: Variable que indica el segundo tipo del Pokémon. Variable de tipo categórica nominal. El tipo de dato es str,

contiene 18 clases (las mismas que hay en Type_1) y 350 no nulos.

- Total: Suma de las estadísticas de batalla del Pokémon. Variable de tipo Numérica continua. El tipo de dato es int64 y la columna contiene 721 no nulos.
- HP: Vida que tiene el Pokémon. Variable de tipo Numérica continua. El tipo de dato es int64 y la columna contiene 721 no nulos.

- Attack: Daño base de los ataques del Pokémon. Variable de tipo Numérica continua. El tipo de dato es int64 y la columna contiene 721 no nulos.

- Defense: Defensa base que tiene el Pokémon. Variable de tipo Numérica continua. El tipo de dato es int64 y la columna contiene 721 no nulos.

- Sp_Atk: Daño base de los super ataques del Pokémon. Variable de tipo numérica continua. El tipo de dato es int64 y la columna contiene 721 nulos.

- Sp_Def: Defensa especial base que tiene el Pokémon. Variable de tipo numérica continua. El tipo de dato es int64 y la columna contiene 721 no nulos.

- Speed: Velocidad de movimiento del Pokémon. Variable de tipo numerica continua. El tipo de dato es int64 y la columna contiene 721 no nulos.

- Generation: Generación en la que fue liberado cada Pokémon. Variable de tipo categórica ordinal. El tipo de dato es int64, contiene las generaciones de la 1 a la 6 y 721 no nulos.

- isLegendary: Característica booleana que indica si el Pokémon es legendario o no. Variable de tipo categórica nominal. El tipo de dato es bool y la columna contiene 721 no nulos.

- Color: Color dominante del Pokémon. Variable de tipo categórica nominal. El tipo de dato es str, contiene 10 opciones de colores y 721 no nulos.

- hasGender: Variable booleana que indica si el Pokémon tiene género o no. Es de tipo categórica nominal. El tipo de dato es bool y la columna contiene 721 no nulos.

- Pr_Male: Variable que indica en caso de que el Pokémon tenga género, la probabilidad de ser macho, siendo la probabilidad de ser hembra 1 menos la probabilidad de ser macho. La variable es de tipo numérica discreta. El tipo de dato es float64 y la columna contiene 644 no nulos.

- Egg_Group_1: Variable que indica el primer grupo de huevos al que pertenece el Pokémon. Es de tipo categórica nominal. El tipo de dato es str, contiene 15 opciones distintas y 721 no nulos.

- Egg_Group_2: Variable que indica el primer grupo de huevos al que pertenece el Pokémon. Es de tipo categórica nominal. El tipo de dato es str, contiene 13 opciones distintas y 191 no nulos.

- hasMegaEvolution: Variable que indica si el Pokémon tiene mega evolución o no. Es de tipo categórica nominal. El tipo de dato es bool y la columna contiene 721 no nulos.

- Height_m: Altura del Pokémon en metros. Variable de tipo numérica continua. El tipo de dato es int64 y la columna contiene 721 no nulos.

- Weight_kg: Peso del Pokémon en kilogramos. Variable de tipo numérica continua. El tipo de dato es int64 y la columna contiene 721 no nulos.

- Catch_Rate (variable objetivo): Indica la probabilidad de captura de cada Pokémon y va desde 3 hasta 255. Variable de tipo numérica continua. El tipo de dato es int64 y la columna contiene 721 no nulos.

- Body_Style: Indica la forma del cuerpo de cada Pokémon. Variable de tipo categórica nominal. El tipo de dato es str, contiene 14 tipos de cuerpos distintos y 721 no nulos.

3-. Limpieza y preparación de los datos:

Lo primero que hago es crear una copia del dataset para poder trabajar con él sin modificar el original.

Tras esto y hacer un breve análisis de las 23 variables, decido eliminar 4 de ellas:

- Number y Name: Estas variables realmente no aportan ninguna información al modelo, ya que ambas constan de 721 variables únicas (las cuales son el nº de la pokedex y el nombre del Pokémon), por lo cual no van a influir de ninguna manera en las en las predicciones, pudiendo incluso confundir al modelo e introducir ruido durante su entrenamiento.

- Type_2 y Egg_Group_2: EL motivo por el que he decidido eliminar estas variables es porque ambas cuentan con un número muy elevado de valores nulos (NaN).

    - Type_2 cuenta con 371 NaN (lo que supone un 51,45% de la columna). Los valores faltantes representan el número de Pokémons que no tienen un segundo tipo, por lo que

para rellenar estos datos tendría que hacerlo de manera artificial e introduciría una información inexistente.

