Universidad Politécnica de Madrid
Escuela Técnica Superior de Ingeniería de Sistemas Informáticos

Análisis de los factores más influyentes en la esperanza de vida mediante Machine Learning

Proyecto Fin de Grado
Grado en Ingeniería del Software

Curso académico 2020-2021

Autor:
Miguel Roca García

Tutor:
Raúl Lara Cabrera

Quiero transmitir mi más sincero agradecimiento a todas las personas que han hecho posible que haya llegado hasta aquí.

En primer lugar, a mi tutor, Raúl, por introducirme en el mundo del Machine Learning y su guía y consejo a lo largo de este trabajo.

En segundo lugar, a todos mis compañeros y amigos realizados en la carrera, que han hecho que la universidad sea una experiencia maravillosa.

Por último, quiero agradecer enormemente todo el apoyo y dedicación de mi familia en mi educación. En especial a mis padres por apostar por mí y permitirme estudiar en Madrid y a mi tía Lelel por haberse esmerado en motivarme para apuntar siempre a lo más alto en mis notas.

Gracias de corazón.

Resumen

En este Trabajo de Fin de Grado se expone un análisis de la influencia de diversos indicadores económicos, de desarrollo, alimenticios, sanitarios y políticos en la esperanza de vida mediante modelos de aprendizaje automático.

Los factores a estudiar se han obtenido mediante un proceso ETL. Provienen de diferentes organizaciones internacionales de prestigio y están organizados por país, año y género. Se ha llevado a cabo un preprocesamiento del conjunto de datos de partida especialmente caracterizado por la imputación de valores desconocidos.

Se han implementado varios modelos de machine learning que sean capaces de obtener la esperanza de vida para cada caso mediante los factores suministrados. Los mejores resultados han sido obtenidos mediante una red neuronal artificial, el perceptrón multicapa.

El estudio del comportamiento de la red de neuronas se ha aplicado mediante tres estrategias con el fin de determinar el efecto de los factores sobre el cálculo de la esperanza de vida.

En el primer enfoque, ha sido la técnica de selección de features *wrapper methods*, con la que se ha determinado el conjunto mínimo de factores con los que se puede obtener un modelo con un error aceptable.

En segundo lugar, se ha implementado una función de calidad para el algoritmo genético que maximizase la esperanza de vida para un caso en concreto con un cierto margen de cambio. Así se determinarán los factores más influyentes para cada situación.

Finalmente, se ha hecho un estudio del modelo con SHAP, obteniendo los factores que más afectan a nivel general, individual y en comparaciones entre casos, pudiendo también visualizar la contribución de cada factor sobre el resultado final en un caso determinado.

Sobre los resultados obtenidos cabe destacar la diferencia entre correlación y causalidad, concluyendo que este estudio identifica de qué forma afectan los factores al cálculo de la esperanza de vida, no pudiendo demostrar que sean su causa, si bien puede servir de base de partida para su estudio por parte de las autoridades sanitarias o políticas.

Abstract

This Final Degree Project uses machine learning models to analyse the influence of various factors on life expectancy, such as economic, development, food diet, health and political indicators.

The factors that are studied have been obtained through an ETL process from different international organizations of prestige and organized by country, year and gender. A preprocessing of the dataset characterized by the imputation of missing values was required.

Several machine learning models have been implemented in order to predict the life expectancy of each case with the provided indicators. The best results were obtained by an artificial neural network, the multilayer perceptron.

The study of the behaviour of the neural network has been approached with three different strategies in order to determine the effect of the indicators on the calculation of life expectancy.

The first approach was the feature selection technique wrapper methods, with which the minimal set of indicators required to get an acceptable error has been determined.

Secondly, a fitness function for the genetic algorithm has been implemented in order to maximize the life expectancy of a specific case with a certain margin change. This algortihm provides the most influential factors for each situation.

Finally, the model has been carried out with SHAP, obtaining the factors that most affect at a general and individual level and in comparisons between cases. The contribution of each factor to the final result in an individual case has been provided as well.

It is important to mention the difference between correlation and causality for the results obtained. This study identifies the effect of the indicators on the calculation of life expectancy, not being able to demonstrate if their values are the cause in real life, although it can serve as a basis for consideration by health or political authorities.

Índice

Agradecimientos I
Resumen II
Abstract III

1. Introducción **1**
1.1. Objetivos 1
1.2. Metodología y estructura 2

2. Estado del arte **3**
2.1. Otros estudios sobre la esperanza de vida 3
2.2. Aprendizaje automático 3
2.2.1. Interpretación de modelos de aprendizaje automático . . . 4

3. Desarrollo del proyecto **5**
3.1. Conjunto de datos 5
3.1.1. Origen de datos 5
3.1.2. Definición del conjunto de datos 6
3.1.3. Procesos ETL para la obtención del dataset 10
3.1.4. Análisis exploratorio y limpieza 12
3.1.5. Tratamiento de valores vacíos 15
3.1.6. Transformaciones aplicadas 20
3.1.6.1. Datos categóricos 20
3.1.6.2. Datos continuos 21
3.2. Selección de features 24
3.3. Análisis de correlación 26
3.4. División en conjunto de entrenamiento y test 27
3.5. Modelos de Machine Learning aplicados 29
3.5.1. Regresión Lineal Múltiple 29
3.5.2. K-Nearest Neighbors Regressor 30
3.5.3. Random Forrest Regressor 31
3.5.4. Red de Neuronas 33
3.5.4.1. Perceptrón simple 33
3.5.4.2. Perceptrón multicapa 34
3.5.5. k-Fold Cross Validation 37
3.6. Técnicas aplicadas para el análisis de resultados 38
3.6.1. Wrapper Methods 38
3.6.2. Algoritmo Genético 39
3.6.3. Shapley Additive Explanations (SHAP) 42
3.7. Predictor 44

4. Resultados **45**
4.1. Conjunto de datos preprocesado 45
4.2. Importancia según la correlación 45
4.3. Error obtenido 46
4.4. Análisis del error obtenido 48
4.5. Interpretación del comportamiento de los modelos 51
4.5.1. Interpretación de Random Forest Regressor 51

4.5.2. Interpretación de la Red de Neuronas 53
4.5.2.1. Wrapper Methods 53
4.5.2.2. Algoritmo genético 57
4.5.2.3. Shapley Additive Explanations (SHAP) 60
4.6. Problemas encontrados 71

5. Conclusiones y trabajos futuros **73**
5.1. Conclusiones 73
5.2. Líneas futuras 76

Bibliografía **77**

Índice de tablas

3.1. Estructura de los datos 12
3.2. Funciones de activación derivables 35

4.1. Las 10 features más correladas con la esperanza de vida 46
4.2. Error sobre el conjunto de entrenamiento 47
4.3. Error sobre el conjunto de test 47
4.4. Features más importantes para la primera iteración 52

Índice de figuras

3.1. Fases de un proceso ETL[1] 11
3.2. Distribución Normal[2] 14
3.3. Distribución exponencial negativa[3] 14
3.4. Histograma del ratio de mortalidad infantil menor que 5 años . . 14
3.5. Histograma del número de valores desconocidos por registro . . . 15
3.6. Histograma del número de valores desconocidos por país 15
3.7. Interpolación del porcentaje de partos atendidos por personal sanitario en Afganistán 16
3.8. Interpolación del ratio de dentistas en República Dominicana . . 16
3.9. En azul: Histograma del número de valores desconocidos por fila original. En rojo: Histograma del número de valores desconocidos por fila tras aplicar las primeras técnicas. 19
3.10. En azul: Histograma del número de valores desconocidos por país original. En rojo: Histograma del número de valores desconocidos por país tras aplicar las primeras técnicas. 19
3.11. Transformaciones sobre una distribución que se asemeja a una normal. Histograma sobre el porcentaje de muertes por heridas. . 23
3.12. Transformaciones sobre una distribución que no se asemeja a una normal. Histograma sobre el PIB per cápita. 23
3.13. Correlación lineal de Pearson entre las features 28
3.14. Comparación de la esperanza de vida y los ingresos per cápita . . 29
3.15. Comportamiento de una regresión lineal[4] 30
3.16. Comportamiento de la regresión KNN[5] 31
3.17. Árbol de decisión para una regresión[6] 32
3.18. Neurona Artificial[7] 34
3.19. Perceptrón multicapa[8] 36
3.20. K-Fold Cross Validation[9] 37
3.21. División de entrenamiento, validación y test[10] 38
3.22. Ejecuciones del modelo para el cálculo de la contribución marginal de cada feature[11] 43
3.23. Ejemplo de la influencia de las features sobre el resultado final mediante *SHAP values* 44

4.1. Distribución del error obtenido 49
4.2. Distribución de la esperanza de vida real por tipo de error 49
4.3. Distribuciones de *Diet Composition Oils And Fats* y *Low CI Value % Death Cardiovascular* respectivamente de valores sobrestimados y no sobrestimados 50
4.4. Distribuciones de *Conflict and Terrorism Deaths %* y *Diet Calories Fat* de valores infraestimados y no infraestimados 50
4.5. Distribución de la predicción de la esperanza de vida por tipo de error 51
4.6. Importancia relativa de las features según *Random Forest* 53
4.7. Evolución del MAE durante el *backward feature selection* 54
4.8. Evolución del MAE durante el *forward feature selection* 55

4.9. Modificaciones sobre los indicadores de Afganistán en 1990 para ambos géneros para maximizar la esperanza de vida 57
4.10. Modificaciones sobre los indicadores de Lituania en 2005 y género masculino para maximizar la esperanza de vida 58
4.11. Modificaciones sobre los indicadores de España en 2019 y ambos géneros para maximizar la esperanza de vida 59
4.12. Media de aportación de cada features sobre el resultado final . . 60
4.13. Comparación de valores de esperanza de vida estandarizados y sin estandarizar 61
4.14. Diagramas de violín sobre los *SHAP values* de cada feature(Parte 1) 63
4.15. Diagramas de violín sobre los *SHAP values* de cada feature(Parte 2) 64
4.16. Gráfica de dependencia sobre la feature *Year* 65
4.17. Gráfica de dependencia sobre la feature *Population 10 Percentage SDG Total* 66
4.18. Gráfica de dependencia sobre la feature *Income per Capita* . . . 66
4.19. Explicación de la influencia de cada feature sobre la predicción final de la red de neuronas sobre el caso de España en 2005 para ambos géneros 67
4.20. Explicación de la influencia de cada feature sobre la predicción final de la red de neuronas en gráfico de cascada sobre el caso de Afganistán en 1990 para género femenino 68
4.21. Gráfico de decisión que compara las predicciones sobre Argelia para ambos géneros año a año, desde 1990 hasta 2019 69
4.22. Gráfica de decisión sobre los países con esperanza de vida mayor que 82 años en 2015 69
4.23. Gráfica de decisión sobre los países con esperanza de vida entre 66 y 68 años en el año 2000 70
4.24. Gráfica de decisión sobre los países con esperanza de vida menor de 50 años en 1990 70

Capítulo 1

Introducción

La salud es un tema que eleva un especial interés y preocupación en la sociedad actualmente. Esta tendencia está promovida por la pandemia que asola el mundo entero y que ha obligado a tomar unas medidas extremas para evitar un muy elevado número de muertes.

El estado sanitario de un país o una población, es una característica muy difícil de medir, puesto que no existe un factor claro que nos pueda dar una idea de este concepto. No obstante, existen algunas métricas que ayudan a entender o hacerse una idea de la situación sanitaria de una población en un determinado espacio temporal. Una de ellas es la esperanza de vida.

La esperanza de vida al nacer es el número medio de años que vivirá un recién nacido de media si los patrones de mortalidad de la población a la que pertenece se mantienen constantes en el futuro respecto a su momento de nacimiento[12]. Esto quiere decir que, la esperanza de vida se calcula para un determinado periodo del tiempo, una determinada población y, generalmente, un determinado sexo y tan solo se referirá a la esperanza de vida para una determinada edad, en nuestro caso, al nacer.

Es una medida extremadamente útil para conocer el nivel sanitario del país o población en un determinado punto temporal, normalmente un año. Su cálculo se realiza mediante lo que se conocen como tablas de mortalidad o *life tables*, que plasman la probabilidad de morir en rangos de edad[13].

La esperanza de vida, por tanto, es una métrica que nace de la mortalidad, una medida que depende de una alta variedad de factores, los cuales podrán influir más o menos dependiendo del caso o de la importancia de dicho factor. Entre los factores a tener en cuenta podemos incluir indicadores de desarrollo, económicos, alimenticios, políticos, geográficos, sanitarios, sociales, etc.

La variedad y abundancia de estos valores a tener en cuenta es tan alta que hace inviable un posible enfoque analítico tradicional sobre los datos con el que ver la influencia de cada factor en el resultado final para cada caso. Este problema encuentra solución gracias a los avances de la inteligencia artificial y el aprendizaje automático.

Mediante modelos de aprendizaje automático o *machine learning*, es posible obtener una salida a partir de un alto número de entradas, pudiendo así introducir al modelo todos los factores que se quieren tener en cuenta para calcular la esperanza de vida, para después analizar el modelo y entender cuánto y cómo afecta cada uno de los factores o indicadores estudiados sobre el cálculo final.

1.1. Objetivos

El objetivo de este trabajo se divide en tres bloques que siguen un orden secuencial:

1. Constituir un conjunto de datos formado la esperanza de vida y factores que puedan influir sobre la misma organizados por país, año y género,

extraídos de bases de datos de organizaciones de alta relevancia internacional.

2. Crear un modelo de *machine learning* que sea capaz de predecir o calcular la esperanza de vida a partir del conjunto de datos conformado.

3. Estudiar e interpretar el funcionamiento del modelo construido mediante distintas estrategias para entender cuánto y cómo influyen los factores establecidos sobre el cálculo final de la esperanza de vida.

1.2. Metodología y estructura

La metodología y herramientas empleadas para conseguir los objetivos establecidos en el apartado anterior establecerán la estructura en la que se organiza el trabajo.

El lenguaje de programación empleado a lo largo de todo el trabajo ha sido *Python 3.8* [14] y el framework elegido para construir y ejecutar el código, *Jupyter Notebook*[15].

El desarrollo del proyecto comienza con la extracción y obtención del conjunto de datos, para lo cual se emplearon archivos CSV y la librería *pandas*[16] de *Python* para crear el dataset unificado con todos los datos a tener en cuenta. Mediante ese mismo lenguaje de programación, se realizó un análisis y preprocesamiento del conjunto de datos.

A continuación, mediante el uso de las librerías de *machine learning scikit-learn*[17] y *TensorFlow*[18], se construyeron y entrenaron los modelos, de menor a mayor complejidad.

Seguidamente, se establecieron varias técnicas para el análisis del modelo más complejo, la red de neuronas. La primera técnica descrita fue la estrategia de selección de features *wrapper methods*. El segundo enfoque de análisis de modelos planteado fue mediante el uso del algoritmo genético, para lo cual se empleó Salga[19], programa desarrollado por la Universidad Politécnica de Madrid para el uso del algoritmo genético con la ayuda de una interfaz visual. La tercer y última técnica aplicada fue mediante el uso de la librería SHAP[20] de *Python*, basada en los *shapely values* para la explicación de modelos.

Para el uso de la red de neuronas y cálculo de la esperanza de vida así como una posible comparación entre casos, se construyó un predictor en una libreta de *Python*.

Terminado el desarrollo, el apartado de resultados nos muestra todo lo obtenido a partir de lo anterior, empezando con un estudio de la correlación de los datos extraídos con la variable objetivo, la esperanza de vida.

Informaremos sobre el error obtenido por cada modelo en el siguiente subapartado y elaboraremos un estudio sobre el error de la red de neuronas para detectar el comportamiento del error.

Finalmente, aplicadas las estrategias de interpretación de modelos descritas en el desarrollo, se describirán los resultados obtenidos según cada enfoque.

Capítulo 2

Estado del arte

2.1. Otros estudios sobre la esperanza de vida

La esperanza de vida es un tema muy estudiado en nuestra sociedad, y prueba de ello es el alto número de investigaciones que, de una forma u otra, tratan este amplio campo de estudio.

Un amplio estudio publicado en *Our World In Data*, trata la evolución y cambios en la esperanza de vida a lo largo de los años y separado por países. Ofrece amplias visualizaciones para ver dicho progreso y comparativas respecto a factores clave como el gasto público en sanidad y el Producto Interior Bruto[12].

El artículo realizado por Casper Worm Hansen va más allá de las visualizaciones y estudia la relación entre la esperanza de vida y un factor como es el tiempo de educación medio, concluyendo que por cada año incrementado en la esperanza de vida, los años de educación aumentan un 3,5 %[21].

Un estudio destacable en el publicado por el diario de salud pública de Irán[22]. Este estudia factores socio-sanitarios sobre la esperanza de vida de países con un nivel de ingresos medio y bajo. Los factores que estudia este artículo son la fertilidad en las mujeres, ingresos per cápita, años de educación, ratio de VIH y la densidad de médicos por habitante. Concluye que para incrementar la esperanza de vida en países con un bajo nivel económico, se debe eliminar la prevalencia del VIH, la tasa de parto por mujeres adolescentes y el analfabetismo.

No obstante, los estudios versados en el campo de la esperanza de vida, estudian el comportamiento de un conjunto reducido de factores, debido a las limitaciones de un análisis tradicional. Mediante el aprendizaje automático podemos analizar la influencia de un número indeterminado de factores.

2.2. Aprendizaje automático

El aprendizaje automático o *machine learning* es una disciplina dentro de la inteligencia artificial en auge gracias a las nuevas tecnologías de cómputo. El *machine learning* se basa en la creación de un modelo que aprende a partir de un conjunto de datos de partida, reduciendo el error respecto al objetivo planteado a lo largo del tiempo. El proceso de aprendizaje se llama entrenamiento. En este proceso el objetivo es encontrar patrones dentro del conjunto de datos que permita un toma de decisiones para predecir futuros datos de entrada. Debido a que la base del aprendizaje automático son los datos, su calidad es crucial para la obtención de unos resultados fiables y precisos.

Dependiendo del problema planteado, dentro del aprendizaje automático existen varios tipos de enfoques.

El **aprendizaje supervisado** es aquel en el que se entrenan un conjunto de datos etiquetados, es decir, se conoce el resultado esperado de cada caso del conjunto de datos que se subministrará al modelo, pudiendo así medir el error de la predicción realizada por el modelo. La ventaja de este enfoque es que

requiere un menor número de datos de entrenamiento y se pueden tener métricas de precisión sobre los resultados. Sin embargo, los datos etiquetados son poco frecuentes y costosos de obtener. Este enfoque se utiliza para problemas de clasificación(cuando la etiqueta es una categoría) y regresión(cuando la etiqueta es un valor numérico).

El **aprendizaje no supervisado** parte de unos datos no etiquetados. El objetivo de este tipo de enfoque es la identificación de patrones y relaciones entre los datos.

2.2.1. Interpretación de modelos de aprendizaje automático

Uno de los mayores retos dentro del aprendizaje automático es la capacidad de explicación del modelo desarrollado. Existen numerosos caso en los que se emplea un algoritmo de *machine learning* para hacer una toma de decisiones de negocio. Esta decisión necesita un justificación, la cual los modelos no siempre son capaces de dar. Un caso práctico de esta situación sería la concesión de un préstamo por parte de un banco a un cliente. En caso de que se emplee un modelo de aprendizaje automático que rechace dicha petición, es necesario proveer una respuesta lógica de la respuesta al cliente.

Dependiendo del modelo aplicado, la capacidad de explicación será mayor o menor. No obstante, cuanto más complejos sean los modelos, menor capacidad tendrán. Cuando un modelo no es capaz de definir su toma de decisiones, se dice que es un modelo de **caja negra**.

El análisis de modelos de caja negra carece de un amplio recorrido, las dos técnicas más destacables en este ámbito son las propuestas por LIME[23] y SHAP[24]. Ambas técnicas se ha desarrollado en la última década y son capaces de explicar el modelo mostrando la aportación de cada una de las entradas al resultado final.

No obstante, LIME presenta una serie de desventajas respecto a su alternativa[25]. Esta técnica de interpretación de modelos tan solo es capaz de explicar un único caso de forma simultánea, mientras que SHAP, provee tanto esta funcionalidad como diferentes gráficas para interpretar de una forma global el efecto de los datos sobre todo el conjunto. Por otro lado, LIME carece de robustez, es decir, un pequeño cambio en los valores de entrada del punto estudiado puede hacer que su explicación cambie bruscamente, dando resultados muy dispares para situaciones similares. Por último, está condicionado a una serie de hiperparámetros requeridos para su uso. Por estos motivos se ha elegido SHAP sobre LIME como técnica de interpretación de modelos.

Se plantea otro enfoque muy interesante para la interpretación de modelos de caja negra en el artículo desarrollado por Federico Piccinini[26], en el cual expone una estrategia para la interpretación mediante el uso del algoritmo genético.

Capítulo 3

Desarrollo del proyecto

El flujo de trabajo seguido en este proyecto ha comenzado con la extracción, transformación, análisis y preprocesamiento de datos, a continuación, la aplicación de modelos de aprendizaje automático para finalizar con el análisis e interpretación de los mismos.

3.1. Conjunto de datos

El dataset o conjunto de datos del que partiremos, consiste en un conjunto de indicadores organizados por año y país. Estos indicadores proveen información sobre la sanidad, pobreza, alimentación, economía, tecnología, acceso a recursos, etc. que caracterizan un país en un determinado año. Además, contaremos con a esperanza de vida asociada a cada caso. El conjunto de datos comprende desde el año 1990 hasta el 2019.

3.1.1. Origen de datos

Los datos han sido extraídos de diversas fuentes, todas ellas organizaciones mundiales de prestigio como son: *World Bank*[27], OMS[28], UNICEF[29], *Our World In Data*[30] y FAOSTAT[31].

Integrado por 189 países, el **World Bank** o *Banco Mundial* es una organización multinacional financiera que tiene como objetivo la reducción de la pobreza mediante préstamos de interés bajo y apoyo económico a las naciones con un menor número de recursos. El *World Bank* ofrece una base de datos de libre acceso con cientos de indicadores sobre el desarrollo mundial. Esta información puede ser extraída mediante su herramienta de visualización y análisis de datos conocida como *DataBank* o Banco de datos, la que permite a su vez generar y almacenar gráficos, tablas y mapas sobre estos datos que ofrecen.

La **Organización Mundial de la Salud**(OMS) es un organismo perteneciente a las Naciones Unidas cuya función es liderar y organizar alianzas en situaciones sanitarias complejas, así como determinar líneas de investigación para adquirir y divulgar nuevo conocimientos en la rama sanitaria. Ofrecen un repositorio de datos público organizado por categorías, indicadores y países.

UNICEF o *Fondo de las Naciones Unidas para la Infancia* es también una organización perteneciente a las Naciones Unidas. Esta provee una ayuda humanitaria para acabar con la pobreza, la discriminación, la violencia y las enfermedades. Opera en países en vías de desarrollo. UNICEF ofrece una web para acceder a diferentes indicadores organizador por tema y país.

Our World in Data es una organización online que presenta y publica datos y estadísticas de investigaciones y análisis empíricos sobre los cambios que se producen en el mundo, con el objetivo de encontrar el motivo de esos comportamientos.

FAOSTAT que son las siglas de Organización de las Naciones Unidas para la Alimentación y la Agricultura se dedica a recoger, analizar y publicar estadísticas sobre alimentación y agricultura para la toma de decisiones. Ofrece un acceso libre a sus datos y estadísticas.

Finalmente, es necesario mencionar **Kaggle**[32], una plataforma gratuita que, entre otras cosas, ofrece datasets elaborados por otros usuarios de esta misma web. Una parte del conjunto de datos ha sido extraída de esta plataforma, que a su vez, fue extraída de los orígenes anteriormente descritos.

3.1.2. Definición del conjunto de datos

Las features son las entradas que contará un modelo de *machine learning* para obtener la salida. Algunos de los indicadores de los que partimos, se han obtenido mediante cálculos estadísticos y estimaciones, por tanto, tienen asociado un intervalo de confianza al 95 %. Este intervalo se reflejará en los datos contando con sus límites inferior y superior.

Las features o características que componen el dataset del que partiremos inicialmente son las especificadas a continuación:

- **Country**: Nombre del país.
- **Year**: Año.
- **Gender**: Género. Puede ser *Female*, *Male* o *Both sexes*.
- **Life Expectancy**: Esperanza de vida. Es la variable objetivo, es decir, la variable que intentaremos obtener mediante el modelo de *machine learning*.
- **Infant Mortality Rate**: Ratio de mortalidad infantil. Número de niños menores de un año fallecidos por cada 1.000 nacidos vivos en el periodo de un año.

  Low CI Value Infant Mortality Rate: Límite inferior del intervalo de confianza.

  High CI Value Infant Mortality Rate: Límite superior del intervalo de confianza.
- **Under 5 Mortality Rate**: Tasa de mortalidad en la niñez(menor de 5 años). Probabilidad de morir de un recién nacido antes de cumplir los primeros 5 años de vida expresado por cada 1.000 nacidos vivos.

  Low CI Value Under 5 Mortality Rate: Límite inferior del intervalo de confianza.

  High CI Value Under 5 Mortality Rate: Límite superior del intervalo de confianza.
- **% Death Cardiovascular**: Probabilidad de morir por enfermedades cardiovasculares, cáncer, diabetes o enfermedades respiratorias crónicas entre los 30 y los 70 años de vida.

  Low CI Value % Death Cardiovascular: Límite inferior del intervalo de confianza.

  High CI Value % Death Cardiovascular: Límite superior del intervalo de confianza.

- **Suicides Rate**: Ratio de suicidio. Número de muertes deliberadas llevadas a cabo por la propia persona con el pleno conocimiento o expectativa de su resultado por cada 100.000 habitantes.
- **Diet Composition Alcoholic Beverages**: Alcohol per capita. Suma total de alcohol consumido por adultos(mayores de 15 años) en el periodo de un año, en litros de alcohol puro por persona.
- **Air Pollution Death Rate**: Ratio de muertes por contaminación. Probabilidad de morir por contaminación del aire doméstica y ambiental. Se divide en:

  Stroke: Derrame cerebral.

  Ischaemic Heart Disease: Enfermedad cardíaca.

  Lower Respiratory Infections: Infección respiratoria menor.

  Chronic Obstructive Pulmonary Disease: Enfermedad pulmonar obstructiva crónica.

  Trachea Bronchus Lung Cancers: Cáncer de tráquea o pulmón.

  Total: Suma total.

  A su vez, están divididos en:

  Aged Standarized: Estandarizado por edad.

  Aged not Standarized: No estandarizado por edad.

