COLEGIO DE CIENCIAS E INGENIERIA

INGENIERIA INDUSTRIAL

IIN - 3007 Analítica de Datos

NRC 2573

NOMBRE DEL ENTREGABLE: Proyecto Final Analítica de Datos

SEMESTRE: Segundo Semestre 2022 – 2023 (202220)

NOMBRE(S) Y CÓDIGO DE ESTUDIANTE(S): Gabriel Vélez 00212365, Joaquín Orbe 00320820, Gabriel Torres 00137882

PROFESOR(A): Gabriela Baldeón

FECHA DE ENTREGA: 02 / 05 / 23

Índice

Introducción

Los ataques cerebrales son una de las causas de muerte principales a nivel mundial. Usando datos de un hospital público, con distintas variables y mostrando si el paciente sufrió o no el ataque cerebral, el objetivo del proyecto es predecir si una persona tiene altas o bajas probabilidades de sufrir de un ataque cerebral en base a estas 10 variables predictivas y una variable respuesta que es la que muestra si un paciente sufrió de un ataque cerebral o no. Para lograr esto es necesario primero manipular la base de datos para luego poder trabajar con esta y encontrar las variables que más influyen en la variable respuesta, probar distintos modelos, finalmente escoger uno y poder con este predecir si una persona va a sufrir de un ataque cerebral o no; y, finalmente se incluyen los costos de aplicar el modelo de predicción en un hospital.

Operaciones de preprocesamiento, EDA y visualizaciones de las variables predictivas.

La base de datos consistía en 5110 observaciones con 11 variables, las cuales estaban divididas en variables numéricas y categóricas. Antes de poder trabajar con esta base de datos, fue necesario realizar una limpieza de esta. Para esto se utilizaron diferentes análisis para cada tipo de variable.

Se realizó un análisis de los datos para cada una de las variables categóricas. Se encontraron algunos valores irrelevantes en las variables “Gender” y “Work”. En el caso de la variable “Gender” existía una observación llamada “Other” la cual se decidió eliminar al no ser una cantidad significativa de observaciones. Para la variable “Work” se encontró un total de siete observaciones incoherentes, las cuales no tenían ningún sentido. Nuevamente al no ser una cantidad de observaciones significativas, se decidió eliminarlas. Finalmente, se transformó a todas las variables categóricas en variables dummies.

Para las variables numéricas se realizaron diagramas de caja para poder visualizar los valores y así encontrar valores atípicos. Como podemos observar en el anexo 20, la variable “age” no presenta valores atípicos, mientras que la variable “avg_glucose_level” si los presenta. Para asegurarnos que estos valores atípicos son erróneos o no, se comparó el valor más alto de esta variable, con el valor real más alto de glucosa en sangre. Al ser el valor máximo de las observaciones menor al valor máximo real reportado, decidimos no eliminar ninguno de estos valores, ya que es probable que estos datos sean correctos y relevantes para la variable de respuesta que tenemos. Por último, se buscaron valores nulos en estas dos variables y no se encontró ninguno.

Para la visualización de datos, se decidió graficar individualmente las variables predictivas para poder observar y analizar su comportamiento. Tal es el caso de la edad donde se entendió que la mayoría de las personas tenían entre 40 y 60 años (como se observa en el anexo 23), así como se encontraban con un nivel de glucosa promedio entre 75 a 100 mg/dL (como se observa en el anexo 24), y así sucesivamente con el resto de los gráficos los cuales nos ayudaron a entender mejor a los clientes reportados en la base de datos.

En el caso de la variable "bmi", el diagrama de caja muestra valores atípicos muy separados del resto de datos, se encontró un total de 4 observaciones con valores de 40000. Estos valores se reemplazaron por valores nulos. Se analizó también la presencia de valores nulos en esta variable y se encontró un total de 204 valores nulos, incluyendo los cuatro valores atípicos que se transformaron a nulos. Al ser una cantidad significativa de observaciones de la base de datos, decidimos imputar estos valores nulos. Para esto utilizamos varios métodos de regresión y se evaluó a cada uno de estos métodos con el valor de R^2.