    - Egg_Group_2 tiene un total de 530 valores nulos (el 73,5% de la columna). Al igual que Type_2, estos datos faltantes son de grupos de huevos que no tienen un segundo tipo. Entre esto y la cantidad de valores NaN en la columna, es más efectivo eliminarla que tratar de rellenarla y conservarla.

- Por último, modifico el contenido de la característica 'Pr_Male', la cual indica la probabilidad de ser macho. Esta contiene 77 valores NaN, que hacen referencia a los pokemons que en la característica 'hasGender' tienen False como atributo, lo cual significa que estos pokemons no tienen género (por eso su probabilidad de ser macho es NaN). Lo que hago es rellenar esas casillas con un -1, para que el modelo pueda diferenciar claramente estos Pokémon de aquellos con probabilidades de ser macho. Esta característica cuenta con 7 valores valores numéricos predefinidos (1.000, 0.875, 0.750, 0.500, 0.250, 0.125, 0.000), por lo que se podría tratar como variable categórica.

4-. Exploración de los datos en busca de patrones:

Para analizar las distintas características del dataset, dividiré el proceso en 2 secciones; variables categóricas y variables numéricas. En ambos casos realizare un análisis individual de cada una de las características para luego centrarme en las relaciones existentes entre las mismas.

4.1-. Variables Categóricas:

Para llevar a cabo el análisis de las variables categóricas, realizare 3 pruebas distintas siguiendo este orden; primero, un histograma de cada una para ver la distribución de las clases; segundo, voy a graficar una tabla de contingencia y llevare a cabo las pruebas de independencia chi-cuadrado y Cramer's V para ver las relaciones existentes entre las distintas columnas; por último, graficaré un histograma de la tasa de captura promedio de las clases de cada caracteristica.

Observaciones en la primera prueba:

- Type_1: Las clases dominantes dentro de esta columna son los tipos Agua (+-100 apariciones), Normal (+- 90 apariciones) y Hierba (+-70 apariciones) respectivamente, habiendo una media de +-50 apariciones en el resto de las clases.
- Generación: Se puede observar que de la Generación 1 a la 5 se liberaron una media de 130 Pokémons por generación, mientras que en la ultima se liberaron solamente unos 70 Pokemons.

- Color: Como se puede ver, el color mas abundante es el Azul, seguido del color Marrón y del Verde, lo cual tiene sentido ya que el color está relacionado con el tipo del Pokemon, y como se puede ver los colores más abundantes van acordes con los tipos más abundantes.

- Egg_Group_1: El tipo Campo es la clase predominante dentro de esta característica, habiendo más de 160 ejemplares dentro de la misma. Le siguen el tipo Monstruo y el tipo Agua_1, con cerca de 80 ejemplares cada uno. El resto de las clases tiene una distribución muy variada, con valores que oscilan entre menos de 10 y 70.

- Body_Style: El tipo de cuerpo que mas se observa en esta característica es el bípedo con cola, seguido del cuadrúpedo y del bípedo sin cola. El resto de las clases tiene una distribución de entre 60 y 15 ejemplares.

- isLegendary: En esta variable se distingue claramente que la cantidad de Pokemons legendarios es sumamente pequeña, habiendo más de 600 Pokemons no legendarios y menos de 100 Pokemons legendarios.

- hasGender: Esta columna tiene una distribución muy parecida a la de isLegendary. Como se puede ver, la probabilidad de encontrar un Pokémon sin genero es muy baja, habiendo solamente alrededor de 100 pokemons sin género.

- hasMegaEvolition: Distribución muy similar a la que se puede observar en las variables isLegendary y hasGender, habiendo menos de 50 ejemplares de Pokemons con Mega Evolución.

Observaciones en la segunda prueba:

- Resultados entre Type_1 y el resto de características;

  - Type_1 - Generation:
    Chi2; 12.51
    P-Value; 0.0000037245
    Cramér's V; 0.208
    La relación es estadísticamente significativa, aunque es débil. Si se observa la tabla de contingencia se puede ver que ciertos tipos principales de Pokemon son mas comunes en generaciones específicas, lo que puede deberse a los cambios de temática en las distintas generaciones.

  - Type_1 – Color:
    Chi2; 32.74

P-value; 0.0000000000
Cramér's V; 0.406
El color está altamente relacionado con el tipo principal del Pokémon, con una relación moderada-alta. Esto se debe a que el color del Pokémon suele basarse en su primer tipo. En la tabla de contingencia se puede observar como los colores Azul, Marrón, Verde y Rojo pertenecen mayoritariamente a Pokemons de tipo Agua, Normal, Planta y Fuego(respectivamente).