  Todas estas features también tienen límite inferior y superior de los intervalos de confianza.
- **Unsafe Wash Mortality Rate**: Ratio de mortalidad por muertes atribuibles a la exposición a servicios de higiene, agua y sanidad no seguros por cada 100.000 habitantes.
- **Poisoning Mortality Rate**: Ratio de mortalidad por muertes atribuidas a envenenamiento no intencionado por cada 100.000 personas.
- **Tobacco Prevalence**: Porcentaje de población superior a 15 años que consumen tabaco.
- **% Population Aged 0-14**: Porcentaje de la población entre 0 y 14 años.
- **% Population Aged 15-64**: Porcentaje de la población entre 15 y 64 años.
- **% Population Aged 65+**: Porcentaje de la población mayor de 65 años.
- **% Population Aged 65-69**: Porcentaje de la población entre 65 y 69 años.
- **% Population Aged 70-74**: Porcentaje de la población entre 70 y 74 años.
- **% Population Aged 75-79**: Porcentaje de la población entre 74 y 79 años.
- **% Population Aged 80+**: Porcentaje de la población mayor de 80 años.

- **Maternal Mortality Ratio**: Ratio de mortalidad materna. Número de muertes maternas al dar a luz por cada 100.000 nacimientos.

  Low CI Value Maternal Mortality Ratio: Límite inferior del intervalo de confianza.

  High CI Value Maternal Mortality Ratio: Límite superior del intervalo de confianza.

- **% of Births Attended By Skilled Personal**: Porcentaje de partos atendidos por personal sanitario.

- **Neonatal Mortality Rate**: Ratio de mortalidad neonatal. Número de muertes durante los primeros 28 días de vida por cada 1.000 nacimientos en un año.

  Low CI Value Neonatal Mortality Rate: Límite inferior del intervalo de confianza.

  High CI Value Neonatal Mortality Rate: Límite superior del intervalo de confianza.

- **Incidence of Malaria**: Incidencia de la malaria. Número de casos de malaria por cada 1.000 habitantes en riesgo por año.

- **Incidence of Tuberculosis**: Incidencia de tuberculosis. Número estimado de casos de tuberculosis por cada 100.000 habitantes en el periodo de un año.

  Low CI Value Incidence of Tuberculosis: Límite inferior del intervalo de confianza.

  High CI Value Incidence of Tuberculosis: Límite superior del intervalo de confianza.

- **Hepatitis B Surface Antigen**: Prevalencia del antígeno de superficie de la hepatitis B.

  Low CI Value Hepatitis B Surface Antigen: Límite inferior del intervalo de confianza.

  High CI Value Hepatitis B Surface Antigen: Límite superior del intervalo de confianza.

- **Intervention Against NTDs**: Número de personas en valor total que han requerido tratamiento y cuidados por cualquier enfermedad tropical desatendida(NTD).

- **Road Traffic Deaths**: Número de muertes por causa de tráfico por cada 100.000 personas.

- **Reproductive Age Women**: Porcentaje de mujeres fértiles casadas o en pareja que emplean anticonceptivos.

- **Adolescent Birth Rate**: Ratio de natalidad adolescente. Número de nacimientos de mujeres entre los 15 y los 19 años de edad por cada 1.000 mujeres en ese rango.

- **Universal Heath Care Coverage**: Índice de servicios de sanidad básica cubiertos en escala de 0 a 100.
- **Population 10 Percentage SDG Total**: Proporción de la población que los gastos en sanidad exceden el 10 % de sus ingresos.

  Population 10 Percentage SDG Urban: Proporción en zona urbana.

  Population 10 Percentage SDG Rural: Proporción en zona rural.
- **Population 25 Percentage SDG Total**: Proporción de la población que los gastos en sanidad exceden el 25 % de sus ingresos.

  Population 25 Percentage SDG Urban: Proporción en zona urbana.

  Population 25 Percentage SDG Rural: Proporción en zona rural.
- **Doctors**: Número de médicos por cada 10.000 habitantes.
- **Nurses and Midwifes**: Número de enfermeros y comadronas por cada 10.000 habitantes.
- **Dentists**: Número de dentistas por cada 10.000 habitantes.
- **Pharmacists**: Número de farmacéuticos por cada 10.000 habitantes.
- **Basic Drinking Water Services**: Porcentaje de la población con acceso a servicios básicos de agua potable.
- **Basic Sanization Services Total**: Porcentaje de la población con acceso a servicios básicos de saneamiento e higiene.

  Basic Sanization Services Urban: Porcentaje en zona urbana.

  Basic Sanization Services Rural: Porcentaje en zona rural.
- **Safely Sanitation Total**: Porcentaje de la población con acceso a servicios de higiene y saneamiento seguros.

  Safely Sanization Urban: Porcentaje en zona urbana.

  Safely Sanization Rural: Porcentaje en zona rural.
- **Basic Hand Washing Total**: Porcentaje de la población con acceso a servicios para el lavado de manos con jabón y agua en el hogar.

  Basic Hand Washing Urban: Porcentaje en zona urbana.

  Basic Hand Washing Rural: Porcentaje en zona rural.
- **Clean Fuel and Technology**: Porcentaje de la población que usa combustibles limpios y tecnologías como principal fuente de energía en el hogar para cocinar.
- **Birth Rate**: Tasa de natalidad. Número de nacimientos por cada 1.000 habitantes.
- **Battle Related Deaths**: Número de muertes relacionadas con la batallas en un conflicto armado.
- **% Injury Deaths**: Porcentaje de muertes ocasionadas por heridas.

- **Death Rate**: Tasa de mortalidad. Número de muertes por cada 1.000 habitantes.
- **GDP per Capita**: Producto Interior Bruto per cápita en dólares.
- **% Population $1.90 a day**: Porcentaje de la población que vive con menos de 1,90 dólares al día.
- **% Population $3.20 a day**: Porcentaje de la población que vive con menos de 3,20 dólares al día.
- **% Population $5.50 a day**: Porcentaje de la población que vive con menos de 5,50 dólares al día.
- **Income per Capita**: Ingresos per cápita en dólares.
- **Total Population**: Número total de habitantes.
- **GNI per Capita**: Producto Interior Bruto per cápita transformado a dólares con el método Atlas.
- **Conflict and Terrorism Deaths**: Número de muertes por conflictos armados y terrorismo.

3.1.3. Procesos ETL para la obtención del dataset

Los procesos ETL (*Extract - Transform - Load*) son procedimientos de recolección de datos a partir de un número indefinido de fuentes, la organización y limpieza de los mismos y la unificación en un repositorio único y unificado. Consta de tres fases como su nombre indica: Extracción, Transformación y Carga.

La primera fase, la **extracción**, consiste en la obtención de los datos desde los sistemas de origen. A continuación, se lleva a cabo un análisis de la estructura de los datos obtenidos para comprobar que estos cumplen con los requisitos establecidos. Finalmente, se convierten los datos a un formato que permita su transformación.

La fase de **Transformación** consiste en la realización de ciertos cambios sobre el contenido o la estructura de los datos para que sigan las pautas y reglas de negocio establecidas. Estas transformaciones suelen consistir en:

- **Limpieza**: Eliminar datos erróneos y duplicados, traducción de códigos o separación de datos discretos.
- **Filtrado**: Eliminar aquellos datos innecesarios según los requisitos establecidos.
- **Clasificación**: División de los datos según su tipo, datos en bruto, audio, vídeo, estructurados, no estructurados, etc.
- **Reestructuración**: Transformaciones necesarias para que lo datos sigan una estructura unificada o especificada. Operaciones como ordenación de filas y columnas, cambio de nombres, división de columnas y filas, etc.
- **Unificación**: Unión de los datos en un único flujo o conjunto.

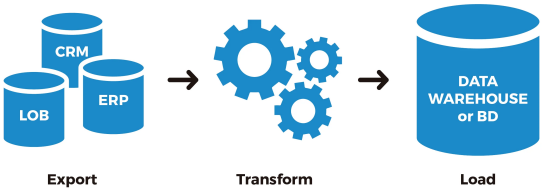

Figura 3.1: Fases de un proceso ETL[1]

Por último, la fase de **Carga** mueve los datos transformados a su destino. Esta carga se puede realizar de forma completa o incremental en función de las características del destino y la cantidad de los datos.

El marco de acción de los procesos ETL es alto, aunque sus usos más comunes son la migración de datos, replicación de datos para copias de seguridad en otras plataformas, unificación de varias fuentes y almacenamiento en un *Data Warehouse* o Almacén de Datos para ingerir, transformar y clasificar para estudios mediante análisis estadística, inteligencia artificial y *machine learning* con el fin de obtener inteligencia de negocio.

El proceso ETL llevado a cabo para obtener el conjunto de datos que se aplicarán sobre los modelos de *machine learning* se ejecutó siguiendo los siguientes pasos:

Fase de extracción

El proceso ETL empezó mediante la extracción de los datos procedentes de las fuentes. Estos **se almacenaron en archivos CSV**, uno por cada indicador extraído.

Un archivo CSV(*Comma Separated Values*) es un archivo de texto en el que cada campo está separado por una coma, punto y coma o carácter establecido, formando una tabla de filas y columnas. Generalmente, los nombres de las columnas, llamadas cabeceras, se especifican al principio del mismo.

Estos archivos extraídos fueron **ordenados y renombrados** para facilitar su uso. El nuevo nombre de cada archivo se hizo según la siguiente regla: *Índice_Nombre del Indicador*.csv. Para aquellos archivos con indicadores similares se les dio un subíndice.

A continuación, se hizo un análisis de los datos obtenidos descartando aquellos archivos que contenían información no relevante para el objetivo del proyecto o les faltaba demasiada información. Finalmente, se cargaron los archivos en un script de python **transformándolos a DataFrames** para así poder aplicar las transformaciones.

Un DataFrame es una estructura de datos en dos dimensiones donde se puede almacenar en cada columna un tipo de dato diferente. Tiene forma de tabla y puede nombrar columnas y filas.

Fase de transformación

Se han llevado a cabo diversas **transformaciones en la estructura** de los DataFrames con el objetivo de que quedasen en una estructura única con el

siguiente formato:

Country	Year	Gender	Indicator

Tabla 3.1: Estructura de los datos

Algunas de las transformaciones aplicadas para esta reestructuración fueron:

- Cambio de nombre de las cabeceras de los CSVs.
- Unificación de datos de un indicador desde varias fuentes para aumentar el número de valores conocidos.
- Unificación de varios CSVs con datos separados por género a un único archivo.
- Creación de la columna *Gender* y asignación de valores según los datos.
- Separación de un CSV en varios archivos según su indicador.
- División de una columna en varias con diferentes valores según el intervalo de confianza en la estimación de un indicador.

Tras la reestructuración, ha habido que **unificar todos los nombres de los países** para quedaran con el mismo valor. Por ejemplo, para aquellos indicadores que tenían Estados Unidos como *United States of America* se transformó para que quedara como *United States*, que es como se encuentra en la variable objetivo.

Finalmente, una vez disponibles todos los indicadores en distintos DataFrames pero con la misma estructura, se ha procedido a una **unión o *mergeo*** según los valores de *Country*, *Year* y *Gender* resultando en un único DataFrame con estos tres valores de columnas más cada uno de los indicadores.

Las transformaciones aplicadas en esta fase son referentes únicamente a la estructura de los datos. La limpieza y filtrado de los datos se llevará a cabo en la fase de Análisis y Preprocesamiento.

Fase de Carga

El conjunto de datos unificado resultante de la fase de transformación se ha cargado en un CSV y se ha almacenado para su posterior accesibilidad.

3.1.4. Análisis exploratorio y limpieza

Se ha llevado a cabo un análisis exploratorio del conjunto de datos unificado con el objetivo de observar las distribuciones que seguían las distintas features, buscar y eliminar valores erróneos, analizar valores atípicos, observar la relación con la variable objetivo y tener una idea general del comportamiento del conjunto de datos y su validez.

Datos erróneos

Mediante un estudio individual de cada feature, se han encontrado valores erróneos que fueron eliminados del conjunto de datos sustituyéndolos por un valor vacío.

Cabe destacar la limpieza de datos de la columna *Country*, en la que un alto número de los valores presentes en el dataset no correspondían a países sino a continentes, regiones, islas, etc. Por ello, se ha extraído una lista con los países reconocidos por la Organización de las Naciones Unidas(ONU) [33] conformada por 193 países miembro y 2 observadores. Se han eliminado los registros de los valores de *Country* no presentes en este conjunto.

Valores atípicos

Los valores atípicos o *outliers* son observaciones numéricamente distantes al resto de datos. Estos valores influyen notablemente en cálculos estadísticos como la media y pueden modificar el comportamiento de los algoritmos de *machine learning*, por lo que es importante detectarlos y estudiarlos.

El procedimiento seguido para detectar los *outliers* de cada característica del conjunto de datos ha consistido en la aplicación de la **regla rango intercuatílico**[34].

El rango intercuatílico(IQR) se define como la resta del primer y el tercer cuartil de una distribución. Los cuartiles son los valores de la división en cuartos de las observaciones. Por ejemplo, el primer cuartil o Q1 corresponde a al valor de la variable tal que la cuarta parte de las observaciones son inferiores o iguales a dicho valor y el resto, superior o igual.

La regla del rango intercuartílico establece como valores atípicos aquellas observaciones inferiores al valor del primer cuartil menos 1.5 veces el rango interecuartílico o superiores al valor del tercer cuartil más 1.5 veces el valor del rango intercuartílico. Es decir, la observación será un valor atípico si está en el rango:

$$(-\infty, Q1 - 1{,}5 \cdot IQR) \cup (Q3 + 1{,}5 \cdot IQR, \infty) \tag{3.1}$$

Tras el estudio de los valores atípicos, se ha llegado a la decisión de mantener estos valores debido a que el conjunto de datos es reducido y estos valores pueden influir positivamente en el aprendizaje del algoritmo destacando casos extremos[35].

Tratamiento de valores absolutos

Para aquellas características o indicadores con valores absolutos, en número de personas, se han pasado a relativos dividiendo entre el valor correspondiente de la feature *Total Population*, obteniendo así el valor relativo en función del tamaño de población del país.

Distribución

La distribución es una característica fundamental a estudiar del conjunto de datos de partida, puesto que un alto número de modelos de *machine learning* asumen su distribución en forma de campana de Gauss[36], también conocida como distribución normal[37].

Para deducir la distribución de los datos de entrada, se ha representado el histograma de cada feature. A pesar de que la distribución gaussiana es la más

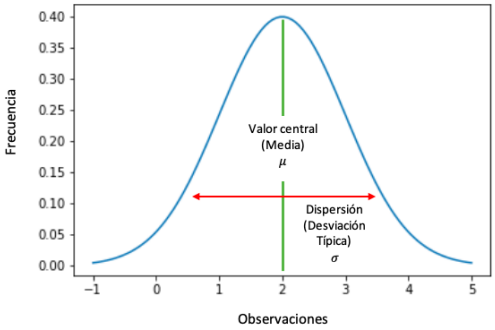

Figura 3.2: Distribución Normal[2]

frecuente[38], en nuestro conjunto de datos tan solo está presente en un reducido grupo de features. La distribución de datos que más abunda en nuestro dataset es la exponencial negativa. Esto se debe a que, para la mayoría de las columnas en nuestro dataset, los valores más frecuentes son pequeños y, a mayor valor, hay un menor número de instancias.

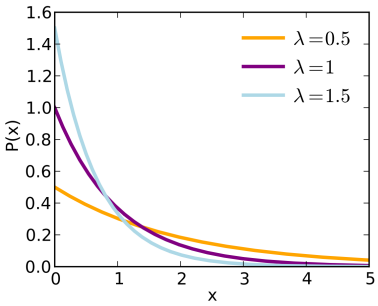

Figura 3.3: Distribución exponencial negativa[3]

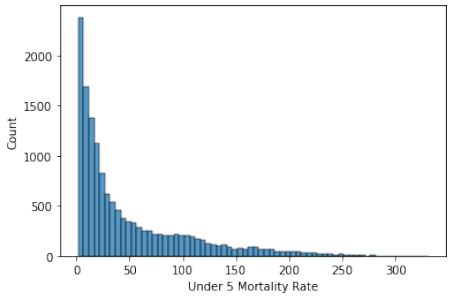

Figura 3.4: Histograma del ratio de mortalidad infantil menor que 5 años

Valores desconocidos

El problema más destacable del conjunto de datos es la ausencia de un alto número de valores. Es decir, se desconoce el valor de muchas de las features en un año concreto o en un país determinado. Se ha estudiado el número de valores descocidos por cada instancia del conjunto de datos, resultando en el histograma de la figura 3.5, que muestra la frecuencia del número de valores desconocidos por fila.

Se ha buscado detectar aquellos países que más valores desconocidos tenían en sus filas. Podemos observar el número de valores vacíos por país en el histograma representado en la figura 3.6.

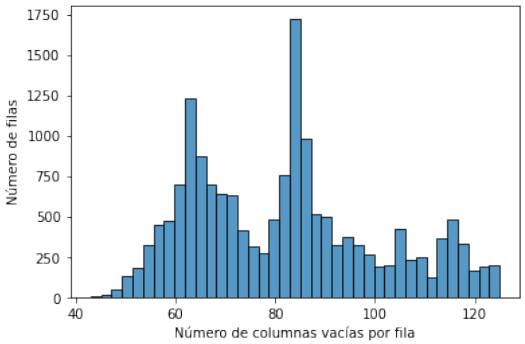

Figura 3.5: Histograma del número de valores desconocidos por registro

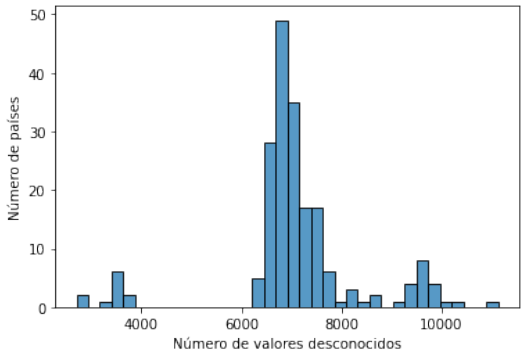

Figura 3.6: Histograma del número de valores desconocidos por país

3.1.5. Tratamiento de valores vacíos

Los modelos de *machine learning* necesitan un conjunto de datos de partida completo para su ejecución. Esto significa que no tolera la presencia de valores desconocidos. Como se ha comentado previamente, en el análisis exploratorio de los datos se encontró un alto número de valores vacíos. Es por dicho motivo, por el que se han aplicado diversas técnicas para dar valor a estos campos o estrategias para eliminarlos.

Interpolación

Se conoce como interpolación al cálculo o estimación de valores intermedios de puntos conocidos.Dichos puntos intermedios se obtienen mediante la aproximación de la función que pasa por los valores conocidos. La interpolación puede ser linear o polinomial(entre otras) dependiendo del tipo de función que se aproxime para obtener los puntos.

La interpolación lineal emplea la función lineal generada por dos puntos exteriores conocidos para calcular los valores intermedios.

La interpolación polinómica define una función polinómica de grado N que pasa por $N+1$ puntos para la obtención de los valores internos a dichos puntos. Podemos observar su comportamiento en la figura 3.7.

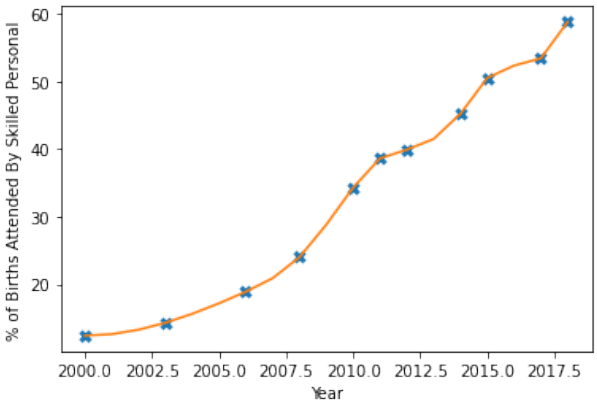

Figura 3.7: Interpolación del porcentaje de partos atendidos por personal sanitario en Afganistán

Se ha aplicado esta técnica matemática[39] para obtener los valores intermedios de las features separándolas por país y género. El eje x de la función de aproximación es el año y en el eje y el valor de la feature. Se ha aplicado una interpolación polinómica de máximo grado 3 para predecir los valores intermedios. Para aquellos casos en los que hay menos de 4 valores conocidos se ha aplicado una interpolación polinómica de grado 2. Por último, para aquellos caso en los tan solo conocíamos los valores de dos años, se ha aplicado interpolación lineal.

Tras la aplicación de esta técnica, se percató de que para ciertos predictores, se habían generado valores imposibles. Como por ejemplo, valores negativos para el número de dentistas por cada 100.000 habitantes. Esto se debía a que, cuando había cambios bruscos que descendían casi a 0, la función de interpolación se salía del rango de valores permitido. Podemos observar este comportamiento en la figura 3.8.

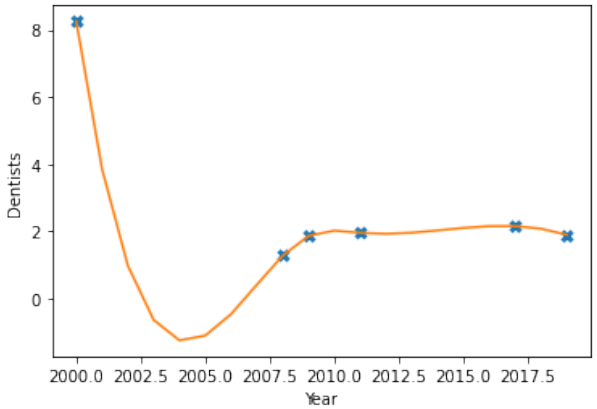

Figura 3.8: Interpolación del ratio de dentistas en República Dominicana

Por este motivo, se modificó esta función para que solo pudiera dar valores dentro del rango establecido por el valor máximo y mínimo de cada feature.

Eliminación de features

En un alto número de indicadores, tan solo se conocían valores de un único año o un conjunto reducido de valores. Debido al alto número de valores desconocidos de estas columnas, se ha procedido a eliminarlas del conjunto de datos. Estas son:

- Low CI Value Air Pollution Death Rate Lower Respiratory Infections
- High CI Value Air Pollution Death Rate Lower Respiratory Infections
- Air Pollution Death Rate Lower Respiratory Infections Age Standarized
- Low CI Value Air Pollution Death Rate Lower Respiratory Infections Age Standarized
- High CI Value Air Pollution Death Rate Lower Respiratory Infections Age Standarized
- Air Pollution Death Rate Chronic Obstructive Pulmonary Disease
- Low CI Value Air Pollution Death Rate Chronic Obstructive Pulmonary Disease
- High CI Value Air Pollution Death Rate Chronic Obstructive Pulmonary Disease
- Air Pollution Death Rate Chronic Obstructive Pulmonary Disease Age Standarized
- Low CI Value Air Pollution Death Rate Chronic Obstructive Pulmonary Disease Age Standarized
- High CI Value Air Pollution Death Rate Chronic Obstructive Pulmonary Disease Age Standarized
- Air Pollution Death Rate Total
- Low CI Value Air Pollution Death Rate Total
- High CI Value Air Pollution Death Rate Total
- Air Pollution Death Rate Total Age Standarized
- Low CI Value Air Pollution Death Rate Total Age Standarized
- High CI Value Air Pollution Death Rate Total Age Standarized
- Air Pollution Death Rate Trachea Bronchus Lung Cancers
- Low CI Value Air Pollution Death Rate Trachea Bronchus Lung Cancers
- High CI Value Air Pollution Death Rate Trachea Bronchus Lung Cancers
- Air Pollution Death Rate Trachea Bronchus Lung Cancers Age Standarized
- Low CI Value Air Pollution Death Rate Trachea Bronchus Lung Cancers Age Standarized

- High CI Value Air Pollution Death Rate Trachea Bronchus Lung Cancers Age Standarized
- Unsafe Wash Mortality Rate
- Hepatitis B Surface Antigen
- Low CI Value Hepatitis B Surface Antigen
- High CI Value Hepatitis B Surface Antigen
- Reproductive Age Women

Eliminación de países

Se ha observado que gran parte de los registros con mayor número de valores desconocidos, se agrupaban en un número reducido de países. Por dicho motivo, se ha procedido a identificar de qué países se desconocían tantos indicadores y, a continuación, han sido eliminados del conjunto de datos. El criterio elegido para seleccionar el número de valores desconocidos que determinarían la eliminación de un país ha sido la mitad del número total de valores. Es decir, si un país tiene la mitad de sus valores vacíos, será eliminado. Siguiendo esa estrategia, los países eliminados han sido los listados a continuación:

- Bahrain
- Bhutan
- Bolivia
- Brunei
- Burundi
- Comoros
- Democratic Republic of Congo
- Equatorial Guinea
- Eritrea
- Libya
- Micronesia (country)
- Montenegro
- North Korea
- North Macedonia
- Palestine
- Papua New Guinea
- Qatar

- Saint Vincent and the Grenadines
- Serbia
- Seychelles
- Singapore
- Somalia
- South Sudan
- Sudan
- Syria
- Timor
- Tonga
- Turkmenistan

Mediana

La mediana es el punto u observación que divide la distribución de la muestra en dos mitades, es decir, deja el mismo número de observaciones a la derecha y a la izquierda de dicho valor.

Mediante el cálculo de este valor estadístico agrupado por país, se ha procedido a rellenar aquellos valores desconocidos de las features de las que conocíamos valor para uno o varios años. Se han dado valor a aquellas observaciones que no se pudieron interpolar debido a que el año estaba fuera del rango conocido.

Tras estas primeras técnicas de imputación de valores desconocidos y estrategias de eliminación de registros, hemos podido observar el decremento del número de valores vacíos. Podemos consultar esta evolución en las gráficas 3.9 y 3.10.

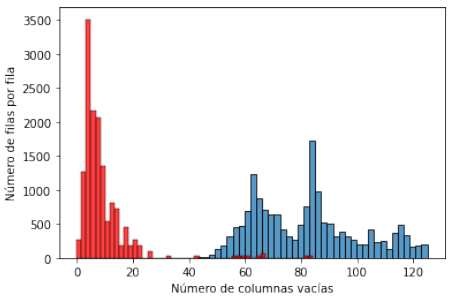

Figura 3.9: En azul: Histograma del número de valores desconocidos por fila original. En rojo: Histograma del número de valores desconocidos por fila tras aplicar las primeras técnicas.

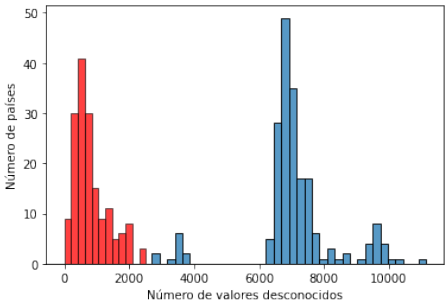

Figura 3.10: En azul: Histograma del número de valores desconocidos por país original. En rojo: Histograma del número de valores desconocidos por país tras aplicar las primeras técnicas.

K-Nearest Neighbors Imputer

El algoritmo *K-Nearest Neighbors Imputer* o Imputador K-Vecinos Más Cercanos, es una técnica de imputación basado en buscar las K muestras más próximas a la observación que queremos estimar[40]. Una vez hallados estos puntos, se le dará el valor siguiendo diferentes posibles técnicas, como la media, la moda o la mediana. Además, este algoritmo es configurable para dar el mismo valor a las K muestras o ponderar su valor en función de la distancia a la que se encuentran de la muestra objetivo.