- Type_1 – isLegendary:
  Chi2; 7.40
  P-value; 0.0000074175
  Cramér's V; 0.275
  La relación entre estas variables es estadísticamente significativa, aun que es una relación moderada. Se puede observar como los tipos menos comunes como Dragon o Psíquico aparecen en un mayor número de Pokemons legendarios.

- Type_1 – hasGender:
  Chi2; 8.96
  P-value; 0.0000000003
  Cramér's V; 0.333
  Relación estadísticamente significativa. Se puede observar como Pokemons con tipos más raros (como Acero, Psíquico o Dragon) tienden a no tener género, mientras que tipos más básicos (Normal, Planta o Agua) tienden a tenerlo.

- Type_1 – Egg_Group_1:
  Chi2; 49.38
  P-value; 0.0000000000
  Cramér's V; 0.491
  Existe una relación estadísticamente significativa entre ambas variables, la cual es moderada-alta. Esto se

debe a que el tipo principal de un pokemon se deberá en gran medida al tipo principal del huevo.

- Type_1 – hasMegaevolution:
  Chi2; 5.26
  P-value; 0.0486156818
  Cramér's V; 0.195
  Aunque la relación es débil, se puede ver la tendencia de ciertos tipos de Pokemons (como Acero o Dragon) a tener una Megaevolucion.

- Type_1 – body_Style:
  Chi2; 30.42
  P-value; 0.0000000000
  Cramér's V; 0.314
  La relación entre las variables es estadísticamente significativa. Se puede observar como el diseño de los Pokemons esta fuertemente influenciado por el tipo. Por eso, los de tipo Agua suelen tener aletas, o los de tipo Volador suelen tener alas.

- Resultados entre Generation y el resto de características:

  - Generation – Color:
    Chi2; 9.88
    P-value; 0.0000092495
    Cramér's V; 0.164
    Relación estadísticamente significativa, pero débil. Se puede ver como los colores de los Pokémon pueden variar ligeramente según la generación.

  - Generation – isLegendary:
    Chi2; 2.69
    P-value; 0.2018760598
    Cramér's V; 0.100

Cuentan con una relación estadísticamente no significativa y muy débil. Esto significa que la liberación de Pokémon legendarios no depende de la generación y tampoco sigue un patron claro.

- Generation – hasGender:
  Chi2; 2.18
  P-value; 0.4449873108
  Cramér's V; 0.081
  Existe una relación estadísticamente no significativa y muy débil.

- Generation – Egg_Group_1:
  Chi2; 11.51
  P-value; 0.0000092131
  Cramér's V; 0.191
  Entre estas variables hay una relación estadísticamente significativa, pero debil.

- Generation – hasMegaevolution:
  Chi2; 5.43
  P-value; 0.0000185988
  Cramér's V; 0.20
  Relación estadísticamente significativa y débil. Esto se debe a que las Megaevoluciones se introdujeron en generaciones especificas (1, 2 y 3).

- Generation – Body_Style:
  Chi2; 8.61
  P-value; 0.2013828183
  Cramér's V; 0.143

Relación estadisticamente no significativa y debil. NO hay patrones claros que relacionen el estilo corporal con la generación.

- Resultados entre Color y el resto de características:

  - Color – isLegendary:
    Chi2; 1.66
    P-value; 0.9723672001
    Cramér's V; 0.062
    Relación estadísticamente no significativa y muy débil. Ser legendario no esta relacionado con el color del Pokémon.

  - Color – hasGender:
    Chi2; 2.67
    P-value; 0.6198648423
    Cramér's V; 0.099
    Tienen una relación no significativa muy debil. No hay indicios de relación entre la ausencia de género y el color de un Pokémon.

  - Color – Egg_Group_1:
    Chi2; 21.64
    P-value; 0.0000000000
    Cramér's V; 0.268
    Existe una relación estadísticamente significativa y moderada. Esto se debe a que el color esta relacionado con el tipo principal del Pokémon, el cual está relacionado con el grupo de huevo principal.

- Color – hasMegaevolution:
  Chi2; 1.84
  P-value; 0.9458319266
  Cramér's V; 0.068
  Relación no significativa y muy débil. El color no esta relacionado con la capacidad de megaevolucionar.

- Color – Body_Style:
  Chi2; 15.42
  P-value; 0.0000000003
  Cramér's V; 0.191
  La relacion es estadisticamente significativa y moderada. Esto puede deberse a que ciertas combinaciones entre color y cuerpo podrían ser más frecuentes (por ejemplo en los tipos Agua, que el color más común es el azul, y el tipo de cuerpo es con aletas)

- Resultados entre isLegendary y el resto de características:

  - isLegendary – hasGender:
    Chi2; 17.06
    P-value; 0.0000000000
    Cramér's V; 0.635
    Existe una relación estadísticamente significativa y muy fuerte. Esto se debe a que los Pokémon legendarios en su mayoría no tienen género, reforzando así su condición de Pokémon únicos.

  - isLegendary – Egg_Group_1:
    Chi2; 20.88
    P-value; 0.0000000000
    Cramér's V; 0.777
    La relacion entre ambas variables es estadisticsmenye signifivstiva y muy fuerte. Esto se debe a que todos los

pokemon legendarios pertenecen al grupo de huevos sin descubrir.

- isLegendary – hasMegaevolution:
  Chi2; 0.97
  P-value; 0.3291114905
  Cramér's V; 0.036
  Relacion estadisticamente no significativa y muy debil. Ser legendario no influye en la capacidad de megaevolucionar (y en general los pokemon legendarios no megaevolucionan).

- isLegendary – Body_Style:
  Chi2; 4.46
  P-value; 0.0962746458
  Cramér's V; 0.166
  La relación está en el límite entre ser significativa y no serlo, no obstante, en la tabla de contingencia se puede ver cierta tendencia en el estilo corporal de los Pokémon legendarios (bipedal tailed, bipedal tailless, quadruped y two wings).

- Resultados entre hasGender y el resto de características:

  - hasGender – Egg_group_1:
    Chi2; 19.06
    P-value; 0.0000000000
    Cramér's V; 0.71
    Relación estadísticamente significativa y muy fuerte, lo cual es muy probable que se deba a las restricciones biológicas por no tener género.

- hasGender – hasMegaevolution:
  Chi2; 0.20
  P-value; 0.8386815792
  Cramér's V; 0.007
  Se podría decir que la relación entre estas variables es nula.

- hasGender – Body_Style:
  Chi2; 6.74
  P-value; 0.0000168978
  Cramér's V; 0.251
  Relación estadísticamente significativa y baja-moderada. Se puede observar en la tabla de contingencia que hay ciertos estilos corporales que tienden a no tener genero (los cuales coinciden con los estilos corporales que tienden a tener los Pokémon legendarios).

- Resultados entre Egg_group_1 y el resto de características:

  - Egg_group_1 – hasMegaevolution:
    Chi2; 4.84
    P-value; 0.0535635520
    Cramér's V; 0.180
    La relación esta en el limite de ser o no ser significativa, además de ser débil. Se puede ver cierta tendencia de algunos tipos de huevos a tener megaevoluciones.

  - Egg_group_1 – Body_Style:
    Chi2; 42.57
    P-value; 0.0000000000
    Cramér's V; 0.439
    Relación estadísticamente significativa, además de moderada-alta. Los grupos de huevos están

moderadamente relacionados con el estilo corporal, lo que significa que el tipo de huevo tiene influencia en el estilo corporal con el que nacerá el Pokémon.

- Resultado entre hasMegaevolution y Body_Style:

  Chi2; 4.19
  P-value; 0.1720961946
  Cramér's V; 0.156
  Relación no significativa y débil. La capacidad de megaevolucionar no parece estar asociada con el estilo corporal.

Observaciones en la tercera prueba:

- Body_Style:
  Se puede ver que el diseño corporal de los Pokémon tiene un impacto significativo en la tasa de captura promedio, siendo head_legs e insectoid los tipos corporales con la tasa de captura promedio más alta, con valores de 157.05 y 156.00 respectivamente. Los estilos four_wings y multiple_bodies presentan los valores más bajos, con 57.22 y 76.87. Luego, la tasa de captura del resto de estilos varía entre los 127 y los 83 puntos, habiendo un mayor número de estos por encima de los 100 puntos. Aquí se puede ver como los Pokémon con diseños más simples o asociados a tipos comunes tienen una tasa de captura

mayor, mientras que los diseños más raros tienden a tener una tasa de captura menor.

- Color: Brown es el color con la mayor tasa promedio con 124.05 puntos, seguido por Pink con 108.32 y White con 107.21. Los colores Blue y Black presentan las tasas más bajas, con valores de 84.71 y 88.88, respectivamente.
  Aquí se puede observar cómo colores más oscuros o raros suelen tener una tasa de captura más baja que otros colores más comunes o claros como el marrón o el amarillo.