Se ha aplicado esta técnica de imputación para dar un valor a las observaciones restantes que tenían valores desconocidos. Estas observaciones son indicadores de los que, para ese país, no se conoce su valor para ningún año, por lo que ha de ser deducido a partir del resto.

Antes de aplicar esta técnica se ha separado la variable objetivo del conjunto, para que no influya en el valor asignado por el algoritmo.

Debido a que este algoritmo de imputación de valores desconocidos trabaja mediante distancias, es necesario normalizar los datos para aplicarlo, puesto que sino, las diferentes escalas en nuestro conjunto de datos harán que se generen valores sesgados para los puntos desconocidos. Por tanto, se han escalado los datos en el rango $[0, 1]$ y, tras la imputación con KNN, se ha deshecho esta transformación, volviendo a los valores originales. Además, para conservar la información de los datos categóricos al aplicar esta técnica, se realizó un *One Hot Encoding*, que se deshizo tras la imputación[41].

Tras la aplicación de esta última técnica de imputación, finalmente el conjunto de datos ha quedado libre de valores desconocidos.

3.1.6. Transformaciones aplicadas

Una vez el conjunto de datos está completo, es necesario realizar ciertas transformaciones para poder aplicar los modelos de *machine learning* elegidos con el fin de obtener unos resultados óptimos.

3.1.6.1. Datos categóricos

Los datos categóricos son aquellos tipos de datos cuyos posibles valores están delimitados a un conjunto finito de valores establecidos como categorías. Este tipo de datos, si son no numéricos, son problemáticos para la mayoría de algoritmos de *machine learning*, pues estos emplean operaciones matemáticas para el aprendizaje. Por ello, es necesario transformar estas variables categóricas textuales a valores numéricos.

Los únicos datos categóricos textuales en el conjunto de datos son *Country* y *Gender*. Para la feature *Country* se ha dedicido eliminarla, para que así no influya en el aprendizaje, puesto que el algoritmo podría aprender sencillamente dividiendo por país, no teniendo en cuenta el resto de indicadores cuya influencia queremos estudiar.

Por otro lado, para la feature correspondiente al género, *Gender*, se ha dividido en dos columnas, una para la categoría de hombre(*Male*) y otra para la categoría de mujer(*Female*). Se han establecido dichas columnas a 1 si el valor de *Gender* correspondía a dicho valor y cero en caso contrario. Este tratamiento se conoce como *One Hot Encoding*. Sin embargo, en nuestro caso, para la categoría

Both sexes se ha establecido a 1 ambas columnas, lo que lo hace una variante de esta técnica.

3.1.6.2. Datos continuos

Los datos continuos son aquellos que pueden tomar cualquier valor numérico real. Estos tipos de datos pueden estar en diferentes escalas o rangos. La forma de aprendizaje de un alto porcentaje de los modelos de *machine learning* implica que los valores más altos tendrán una mayor importancia y peso en el modelo. Sin embargo, para dos features diferentes en dos escalas diferentes, el algoritmo no debe dar más importancia a un valor por estar en una escala superior a la otra.

Por este motivo, es importante transformar los datos continuos para que estén comprendidos en el mismo rango y en un orden de magnitud equivalente[42]. Esta transformación será más o menos relevante dependiendo del modelo aplicado. Por ejemplo, para el uso de modelos basados en los árboles de decisión, esta transformación será innecesaria[43]. En cambio, para las redes de neuronas, se trata de un proceso crucial de preprocesado de datos, puesto que hará que la red aprenda mucho más rápido, mediante la aceleración de la convergencia del gradiente.

Normalización

La normalización, en estadística, consiste en la transformación de escala o rango de valores de una distribución de una variable, con el principal propósito de poder hacer comparaciones a un mismo orden de magnitud con otras posibles distribuciones. Existen varias opciones a la hora de aplicar una normalización, dependiendo de las características y distribución de los datos a transformar. Las dos más utilizadas son:

- **Estandarización**: También conocida como puntuación estándar, la estandarización transformará la distribución de datos acomodándola a una distribución normal con media 0 y desviación típica 1. De esta forma, todos los datos inferiores a la media, serán negativos, mientras que los que la superen quedarán positivos. Esta transformación es idónea para aquellas distribuciones que tengan(o se asemejen) a una campana de Gauss. La fórmula para su aplicación es:

$$x' = \frac{x - \mu}{\sigma}$$

  Siendo x el valor a estandarizar, x' el nuevo valor obtenido, μ la media de la distribución original y σ la desviación típica.

- **Escalado Mínimo-Máximo**: Especialmente utilizado en preprocesamiento, el escalado consiste en reescalar el rango de los datos a, generalmente, [0,1] o [-1,1], sin embargo esta transformación permite elegir ese rango según cualquier tupla de valores. La selección de este rango dependerá de la naturaleza de los datos. Se suele aplicar a distribuciones que no permiten una estandarización. La fórmula para el rango [0,1] es:

$$x' = \frac{x - min(x)}{max(x) - min(x)}$$

Siendo x el valor a estandarizar, x' el nuevo valor obtenido, $max(x)$ el máximo valor del conjunto inicial y $min(x)$ el mínimo.

Ajuste de distribuciones

Los algoritmos de *machine learning* tienden a funcionar notablemente mejor cuando los datos siguen una distribución normal[44], puesto que asumen dicha distribución de los datos para hacer las operaciones. Como se ha comentado previamente, gran parte de los datos de los que partimos, al contrario que de costumbre, no siguen ni se asemejan a una distribución normal. Con el objetivo de mejorar el aprendizaje de los modelos, existen diferentes técnicas para transformar la distribución de los datos para que se asemejen a una distribución con forma de campana de Gauss.

Box-Cox Transformation La transformación Box-Cox [45] engloba un conjunto de transformaciones empleadas con el objetivo de corregir los sesgos en la distribución de errores, varianzas desiguales y mejorar la correlación entre las variables. A rasgos generales, lo que se consigue aplicando este tipo de transformaciones es convertir una distribución no normal a una forma normal. Esto se consigue mediante la siguiente lógica:

$$y_i^{(\lambda)} = \begin{cases} \frac{y_i^{\lambda}-1}{\lambda} & si \quad \lambda \neq 0 \\ \\ log(y_i) & si \quad \lambda = 0 \end{cases} \tag{3.2}$$

Siendo y el conjunto de datos inicial, i el índice del conjunto y λ un hiperparámetro a configurar.

El valor óptimo de λ será aquel entre -5 y 5 que minimice la desviación estándar de los datos transformados. No obstante, para poder aplicar esta transformación, es necesario asegurar que los datos no contienen ningún valor negativo o igual a cero, es decir, esta transformación solo es aplicable a datos exclusivamente positivos.

Yeo-Johnson Transformation La transformación Yeo-Johnson [46] es una evolución de Box-Cox que permite transformar datos con valores negativos. Esta transformación se consigue mediante la aplicación de la siguiente fórmula:

$$y_i'^{(\lambda)} = \begin{cases} \frac{(y_i+1)^{\lambda}-1}{\lambda} & si \quad \lambda \neq 0, y_i \geq 0 \\ log(y_i+1) & si \quad \lambda = 0, y_i \geq 0 \\ -\frac{(-y_i+1)^{2-\lambda}-1}{2-\lambda} & si \quad \lambda \neq 2, y_i < 0 \\ -log(-y_i+1) & si \quad \lambda = 2, y_i < 0 \end{cases} \tag{3.3}$$

Siendo y el conjunto de datos inicial, i el índice del conjunto y λ un hiperparámetro a configurar.

Como podemos observar, el comportamiento de esta transformación es muy similar a la Box-Cox para valores positivos, a excepción del incremento en uno de y. Esto sería equivalente a la transformación Box-Cox de $y + 1$. En caso de

que la y sea negativa, la transformación sería una Box-Cox pero en este caso de $-y + 1$ y con lambda $= 2 - \lambda$.

En la figuras 3.11 y 3.12 podemos observar los histogramas de distribución de antes y después de las transformaciones, aplicado a una distribución que se asemeja a una normal y una que no. Podemos apreciar la sutil diferencia entre la transformación Box-Cox y la Yeo-Johnson. Tras dichas transformaciones, también se ha aplicado una estandarización.

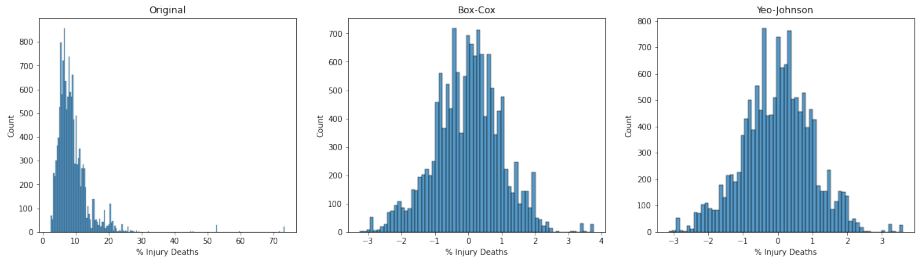

Figura 3.11: Transformaciones sobre una distribución que se asemeja a una normal. Histograma sobre el porcentaje de muertes por heridas.

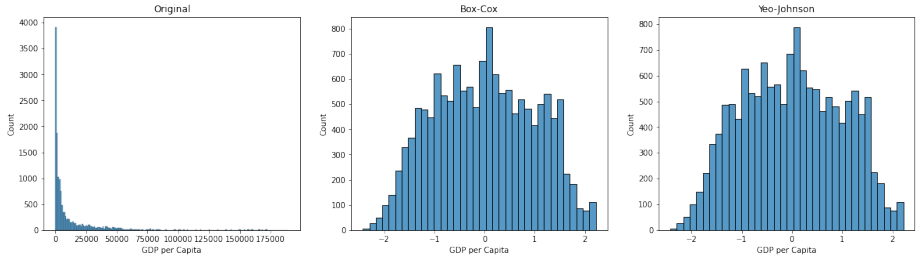

Figura 3.12: Transformaciones sobre una distribución que no se asemeja a una normal. Histograma sobre el PIB per cápita.

En nuestro conjunto de datos, se han aplicado la estandarización para todas aquellas distribuciones que seguían o, por lo menos, se asemejaban a una distribución normal. A continuación, se ha aplicado la transformación Yeo-Johnson a todas las características excepto las referentes al género. Tras esta última transformación, con todas la features con forma de campana de Gauss, se a aplicado una estandarización resultado todas las características en el mismo orden de magnitud con media 0 y desviación típica 1. La estandarización anterior a Yeo-Johnson para aquellas variables que siguen una distribución normal se aplica debido a que tiende a mejorar los resultados[47].

Cabe destacar el trato especial que se ha tenido que aplicar a la variable objetivo *Life Expectancy*. A esta feature no se le ha aplicado la transformación Yeo-Johnson, tan solo se ha estandarizado. El motivo de esta decisión viene dado por las limitaciones de la función de transformación. Con el objetivo de medir de forma más representativa el error de los modelos, es necesario deshacer la transformación aplicada al valor obtenido. Para invertir la transformación aplicada se deshará la estandarización que se hace posteriormente y se empleará la función inversa a Yeo-Johnson, es decir:

$$y_i^{(\lambda)} = \begin{cases} (y' \cdot \lambda + 1)^{\frac{1}{\lambda}} - 1 & si \quad \lambda \neq 0, y_i' \geq 0 \\ 10^{y'} - 1 & si \quad \lambda = 0, y_i' \geq 0 \\ 1 - (-(2-\lambda) \cdot y' + 1)^{\frac{1}{2-\lambda}} & si \quad \lambda \neq 2, y_i' < 0 \\ 1 - 10^{-y'} & si \quad \lambda = 2, y_i' < 0 \end{cases} \tag{3.4}$$

Siendo y el conjunto de datos inicial, i el índice del conjunto y λ un hiperparámetro a configurar.

No obstante, a diferencia de la original, esta función no es continua para todo el conjunto de números reales. En la práctica, esto resultaba en que, para aquellos valores de la esperanza de vida predichos que fueran menores de un cierto valor(concretamente el mínimo valor previo a la transformación), se efectuaría un exponente fraccionario de un número negativo para hacer la inversión, lo que resultaría en un número complejo y, por tanto, no válido.

Una vez concluido el preprocesamiento, ya disponemos de un conjunto de datos unificado, limpio, correcto, ajustado y completo. Por tanto, ya es posible aplicar los modelos de *machine learning*.

3.2. Selección de features

Tras ejecutar una primera iteración del flujo de trabajo sobre el conjunto de datos, se ha detectado, a partir de los resultados preliminares obtenidos (4.5.1), que algunas de las variables más importantes están demasiado relacionadas con la esperanza de vida y su valor depende de factores que no son claramente reconocibles. Por tanto, se ha llegado a la decisión de eliminar aquellos indicadores que no sean influenciables, tratables o modificables mediante políticas de seguimiento de una forma factible y clara. Se han mantenido los factores inalterables como el género y el año.

El objetivo de este cambio es determinar qué indicadores es posible modificar en busca de alcanzar una variación positiva en la predicción de la esperanza de vida del país. De esta forma se podrán determinar puntos clave que podría abordar un país con el objetivo de mejorar el tiempo de vida esperado de dicha población.

Por tanto, se ha procedido a la eliminación de las siguientes features:

- Infant Mortality Rate (Y sus Low CI Value y High CI Value)
- Under 5 Mortality Rate (Y sus Low CI Value y High CI Value)
- % Population Aged 0-14
- % Population Aged 15-64
- % Population Aged 65+
- % Population Aged 65-69
- % Population Aged 70-74
- % Population Aged 75-79
- % Population Aged 80+
- Neonatal Mortality Rate (Y sus Low CI Value y High CI Value)

- Maternal Mortality Rate (Y sus Low CI Value y High CI Value)
- Death Rate
- Total Population

A su vez, con el objetivo de ampliar el rango de factores que caracterizan a un país en un determinado espacio temporal, se han añadido los siguientes nuevos indicadores extraídos de las mismas fuentes que los anteriores:

- **Homicide Rate**: Ratio de homicidios. Número de homicidios por cada 100.000 habitantes.
- **Government Expenditure Education**: Gasto público en educación en porcentaje sobre el gasto público total.
- **Government Expenditure Military**: Gasto público en defensa en porcentaje sobre el gasto público total.
- **Government Expenditure Education**: Gasto público en sanidad en porcentaje sobre el PIB.
- **Diet Composition**: Consumo de tipos de alimentos en kilocalorías por persona al día. Se divide en:

  Sugar: Azúcar

  Oil and Fats: Aceite y grasas

  Meat: Carne

  Dairy and Eggs: Productos lácteos y huevos

  Fruit and Vegetables: Fruta y verduras

  Starchy Roots: Almidón

  Pulses: Legumbres

  Cereal and Grains: Cereal y grano

  Alcoholic Beverages: Bebidas alcohólicas

  Other: Otros alimentos
- **Vegetable Consumption**: Consumo de verduras en kilogramos por persona al año.
- **Fruit Consumption**: Consumo de fruta por tipo en kilogramos por persona al día. Se divide en:

  Bananas: Bananas

  Dates: Dátiles

  Other Citrus: Otros cítricos

  Orange and Mandarines: Naranjas y mandarinas

  Apple: Manzana

  Lemons and Limes: Limas y Limones

  Grapes: Uvas

Grapefruit: Pomelo

Pineapple: Piña

Platains:Plátanos

Other: Otras frutas

- **Cereal Consumption**: Consumo de cereales en kilocalorías por personal al día. Se divide en:

  Oats: Avena

  Rye: Centeno

  Barley: Cebada

  Sorghum: Sorgo

  Maize: Maíz

  Wheat: Trigo

  Rice: Arroz

- **Diet Calories**: Dieta por macronutrientes en kilocalorías por persona al día. Se divide en:

  Animal Protein: Proteína animal

  Plant Protein: Proteína vegetal

  Fat: Grasa

  Carbohydrates: Carbohidratos

3.3. Análisis de correlación

El análisis de correlación entre variables es una de las técnicas más empleadas para interpretar el comportamiento del conjunto de datos respecto a la variable objetivo. Es un primer paso necesario para la construcción de modelos predictivos más complejos.

La correlación entre dos variables evalúa la relación directa entre estas. El índice de correlación es un valor comprendido en el rango $[-1, 1]$. Un valor positivo de este índice indicará una relación directamente proporcional, mientras que los valores negativos señalarán la relación como inversamente proporcional. Cuanto más cerca este del valor cero, menos relación habrá entre las variables estudiadas.

El cálculo de la correlación se puede realizar siguiendo diferentes técnicas. Las más comunes son la **correlación lineal de Pearson** y la **correlación de Spearman**. El coeficiente de correlación lineal de Pearson mide la tendencia lineal entre las variables estudiadas, mientras que el coeficiente de correlación de Spearman mide la tendencia monótona(creciente o decreciente), es decir, que ambas variables se desplacen hacia la misma dirección relativa, aunque no sea de forma constante.

Se ha aplicado este análisis de correlación sobre el conjunto de datos mediante el coeficiente de correlación lineal de Pearson. Este, como se ha comentado previamente, asume la linealidad de los datos, por lo que solo destacará las relaciones lineales, lo que significa que no será excesivamente útil en caso de

que el problema no sea lineal. Sin embargo, es muy útil para tomar un primer contacto con el posible tipo de problema que presentan los datos, pudiendo hacer un análisis preliminar de cuáles serán las features a las que los modelos más sencillos darán una mayor importancia. Además, podremos descartar si se trata o no de un problema lineal.

Podemos observar la matriz de correlación en la figura 3.13. Analizando la matriz, podemos concluir que hay grupos de indicadores que tienen una alta relación entre ellos, como puede ser los referentes a los ratios de personal sanitario por habitante. En cuanto a la variable objetivo, se puede observar una alta correlación con un notable número de features mientras que, con algunas otras, el coeficiente de correlación es cercano a cero. Existen técnicas de filtrado de features para solo seleccionar aquellas que tengan una alta correlación con la variable dependiente, sin embargo, como desconocemos la linealidad del problema, se ha decidido no eliminar estas variables puesto que podríamos estar perjudicando el rendimiento del modelo.

Se ha analizado la pureza de la correlación lineal entre la esperanza de vida y las features más proporcionales a ella observando su comportamiento mediante los gráficos de dispersión desarrollados en el análisis previo de los datos. En el gráfico de dispersión de la figura 3.14 podemos observar de que, para *Income per Cápita*, a pesar de que el coeficiente de correlación es muy alto(0,798), la relación entre ambas variables es claramente no lineal.

Este análisis nos indica que las variables presentes en el conjunto de datos son, en su mayoría, significativas, puesto que tienen una correlación notable con la variable a predecir, aunque no es una relación directa.

3.4. División en conjunto de entrenamiento y test

El conjunto de datos se ha separado en dos, el conjunto de entrenamiento y el conjunto test. Esta división se debe a que necesitamos un conjunto de datos con el que valorar la calidad del modelo. Para ello, esos datos no pueden haberse usado en el aprendizaje.

El conjunto de pruebas debe ser lo suficientemente grande como para poder generar resultados significativos y que sea un conjunto representativo del total para obtener un error válido. De esta forma, el objetivo del modelo será generalizar para obtener las predicciones del conjunto de datos de prueba de forma correcta.

La división suele hacerse en una proporción de 80 % para el conjunto de entrenamiento y 20 % para el conjunto de test. En nuestro caso se ha dividido siguiendo estas proporciones. Dicha separación, se ha realizado de forma aleatoria, confiando en que el azar haga una división equitativa y significativa del conjunto de datos.

No obstante, para el entrenamiento de algunos de los modelo de *machine learning*, es necesario validar cómo de bien está funcionando el modelo. esta comprobación no se puede realizar sobre el conjunto de test, pues debe ser completamente ajeno al entrenamiento y no debe haberse cruzado en ningún momento antes de la evaluación. Por este motivo, para dichos modelos, el conjunto de entrenamiento se ha separado nuevamente, creando así el conjunto de validación, con una proporción menor que el de entrenamiento. Este nuevo

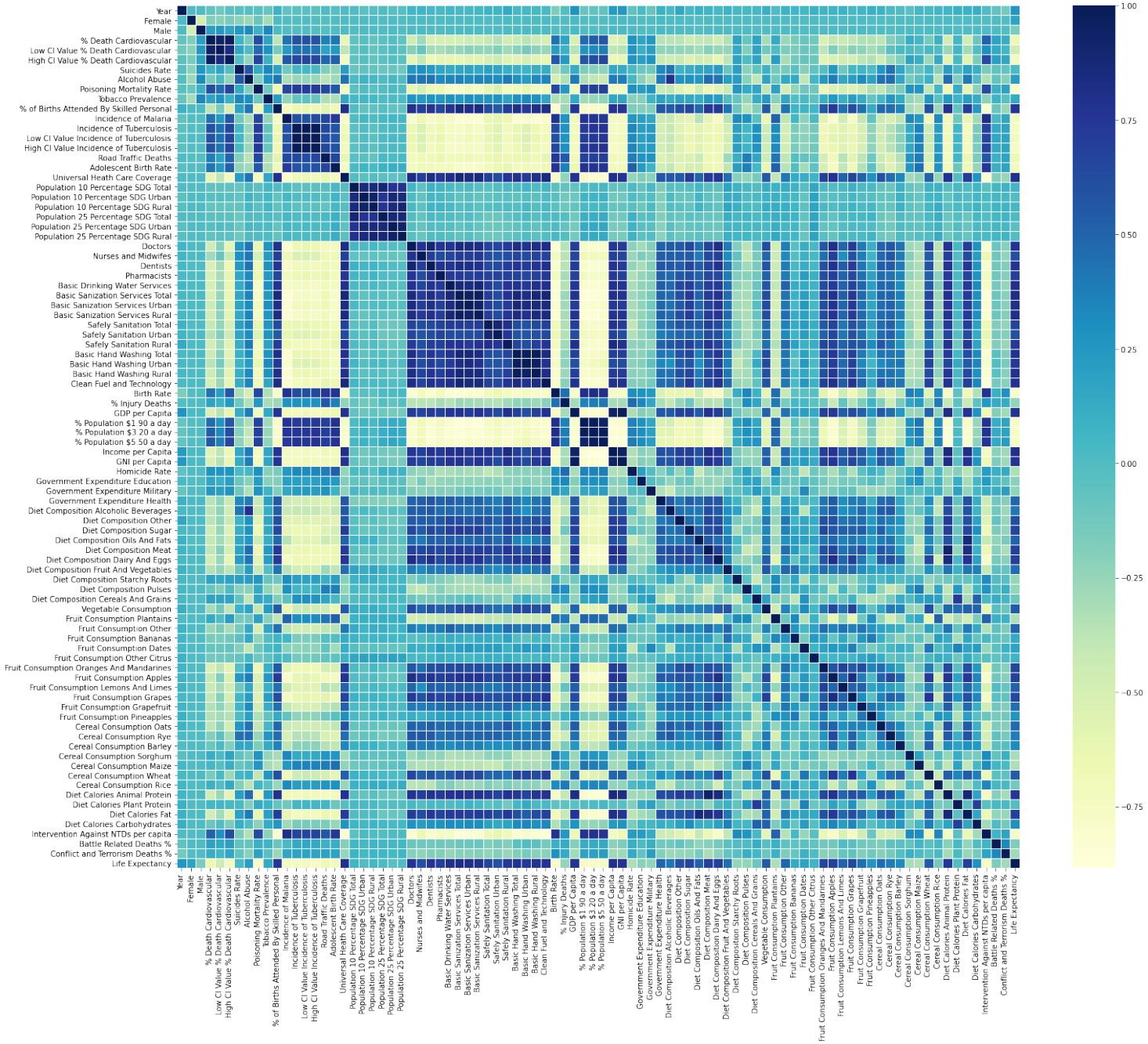

Figura 3.13: Correlación lineal de Pearson entre las features

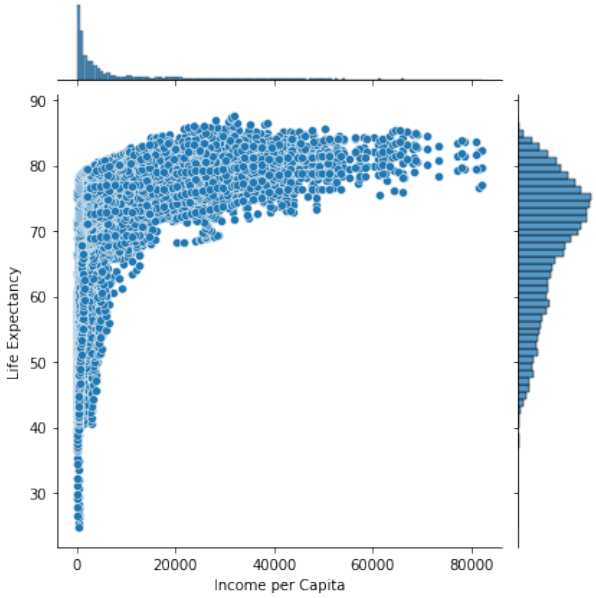

Figura 3.14: Comparación de la esperanza de vida y los ingresos per cápita

conjunto nos permitirá evaluar y validar el modelo durante su entrenamiento, manteniendo el conjunto de pruebas completamente independiente.

3.5. Modelos de Machine Learning aplicados

Para abordar el problema, se decidió aplicar diversos modelos de regresión, de más sencillos a más complejos, para ver gradualmente la progresión del error y el comportamiento distintivo de cada enfoque. Los modelos aplicados han sido los descritos a continuación.

3.5.1. Regresión Lineal Múltiple

La regresión lineal busca encontrar una relación directa y lineal entre la variable de entrada (X) y la variable objetivo (y), también conocida como la variable dependiente[48]. La ecuación que define este modelo es:

$$y = \beta_0 + \beta_1 x \tag{3.5}$$

El algoritmo buscará optimizar los valores desconocidos de β_0 y β_1 que disminuyan la diferencia entre el valor predicho y el valor objetivo. De esta forma, intenta dibujar una recta que aproxime los valores deseados. Podemos observar el comportamiento de este modelo con un conjunto de entrada mediante la gráfica de la figura 3.15

La regresión lineal múltiple es una extensión de la regresión lineal, siendo uno de los modelos más sencillos y utilizados dentro del aprendizaje automático. Este modelo extiende la entrada a un número indeterminado de dimensiones, siguiendo la siguiente ecuación:

$$y = \beta_0 + \beta_1 x_1 + \ldots + \beta_n x_n \tag{3.6}$$

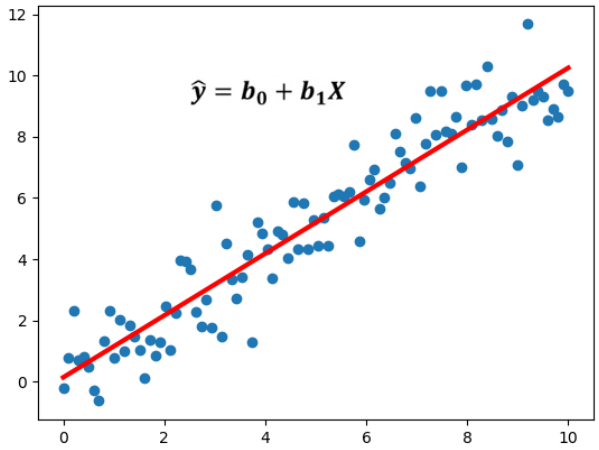

Figura 3.15: Comportamiento de una regresión lineal[4]

Siendo n el número de dimensiones de entrada, $x_1...x_n$ los valores de la entrada de cada dimensión y $\beta_0...\beta_n$ los coeficientes a determinar por el algoritmo. Se ha aplicado este algoritmo sencillo para observar la linealidad entre los predictores y la variable dependiente, además de para obtener una pequeña concepción de la calidad de los datos y la complejidad del tipo de problema al que nos enfrentamos.