- Generation:
  El promedio de las tasas de captura varía entre las distintas generaciones. Las que tienen un mayor valor son la 3ª con 113.36, la 1ª con 106.19, y la 5ª con 103.10.
  La generación 4 tiene la tasa más baja (78.86).
  No parece haber alguna relación entre Generation y la tasa de captura, pero se puede ver que la 4ª generación se sale del promedio de las otras generaciones, lo que significa que en ella se pudo liberar un mayor número de Pokémon de tipo acero, dragón, legendarios o con megaevolucion.

- Type_1:
  Poison y Normal son los tipos con la tasa de captura más alta, con valores de 131.43 y 123.06, respectivamente. En la parte contraria, están tipos como Dragon, con 36.37 de tasa de captura, y Steel, con 63.00. Estas son las tasas de captura más bajas. Luego, hay tipos más comunes como Water (101.86), Grass (106.33) o Fighting (103.00) con valores intermedios.
  Se puede apreciar como tipos más defensivos como Rock o asociados a Pokémon legendarios como Dragon o Steel tienen una tasa de captura mucho menor que otros tipos asociados a Pokémon comunes como Poison o Bug.

- HasGender:
  Se puede apreciar una grandísima diferencia entre los Pokémon con género y sin él, teniendo los primeros una tasa de captura promedio de 107.45, mientras que los que no tienen género presentan una tasa mucho más baja de 39.99.
  Como he comprobado antes, la ausencia de género es una característica que se da mayoritariamente en Pokémon legendarios o muy poderosos, por lo que es normal que los Pokémon que cuentan con esta característica tengan una tasa mucho menor.

- Egg_Group_1:
  Los grupos de huevos con mayor promedio son Water_2, Fairy, y Grass con valores de 130.67, 129.33, y 128.70, respectivamente, mientras que los que tienen el menor promedio son Ditto, Undiscovered y Dragon con valores de 35.00, 41.62 y 45.00.
  Esto se debe a que los grupos relacionados con Pokémon comunes como Water_2 tienen tasas más altas, mientras que grupos inusuales como Ditto o asociados a Pokémon legendarios como Undiscovered son más difíciles de capturar.

- 6. hasMegaEvolution
  La megaevolución afecta significativamente la tasa de captura ya que los Pokémon sin megaevolución tienen una tasa promedio de 103.71, mientras los que si la tienen presentan una tasa mucho menor de 49.46. Esto se debe a que la megaevolucion es una característica que suelen tener Pokémon fuertes.

- 7. isLegendary

La diferencia aquí es extremadamente notable, ya que los Pokémon no legendarios tienen una tasa promedio de 106.63, mientras que los legendarios presentan una tasa promedio bajísima, con un valor de 6.65.
Esto es algo normal ya que los Pokémon legendarios son únicos y muy poderosos, lo que hace que sean extremadamente difíciles de capturar.

4.2-. Variables Numéricas:

Para realizar el análisis de las variables numéricas, hare 3 pruebas distintas en el siguiente orden: primero comprobare la distribución de estas mediante un Kernel Density Plot y un gráfico Q-Q, además de calcular la media, la mediana y la moda y realizar las pruebas de asimetría y curtosis; en segundo lugar, hare un gráfico boxplot para detectar outliers ; en tercer lugar, mostraré una matriz de dispersión entre las distintas variables, calcularé la correlación existente entre ellas y graficaré un heatmap con los valores obtenidos para verlos con más claridad.

Observaciones en la primera prueba:

- Catch_Rate:
  - Asimetría: 0.801
  - Curtosis: -0.694
  - Media: 100.25, Mediana: 65.0, Moda: 45
  - La simetría y la curtosis sugieren que la distribución tiene un leve sesgo hacia valores más altos, además de haber una menor concentración alrededor de la media en comparación con una distribución normal, lo cual se

confirma al ver que la media tiene un valor bastante mas alto que la mediana y la moda, por lo que se podría decir que no sigue una distribución normal.

- Total:
    - Asimetría: 0.061
    - Curtosis: -0.646
    - Media: 417.95, Mediana: 424, Moda: 405 y 600.
    - La variable total muestra una distribución cercana a la normal, ya que tiene unos valores de asimetría y curtosis cercanas a 0 y una media, mediana y moda similares.

- HP:
    - Asimetría: 1.666
    - Curtosis: 7.718
    - Media: 68.38, Mediana: 65, Moda: 60.
    - La columna presenta una asimetría positiva y una curtosis alta. Esto indica que la distribución está sesgada hacia valores más altos y, aunque la mayoría de los datos están centrados alrededor de la media (lo que hace que en la gráfica se vea ese pico), hay bastantes valores muy por encima de esta. El que la media sea mayor que la mediana y la moda confirma el sesgo positivo. Por lo tanto, la columna sigue una distribución cercana a la normal, aunque con una leve tendencia a valores por encima de la media .
- Attack:
    - Asimetría: 0.308
    - Curtosis: -0.274
    - Media: 75.01, Mediana: 74, Moda: 50 y 80.
    - La grafica es casi simétrica, pero con una leve tendencia a valores más altos. La curtosis y la asimetría cercanas a 0 sugieren una distribución casi gaussiana,

lo cual es reforzado por la cercanía entre los valores de la media, la mediana y la moda.

- Defense:
  - Asimetría: 1.119
  - Curtosis: 2.46
  - Media: 70.81, Mediana: 65, Moda: 50 y 70.
  - La columna tiene una asimetría y una curtosis positivas, pero no extremadamente altas. Esto indica que hay un mayor nº Pokémon con defensas bajas, pero existen algunos casos con defensas muy altas que generan un sesgo positivo, además de tener una concentración moderada de valores cerca de la media. Aunque la distribución esta ligeramente sesgada a la derecha, la distribución es cercana a la gaussiana.

- Sp_Atk:
  - Asimetría: 0.526
  - Curtosis: -0.26
  - Media: 68.74, Mediana: 65, Moda: 60.
  - La asimetría positiva y la curtosis negativa, junto con la cercanía entre la media, la mediana y la moda, indican que Sp_Atk sigue una distribución muy similar a la normal, aunque con una ligera tendencia hacia valores más altos.
- Sp_Def:
  - Asimetría: 1.026
  - Curtosis: 2.37
  - Media: 69.29, Mediana: 65, Moda: 50.
  - Tiene prácticamente la misma distribución que Defense, pero con una tendencia menor que esta hacia valores altos, además de una menor concentración de valores alrededor de la media. Teniendo en cuenta esto

y la cercanía entre la media, la mediana y la moda, se podría decir que tiene una distribución cercana a la normal.

- Pr_Male:
    - Asimetría: -1.917
    - Curtosis: 2.78
    - Media: 0.39, Mediana: 0.5, Moda: 0.5.
    - La asimetría muestra una fuerte tendencia hacia valores más bajos, aunque esto seguramente se deba a que los Pokémon que no tienen genero tienen la probabilidad de ser macho en -1, lo que hace que la asimetría sea negativa. La curtosis indica que hay una alta concentración de valores alrededor de la mediana. La distribución no es normal debido a la marcada asimetría y la curtosis (lo cual se debe a esos Pokémon sin género que tienen la probabilidad en -1).
- Speed:
    - Asimetría: 0.278
    - Curtosis: -0.451
    - Media: 65.71, Mediana: 65, Moda: 50 y 60.
    - La asimetría ligeramente positiva indica que hay una ligera inclinación hacia valores altos fuera de la media, y la curtosis negativa indica que los valores no están del todo centrados alrededor de la media, si no más bien algo dispersos. La cercanía entre la media, la mediana y las modas, además de los valores de asimetria y curtosis, son indicadores de que la columna tiene una distribución casi normal.

- Height_m:
    - Asimetría: 5.497
    - Curtosis: 49.77

- Media: 1.14, Mediana: 0.99, Moda: 0.61.
- La altura muestra una asimetría positiva alta, lo que significa que la gran mayoría de Pokémon tienen una altura relativamente baja, habiendo unos pocos casos con alturas extremadamente altas. La curtosis es extremadamente alta, lo cual indica que la gran mayoría de valores están centrados alrededor de la media, pero que hay unos pocos casos muy por encima de esta. La distribución de esta columna está muy lejos de ser normal debido a esos pocos casos de altura extremadamente alta que la distorsionan.

- Weight_kg:
    - Asimetría: 4.000
    - Curtosis: 23.71
    - Media: 56.77, Mediana: 28, Moda: 5.0.
    - Los valores de asimetría y curtosis indican prácticamente lo mismo que en la columna Height_m, lo cual es normal ya que a mayor tamaño mayor peso. La gran mayoría de Pokémon están alrededor de la media, aunque la curtosis muestra una menor concentración alrededor de esta que en Height_m. Se puede ver que en la columna existen algunos casos excepcionales con pesos muy altos que alteran su distribución, la cual tampoco es gaussiana.