3.5.2. K-Nearest Neighbors Regressor

Este método de aprendizaje automático basa su comportamiento en las muestras más cercanas o similares de la que se quiere predecir su valor[48]. Se trata de un modelo basado en instancias, lo que quiere decir que no aprende expresamente un modelo sino que memoriza el conjunto de entrenamiento para predecir el conjunto de test. Su funcionamiento consta de los siguientes pasos:

1. Determinar la distancia entre la muestra que se quiere predecir y el conjunto de entrenamiento.
2. Seleccionar los k puntos más cercanos, conocidos como vecinos o *neighbors*.
3. Predecir el valor de la muestra mediante la media de dichos vecinos, según la ponderación elegida.

Los hiperparámetros a determinar para este modelo son dos:

- k: Número de vecinos a utilizar para hacer la predicción.
- *weights*: Tipo de ponderación elegida para establecer la predicción. Se puede ponderar de forma *uniforme*, de tal manera que todos los vecinos tienen el mismo peso sobre el resultado obtenido; o por *distancia*, es decir, dando más valor a aquellos vecinos más cercanos.

Podemos observar el comportamiento de este modelo en la gráfica de la figura 3.16 para un problema de una dimensión de entrada.

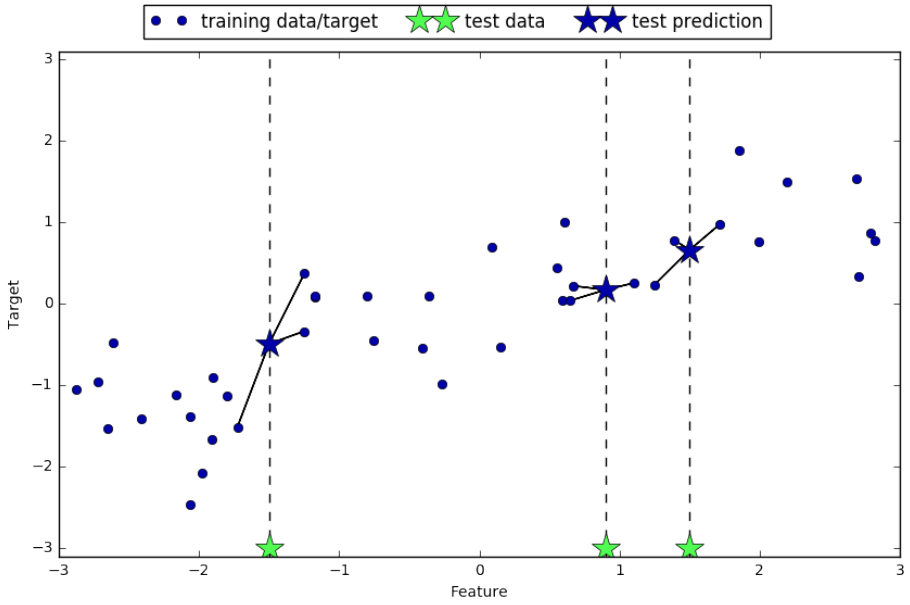

Figura 3.16: Comportamiento de la regresión KNN[5]

Se ha aplicado *k-Nearest Neighbor* para observar los resultados con un modelo sencillo que, a diferencia de la regresión lineal múltiple, no exige linealidad en la variable objetivo. Sin embargo, cabe destacar que el aprendizaje y predicción de este modelo no es recomendable para conjuntos de datos de un alto número de filas o de predictores, siendo este último nuestro caso. La alta dimensionalidad hace que el tiempo de ejecución, tanto de aprendizaje como de predicción, sea notablemente alto y exige una alta capacidad computacional.

GridSearch

Determinar los hiperparámetros de un modelo no es sencillo y en numerosas ocasiones el mejor método de optimización de hiperparámetros es la realización de pruebas con la combinación de múltiples posibles valores.

Este ha sido el procedimiento seguido para establecer los hiperparámetros para el modelo de KNN Regressor. Estas pruebas se han implementado mediante el método *GridSearch*, disponible en la librería de *sklearn*[17]. Se han efectuado estas pruebas con los valores de número de vecinos: 2,3,4,5 y 10 y con los dos posibles tipos de pesos, por distancia y uniforme, resultando en el menor error con k=4 y peso por distancia.

3.5.3. Random Forrest Regressor

Un **árbol de decisión** es un algoritmo de aprendizaje automático que basa su comportamiento en la toma de decisiones en forma de árbol. En este árbol de decisión, cada nodo establecerá una condición sobre una feature del conjunto de entrada, cuya respuesta será booleana, es decir, verdadera o falsa. Dicho nodo se dividirá en dos ramas, una para cada posible respuesta. Finalmente, a través

de los nodos de condición se llegará al nodo hoja, el cual establecerá el valor de la predicción sobre la muestra analizada. El árbol de decisión para regresión se construye usando una algoritmo voraz que optimiza la siguiente función de coste:

$$J(a, l_a) = \frac{m_{izquierdo}}{m} MSE_{izquierdo} + \frac{m_{derecho}}{m} MSE_{derecho} \tag{3.7}$$

Siendo a un atributo o feature, l_a el límite del atributo, m el número de muestras y MSE el error cuadrático medio. El algoritmo voraz establecerá qué atributos y qué límites son los mejores para la toma de decisiones. Podemos ver el comportamiento de un árbol de decisión para una regresión en la figura 3.17.

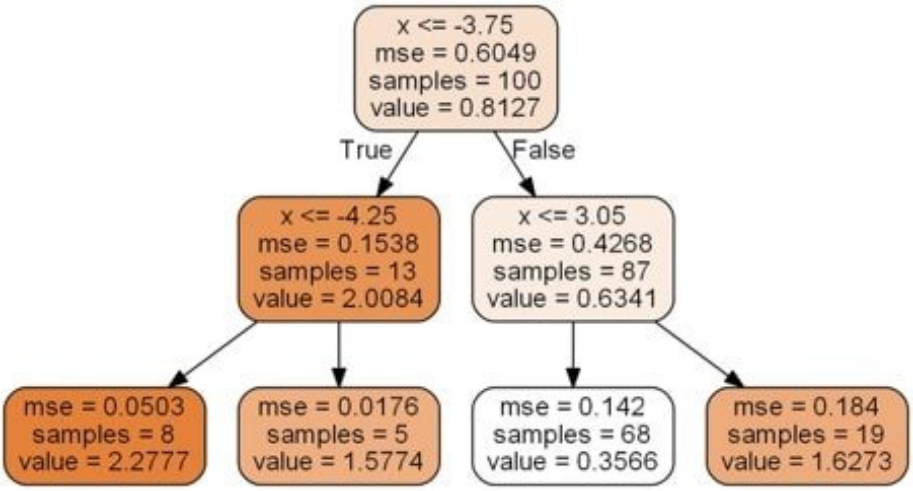

Figura 3.17: Árbol de decisión para una regresión[6]

El regresor **Random Forest** o **Bosque aleatorio** consiste en un conjunto o *ensemble* de árboles de decisión con *bagging*, lo que significa que se generan árboles con porciones distintas del conjunto de entrenamiento. Estos árboles se generarán de manera aleatoria, acelerando así su ejecución. El resultado final para este regresor será la media aritmética del resultado de cada árbol aleatorio. Los hiperparámetros a destacar son:

- Número de árboles en el bosque
- Profundidad máxima
- Número mínimo de muestras para ser dividir un nodo interno
- Número mínimo de muestras requerido para ser un nodo hoja

A diferencia de lo que puede parecer intuitivamente, este modelo de *machine learning* suele obtener notablemente buenos resultados, puesto que al haber un número tan alto de árboles, el error se compensa entre ellos. Además, cabe destacar su mayor capacidad de generalización respecto a árboles de decisión comunes y su capacidad de procesar un alto número de dimensiones. Otra ventaja de este modelo es que puede devolver la importancia que le da a cada una de las features que usa, muy útil para analizar su comportamiento.

No obstante, este modelo tiene la desventaja de no poder predecir valores fuera del rango de entrada, limitándolo notablemente.

El bosque aleatorio aplicado para obtener la esperanza de vida se ha implementado con una profundidad máxima de árbol de 100 nodos y un tamaño de bosque de 100 árboles(puesto que aumentando este número no se obtenían resultados significativamente mejores y aumentaba el tiempo de cómputo).

3.5.4. Red de Neuronas

Las redes de neuronas se basan en el comportamiento neuronal del cerebro humano[49]. Lo forman un conjunto de nodos llamados neuronas artificiales que están conectadas entre sí y transmiten una señal. Según la topología de la red, existen diversos tipos de redes neuronales como son:

- Perceptrón simple
- Perceptrón multicapa
- Red Neuronal Convolucional
- Red Neuronal Recurrente
- Redes de Base Radial

El tipo de red de neuronas aplicada en este trabajo ha sido el perceptrón multicapa. El perceptrón multicapa es una extensión del perceptrón simple.

3.5.4.1. Perceptrón simple

El perceptrón simple es una red de neuronas con una única capa, la capa de salida. Todas las entradas se transmiten a las neuronas que conforman esta capa. Una neurona consta de unas entradas, cada cual tendrá un peso asociado que se irá modificando; un bias, que será un número real que también se actualizará; y una función de activación, la cual para el perceptrón simple siempre será la función escalón, que se define como:

$$f(x) = \begin{cases} 0 & si \quad x < 0 \\ 1 & si \quad x \geq 0 \end{cases} \tag{3.8}$$

La inicialización tanto de los pesos de las entradas como del bias se hará, comúnmente, de forma aleatoria.

El funcionamiento de una neurona de la capa de salida consta de dos operaciones: propagación y actualización de pesos.

Propagación

La propagación determinará la salida de la red, y se obtendrá mediante el cálculo de la siguiente operación:

$$s = \sum_{i=1}^{n} F_{activacion}(e_i \cdot w_i) + b \tag{3.9}$$

Siendo $F_{activacion}$ la función de activación, n el número de entradas, e el valor de la entrada, w el valor de los pesos y b el bias. Podemos observar el comportamiento de una neurona de un perceptrón simple en la figura 3.18.

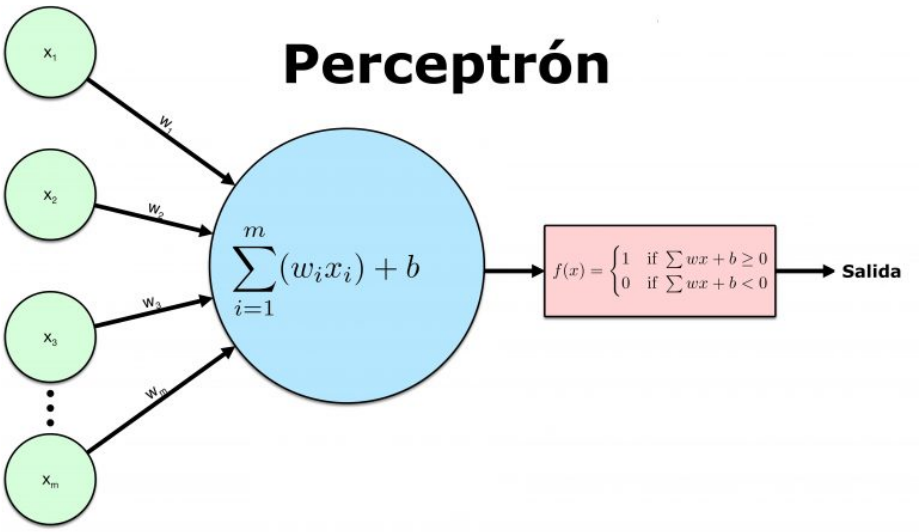

Figura 3.18: Neurona Artificial[7]

Actualización de pesos

Los pesos se actualizarán conforme a al error otorgado por la función de error o *loss*, la cual suele ser la diferencia entre el valor esperado y el obtenido($d - s$). La función que determina la actualización de los pesos es:

$$\Delta W = \alpha \cdot e \cdot F_{loss} \tag{3.10}$$

Con W como los pesos, e como la entrada, F_{loss} como la función de error y α el factor de aprendizaje.

El factor de aprendizaje es un escalar de un valor reducido que controla la variación del incremento de los pesos. Cuanto mayor sea el factor de aprendizaje más rápido será el modelo, sin embargo será más sencillo que alcance un mínimo local o que el error oscile. Cuanto más reducido sea este valor, más lento será el aprendizaje, sin embargo, mejor resultados se obtendrán.

Para la actualización del bias se hará:

$$\Delta b = \alpha \cdot F_{loss} \tag{3.11}$$

Las operaciones de propagación y retropropagación se realizarán para todo el conjunto de entradas. Esto se conoce como un *epoch*. Cuanto mayor sea el número de *epochs* más aprenderá la red, sin embargo, sobrepasado un cierto límite, se puede llegar a alcanzar sobreajuste.

El perceptrón simple es la más sencilla de las redes de neuronas, aunque asienta la base para el perceptrón multicapa. Se caracteriza porque solo es capaz de resolver problemas lineales.

3.5.4.2. Perceptrón multicapa

El perceptrón multicapa extiende el comportamiento del perceptrón simple añadiendo capas intermedias entre las entradas y la capa de salida, estas son

conocidas como capas ocultas. El problema que plantean las capas ocultas es el desconocimiento del valor de la salida deseada para esas capas, por tanto no se puede calcular la actualización de los pesos.

La solución que implementa el perceptrón multicapa para sortear este problema es la retropropagación del gradiente.

Retroprogación del gradiente

El gradiente es la derivada direccional que resulta en la dirección de máximo crecimiento de la función, pudiendo obtener así obtener la dirección vectorial para converger hacia donde el error es menor. Esta técnica utiliza la derivada de la función de activación. Sin embargo, la función escalón previamente descrita no posee derivada, por lo tanto no es aplicable al perceptrón multicapa. En sustitución, las más utilizadas son las citadas en la tabla 3.2.

Función	Fórmula	Derivada
Lineal	$f(x) = x$	$f'(x) = 1$
Sigmoidal	$f(x) = \frac{1}{1+e^{-x}}$	$f'(x) = f(x) \cdot (1 - f(x))$
ReLU	$max(0, x)$	$f'(x) = 1$ si $x > 0$

Tabla 3.2: Funciones de activación derivables

Normalmente, para redes de neuronas con un número reducido de capas ocultas, se suele utilizar la función sigmoidal como función de activación. No obstante, conforme el número de capas ocultas aumenta, al utilizar esta función o la función lineal, se presenta el problema de desvanecimiento del gradiente.

El desvanecimiento de gradiente afecta en la retropropagación del gradiente, haciendo que, cuantas más capas se retropropagan, menor es el el valor del gradiente y, por ende, la propagación de los pesos. De tal forma, la red neuronal no es capaz de aprender.

Este problema no ocurre con la función lineal, puesto que su derivada es una constante. El defecto de esta es que, al ser precisamente lineal, al aplicarlo, todas sus capas serían equivalentes a una única operación matricial, lo que equivaldría a un perceptrón simple, el cual tan solo puede resolver problemas lineales, como se comentó previamente.

Por tanto, la función necesaria había de ser derivable, no lineal y diferenciable.

Como solución se halló el uso de la función ReLU, es decir, la Función de Activación Lineal Rectificada. La cual permitía la retropropagación sin su desvanecimiento y no era lineal. La desventaja de esta función de activación es que nunca podrá devolver valores negativos[50].

Finalmente, las ecuaciones de retropropagación de gradiente quedarían:

$$\Delta W^{(k)} = \alpha \cdot s^{(k-1)} \cdot \delta^{(k)}$$

$$\Delta b^{(k)} = \alpha \cdot \delta^{(k)}$$

Para la capa de salida:

$$\delta^{(n)} = F_{loss} \cdot f'_{activacion}(s^{(n)})$$

Para las capas ocultas:

$$\delta^{(k)} = W^{(k+1)} \cdot \delta^{(k+1)} \cdot f'(s^{(k)})$$

Siendo W los pesos, k el número de capa, α el factor de aprendizaje, b el bias, n el número total de capas y s la salida.

En la figura 3.19 podemos ver la estructura de un perceptrón multicapa con una capa oculta densa.

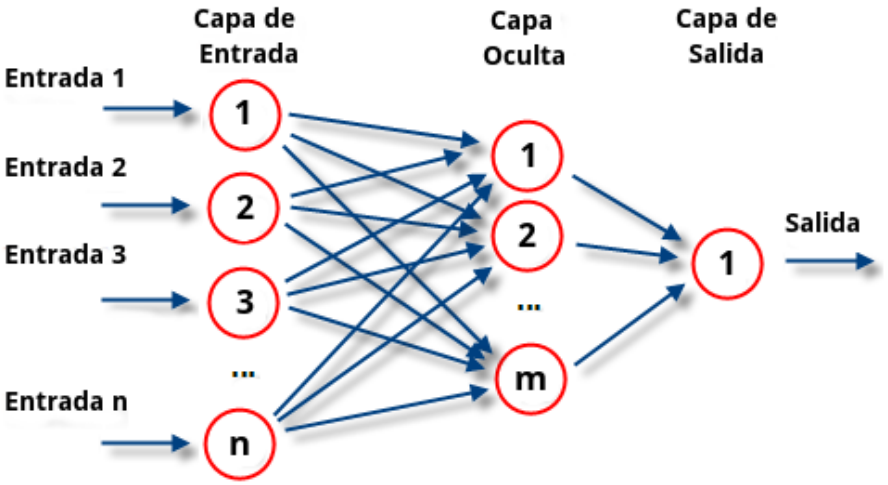

Figura 3.19: Perceptrón multicapa[8]

Aplicación

La red de neuronas utilizada para predecir la esperanza de vida consiste en un perceptrón multicapa que cuenta con dos capas ocultas de 350 neuronas y una función de activación ReLU. La capa de salida tiene una única neurona que devolverá la esperanza de vida. Esta última capa tiene una función de activación linear. Esto es debido a que los datos han sido estandarizados, por tanto, la esperanza de vida comprende en un rango de valores tanto positivos como negativos, motivo por el cual no es posible utilizar la función ReLU, que solo devolverá valores positivos como se comentó anteriormente. Tras la realización de pruebas con más y menos capas ocultas y neuronas, se llegó a la conclusión de que esta era la mejor estructura debido a que es suficientemente compleja para resolver el problema pero no demasiado como para caer en *overfitting*, es decir, que el modelo se ajuste excesivamente a los datos de entrenamiento.

La función de *loss* aplicada ha sido el error cuadrático medio, conocido como MSE.

El factor de aprendizaje elegido, buscando reducir la oscilación del error y evitar alcanzar mínimos locales, ha sido muy bajo, de 0.0001. Además, se ha usado el optimizador Adam[51]. El optimizador Adam(*adaptive moment estimation*) es una extensión del descenso de gradiente, es decir, se usa en la actualización de los pesos de la red. Mientras que el descenso de gradiente clásico mantiene un factor de aprendizaje para todas las actualizaciones de los pesos,

Adam varía este valor según los momentos del gradiente. Existen otras técnicas de optimización para redes de neuronas muy conocidas como RMSprop(de hecho Adam se basa en esta técnica), sin embargo, se obtienen peores resultados para nuestro problema.

El aprendizaje de la red está muy condicionado por el número de *epochs* que se entrenará la red. La estrategia aplicada para determinar este número ha sido la parada temprana o *early stop*[52]. Para aplicar esta técnica se reservará una pequeña proporción del conjunto de entrenamiento como conjunto de validación, con el que la red no aprenderá. Se probará la red de neuronas sobre conjunto de validación al finalizar cada *epoch*. *Early stop* establece que, si el error de validación aumenta durante un determinado número de *epochs*(en nuestro caso se ha elegido 30 debido a la alta oscilación del error), se dará por finalizado el aprendizaje de la red de neuronas, antes de completar el número de *epochs* establecido, de ahí *early stop*. En caso de que el error de validación nunca empeore, se ejecutarían tantos *epochs* como estuvieran establecidos, en nuestro caso 1000.

3.5.5. k-Fold Cross Validation

La validación cruzada o *cross validation* surge de la base de que, al realizar la separación en conjunto de entrenamiento y conjunto de test de forma aleatoria, es posible que se produzca un sesgo, no habiendo en el conjunto de entrenamiento un tipo de dato conciertas características[48].

Para evitar este azar, se utiliza la validación cruzada. Esta divide en k separaciones diferentes de forma aleatoria, a las cuales aplica el modelo y mide el error, la media de estas métricas será el error del modelo. Podemos observar el comportamiento de esta división en la figura 3.20.

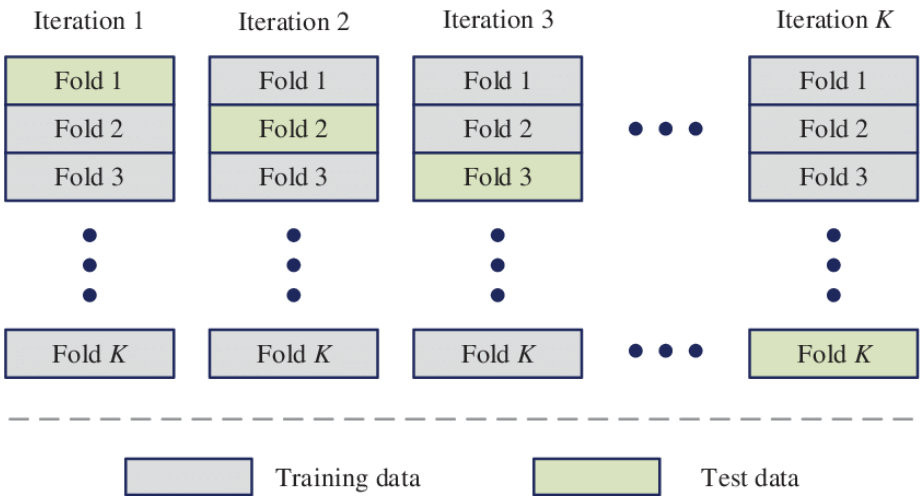

Figura 3.20: K-Fold Cross Validation[9]

No ha sido posible aplicar validación cruzada a los modelos de *Random Forest* y perceptrón multicapa debido a la complejidad computacional de ambos modelos. En cuanto a la regresión lineal, debido a que tan solo se ha usado como referencia para aplicar modelos más complejos, tampoco se ha visto la

necesidad de aplicar esta validación. Sí se ha aplicado en cambio en el modelo de *KNN Regressor*, con una k igual a 5. Es importante destacar que esta división se ha hecho sobre el conjunto de entrenamiento, dejando el conjunto de test completamente libre de todo contacto con el modelo, para así hacer un error más fiable. Esta división del conjunto de entrenamiento se llama conjunto de validación.

Se puede observar al detalle el proceso seguido en la figura 3.21.

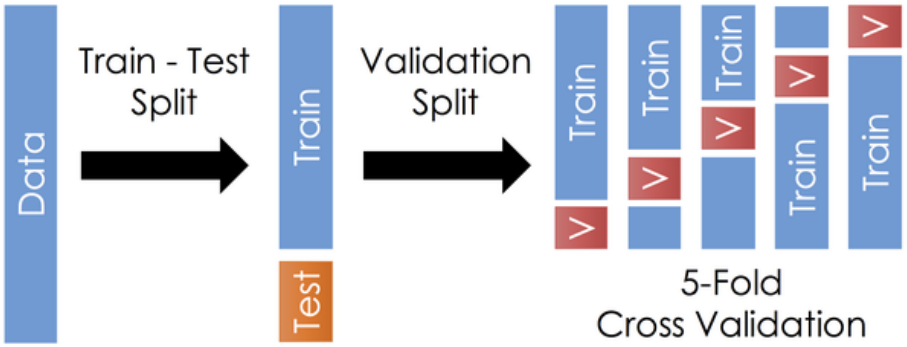

Figura 3.21: División de entrenamiento, validación y test[10]

3.6. Técnicas aplicadas para el análisis de resultados

3.6.1. Wrapper Methods

Los *wrapper methods* son una técnica de selección de features en la que se valoran las features en función de cuánto mejora el error del modelo con o sin ella[49]. Funcionan entrenando y evaluando el modelo con un subconjunto de features, de forma que se elegirán aquel subconjunto de features que menor error tengan o más lo reduzcan.

La selección u orden de elección de estos subconjuntos de predictores con los que se entrenará y evaluará el modelo, se puede realizar desde varios enfoques posibles:

- **Forward Feature Selection**: Irá añadiendo de una en una cada feature al modelo partiendo desde cero.
- **Backward Feature Elimination**: Al contrario que la estrategia anterior, esta partirá del modelo con todas las features e irá eliminando una a una del subconjunto.
- **Exhaustive Feature selection**: Es el equivalente a un enfoque por fuerza bruta, irá probando cada combinación posible de features hasta encontrar el subconjunto óptimo.

Debido a la complejidad computacional y el tiempo de ejecución que lleva el entrenamiento de una red de neuronas, además de la enorme cantidad de features de la que se compone el conjunto de datos, se ha descartado el método de selección exhaustivo. Por tanto, se ha optado por aplicar los otros dos enfoques.

El orden de la creación de subconjuntos, ya sea añadiendo o eliminado features puede o no puede tener un cierto criterio. En nuestro caso, para intentar captar el subconjunto más relevante, se ha decidido ordenar las features según dos criterios:

1. Correlación con la variable objetivo.
2. Según la importancia obtenida a partir de los resultados del modelo de *Random Forest Regressor*.

3.6.2. Algoritmo Genético

El algoritmo genético[53] es una técnica de búsqueda y optimización basada en la teoría del origen de las especies de Charles Darwin, también conocida como la selección natural[54]. Esta enuncia tres principios fundamentales:

1. Cada individuo tiende a transmitir sus rasgos a su progenie.
2. La naturaleza produce individuos con rasgos diferentes.
3. Los individuos más adaptados tienden a producir más progenie.

Este algoritmo parte de una población inicial, cuyos individuos son posibles soluciones al problema, ya sean válidas, no válidas, de mayor o menor calidad. La calidad de la solución, conocida como *fitness*, será el valor a maximizar por el algoritmo. Para ello, hará uso de sus tres operaciones fundamentales: selección, emparejamiento y mutación.