- Tras este análisis, he divido las variables en 2 grupos, las que tienen una distribución normal o próxima a esta y las que tienen una distribución distinta a la normal; en el primer grupo están Total, HP, Attack, Sp_Attack, Defense, Sp_Defense y Speed, mientras que en el segundo están Weight_kg, Height_m y Pr_Male.

Resultados en la segunda prueba:

- En los boxplots de las variables HP, Defense, Sp_Defense, Pr_Male, Height_m y Weight_kg, que son las que tienen una mayor asimetria y curtosis, se puede observar que son las que tienen el mayor número de outliers. Las mas notables son las 3 ultimas mencionadas, las cuales siguen una distribución distinta a la normal. Estos gráficos respaldan los datos obtenidos en la prueba anterior que indican una distribución sesgada, especialmente hacia valores por encima de la media, a excepción de Pr_Male, la cual tenía el sesgo hacia valores por debajo de la media debido a los ejemplos sin género. Las variables Height_m y Weight_kg destacan por su alto nº de outliers, que son el motivo de la alta curtosis obtenida en la prueba anterior, y que seguramente señalan a casos de Pokémon excepcionales con valores fuera de lo común, los cuales probablemente sean Pokémon Legendarios, más poderosos o de tipos más raros.

- Resultados en la tercera prueba:

  Para analizar los datos obtenidos en la prueba de correlación, dividiré los resultados en 3 partes; relaciones fuertes, relaciones moderadas y relaciones débiles:

  - Relaciones fuertes:

    1. Catch_Rate y Total (-0.738): Esta es la correlación negativa mas fuerte. Se puede ver que cuanto mas alta es la suma de todas las estadísticas base de un Pokémon, menor es la tasa de captura del mismo, lo

cual es normal ya que cuanto mas poderoso es un Pokémon suele ser más difícil capturarlo.

2. Total y Sp_Atk (0.724): Esta es la correlación positiva mas alta, la cual indica que aquellos Pokémon con un valor mayor en Total tienden a tener valores más altos en Sp_Atk. Esto refleja que los Pokémon que tienen un mayor valor ofensivo suelen tener mejores estadísticas globales.

3. Sp_Def y Total (0.707): La relación entre estas dos variables indica que en general, aquellos Pokémon con una alta defensa especial suelen tener un Total más elevado.

4. Attack y Total (0.704): La alta correlación en esta variable indica que aquellos Pokémon con mayor poder ofensivo tienden a tener un Total mas elevado, lo cual confirma lo dicho anteriormente en la relación entre Total y Sp_Atk.

5. Weight_kg y Height_m (0.661): La altura y el peso tienen una alta correlación positiva, lo cual tiene sentido ya que cuanto mas grande es un ser vivo, este suele pesar más.

- Relaciones moderadas:

1. Defense y Total (0.606): Estas variables tienen una relación similar a Attack y Total. Los Pokémon con buena defensa física tienden a tener mejores valores en general, aunque esta característica no es tan decisiva como el ataque.

2. Speed y Total (0.549): La velocidad también tiene una relación bastante positiva con el Total, aunque esta influye de forma más débil que otras características.

3. Catch_Rate y Sp_Def (-0.513): Este dato refleja que los Pokémon con una alta defensa especial suelen ser más difíciles de capturar, lo cual refuerza la idea de que las estadísticas defensivas contribuyen en gran medida a aumentar la dificultad de captura de un Pokémon.

4. Sp_Def y Sp_Atk(0.492): La correlación moderadamente positiva entre estas variables indica que algunos Pokémon con una alta defensa especial tienden a tener también un buen ataque especial, señalando así un equilibrio entre ambas características.

5. Weight_kg y Defense (0.476): La correlación entre estas variables, a pesar de no ser muy alta, es una clara señal de que los Pokémon mas grandes o pesados suelen tener mejores valores de defensa básica.

6. Height_m y Weight_kg con HP (0.442 y 0.431 respectivamente): Los valores obtenidos en la prueba de correlación de la altura y el peso con HP muestran que aquellos Pokémon que tienen un mayor tamaño y peso tienden a tener un valor de HP más alto, reforzando así que los Pokémon más grandes y pesados suelen ser mas resistentes en combate.

- Relaciones débiles:

    1. Speed y Defense (-0.009): Esta es la relación mas débil de todas, la cual es prácticamente nula. La velocidad del pokemon no parece tener ningún tipo de relación con las defensas físicas.