Inicialización

La población inicial estará formado por N individuos. Cada individuo corresponde a un cromosoma, una cadena de genes que representan una posible solución al problema. Dependiendo del problema, el tipo de cada gen puede variar. En nuestro caso, cada gen del cromosoma corresponderá al valor de una feature. Por tanto, cada cromosoma será un número real dentro de un rango a especificar, este se llamará alfabeto. El alfabeto es el conjunto de posibles valores de un gen.

Función de calidad

La calidad de un individuo viene determinado por la función calidad o *fitness*, que el algoritmo tratrá de maximizar. Dado un cromosoma, esta función calculará la calidad de dicha solución y, por tanto, la calidad del individuo. Cuanto mayor sea este valor, más adaptado al medio se considerará dicho individuo y, siguiendo las leyes de Darwin, más descendencia dejará.

Selección

La selección es el proceso por el cual se eligen ciertos individuos de la población para emparejarse y crear descendencia. Existen varias estrategias para afrontar este proceso de selección, las más conocidas son:

- **Aleatorio**: Los individuos se eligen de forma aleatoria equiprobable.

- **Ruleta**: Los individuos más adaptados tienen una mayor probabilidad de ser seleccionados. Debido a la mala calidad inicial de los individuos, esta técnica puede dar malos resultados. Se puede mejorar aplicando lo que se conoce como normalización, la cual aplica la función exponencial de la calidad del individuo para realizar la selección.
- **Selección por torneo**: Se eligen aleatoriamente T individuos que competirán para determinar quién se reproducirá, ganando el que mayor calidad tenga.

Emparejamiento

El emparejamiento consiste en la división y unión de dos individuos para crear descendencia. Cada combinación de individuos producirá dos hijos, que transmitirán el material genético de los padres. La estrategia de emparejamiento variará dependiendo del tipo de problema, ya sea clásico, permutación, numérico, etc. Para nuestro caso, el emparejamiento se realiza siguiendo las siguientes fórmulas:

$$\begin{aligned} a' &= \beta \cdot a + (1 - \beta) \cdot b \\ b' &= (1 - \beta) \cdot a + \beta \cdot b \end{aligned} \tag{3.12}$$

Donde a y b son los cromosomas padre, a' y b' los hijos y β un número aleatorio dentro del rango $(0, 1)$.

Mutación

La mutación es el proceso con el que se introduce nuevo material genético en la población. Mutar consiste en variar los genes de un cromosoma mediante una probabilidad. Para ello se recorrerá el cromosoma gen a gen. La probabilidad de mutación determinará si se mutará o no el gen. En caso de mutación, el nuevo valor será un elemento del alfabeto elegido de forma aleatoria de forma equiprobable.

Estas tres operaciones se ejecutarán sobre toda la población, creando una nueva generación.

Aplicación

La estrategia aplicada para determinar la importancia de los indicadores mediante este algoritmo se ha apoyado en el artículo de Federico Piccinini[26] sobre este tema y ha sido la siguiente lógica:

Dado un caso real de un país en un año determinado y un género concreto(o ambos), se buscará optimizar las features que maximicen la esperanza de vida modificando lo menor posible los valores originales de las features. De tal forma que aquellas features que más se vean modificadas, significarán una mayor importancia sobre el cálculo de la variable objetivo.

Por tanto, necesitaremos que la solución sea lo más parecida a la entrada, pero la salida sea lo más alejada(positivamente) de la predicción original. Para ello, se ha implementado la función de fitness que tendrá los siguientes hiperparámetros:

- *margin_input*: Margen de modificación de la entrada. Máxima diferencia permitida entre la entrada original y los valores del cromosoma(en suma

total). Es decir, cuánto se pueden diferenciar la entrada original de la solución propuesta.

- w_{input} y w_{output}: Pesos de la importancia sobre la similitud de la entrada y la salida del cromosoma. En total sumarán 1.

El cálculo de la calidad sigue el siguiente procedimiento:

Entrada Para determinar la calidad de la entrada se ha calculado la suma de las diferencias de cada feature respecto al valor original. Las features referentes al año y al género no deben verse modificadas, puesto que se está buscando analizar la influencia de los indicadores, no de estos dos factores. Por tanto, con el objetivo de penalizar la diferencia de estas features no mutables respecto a la entrada original, se multiplicará su diferencia por 100.

Como este valor obtenido se quiere minimizar, se le ha aplicado el inverso para que se pueda maximizar, puesto que el algoritmo genético solo maximiza la calidad.

Una vez la diferencia esté por debajo del umbral impuesto como *margin_input*, la calidad será igual a 1.

$$dif_{input} = \left(\sum_{no\ mutables} |x_{original} - x_{cromosoma}| \right) \cdot 100 + \sum_{mutables} |x_{original} - x_{cromosoma}|$$

$$f_{input} = \begin{cases} \frac{1}{1+dif_{input}} & si \quad dif_{input} \geq magin\_input \\ 1 & si \quad dif_{input} < magin\_input \end{cases} \tag{3.13}$$

Siendo x el valor de cada feature, el conjunto *no mutables* las features *Year, Male* y *Female* y el conjunto *mutables* el resto.

El cromosoma será una lista que contiene los valores de cada una de las features que el modelo toma como entrada.

Predicción La calidad de la salida será siempre cero, hasta que la similitud de la entrada llegue hasta el valor de margin_input, que determinará cuánto se permite cambiar la entrada como máximo. Una vez se alcance este punto, se calculará la calidad de la salida como la diferencia entre el valor calculado por la red de neuronas y el valor original de la esperanza de vida. Para da una mayor importancia a esta diferencia a maximizar, se eleva al cubo dicho valor tras haber sumado 1(para no penalizar los valores entre 0 y 1).

$$f_{output} = \begin{cases} (le\_predicted - orginal\_le + 1)^3 & si \quad difinput < margin\_output \\ 0 & si \quad dif_{input} \geq margin\_input \end{cases} \tag{3.14}$$

Donde *le_predicted* es el valor de la esperanza de vida del cromosoma y *original_le* el valor original.

Unión Finalmente, se suman las diferencias multiplicadas por sus pesos.

$$f = w_{input} \cdot f_{input} + w_{output} \cdot f_{output} \tag{3.15}$$

Los valores de los pesos de entrada y salida escogidos han sido 0.1 para w_{input} y 0.9 para w_{output} para dar una mayor importancia al resultado de salida una vez se han alcanzado unos valores lo suficientemente parecidos a las entradas. La técnica de selección elegida ha sido ruleta y normalización con probabilidad de cruce de 0.7. La probabilidad de mutación establecida ha sido de 0.75. Se ha escogido una probabilidad tan alta para que el algoritmo pruebe un mayor número de soluciones diferentes, ampliando el espacio de búsqueda. Para contrarrestar esta decisión y no perder la mejor solución se ha usado elitismo, por el cual el mejor individuo de una generación pasa a la siguiente siempre sin verse modificado. Por último, se ha escogido un tamaño de población de 100 individuos.

La ejecución del algoritmo genético se ha llevado a cabo mediante Salga, implementación con interfaz visual del algoritmo genético implementada por la Universidad Politécnica de Madrid[19].

3.6.3. Shapley Additive Explanations (SHAP)

SHapley Additive exPlanations o SHAP para abreviar, es una técnica de interpretación de modelos de caja negra aplicable a cualquier tipo de modelo de *machine learning*. Se entiende por caja negra, aquellos modelos que disponen de unas entradas y devuelven unas salidas, pero es posible entender la toma de decisiones que el modelo lleva a cabo para determinar el *output*.

El algoritmo, desarrollado por Lundberg and Lee en 2017[24], se basa en los valores de Shapley[55], un concepto extraído de la teoría de juegos o *game theory*. La teoría de juegos lo conforman un juego y sus jugadores. La suma de las aportaciones de cada jugador determinarán el estado en el que acabará el juego. Esta teoría se traspasa a la explicación de modelos interpretando la resolución del juego como la salida del modelo de caja negra y los jugadores como las distintas features que conforman las entradas.

El valor que aporta cada jugador al resultado final lo especifica su valor de Shapley. En nuestro caso, los valores que determinan la contribución cuantificativa de cada feature al resultado final calculado por el modelo se conocen como *SHAP values* o valores SHAP.

El juego al que aportan ese valor, para nuestro algoritmo, se corresponde a una única predicción. De forma que para cada conjunto de entradas se tendrá un valor SHAP diferente en cada feature.

El cálculo de los *SHAP values* se realiza mediante la contribución marginal de cada feature al resultado final. La contribución marginal será la diferencia entre la predicción del modelo incluyendo una feature y no incluyéndola.

Para calcular la contribución marginal de cada feature, se ejecutará el modelo 2^F veces, siendo F igual al número de features. En cada ejecución se incluirá el valor de diferentes combinaciones features, empezando desde cero hasta acabar con todas. El valor de una feature no incluida en la ejecución será igual a la media aritmética. En nuestro caso, al estar todas estandarizadas, ese valor será siempre igual a 0.

Finalmente, nos queda un árbol de ejecución como el de la figura 3.22 para $F = 3$, donde f es cada feature con y R cada resultado, obteniendo $2^F = 2^3 = 8$

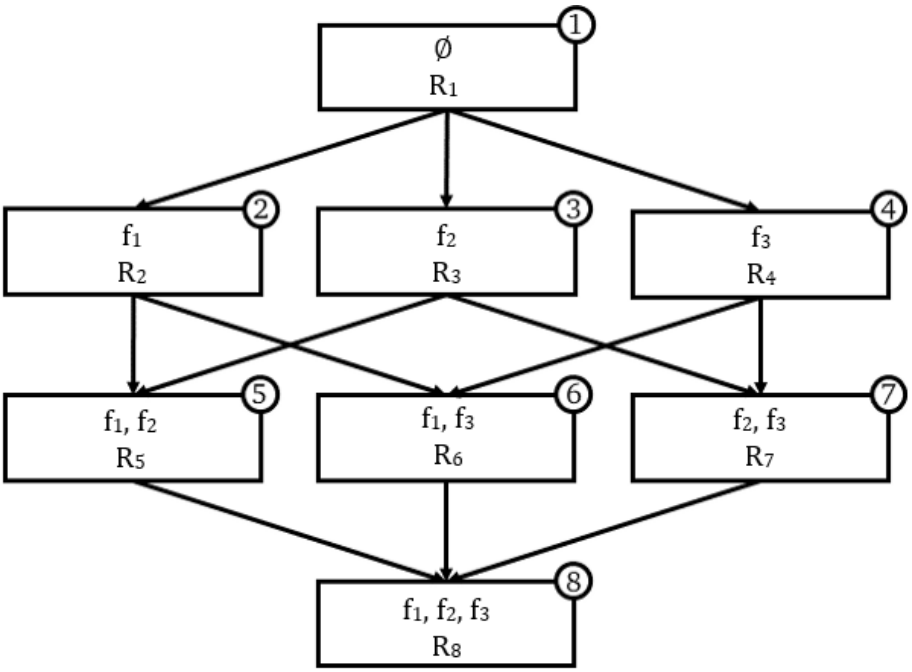

Figura 3.22: Ejecuciones del modelo para el cálculo de la contribución marginal de cada feature[11]

resultados. Cada conexión entre los nodos simboliza la contribución marginal de una feature de un nivel a otro. Por ejemplo, en el nodo 5, la contribución marginal de la feature 2 se obtendría mediante la resta $CM_{f_2,\{f_1,f_2\}} = R_5 - R_2$, es decir, la diferencia entre predicción del modelo con la feature 2 y si ella.

Para calcular el error marginal total y, por tanto, el *SHAP value* de una feature, hay que sumar, de forma ponderada, todas sus contribuciones marginales:

$$SHAPvalue_{f_i} = w_1 \cdot CM_{f_i,\{f_i\}} + ... + w_{F-1} \cdot CM_{f_i,\{f_1,...,f_F\}} \tag{3.16}$$

Donde f_i es la feature, w los pesos y F el número total de features. El valor de los pesos ha de sumar 1 y sumar lo mismo para cada nivel. Por tanto, su valor es igual es:

$$w_{nivel} = nivel \cdot \binom{F}{nivel}. \tag{3.17}$$

Finalmente, el cálculo de un valor SHAP para una feature para una entrada del modelo se resume como:

$$SHAP_{feature}(x) = \sum_{set:feature \in set} \left[|set| \cdot \binom{F}{|set|} \right]^{-1} [Prediccion_{set}(x) - Prediccion_{set-feature}(x)] \tag{3.18}$$

Siendo x la entrada, *set* el conjunto de features y F el número total de features.

La suma de los valores SHAP de todas las features resultará en la predicción calculada por el modelo sobre la muestra dada.

Por ejemplo, en la figura 3.23 podemos analizar la influencia que cada valor SHAP de cada feature tiene en la predicción final. Siendo las barras de color azul aquellos valores SHAP negativos que reducirán el valor de la esperanza de vida y los rojos los que influyen para incrementarla para sumar en el valor calculado por el modelo, en este caso 0,87 años en su valor estandarizado.

Figura 3.23: Ejemplo de la influencia de las features sobre el resultado final mediante *SHAP values*

Es necesario entender que los valores SHAP son únicos para cada predicción y su valor dependerá del resto de features. De tal forma que es posible que para dos valores iguales de entrada de una misma feature, resulten en valores SHAP completamente dispares, uno pudiendo influir positivamente en el resultado y otro negativamente.

Realizar este cálculo cuando el número de features es extremadamente alto es computacionalmente imposible, puesto que habría que ejecutar el modelo 2^F veces solo para explicar el resultado de una muestra. Para nuestro conjunto de datos tendría que hacer 2^{87} ejecuciones, lo que resulta en $1{,}5 \cdot 10^{26}$. No obstante, la librería utilizada implementa una serie de aproximaciones que permite obtener resultados. Aun más, ofrece una optimización muy potente para el caso concreto de redes neuronales.

3.7. Predictor

Para finalizar, se ha elaborado un predictor en una libreta de *Jupyter Notebook*, en el cual se puede especificar los valores correspondientes a cada indicador para obtener su esperanza de vida a través de la red de neuronas y, además, la explicación de la aportación de cada una de las features de entrada mediante SHAP.

Como segunda opción, se pueden elegir los valores de los indicadores de un país en un año concreto para un género y, tras ello, modificar los factores deseados para comparar el valor real, el original calculado por el modelo de *machine learning* y el resultado final tras los cambios.

Capítulo 4

Resultados

Los resultados obtenidos se han dividido según los distintos enfoques que abarca este trabajo, incluyendo determinar qué indicadores de desarrollo de un país son los más influyentes para el cálculo o resultado de la esperanza de vida, el error obtenido en los modelos de aprendizaje automático desarrollados, la optimización de los indicadores para maximizar la esperanza de vida y la interpretación del funcionamiento de los modelos.

4.1. Conjunto de datos preprocesado

Tras el tratamiento de los datos se han obtenido tres datasets correspondientes a tres fases del flujo de proceso.

- El primero es el conjunto de datos unificado con todos los indicadores de desarrollo organizados por país, año y género, conseguido tras el proceso de extracción, transformación y carga.
- El segundo dataset resultante ha sido el obtenido al aplicar la limpieza del conjunto anterior y la eliminación e imputación de los valores desconocidos, resultado en un conjunto de datos completo.
- El tercer y último conjunto de datos que se ha obtenido, ha sido el correspondiente tras aplicar las transformaciones de distribución, estandarización y tratamiento de datos categóricos sobre los datos completos.

4.2. Importancia según la correlación

El análisis de correlación entre las variables independientes y la variable objetivo ha mostrado una alta relación entre ellas. Este primer análisis nos ha permitido dilucidad cuales son, a priori, las variables más relevantes para determinar la esperanza de vida. Podemos observar las diez features más correladas por orden en la tabla 4.1.

Es destacable que todos esto indicadores, a excepción del cuarto y el último, son referentes a temas sanitarios e higiénicos. Por lo tanto, dedujimos que este tipo de indicadores tendrían un papel fundamental en los modelos a implementar. Por otro lado, observamos la alta correlación inversamente proporcional de la variable objetivo con la tasa de natalidad, es decir, cuanta más gente nazca, menor será la esperanza de vida. Esta será otra feature sobre la que hubo que hacer un seguimiento especial.

Posición	Feature	Coeficiente de correlación
1	Basic Sanization Services Total	0,817128
2	Basic Sanization Services Urban	0,802336
3	Basic Drinking Water Services	0,800003
4	Income per Capita	0,798284
5	Basic Sanization Services Rural	0,794671
6	Universal Heath Care Coverage	0,794614
7	Poisoning Mortality Rate	−0,790273
8	Basic Hand Washing Rural	0,786186
9	Basic Hand Washing Total	0,782979
10	Birth Rate	−0,782901

Tabla 4.1: Las 10 features más correladas con la esperanza de vida

4.3. Error obtenido

La diferencia entre el valor obtenido por un modelo de *machine learning* y el valor esperado es lo que se conoce como el error. Sin embargo el cálculo del error se puede implementar desde varios enfoques diferentes. Para una regresión, como es nuestro caso, las métricas de error más comunes y que se han aplicado para evaluar los modelos desarrollados son:

Mean Absolute Error (MAE)

El Error Absoluto Medio es la media de los errores cometidos en valor absoluto. Se calcula mediante la siguiente fórmula:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y}_i| \tag{4.1}$$

Root Mean Squared Error (RMSE)

Traducido como raíz del error cuadrático medio, el RMSE se obtiene mediante la raíz del cuadrado de la diferencia entre el valor esperado y el obtenido.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2} \tag{4.2}$$

Una particularidad del RMSE es que, al aplicar el cuadrado del error, penalizará más los errores grandes. Es interesante calcular el error mediante esta técnica cuando se busca aumentar la influencia de los errores grandes, para aquellos casos en los que estos se quieren evitar.

Coefficient of Determination (R^2)

Los errores descritos anteriormente son técnicas de evaluación del error cuyo rango variará dependiendo del problema planteado. Por este motivo es necesario tener unos valores de referencia para conocer el rendimiento de los modelos de regresión.

$$R^2 = 1 - \frac{\sigma^2_{error}}{\sigma^2_{salida}} \tag{4.3}$$

El Coeficiente de Determinación o Ratio de Varianza Explicada, conocido comúnmente como R^2, calcula el error respecto a la varianza. El valor de este error oscilará en el rango $[0, 1]$, de tal manera que, cuanto menor sea el error, más cercano a 1 será el valor de R^2.

Debido al proceso de transformación de distribuciones y estandarización, el error obtenido para RMSE y MAE no aportaba información, puesto que se carecía de un valor de referencia, necesario para la interpretación de estos errores como se ha comentado previamente. Por este motivo, para la interpretación del error, se ha aplicado una desestandarización, obteniendo así el error para RMSE y MAE en años.

Una vez entrenados los diferentes modelos implementados, se ha procedido a evaluarlos mediante la predicción del conjunto de pruebas y el cálculo del error, para conocer la capacidad de generalización y el rendimiento de los modelos con valores que el modelo no ha afrontado previamente en el entrenamiento.

El error obtenido por cada modelo, se muestra en la tabla 4.2 para el conjunto de entrenamiento y en la tabla 4.3 para el conjunto de test. Se ha añadido el valor del error en años de vida para ayudar en la interpretación de los resultados sobre el conjunto de pruebas.

Modelo	RMSE	MAE	R^2
Regresión Lineal	0,2729	0,199	0,9247
KNN	$2{,}015 \cdot 10^{-8}$	$5{,}54164 \cdot 10^{-9}$	0,9999
Random Forest	0,0281	0,0172	0,9991
Red de Neuronas	0,0276	0,0177	0,9978

Tabla 4.2: Error sobre el conjunto de entrenamiento

Modelo	RMSE	MAE	R^2	RMSE en años	MAE en años
Regresión Lineal	0,2808	0,2030	0,924	2,8179	2,0367
KNN	0,1536	0,1038	0,9772	1,5414	1,0417
Random Forest	0,0706	0,0453	0,9951	0,7084	0,4551
Red de Neuronas	0,0469	0,0294	0,9978	0,4709	0,2952

Tabla 4.3: Error sobre el conjunto de test

Podemos observar que el error de entrenamiento del modelo *KNN Regressor*

es extraordinariamente bajo, sin embargo, el error obtenido en el conjunto de pruebas, sobre datos nunca antes vistos por el modelo, es muy superior al primero. Esto significa que el modelo hace *overfitting* sobre los datos de entrenamiento, perdiendo así capacidad de generalización.

El *overfitting* o sobreajuste se define como el comportamiento de un modelo que aprende en exceso los datos de entrenamiento, obteniendo un valor muy preciso para estos pero es incapaz de dar una buena predicción para los nuevos datos puesto que no tiene capacidad de generalización al haber ajustado en exceso a los datos dados en el entrenamiento.

Este es un comportamiento relativamente común en este tipo de modelos, puesto que está basado en instancias, por lo que memoriza el conjunto de entrenamiento. No obstante, a pesar de su *overfitting*, el modelo tiene un mejor rendimiento que la regresión lineal.

Por otro lado, pasa algo similar con los resultados de la red de neuronas y el regresor de árboles aleatorios. Mientras que *Random Forest Regressor* obtiene unos resultados ligeramente mejores para el conjunto de entrenamiento, en el conjunto de pruebas la red de neuronas lo supera notablemente. Esto nos hace comprender que la capacidad de generalización de la red de neuronas es superior a la del regresor de árboles aleatorios.

Finalmente se concluye que, como se previó en un principio, el modelo más complejo, la red de neuronas, ha sido el que mejor resultados ha obtenido.

4.4. Análisis del error obtenido

Mediante las diferentes métricas de error, podemos hacernos una idea de qué error podemos asumir al utilizar el modelo. Sin embargo, es muy interesante estudiar el error para dilucidar qué valores tienden a fallar más. Así podremos predecir la probabilidad de error sobre una muestra a predecir.

Debido a que los mejores resultados obtenidos se ha conseguido con la red neuronal, solo se ha aplicado el estudio a las predicciones de este modelo.

Para este análisis, se ha calculado el error como la diferencia entre el valor esperado y el valor obtenido por la red de neuronas.

En la figura 4.1 se muestra la distribución del error. Este es divisible en dos según su tipo.

El error por **sobrestimación** o *overestimation* es aquel en el que el valor calculado es mayor que el valor obtenido, mientras que el error por **infraestimación** o *underestimation* es el opuesto, errores donde el valor obtenido es menor que el esperado.

Según la gráfica de la figura 4.1, los errores positivos se clasifican como sobrestimación y los valores negativos como infraestimación.

Para analizar los errores más significativos, se ha seleccionado el 2,5 % de los mayores errores por sobrestimación y por infraestimación aplicando cuantiles y se les a etiquetado como tal, dejando el resto como *Middle*.

- **Error medio** por sobrestimación en años es: 1,2760 años
- El **mayor error** por sobrestimación en años es: 3,7217 años
- **Cuantil 0.975** para determinar sobrestimación: 0,7938 años

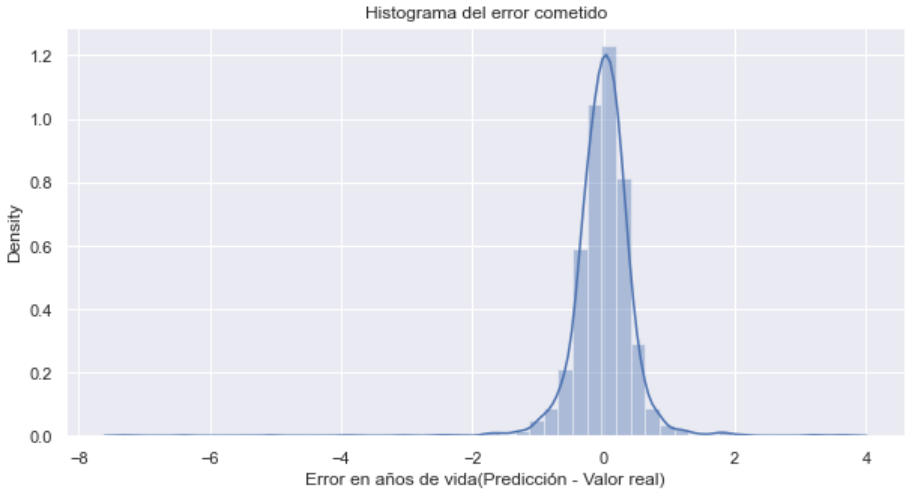

Figura 4.1: Distribución del error obtenido

- **Error medio** por infraestimación en años es: -1,4538 años
- El **mayor error** por infraestimación en años es: -7,3143 años
- **Cuantil 0.025** para determinar infraestimación: -0,8443 años

Mediante la comparación de la figura 4.2, se ha apreciado que, mientras que los mayores errores por sobrestimación se producen para esperanzas de vida entre 60 y 65 años, los errores por infraestimación más notables se presentan para esperanzas de vidas bajas.

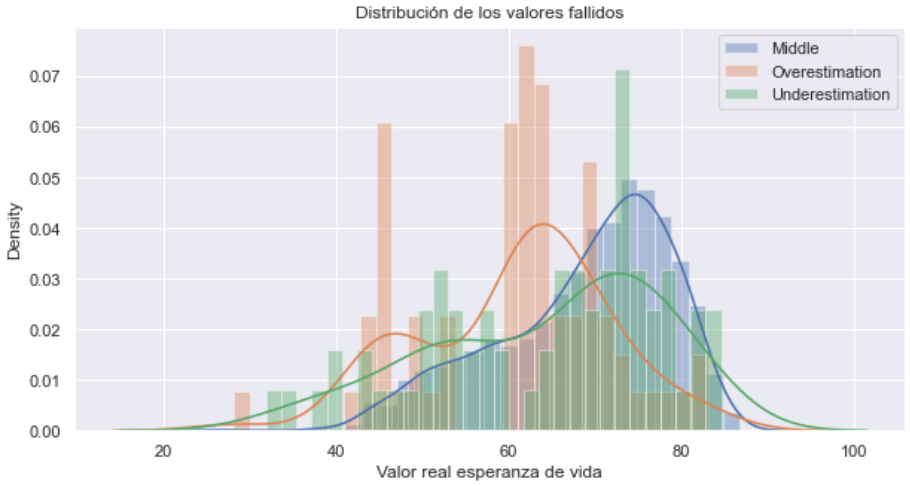

Figura 4.2: Distribución de la esperanza de vida real por tipo de error

A continuación, se ha analizado cada clase de error por separado para determinar la features que caracterizan a los registros con mayor tasa de error.

Sobrestimación

Restando la media de cada feature de todo el conjunto con el conjunto de registros que producen sobrestimaciones, se ha detectado que la feature *Diet Composition Oils And Fats* excede la media en 0,705614 y *Low CI Value % Death Cardiovascular* está 0,649977 por debajo de la media.

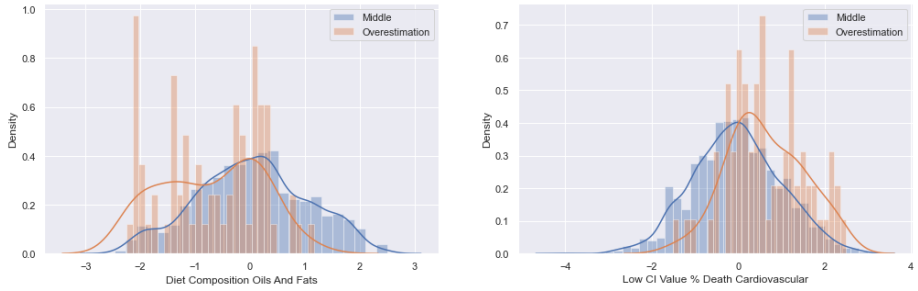

Figura 4.3: Distribuciones de *Diet Composition Oils And Fats* y *Low CI Value % Death Cardiovascular* respectivamente de valores sobrestimados y no sobrestimados

A partir de las gráficas de la figura 4.2 y 4.3 podemos concluir que el modelo tiende a sobrestimar cuando el valor a predecir es muy bajo, está en torno a 45-55 o 60-75 años de vida, el valor de *Diet Composition Oils And Fats* es más bajo de la media y/o también cuando el valor de *(Low CI Value) % Death Cardiovascular* es más alto que la media.

Infraestimación

Las features que más distan por encima de la media en la infraestimación son *Conflict and Terrorism Deaths %* y *Diet Calories Fat*.

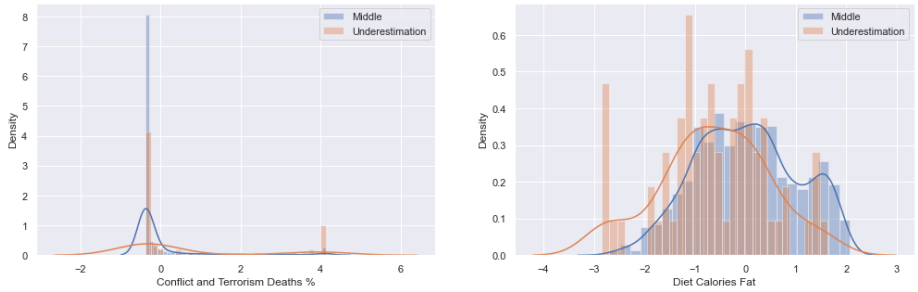

Figura 4.4: Distribuciones de *Conflict and Terrorism Deaths %* y *Diet Calories Fat* de valores infraestimados y no infraestimados

Tras el análisis de las gráficas de las figuras 4.2 y 4.4 se ha llegado a la conclusión de que el modelo tiende a infraestimar cuando el valor a predecir es muy bajo, cuando *Conflict and Terrorism Deaths %* es muy alto y/o cuando *Diet Calories Fat* es muy bajo.

Probabilidad de error

Tras el estudio de qué valores de entrada tienden provocar un mayor error en la predicción del modelo, se ha efectuado un análisis para observar la probabilidad de qué predicciones pueden estar más equivocadas.

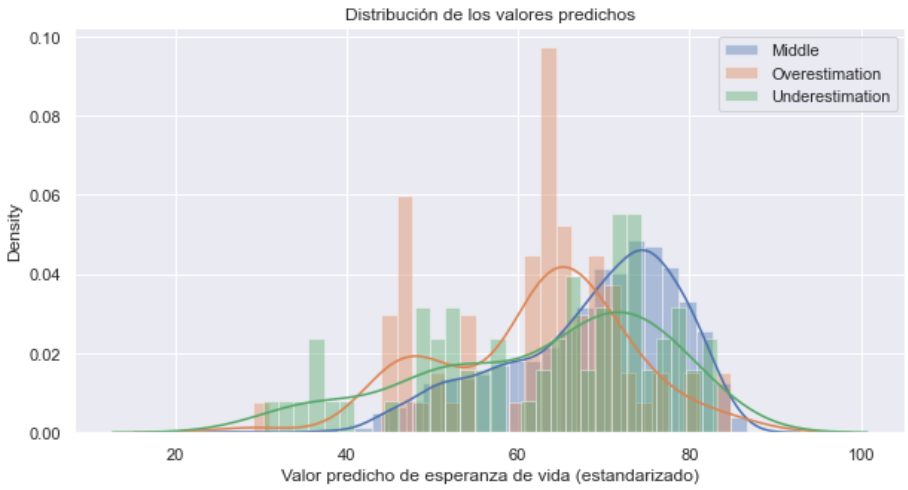

Figura 4.5: Distribución de la predicción de la esperanza de vida por tipo de error

Mediante el estudio de la gráfica 4.5 se concluyeron las siguientes afirmaciones:

- La probabilidad de que la predicción tenga un error alto por sobrestimación, será mayor si esta está entre 45-50 años, 60-65, o valores muy bajos.
- La probabilidad de que la predicción tenga un error alto por infraestimación, será mayor si esta es un valor bajo(menor de 40 años).
- Podremos asegurar con más certeza de que la predicción es más exacta cuando es mayor de 70 años.

4.5. Interpretación del comportamiento de los modelos

Una vez construidos y evaluados los modelos de *machine learning*, se ha llevado a cabo un análisis de interpretación de comportamiento de los modelos. Este se ha aplicado según diferentes enfoques dependiendo del modelo. El principal objetivo era determinar qué indicadores de desarrollo son los que más influencian en el cálculo de la esperanza de vida del país.

4.5.1. Interpretación de Random Forest Regressor

El análisis sobre *Random Forest*, se ha realizado mediante el atributo otorgado por este mismo modelo que provee la importancia de cada feature. La importancia de cada feature viene establecida por la media de la capacidad de disminución la

impureza en la división de un nodo del árbol, es decir, cuánto reduce la varianza de la solución dicha feature al ser usada como nodo en el árbol de decisión. Este coeficiente, será un valor entre 0 y 1, teniendo en cuenta que la suma de las importancias de todas las features resultarán en 1. Por tanto, es una importancia relativa.

Primera iteración

En este trabajo se ha realizado una primera iteración, obteniendo unos resultados de error aceptables. No obstante, se estudió la importancia de las features a partir de la explicación que ofrece *Random Forest*. La importancia de las features según este enfoque que explicaron la toma de decisión de selección de features(3.2) fueron las especificadas en la figura 4.4.

Posición	Feature	Importancia
1	High CI Value Under 5 Mortality Rate	0,632327
2	Death Rate	0,138211
3	Low CI Value Under 5 Mortality Rate	0,087504
6	% Population Aged 80+	0,010222
8	Infant Mortality Rate	0,006056
13	% Population Aged 65+	0,004455
14	High CI Value Infant Mortality Rate	0,003856

Tabla 4.4: Features más importantes para la primera iteración

Segunda iteración

En la segunda iteración del proyecto se volvió a estudiar la importancia de las features tras la aplicación de *Random Forest*. Los resultados obtenidos fueron los descritos en el gráfico de la figura 4.6 para las features más significativas.

Como podemos apreciar en dicho gráfico, la feature más importante para determinar la esperanza de vida, destacando ampliamente respecto al resto es *Basic Sanization Services Total*. Esto significa que este indicador de desarrollo será el que más distinga los casos, separándolos en dos grupos con una mayor reducción de la varianza de la esperanza de vida de dichas situaciones.

A continuación, destaca la importancia la probabilidad de morir por enfermedades cardiovasculares, la tasa de natalidad, el porcentaje de muertes por heridas, etc.

No obstante, este primer análisis de importancia de indicadores solo nos permite un limitado vistazo sobre el peso relativo de cada indicador, puesto que solo sabremos cual será más importante para el cálculo, pero no cómo afecta al resultado final. Es decir, mediante este análisis no podremos conocer que impacto supone el aumento o disminución de los valores de estos indicadores que destacan sobre el resto.

Además, la importancia calculada mediante esta estrategia no es perfecta, puesto que da problemas para aquellas features que estén correladas, seleccionando

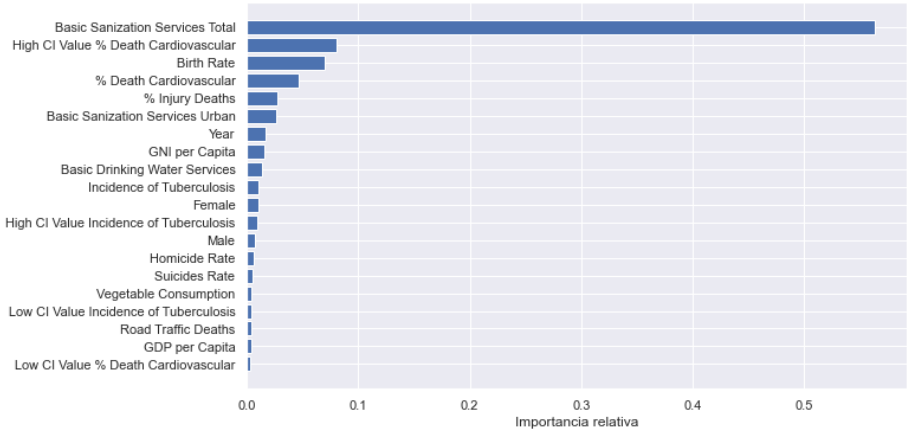

Figura 4.6: Importancia relativa de las features según *Random Forest*

una y renegando la importancia de la segunda, pudiendo inducir a conclusiones erróneas[56]. Por este motivo se ha decidido interpretar la importancia mediante otras estrategias, aunque teniendo en cuenta estos resultados para futuros análisis.

4.5.2. Interpretación de la Red de Neuronas

Las redes de neuronas son conocidas por su comportamiento de **caja negra**, es decir, son modelos que no explican su comportamiento, sino que sencillamente disponiendo de unas entradas podemos conocer su salida. La interpretación de este modelo es la más importante puesto que es el modelo que, como se expuso anteriormente, obtenía mejores resultados gracias a una mayor capacidad de generalización. Por tanto, el entender su comportamiento y la importancia relativa que da a cada indicador ha supuesto un reto, para el cual se han afrontado tres enfoques diferentes.

4.5.2.1. Wrapper Methods

Mediante los *wrapper methods* se ha analizado la evolución del error con un número incremental o decremental de indicadores siguiendo dos enfoques.

Backward Feature Selection

Se ha entrenado y validado la red de neuronas mediante la técnica de *backward feature selection* siguiendo el orden de eliminación de menos a más correlación con la esperanza de vida. Obteniendo así 87 redes de neuronas con un error incremental.

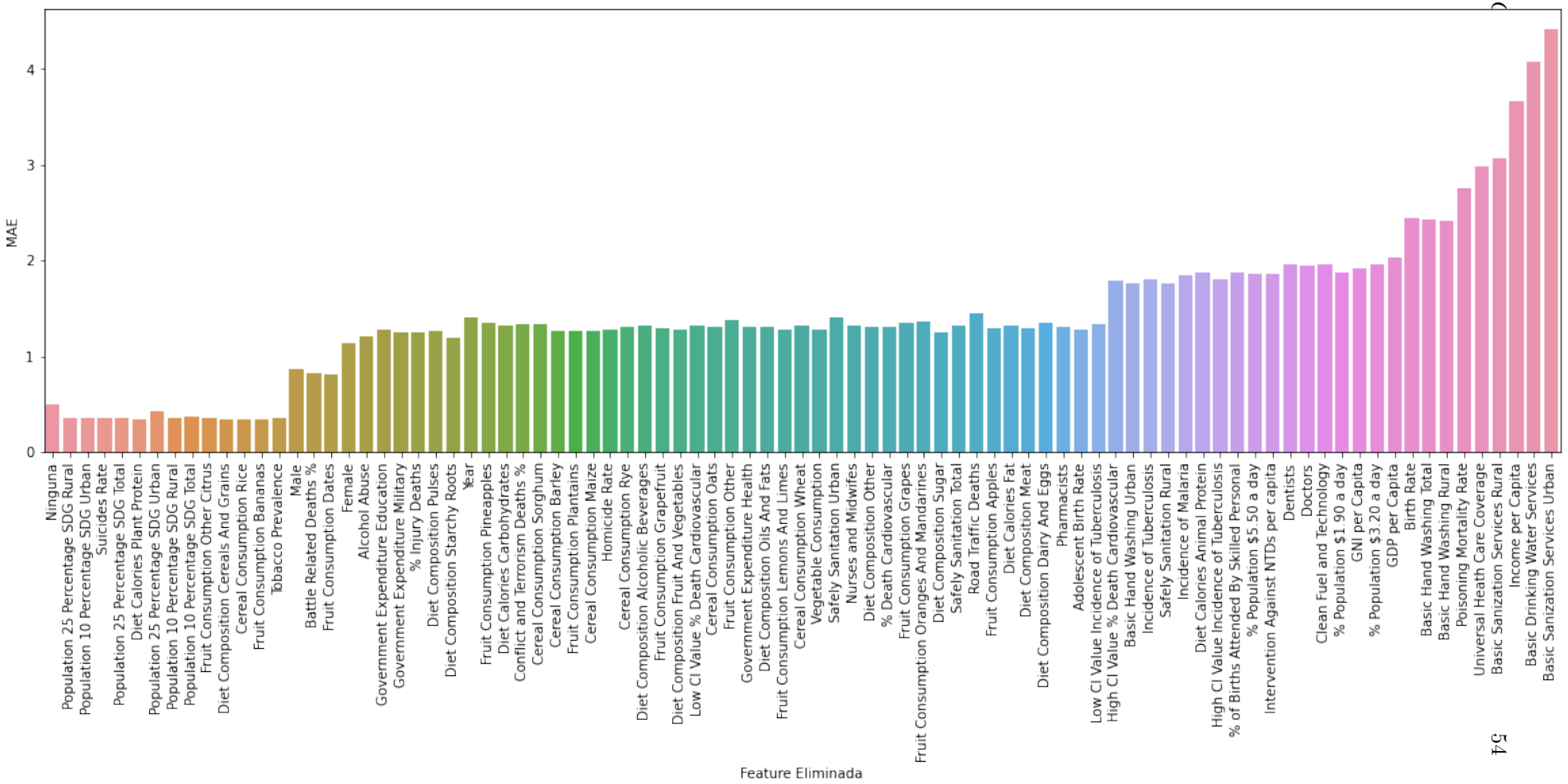

Figura 4.7: Evolución del MAE durante el *backward feature selection*

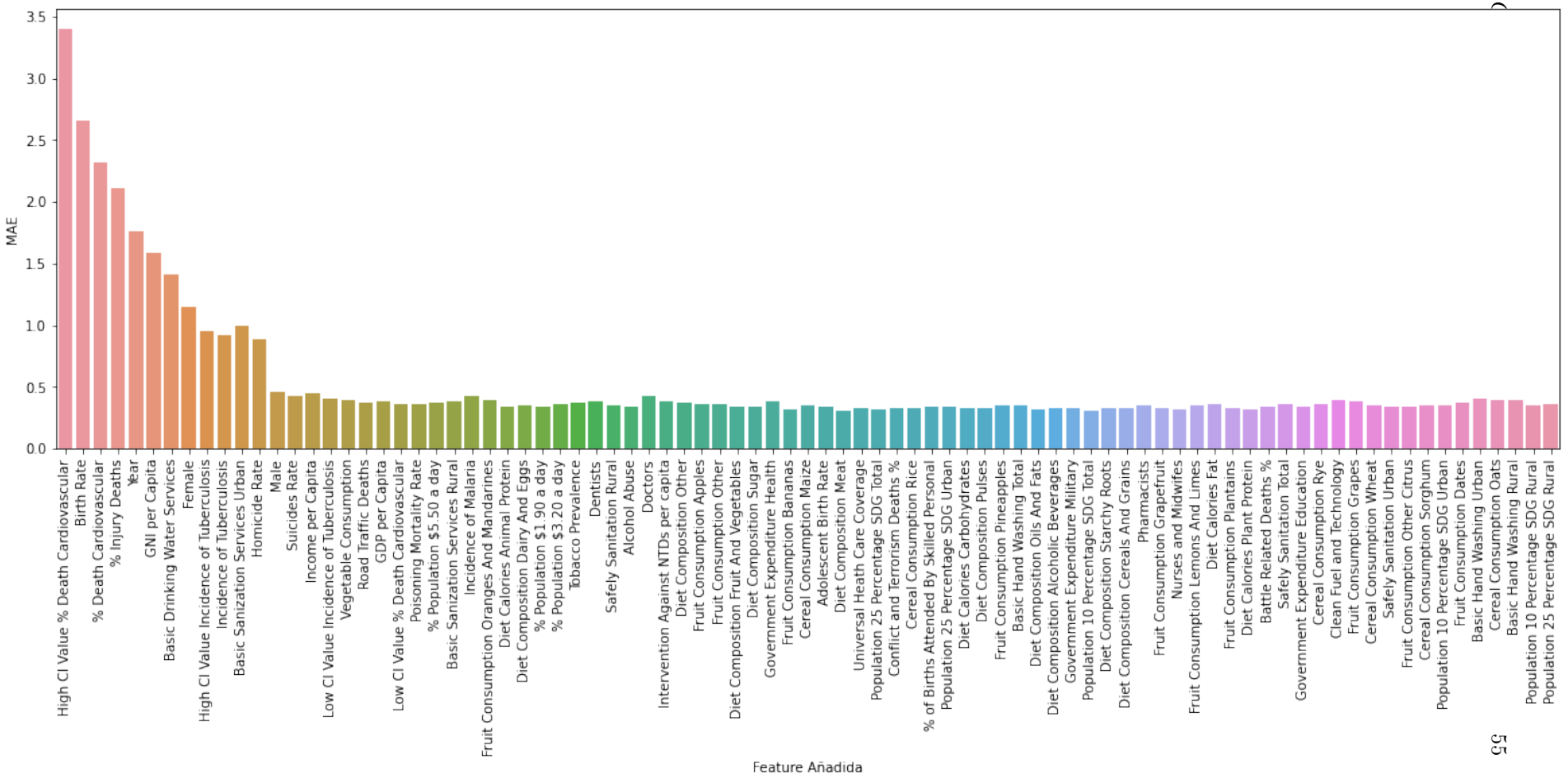

Figura 4.8: Evolución del MAE durante el *forward feature selection*

Se ha analizado en más detalle el error absoluto medio(MAE), como se muestra en la figura 4.7, donde se dibuja el error por cada feature eliminada, llegando a la conclusión de que los indicadores eliminados que más han aumentado el error y por tanto, más importantes son:

- Male
- Female
- High CI Value % Deaths Cardiovascular
- Birth Rate
- Poisoning Mortality Rate
- Universal Health Care Coverage
- Basic Sanization Services Rural
- Income per cápita
- Basic Drinking Water Services
- Basic Sanization Service Urban

Forward Feature Selection

El orden elegido para ir añadiendo cada feature ha sido la importancia según el modelo de *Random Forest Regressor*. En este caso la evolución del error para cada nueva red neuronal creada irá en decremento.

Tras el análisis de la evolución del MAE de la figura 4.8, se ha concluido que las features más relevantes según esta técnica de selección de features son:

- High CI Value % Death Cardiovascular
- Birth Rate
- % Death Cardiovascular
- % Injury Deaths
- Year
- GNI per cápita
- Basic Drinking Water Services
- Female
- Male

Como podemos observar, en ambas estrategias obtenemos resultados comunes, destacando por ejemplo la importancia de la probabilidad de morir por enfermedades cardiovasculares y el género. Sin embargo, la influencia del orden de creación de subconjuntos está muy latente en estas decisiones. Además, debido al alto tiempo de ejecución que conlleva aplicar estas técnicas, la red de neuronas implementadas se ha entrenado con 100 *epochs*, cuando quizá se podría haber

obtenido mejores resultados con un número diferente de iteraciones(ya sea mayor en caso de *underfitting* o menor en caso de *overfitting*). Por último, cabe destacar que en estos entrenamientos no se ha aplicado validación cruzada, por lo que los pequeños aumentos o decrementos del error no son interpretables. Por estos motivos se decidió estudiar la red de neuronas con otros enfoques.

Sin embargo, este enfoque nos ha ayudado a tener una visión preliminar de la importancia de las features, ya pudiendo destacar y prever el peso de algunas de ellas sobre la red de neuronas implementada.

4.5.2.2. Algoritmo genético

Debido a la lentitud de ejecución de la optimización, se ha aplicado esta estrategia sobre tan solo tres casos, donde la esperanza de vida es baja, media y alta. Para todos los casos se ha escogido un margen de maniobra(*margin_input*) igual a 5. Dicho valor permitirá diferenciar la solución del estado original lo suficiente como para que haya un aumento significativo en la esperanza de vida.

Caso sobre baja esperanza de vida

El primer caso escogido, correspondiente a una esperanza de vida baja, ha sido Afganistán en 1990 para ambos géneros, a partir del cual, tras más de 12.500 generaciones, el mejor resultado obtenido ha sido un incremento de la esperanza de vida de 4,09 años, pasando de los 50,3 originales a 54.4 según la red de neuronas.

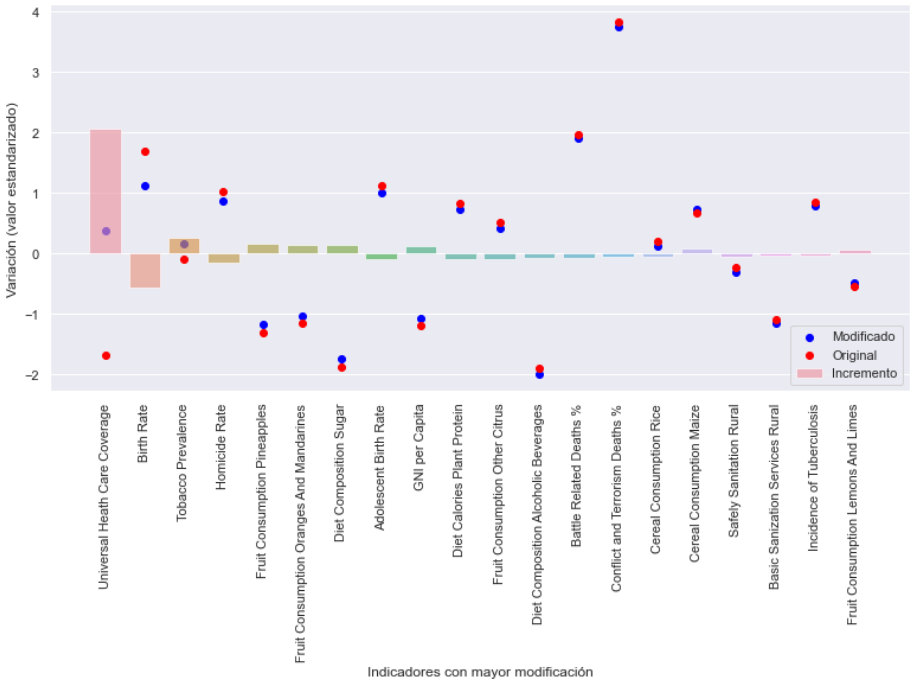

Figura 4.9: Modificaciones sobre los indicadores de Afganistán en 1990 para ambos géneros para maximizar la esperanza de vida

La modificación realizada sobre los indicadores, con valores estandarizados, se representan en la figura 4.9. Las barras representan cuánto se ha modificado este valor, los puntos rojos marcan el valor original de la feature y los puntos azules el valor final.

Para este caso en concreto, el indicador cuya modificación maximiza la esperanza de vida, con un peso altamente diferenciativo, es *Universal Health Care Coverage*, es decir, la accesibilidad a servicios sanitarios básicos. Este valor originalmente estaba muy por debajo de la media(cero), para colocarse por encima.

Los siguiente dos indicadores que se han visto más modificados han sido, la tasa de natalidad y la permanencia del tabaco. De de forma contraria a lo que se puede intuir, esta solución propone un descenso de la natalidad y un aumento del consumo de trabajo para aumentar la esperanza de vida.

Esta solución nos indica, por tanto, que en los países donde hay una menor natalidad y mayor consumo de tabaco existe una mayor esperanza de vida, por tanto nuestra red de neuronas considera estos dos factores para aumentarla. Lo que significa que nuestro conjunto de datos está correlado con la variable objetivo, sin embargo, descubrimos que correlación no implica causalidad, es decir, que dos cosas estén correladas como en este caso, no quiere decir que una provoque la otra.

Caso sobre esperanza de vida media

El segundo caso elegido para aplicar el algoritmo genético para maximizar la esperanza de vida ha sido Lituania en 2005 para género masculino.

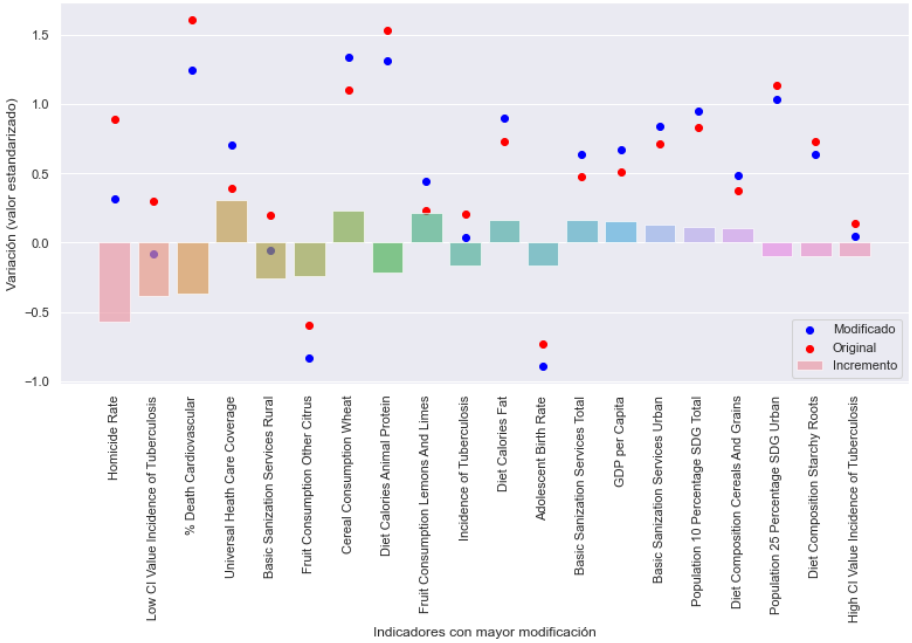

Figura 4.10: Modificaciones sobre los indicadores de Lituania en 2005 y género masculino para maximizar la esperanza de vida

En esta situación, se ha conseguido un aumento de la esperanza de vida de 3,4 años, pasando de 65,6 a 69.

En este caso presente en la figura 4.10, el indicador a modificar más destacable ha sido la tasa de homicidios, que desciende desde un punto inicialmente muy por encima de la media. Ocurre una situación similar para los siguientes indicadores sobre la incidencia de tuberculosis y la probabilidad de morir por enfermedades cardiovasculares.

Caso sobre alta esperanza de vida

Como tercer y último caso a aplicar, se ha escogido el más cercano posible a nuestra situación, es decir, España en 2019 para ambos géneros. Partiendo de una esperanza de vida de 83,4 años, se ha llegado mediante los cambios a 87,1, consiguiendo aumentarla 3,63 años.