    2. Pr_Male con la mayoría de las variables: Las correlaciones entre la probabilidad de ser macho y el resto de las características son en general muy bajas, lo cual indica que el genero no influye ni positiva ni negativamente en las estadísticas de los Pokémon.

4.2-. Variable Objetivo:

Mi variable objetivo cuenta con 33 valores predefinidos los cuales son; 3, 15, 25, 30, 35, 45, 50, 55, 60, 65, 70, 75, 80, 90, 100, 120, 125, 127, 130, 140, 145, 150, 155, 160, 170, 180, 190, 200, 205, 220, 225, 235, 255. Por ello, aunque es una variable de tipo numérico, la trataré como si fuese una variable categórica para así poder utilizar un modelo de clasificación.

Ahora, voy a crear un gráfico de mi variable objetivo para analizar su distribución y ver como de desbalanceadas se encuentran las distintas clases:

La clase que más se repite es con diferencia 45, con 223 apariciones, seguida de 190 y 255 con 69 y 64 apariciones respectivamente; luego están las clases 3, 60, 75, 90 y 120 que rondan las 50 apariciones. El

resto de las clases tiene un número muy inferior que ronda entre 1 y 20 apariciones, por lo que tenemos un desbalance muy grande en los datos. Para compensar este desbalance, voy a usar la librería imblearn para crear nuevos ejemplos de datos sintéticos, aunque la función que voy a utilizar para ello necesita que haya por lo menos 2 ejemplos de la misma clase para crear nuevos ejemplos, por lo que filtraré y eliminare todas aquellas clases que tengan 1 única manifestación (que son en total 8), e igualare al resto en apariciones con la clase 45, que es la que tiene el mayor nº de manifestaciones.

Antes de llevar esto a cabo, he hecho un mapeo de todas las variables categóricas sustituyendo las clases por números, además de añadir una nueva columna llamada Total+Catch_Rate que contiene la suma de mi variable objetivo y la variable Total, para que el modelo destaque la importancia de esta. Tras esto, hago una codificación de la variable objetivo, sustituyendo las clases por los números del 1 al 25.

5-. Definición del modelo que vamos a utilizar:

Para elegir el modelo que voy a utilizar, he hecho una selección de modelos de clasificación los cuales son: LogisticRegression, RandomForestClassifier, ExtraTreesClassifier, BernoulliNB, GaussianNB, DecisionTreeClassifier, ExtraTreeClassifier, KNeighborsClassifier, LinearSVC, NuSVC y SVC, los cuales voy a probar usando una función que he creado y que devuelve los 3 modelos con mayor precisión.

Primero, antes de usar la función, hago un escalado de las variables que siguen una distribución normal o casi normal y una normalización de aquellas con una distribución distinta, divido mi set de datos en

test y train y uso la función para testear los modelos. Los que me dan un mejor resultado son: ExtraTreesClassifier con 0.978 de precisión, RandomForestClassifier con 0.976 y DecisionTreeClassifier 0.935, por lo que el modelo que utilizare es ExtraTreesClassifier.

Lo siguiente que hago es escoger los mejores parámetros para mi modelo utilizando la función GridSearchCV de scikit-learn, a la cual le paso la siguiente rejilla de parámetros para que busque la mejor combinación:

{'n_estimators': [100, 150, 200],

'criterion': ['gini', 'entropy', 'log_loss'],

'max_depth': [None, 15, 30],

'min_samples_split': [2, 4, 8, 16],

'min_samples_leaf': [1, 2, 4, 8]}

La mejor combinación de hiperparámetros que ha encontrado la función es la siguiente: {'criterion': 'entropy', 'max_depth': None, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 150} y las métricas que ha obtenido mi modelo con esos parámetros son: 0.978 de precisión, 0.98 de sensibilidad, 0.979 en la prueba F1-score y 0.961 en la prueba de Jaccard.

6-. Representación de los resultados:

Para hacer una representación de los resultados de mi modelo, empiezo haciendo un mapa de calor de la matriz de confusión, donde se puede ver que el nº de errores de mi modelo es mínimo, siendo en la clase 3 donde más ha fallado clasificando, con 16 clasificaciones erróneas.

Después, hago un gráfico de puntos en donde muestro los valores reales de color negro y las predicciones de color rojo, para ver si estas están muy alejadas de los valores reales. En él se puede ver que casi todos los puntos rojos están situados encima de un punto negro.

Por último, calculo el nº de predicciones fallidas y de predicciones acertadas, y el resultado que obtengo es el siguiente:

Predicciones Correctas: 1091

Predicciones Fallidas: 24

Predicciones Totales: 1115