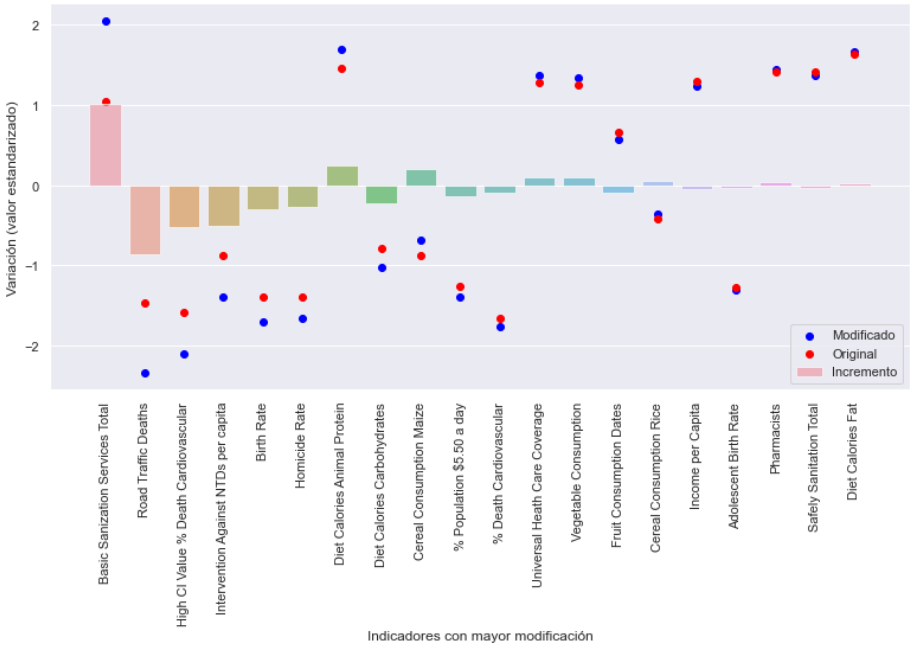

Figura 4.11: Modificaciones sobre los indicadores de España en 2019 y ambos géneros para maximizar la esperanza de vida

La figura 4.11 nos muestra que para países con una esperanza de vida alta, la feature que requiere una mayor modificación es, curiosamente, *Basic Sanization Services*, que partía de un valor inicial notablemente por encima de la media, lo que nos indica que es un valor que será directamente proporcional a la esperanza de vida, es decir, cuanto más aumente este indicador más aumentará la esperanza de vida. Ocurre lo inverso con las features que van a continuación, el número de muertes por accidentes de tráfico, la probabilidad de morir por enfermedades cardiovasculares, el número de intervenciones de enfermedades tropicales desatendidas, la tasa de natalidad y el ratio de homicidios. Todas ellas parten de un valor inicial muy por debajo de la media, a lo que cuanto menor sean, más maximizará la esperanza de vida.

A pesar de que el análisis individual se ha basado en las features más importantes, es necesario destacar la importancia de los indicadores relativos a la alimentación que, aunque no se colocan en primer lugar están muy presentes en los cambios a realizar.

A partir de los resultados obtenidos en los diferentes casos, interpretamos que el funcionamiento y la toma de decisiones del modelo se apoya notablemente en los casos con una alta esperanza de vida, buscando maximizar todos los valores de sus features(observamos que los puntos iniciales siempre están más alejados de la media), mientras que para aquellos casos con una baja esperanza de vida, se tiende a acercar más los valores hacia la media.

Una de las ventajas de este enfoque para la interpretación del modelo es que, no solo nos muestra las features que más influyen en el caso estudiado, sino que además nos permite conocer qué cambios han de promoverse para aumentar la esperanza de vida. Sin embargo, tiene dos grandes desventajas, una sería el límite de algunas de las features, es decir, aquellas referidas a porcentajes tendrán un tope máximo y mínimo, por lo que habría que modificar el algoritmo genético para penalizar que se sobrepasen esos límites. La segunda debilidad es el modelo en sí, puesto que como se ha mencionado previamente, correlación no implica causalidad y características de los países como una baja tasa de natalidad no implicará el aumento de la esperanza de vida.

4.5.2.3. Shapley Additive Explanations (SHAP)

Mediante el uso de esta técnica, se han obtenido diversas gráficas para entender la influencia de las features sobre los resultados otorgados por la red de neuronas tanto a nivel general como en casos concretos.

Análisis general

En la figura 4.12 se muestra en orden las 15 features más influyentes sobre la predicción del modelo ordenadas de mayor a menor. La influencia de las features se calculan mediante la media del valor absoluto de sus *SHAP values*.

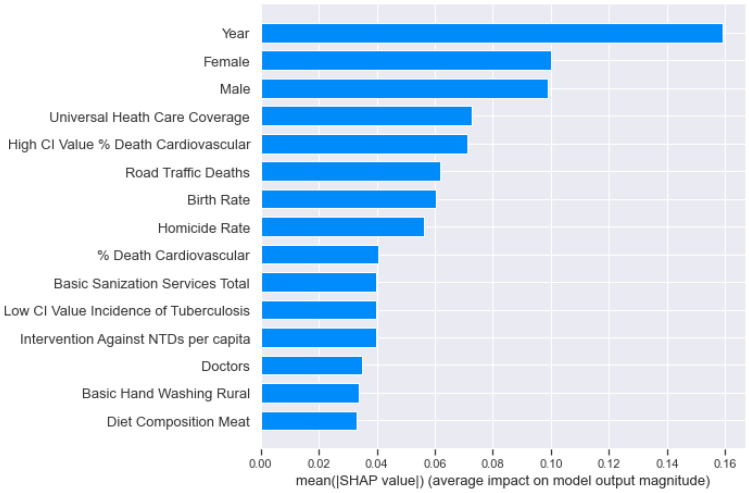

Figura 4.12: Media de aportación de cada features sobre el resultado final

Según esta estrategia de interpretación de modelos de caja negra, las features cuyo valor produce más impacto de media son el año y el género, con una gran diferencia sobre el resto. Estas son dos features que podemos calificar como no mutables, es decir, no pueden ser modificadas con el objetivo de un aumento de la esperanza de vida. Sin embargo, son altamente necesarias para el cálculo de la esperanza de vida puesto que son determinantes como bien indican los valores SHAP.

El resto de los indicadores más determinantes son relativos a diversos campos como sanidad, natalidad, accidentes, seguridad y alimentación. Analizaremos su comportamiento más adelante.

Para poder interpretar de forma más clara el peso de los valores SHAP, hay que tener en cuenta que estos suman y restan sobre el valor final. Sin embargo, estas cifras no son interpretables debido a que el resultado de la red de neuronas está estandarizado. Para poder interpretar tanto la gráfica 4.12 como el resto, podemos hacer la conversión mediante la figura 4.13, donde se comparan las dos escalas de la esperanza de vida estandarizada y sin estandarizar.

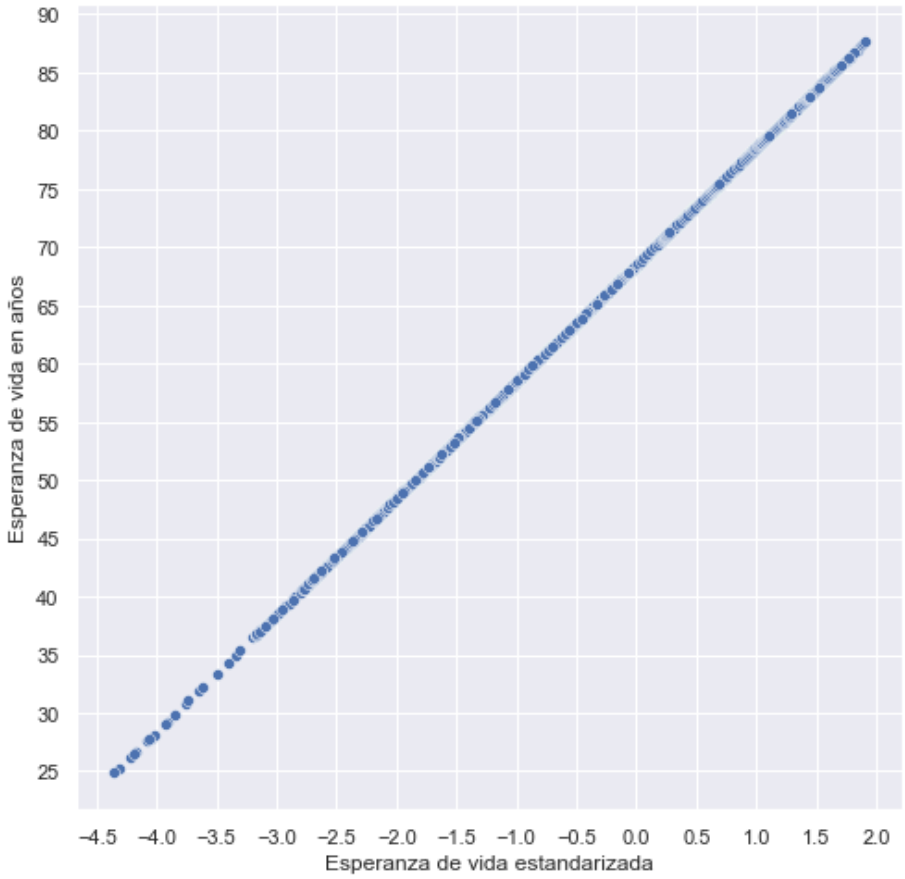

Figura 4.13: Comparación de valores de esperanza de vida estandarizados y sin estandarizar

Una vez presente esta conversión, cabe destacar que la feature que más

influye en los datos lo hace con una media de 0.16 años estandarizados, lo que equivale en torno a 1.6 años. Esto significa que la influencia que puede tener una única feature sobre la predicción final no será excesivamente relevante, sino que será la suma de todos los indicadores los que poco a poco determinarán el resultado.

La gráfica que muestra la mejor visión para la interpretación del comportamiento e influencia de cada feature sobre las predicciones de la red de neuronas es el correspondiente a la figura 4.14 y 4.15. Se trata de un conjunto de diagramas de violín, habiendo uno para cada feature, que muestran los *SHAP values* de diferentes muestras para una feature. El grosor vertical de la figura indica el número de casos con ese valor SHAP para esa feature. Por último, el color indica el valor original de esa feature, siendo más azul para un valor más bajo y rojo para un valor más alto.

Por ejemplo, la gráfica nos indica que para *Year*, un valor alto de la feature significará un valor SHAP alto, incrementado el resultado de la esperanza de vida. Sucede lo contrario para un valor bajo de esta misma feature.

Mediante este gráfico podemos entender la influencia positiva o negativa del aumento o disminución de un cierto indicador. En su mayoría el comportamiento es directa o inversamente proporcional, es decir, un mayor valor de la feature corresponderá a un mayor(o menor en caso de inversamente proporcional) valor de SHAP. No obstante, el valor SHAP de una feature está condicionado por el valor del resto de features. De tal manera, hay ciertas features cuya influencia variará positiva o negativamente según el resto de valores. Podemos observar este comportamiento, por ejemplo, en *Vegetable Consumption*.

Por último, para muchas features, observamos que el valor correspondiente a la media(donde se unen el azul y el rojo en color morado) tiende a reducir el valor de la predicción.

Análisis individual de features

Para analizar la progresión del valor de SHAP de una determinada feature conforme varía su valor de entrada, además de la correlación e influencia de unas features sobre otras para el cálculo de este valor, se han empleado lo que se conocen como gráficas de dependencia.

En las gráficas de dependencia podemos observar en los ejes el valor original de entrada de la feature y su valor SHAP correspondiente. Además podremos ver cómo varían dependiendo del valor de otra feature mediante los colores de los puntos. Esta feature la elige automáticamente el algoritmo, seleccionando aquella que más influye(o de la que más depende) sobre la feature analizada.

Analizamos el comportamiento de la feature más relevante según esta estrategia, el año, en la gráfica de dependencia de la figura 4.16. Esta imagen corrobora lo mencionado anteriormente, a mayor año, mayor influencia positiva sobre la esperanza de vida y viceversa. No obstante, los valores SHAP varían aun teniendo la misma entrada, debido a la influencia del resto de entradas. La feature de la que más depende *Year* al establecer sus valores SHAP es *% of Births Attended By Skilled Personal*. El color rojo sobre los puntos representa un mayor valor de esta feature y, gradualmente, el color azul lo contrario.

Se concluye, por tanto que, el año será un factor mucho más determinante(puesto que tendrá un *SHAP value* mayor en valor absoluto) para aquellos casos en los que el porcentaje de partos atendidos por personal sanitario sea menor. Lo que

Figura 4.14: Diagramas de violín sobre los *SHAP values* de cada feature(Parte 1)

Figura 4.15: Diagramas de violín sobre los *SHAP values* de cada feature(Parte 2)

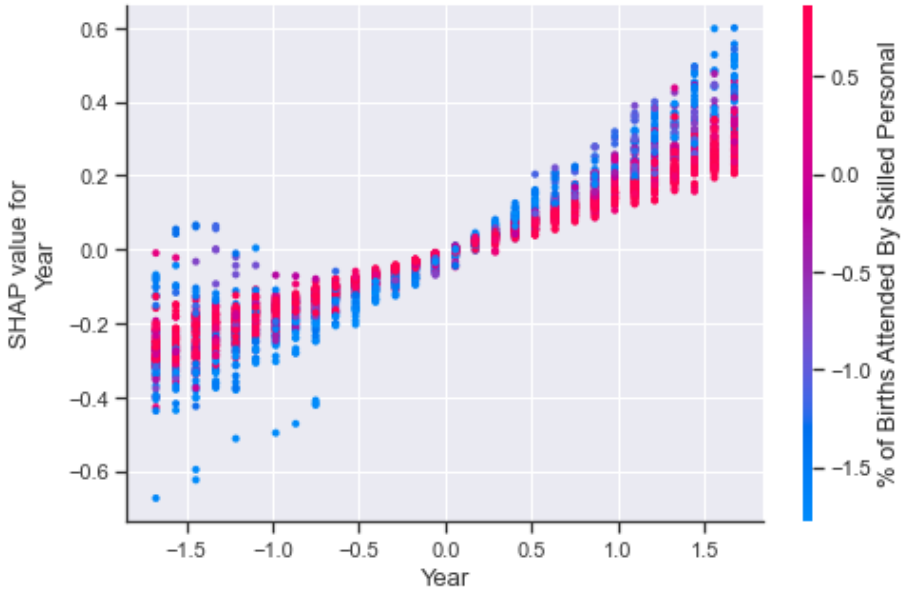

Figura 4.16: Gráfica de dependencia sobre la feature *Year*

quiere decir, que el año influirá mucho más en el resultado final en países con una sanidad más precaria.

Para otras features en cambio, la influencia positiva o negativa en el resultado final no sigue una lógica directa sobre el valor de entrada como la anterior, sino que dependen en mucha mayor medida del resto de entradas. Este es el caso de la feature *Population 10 Percentage SDG Total*, es decir, el porcentaje de la población cuyos los gastos en sanidad exceden el 10 % de sus ingresos. En su gráfico de dependencia de la figura 4.17, se puede vislumbrar que un mismo valor de entrada de esta feature, puede influir tanto positiva como negativamente en la esperanza de vida calculada. Esa diferencia está directamente relacionada con el valor de las intervenciones contra enfermedades tropicales desatendidas per cápita.

Los valores altos de *Population 10 Percentage SDG Total* influirán positivamente en el resultado si *Interventions Against NTDs per Capita* es baja y lo hará negativamente cuando el valor sea alto. Contrariamente, los valores bajos influirán positivamente si las intervenciones per cápita son altas y viceversa.

Gracias a este tipo de gráficas, se han podido detectar comportamientos extraños o no esperados, como sucede con la feature *Income per Capita*, es decir, los ingresos per cápita. Se puede intuir, que a mayor ingresos, más aportará positivamente a la esperanza de vida. Sin embargo, se puede observar en su diagrama de dependencia, presente en la figura 4.18, que no se cumplen estas expectativas.

En cambio, valores bajos de *Income per Capita* influyen positivamente en el modelo, mientras que valores altos pueden llevar a influir ligeramente disminuyendo el valor calculado de la esperanza de vida. Además observamos que esta influencia contraria a la esperada en el resultado final será más extrema según el valor del consumo de productos lácteos y huevos.

Figura 4.17: Gráfica de dependencia sobre la feature *Population 10 Percentage SDG Total*

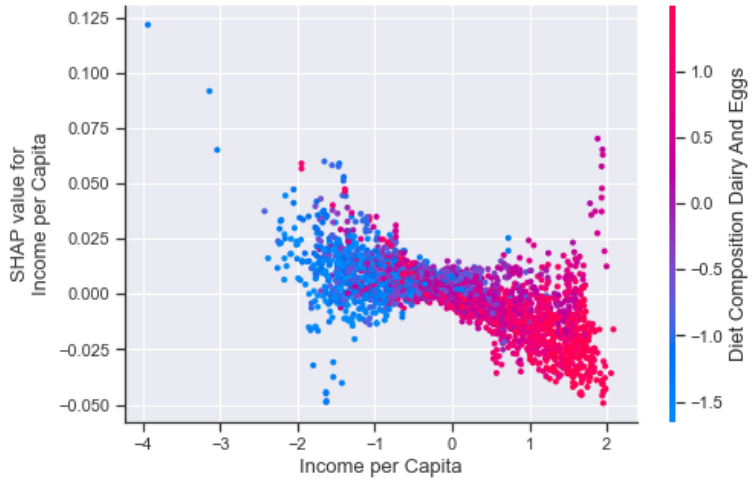

Figura 4.18: Gráfica de dependencia sobre la feature *Income per Capita*

Análisis de casos individuales

La mejor funcionalidad que nos otorga esta estrategia de análisis e interpretación de modelos, es la explicación de casos individuales. Para un caso concreto, se puede ver cuánto y cómo aporta cada una de las features hasta llegar al resultado final.

Por ejemplo, analizamos el caso de España en 2005 para ambos géneros. La esperanza de vida en este año es 80,553, la red de neuronas da un resultado muy similar, 80,603 años, lo que estandarizado queda como 1,21. En la gráfica dispuesta como figura 4.19 se puede apreciar la influencia de las features más relevantes en el cálculo. Podemos destacar la alta influencia positiva sobre el resultado de *Road Traffic Deaths* y de la probabilidad de morir por enfermedades

cardiovasculares.

Figura 4.19: Explicación de la influencia de cada feature sobre la predicción final de la red de neuronas sobre el caso de España en 2005 para ambos géneros

Para poder visualizar la aportación de cada feature sobre la predicción, usamos las gráficas en cascada, como la de la figura 4.20, que muestra individualmente cada feature con su valor SHAP. En el gráfico de cascada, partiendo del valor medio del resultado(en nuestro dataset 0, debido a la estandarización), cada feature irá aportando su *SHAP value* hasta llegar al valor obtenido por la red de neuronas. En este caso, se estudia la predicción sobre Afganistán en el año 1990 para el género femenino, donde la esperanza de vida real es de 51,442 años y la calculada por el modelo es 51,573 años.

Para este caso, podemos ver la participación de la feature *Conflict and Terrorism Deaths %*, el cual juega un papel negativo en el resultado final. Es destacable el hecho de que, a pesar del alto número de features que componen la entrada, casi todas ellas aportan a la salida, de forma más influyente o menos.

Mediante estos gráficos, se puede explicar de forma exacta la toma de decisiones que lleva a cabo nuestro modelo para determinar una salida.

Comparaciones entre casos

Para comparar varios casos, es muy interesante el gráfico de decisión. Este gráfico nos muestra en forma de línea vertical el progreso del valor calculado tras aplicar el valor de SHAP feature a feature, partiendo desde cero hasta llegar al resultado obtenido.

En el diagrama de decisión de la figura 4.21, podemos observar la aportación de algunas de las features de entrada del modelo sobre las predicciones de la esperanza de vida en Argelia con la **evolución de los años**. Las features están ordenadas de menor a mayor diferencia de desviación típica de sus valores SHAP, para poder observar qué features tienen mayor diferencia y cuanto suponen.

Es percatable que, para este caso, lo que más marca la diferencia en la predicción de un año a otro, a parte de las features más comúnmente influyentes, son *Population 25 Percentage SDG Rural*, *Diet Calories Plant Protein* y *Diet Composition Alcoholic Beverages*, entre otras observables en la gráfica de decisión.

A continuación, se ha estudiado el comportamiento del modelo sobre **diferentes grupos divididos según la esperanza de vid**a. Se han formado tres grupos, uno con esperanza de vida baja, otro media y otro alta, con el objetivo de ver qué features marcan la diferencia entre estos grupos.

Los grupos elegidos se han seleccionado sobre unos rangos definidos de esperanza de vida en un año único para ambos géneros, para que estas últimas features no mutables no lideren los gráficos de estudio. Los valores de los rangos afectan sobre el valor real, no del valor predicho por la red de neuronas.

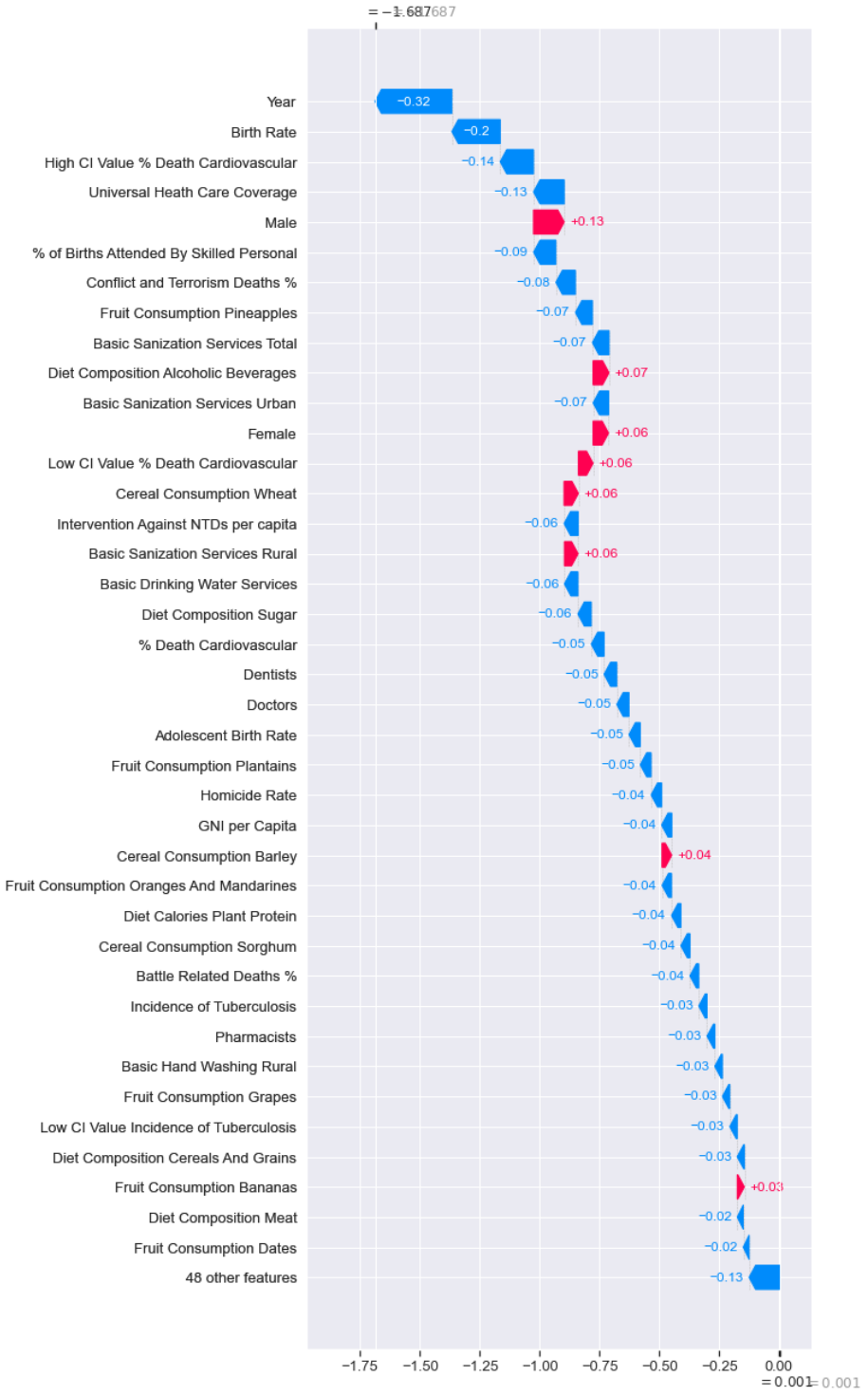

Figura 4.20: Explicación de la influencia de cada feature sobre la predicción final de la red de neuronas en gráfico de cascada sobre el caso de Afganistán en 1990 para género femenino

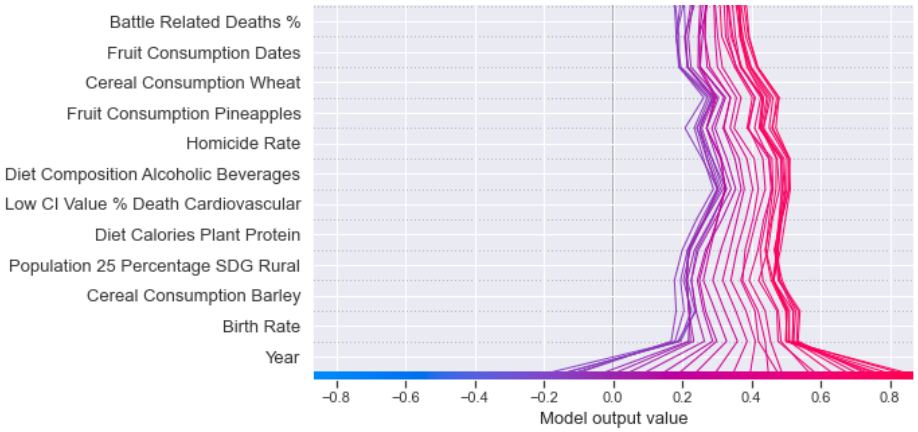

Figura 4.21: Gráfico de decisión que compara las predicciones sobre Argelia para ambos géneros año a año, desde 1990 hasta 2019

Para **países con una alta esperanza de vida** se ha seleccionado el rango de mayor de 82 años para el año 2015. Se muestran los resultados de las features con *SHAP values* más dispares en la gráfica de decisión de la figura 4.22.

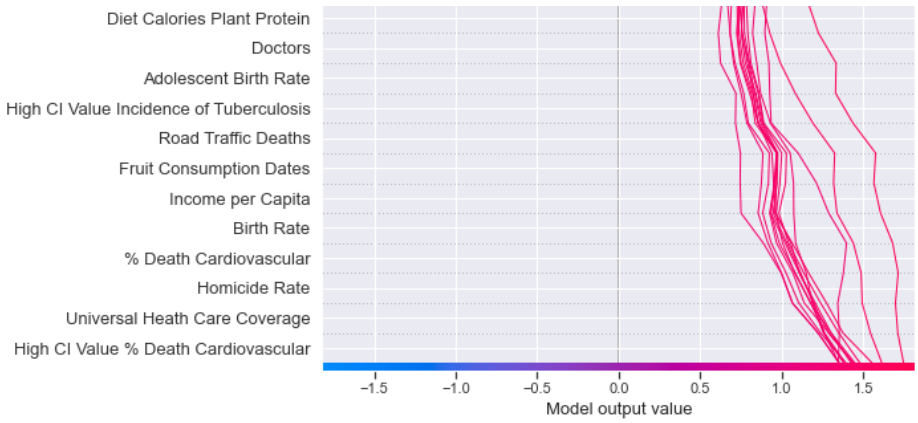

Figura 4.22: Gráfica de decisión sobre los países con esperanza de vida mayor que 82 años en 2015

En dicha gráfica podemos observar la alta influencia de features en un principio menos relevantes como el consumo de dátiles, la incidencia de tuberculosis o las calorías consumidas por proteína animal.

En el caso con **casos con una esperanza de vida media**, se ha escogido el rango de 66 a 68 años en el año 2000. En estas situaciones hay una disparidad de origen mucho mayor como se aprecia en la figura 4.23. La aportación de las feature sobre el resultado final son mucho mayores y relevantes. Estas features son *Basic Sanization Services, Doctors, % Deaths Cardiovascular, Universal Health Care Coverage, Interventions Against NTDs, Homicide Rate* y *Birth Rate*.

Por último, analizamos el grupo de **casos con una esperanza de vida**

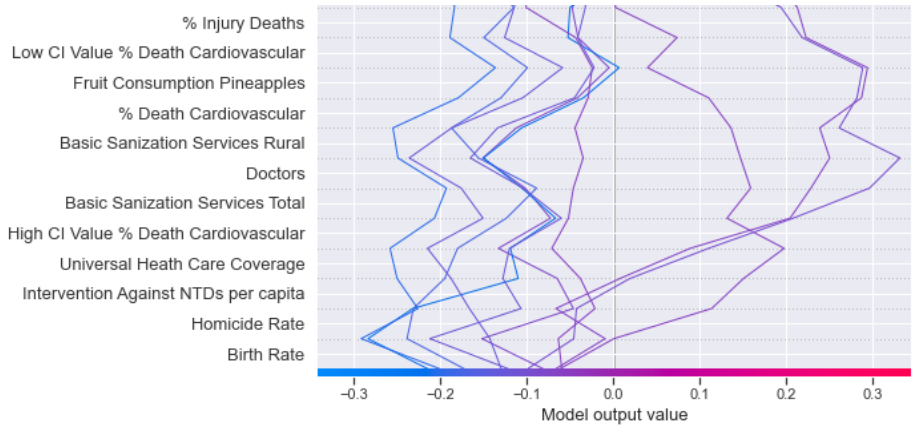

Figura 4.23: Gráfica de decisión sobre los países con esperanza de vida entre 66 y 68 años en el año 2000

baja, situando el rango como menor de 50 años para el año 1990. En este caso también podremos observar los valores más atípicos, pudiendo determinar el origen de este comportamiento. Mediante la figura 4.24 se ha concluido que los indicadores que más afectan para originar estos valores atípicos es principalmente el porcentaje de muertes por conflictos y terrorismo, seguido de la alimentación.

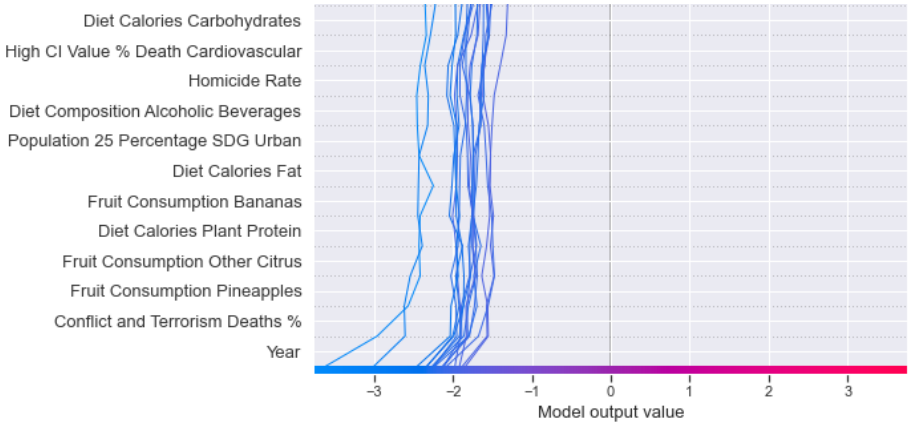

Figura 4.24: Gráfica de decisión sobre los países con esperanza de vida menor de 50 años en 1990

Finalmente, mediante el análisis con SHAP podemos concluir que, a pesar de que las features que más afectan al resultado son las presentes en la figura 4.12 y su efecto sobre el resultado final es observable en la figuras 4.14 y 4.15, al final, en cada caso en particular influirán más unas features u otras, especialmente dependiendo de la esperanza de vida que predice nuestra red de neuronas.

4.6. Problemas encontrados

A lo largo de la elaboración de este trabajo, se han encontrado diversos obstáculos que ha habido que sobrepasar, para ello, se han tomado diversas decisiones y enfoques para abarcar la solución.

Conjunto de datos unificado

Al haber extraído datos de diversas fuentes, no disponíamos de un único dataset, sino de múltiples archivos con la información que se deseaba tratar. Por ello se tuvo que llevar a cabo un proceso ETL para la unificación de todos los datos a un dataset único que contuviera los indicadores de desarrollo organizados por país y año. Además en este proceso, debido a que ciertos indicadores refieren al género, a su vez, ha habido que organizarlo por esta categoría, resultado un dataset que contuviera una fila por país, año y género.

Calidad del dato

A parte de la necesaria análisis y limpieza de datos erróneos requeridos para el dataset, el principal problema afrontado con el conjunto de datos han sido los valores desconocidos. La cantidad de datos vacíos era sobrecogedoramente alta. Por este motivo se tuvo que llevar a cabo un plan para solucionarlo. El plan elaborado se ha apoyado en técnicas de imputación de valores desconocidos como son la interpolación y KNN y en la eliminación tanto de filas como de indicadores.

Selección de features

Tras aplicar una primera iteración con modelos de *machine learning* sobre el conjunto de datos inicial, destacó la importancia de ciertos indicadores que, tras un análisis de los mismo, se concluyó que su cálculo estaba demasiado relacionado con la esperanza de vida, no siendo factores causantes sino causados por la esperanza de vida. Por tanto, se tomó la decisión de eliminar un conjunto de indicadores del dataset, a la vez que se extrajo y añadió otro grupo más acorde a los intereses del proyecto.

Análisis de los modelos

El modelo de *machine learning* con el que se obtuvo mejor resultado fue la red de neuronas, no obstante, este modelo funciona como una caja negra, por tanto no es posible entender mediante sus operaciones la toma de decisiones realizada para obtener el resultado calculado. Para este problema se propusieron tres soluciones, tres estrategias para la interpretación de modelos de caja negra: *wrapper methods*, algoritmo genético y SHAP.

Limitación computacional

La limitación computacional ha sido un problema recurrente en este proyecto para el análisis de la red de neuronas. El enfoque de *wrapper methods* se ha visto altamente limitado al orden de las features para su ejecución, cuando la solución óptima hubiese sido todas las posibles combinaciones de features o, al menos,

todas las posibles combinaciones de 10 features, ambos casos completamente inviables por cuestiones de tiempo de ejecución. Por este motivo se decidió elegir dos enfoques para el orden de features en la aplicación de esta estrategia: por correlación con la variable objetivo y según los resultados de *Random Forest*.

Nos hemos vuelto a enfrentar a este problema en la aplicación del algoritmo genético. Debido a su funcionamiento, durante la ejecución de esta técnica de aprendizaje automático, se ha de ejecutar la red de neuronas una vez por individuo(en nuestro caso 100) por generación, para así calcular su calidad. La ejecución de la red de neuronas es computacional costoso, lo que relentecía notablemente el aprendizaje del modelo. Para acelerar la ejecución y no tener que ejecutar la red para cada individuo, se modificó la lógica de la calidad del individuo para que, en caso de no parecerse demasiado al caso de entrada, no se calculara la esperanza de vida sino que la calidad referida a esa parte fuera mínima. A pesar de estos cambios, el tiempo de ejecución para cada caso continuó siendo muy elevado, privándonos de realizar un análisis más completo y exhaustivo de la red de neuronas aplicando esta estrategia.

Correlación y causalidad

Los resultados obtenidos en la interpretación del funcionamiento de la red de neuronas nos muestra que para ciertos indicadores de desarrollo el algoritmo toma decisiones que no corresponden a la realidad. Este es el caso, por ejemplo, de la feature referida a la tasa de natalidad. El valor de este indicador puede parecer que no tendrá influencia en la esperanza de vida, sin embargo, tras el análisis realizado, se ha observado que para casos con tasa de natalidad alta da una esperanza de vida menor a si la tasa es baja. Esto es debido a que los países con una natalidad alta son, generalmente, países desarrollados con una alta esperanza de vida, mientras que aquellos menos desarrollados tienen una tasa de natalidad más alta. Es decir, la tasa de natalidad y la esperanza de vida están correladas, por tanto el modelo usará la primera para calcular la segunda. No obstante, esta correlación no implica causalidad.

Al igual que sucede en el caso de la natalidad, puede ocurrir con el resto de indicadores, por tanto, no podemos concluir que la modificación de los factores estudiados impliquen una modificación en la esperanza de vida. Sin embargo, si podemos plantear que aquel país con unas determinadas características dadas por los indicadores empleados, tendrá una cierta esperanza de vida asociada.

Capítulo 5

Conclusiones y trabajos futuros

5.1. Conclusiones

Tras la realización de todo el flujo de trabajo y la obtención de resultados, se han llegado a unas conclusiones respecto a los objetivos iniciales planteados en este trabajo.

Como primer punto cabe destacar que se ha conseguido el conjunto de datos unificado que se pretendía con indicadores de desarrollo, alimenticios, económicos, políticos y sanitarios de un amplio número de países organizados por año y género. No obstante, dicho conjunto de datos contenía un alto número de valores vacíos que ha habido que imputar, lo cual es un factor a tener en cuenta.

Se ha logrado obtener un modelo de aprendizaje automático basado en un perceptrón multicapa capaz de predecir la esperanza de vida a partir de los factores suministrados. Este tipo de modelo de regresión ha sido de forma clara el mejor de todos los planteados, obteniendo un error absoluto medio (MAE) de menos de un tercio de año.

El análisis de una red de neuronas es un proceso complicado, en el cual se han planteado diferentes enfoques que nos han permitido obtener un rango más amplio de conclusiones.

No obstante, la primera conclusión referente al análisis de cuales son los factores que influyen más en la esperanza de vida de un país mediante un modelo de *machine learning* es, sin duda, la relación entre **correlación y causalidad**. A partir de los resultados obtenidos en las diferentes interpretaciones del modelo de caja negra planteado, se ha hecho evidente la importante influencia de ciertos factores como la tasa de natalidad. Estos factores que, en un principio puede parecer que no deberían influir en el cálculo del resultado, tienen un gran peso en el modelo. Esto es debido a que guardan una alta correlación con la variable objetivo, sin embargo, no quiere decir que provoquen ese valor, es decir, que sean la causa. Siguiendo el ejemplo de la tasa de natalidad, según nuestro modelo, a mayor natalidad, menor será la esperanza de vida y viceversa. Esto se debe claramente a que los países en los que la tasa de natalidad es baja, suelen ser países desarrollados, con una mayor esperanza de vida y, en cambio, los países con una mayor natalidad suelen ser subdesarrollados, con una menor esperanza de vida.

Por tanto, los factores que más influyen en el modelo se han interpretar como aquellos que más caracterizan la esperanza de vida.

Gracias a la aplicación de ***wrapper methods*** se ha podido concluir que las features clave con las que se puede obtener un error en la predicción de la esperanza de vida aceptable son únicamente las nueve que se listan a continuación:

- Probabilidad de morir por enfermedades cardiovasculares, cáncer, diabetes o enfermedades respiratorias crónicas entre los 30 y los 70 años de vida y su intervalo de confianza al 95 %

- Tasa de natalidad
- Porcentaje de muertes ocasionadas por heridas
- Año
- GNI per cápita
- Porcentaje de la población con acceso a servicios básicos de agua potable
- Género
- Incidencia de tuberculosis
- Ratio de homicidio

Al igual que tan solo con estas features del conjunto de datos se puede obtener un error aceptable, también es probable que se pueda con otro conjunto, sin embargo, las pruebas requeridas para encontrar otro conjunto con este número de factores que obtengan un buen error es computacionalmente muy costoso.

Mediante el **algoritmo genético** se ha desarrollado un modelo que, dando un margen de cambio como parámetro sobre el valor de los factores iniciales de un país en un año y un género, es capaz de optimizar dichas variaciones para maximizar la esperanza de vida. Mediante la interpretación de los cambios que propone este algoritmo para cada caso, se ha concluido que la red de neuronas se basa en los valores de los casos con una alta esperanza de vida, de tal forma que cuanto más se parezcan los valores de las features a dichos casos, más alta será la esperanza de vida. Y, para las features que en dichos casos tengan un valor alto(en valor absoluto), cuanto más extremo sea dicho valor más aumentará la esperanza de vida. Estos indicadores son:

- Porcentaje de la población con acceso a servicios básicos de saneamiento e higiene
- Muertes por causa de tráfico
- Probabilidad de morir por enfermedades cardiovasculares, cáncer, diabetes o enfermedades respiratorias crónicas entre los 30 y los 70 años de vida y su intervalo de confianza al 95 %
- Intervenciones contra enfermedades tropicales desatendidas(NTD)
- Tasa de natalidad
- Tasa de homicidio
- Dieta por macronutrientes

Gracias al último enfoque planteado para la interpretación de la red de neuronas, se ha podido descubrir la gran utilidad de SHAP sobre modelos de caja negra. SHAP nos permite entender de forma muy sencilla, la influencia de cada feature sobre el resultado final, al igual que poder observar cómo influyen los valores de ciertas features sobre otras.

Gracias a este enfoque se ha podido concluir el tipo de influencia, ya sea positiva o negativa, sobre el resultado final de cada factor mediante los diagramas de violín de las figuras 4.14 y 4.15. A su vez, se concluye que los indicadores que mayor influencia tienen en el cálculo son, en orden:

- Servicios de sanidad básica cubiertos
- Probabilidad de morir por enfermedades cardiovasculares, cáncer, diabetes o enfermedades respiratorias crónicas entre los 30 y los 70 años de vida y su intervalo de confianza al 95 %
- Muertes por causa de tráfico
- Tasa de natalidad
- Tasa de homicidio
- Porcentaje de la población con acceso a servicios básicos de saneamiento e higiene
- Incidencia de tuberculosis
- Intervenciones contra enfermedades tropicales desatendidas(NTD)
- Número de médicos por cada 10.000 habitantes
- Porcentaje de la población con acceso a servicios para el lavado de manos con jabón y agua en el hogar
- Consumo de carne

Sin embargo, a pesar de que estos factores son más relevante a nivel general, para cada caso o comparación influirán unos u otros. Por ejemplo, para la evolución de la esperanza de vida de año en año, los indicadores que más influyen varían, teniendo un peso mucho más importante, por ejemplo, los relativos a la alimentación en el caso planteado en la figura 4.21.

Si la comparación se realiza por conjuntos de países con esperanza de vida similar concluimos que, mientras que lo que marca la diferencia para los países con una alta esperanza de vida son los factores que más afectan a nivel general, los que más influyen en los que tienen una baja esperanza de vida son las muertes por conflictos armados y terrorismo y los relativos a la alimentación como son:

- Consumo de piña
- Consumo de fruta: otros cítricos
- Consumo de proteína vegetal
- Consumo de bananas
- Consumo de grasa
- Proporción de la población que los gastos en sanidad exceden el 25 % de sus ingresos
- Consumo de bebidas alcohólicas

Por último, gracias al análisis de casos individuales que nos proporciona SHAP, se ha concluido que, a pesar de que hay factores que tienen un mayor peso sobre la esperanza de vida, en general, casi todos los contenido en el conjunto de datos tienen, aunque sea, una pequeña influencia en el resultado final.

5.2. Líneas futuras

Este trabajo ofrece una serie de posibles ampliaciones y mejoras a realizar.

Despliegue

Se podría realizar un despliegue a una aplicación web con interfaz con la cual poder experimentar con el modelo implementado observando la influencia de los factores sobre el resultado final. Se podría habilitar el estudio de casos hipotéticos con valores introducidos por el usuario, modificaciones sobre casos ya existentes para ver el impacto de los cambios o comparativas entre diferentes casos para ver la influencia y diferencia de los indicadores de entrada.

La implementación de esta propuesta se podría hacer, por ejemplo, mediante el servicio cloud *Azure Function*[57], que proporciona la infraestructura para su realizarlo y ofrece un servicio que no necesita servidor.

Diferente conjunto de datos

Se podría aplicar este enfoque mediante un conjunto de datos diferente, ya sea ampliando o reduciendo el número de factores de entrada. Ya sea con diferentes indicadores o extraídos de otras fuentes, con menos valores desconocidos para no condicionar tanto el modelo.

Además, se podría separar en varios modelos diferentes, por ejemplo para países desarrollados, subdesarrollados y en vías de desarrollo. De esta forma, indicadores como la tasa de natalidad quizá podrían tener una influencia más significativa.

Otra posibilidad sería seleccionar un conjunto reducido de factores especialmente interesantes, eliminando features de mayor influencia como el año, para estudiar con qué exactitud son capaces de determinar la esperanza de vida.

Modificación del algoritmo genético

Mediante los nuevos enfoques se podría aplicar el algoritmo genético, pero esta vez teniendo en cuenta los valores máximo y mínimo posibles de cada indicador, penalizando que sobrepasen esos valores determinados, por ejemplo, por el máximo y mínimo valor existente de cada feature dentro del conjunto de datos.

Bibliografía

[1] bismart, "¿qué hacemos? - etl." https://blog.bismart.com/es/que-hacemos-etl.

[2] P. Rodó, "Distribución normal." https://economipedia.com/definiciones/distribucion-normal.html, 2019.

[3] Wikipedia, "Distribución exponencial." https://es.wikipedia.org/wiki/Distribuci%C3%B3n_exponencial, 2021.

[4] P. N. Roldán, "Modelo de regresión." https://economipedia.com/definiciones/modelo-de-regresion.html, 2016.

[5] C. E. Ouyang, "Introduction to machine learning with python - chapter 2 - datasets and knn." https://elvinouyang.github.io/study%20notes/python-datasets-and-knn/, 2017.

[6] J. M. Heras, "Árboles de decisión con ejemplos en python." https://www.iartificial.net/arboles-de-decision-con-ejemplos-en-python/#Arboles_de_Decision_para_Regresion, 2020.

[7] J. M. Alvarez, "El perceptrón como neurona artificial." http://blog.josemarianoalvarez.com/2018/06/10/el-perceptron-como-neurona-artificial/, 2018.

[8] Atria Innovation, "Qué son las redes neuronales y sus funciones." https://www.atriainnovation.com/que-son-las-redes-neuronales-y-sus-funciones/, 2019.

[9] M. Li, "Tectonic discrimination of olivine in basalt using data mining techniques based on major elements." https://www.researchgate.net/figure/K-fold-cross-validation-method_fig2_331209203.

[10] D. Nieves, "¿qué es el escenario de entrenamiento, validación y prueba de conjuntos de datos en aprendizaje automático?." https://es.quora.com/Qu%C3%A9-es-el-escenario-de-entrenamiento-validaci%C3%B3n-y-prueba-de-conjuntos-de-datos-en-aprendizaje-autom%C3%A1tico, 2021.

[11] S. Mazzanti, "Shap values explained exactly how you wished someone explained to you." https://towardsdatascience.com/shap-explained-the-way-i-wish-someone-explained-it-to-me-ab81cc69ef30, 2020.

[12] E. O.-O. Max Roser and H. Ritchie, "Life expectancy," *Our World in Data*, 2013. https://ourworldindata.org/life-expectancy.

[13] M. Evaluation, "Lesson 3: Life tables." https://www.measureevaluation.org/resources/training/online-courses-and-resources/non-certificate-courses-and-mini-tutorials/multiple-decrement-life-tables/lesson-3.

[14] G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.

[15] T. Kluyver, B. Ragan-Kelley, F. Pérez, B. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. Hamrick, J. Grout, S. Corlay, P. Ivanov, D. Avila, S. Abdalla, and C. Willing, "Jupyter notebooks – a publishing format for reproducible computational workflows," in *Positioning and Power in Academic Publishing: Players, Agents and Agendas* (F. Loizides and B. Schmidt, eds.), pp. 87 – 90, IOS Press, 2016.

[16] W. McKinney *et al.*, "Data structures for statistical computing in python," in *Proceedings of the 9th Python in Science Conference*, vol. 445, pp. 51–56, Austin, TX, 2010.

[17] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, "API design for machine learning software: experiences from the scikit-learn project," in *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pp. 108–122, 2013.

[18] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. Software available from tensorflow.org.

[19] F. Serredilla, "Salga."

[20] S. Lundberg., "Documentación shap en python." `https://shap.readthedocs.io/en/latest/index.html`, 2018.

[21] C. W. Hansen, "The effect of life expectancy on schooling: Evidence from the international health transition," *University of Southern Denmark*, 2012.

[22] M. S. Md. Nazrul Islam MONDAL, "Impact of socio-health factors on life expectancy in the low and lower middle income countries," *Iran J Public Health*, 2013. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4441932/.

[23] C. G. Marco Tulio Ribeiro, Sameer Singh, ""why should i trust you?": Explaining the predictions of any classifier." `https://arxiv.org/abs/1602.04938`, 2016.

[24] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting modelpredictions." `https://arxiv.org/abs/1705.07874`, 2017.

[25] D. Vorotyntsev, "What's wrong with lime." `https://towardsdatascience.com/whats-wrong-with-lime-86b335f34612`, 2020.

[26] F. Piccinini, "Interpreting black-box machine learning models with genetic algorithms." https://towardsdatascience.com/interpreting-black-box-machine-learning-models-with-genetic-algorithms-a803bfd134cb, 2020.

[27] World Bank, "World development indicators." https://databank.worldbank.org/source/world-development-indicators, 2021.

[28] World Health Organization, "World health organization datasets." https://www.who.int/data/collections, 2021.

[29] UNICEF, "Dataset archives." https://data.unicef.org/resources/resource-type/datasets/, 2021.

[30] Our World In Data, "World indicators." https://ourworldindata.org/, 2021.

[31] FAOSTAT, "Food and agriculture organization of the united nations data." http://www.fao.org/faostat/en/#data, 2021.

[32] U. Sharma, "World health statistics 2020 complete geo-analysis." https://www.kaggle.com/utkarshxy/who-worldhealth-statistics-2020-complete, Enero 2021.

[33] Naciones Unidas, "Estados miembro." https://www.un.org/es/about-us/member-states, 2021.

[34] Greelane, "Detecte la presencia de valores atípicos con la regla del rango intercuartílico." https://www.greelane.com/es/ciencia-tecnolog%C3%ADa-matem%C3%A1ticas/mates/what-is-the-interquartile-range-rule-3126244, 2018.

[35] N. N. Taleb, *The Black Swan: The Impact of the Highly Improbable*. Random House, 2007.

[36] A. Barai, "Normal distribution and machine learning." https://medium.com/analytics-vidhya/normal-distribution-and-machine-learning-ec9d3ca05070, 2020.

[37] ICHI.PRO, "Distribución normal y aprendizaje automático." https://ichi.pro/es/distribucion-normal-y-aprendizaje-automatico-260160071159920.

[38] B. AI, "Funciones de probabilidad." https://bootcampai.medium.com/funciones-de-probabilidad-fbd59eb55b59, 2020.

[39] J. L. Cano, "Ajuste e interpolación unidimensionales básicos en python con scipy," *Pybonacci*, 2013.

[40] J. Brownlee, "knn imputation for missing values in machine learning," *Machine Learning Mastery*, 2020.

[41] Mahomet, "Reverse a get dummies encoding in pandas," *Stack Overflow*, 2020.

[42] J. Brownlee, "How to use data scaling improve deep learning model stability and performance." https://machinelearningmastery.com/how-to-improve-neural-network-stability-and-modeling-performance-with-data-scaling/, 2019.

[43] B. Roy, "All about feature scaling." https://towardsdatascience.com/all-about-feature-scaling-bcc0ad75cb35, 2020.

[44] V. N. G. Raju, K. P. Lakshmi, V. M. Jain, A. Kalidindi, and V. Padma, "Study the influence of normalization/transformation process on the accuracy of supervised classification," in *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, pp. 729–735, 2020.

[45] G. Box and D. R. Cox, "An analysis of transformations, journal of the royal statistical society." https://www.ime.usp.br/~abe/lista/pdfQWaCMboK68.pdf, 1964.

[46] Yeo and R. Johnson, "A new family of power transformations to improve normality or symmetry," 2000.

[47] J. Brownlee, "How to use power transforms for machine learning." https://machinelearningmastery.com/power-transforms-with-scikit-learn/, 2020.

[48] R. L. y Fernando Ortega, "Material de la asignatura de "machine learning"del grado de ingeniería del software en la escuela técnica superior de sistemas informáticos de la universidad politécnica de madrid," 2020.

[49] F. Serradilla, "Material de la asignatura de .ªgentes inteligentes"del grado de ingeniería del software en la escuela técnica superior de sistemas informáticos de la universidad politécnica de madrid," 2021.

[50] G. Perez, "¿por qué es relu la función de activación más común utilizada en redes neuronales?." https://es.quora.com/Por-qu%C3%A9-es-ReLU-la-funci%C3%B3n-de-activaci%C3%B3n-m%C3%A1s-com%C3%BAn-utilizada-en-redes-neuronales, 2021.

[51] J. B. Diederik P. Kingma, "Adam: A method for stochastic optimization." https://arxiv.org/abs/1412.6980, 2014.

[52] J. Brownlee, "A gentle introduction to early stopping to avoid overtraining neural networks." https://machinelearningmastery.com/early-stopping-to-avoid-overtraining-neural-network-models/, 2018.

[53] J. Holland, "Adaptation in natural and artificial systems," 1975.

[54] C. Darwin, *On the Origin of Species*. John Murray, 1859.

[55] L. Shapley, "Shapley values," 1953.

[56] P. Płoński, "Random forest feature importance computed in 3 ways with python." https://mljar.com/blog/feature-importance-in-random-forest/, 2020.

[57] Microsoft, "Documentación de azure functions." `https://docs.microsoft.com/es-es/azure/azure-functions/`.