Modelo utilizado	**R^2**
Regresión lineal con variables predictivas numéricas	0.12
Regresión lineal con todas las variables predictivas	0.03
Regresión Lasso	0.24
Regresión Ridge	0.24
Árbol de decisión	0.46
K- Nearest Neighbors	1.00

Como podemos ver en la Tabla 1, el mejor modelo de imputación fue el K-Nearest Neighbors, el cual obtuvo un valor r cuadrado de 1. Los valores faltantes fueron reemplazados con los valores generados por K-Nearest Neighbors.

Una vez finalizada la limpieza de datos, nuestra base de datos consiste en un total de 5102 observaciones y 16 variables.

Correlación entre las variables predictivas y su importancia para la variable de respuesta.

Como se puede ver en el anexo 1 (ver sección de anexos), se muestran las diferentes correlaciones entre las variables predictivas con la variable de respuesta que en este caso es Stroke la que nos interesa, entonces se puede ver que la variable predictiva que más influye

es la edad (age), y le sigue el nivel promedio de glucosa (avg_glucose_level), hipertensión (hypertension) y enfermedad al corazón (heart disease). Las características con mayor importancia serán las que contribuyan más a la predicción del modelo. Cabe recalcar que esto solo analiza una correlación lineal entre las variables, es decir, se analiza uno a uno con las variables, es por eso que no se confían de esos datos si no solo es para darse una idea, entonces más adelante se analizará esta sección detalladamente.

Bases de Datos No Balanceadas

Una base de datos no balanceada o unbalanced dataset se da cuando, "en una base de datos, el número de observaciones de una clase o grupo es significativamente mayor que el de otras clases" (Badr, 2019). Esto puede generar varios problemas por el desbalance en la base de datos: "El algoritmo puede tener un sesgo hacia las clases con mayores observaciones, ignorando a las que tienen un menor número" (AprendeIA, 2019). Además, se puede generar overfitting ya que el algoritmo puede sobre ajustar los datos de la clase con mayor número de observaciones. Sin embargo, existen dos soluciones principales durante el entrenamiento del algoritmo: Oversampling y undersampling. El oversampling se refiere a generar datos sintéticos aleatorios para la clase minoritaria. Una de las técnicas usadas para hacer esto es conocida como SMOTE (Synthetic Minority Over-sampling Technique) o K-Nearest Neighbors, encontrando así observaciones cercanas para predecir una nueva observación. Por otro lado, el undersampling quiere decir que se eliminan algunas observaciones de forma aleatoria de las clases mayoritarias para que se igualen al número de observaciones de las clases minoritarias.

Prueba Estadística

Se ha determinado que la base de datos no está balanceada y que la variable no balanceada es la variable respuesta: stroke, como se puede ver en el anexo 3 (ver anexos). Para solucionar el problema, se ha decidido aplicar tanto el oversampling como el undersampling. Después de aplicar ambas técnicas, se ha decidido probar con una prueba estadística cuál de estas es mejor. Entonces, se ha decidido usar la prueba de Wilcoxon, que es una prueba no paramétrica permite comparar dos muestras relacionadas. En este caso se ha elegido usar esta prueba porque no asume una distribución normal de los datos y puede manejar datos con valores atípicos y de tamaño pequeño. Además, esta prueba estadística

compara las clasificaciones ordenadas de las muestras, entonces es adecuado para evaluar la efectividad de técnicas de clasificación como el oversampling y el undersampling, comparando así la precisión promedio de los modelos. A través de una prueba de hipótesis y la prueba de Wilcoxon se ha concluido que se rechaza la hipótesis nula, es decir que el puntaje promedio para los modelos entrenados con oversampling y undersampling son distintos, y que el modelo con mayor puntaje es el de undersampling, por lo que se elige este modelo como solución a la base de datos no balanceada.

División de la base de datos

Se dividió a la base de datos en una proporción de 60-20-20 para los sets de entrenamiento, validación y prueba respectivamente. Basamos esta decisión en el artículo "How much data is needed to train a medical image deep learning system, to achieve necessary high accuracy" (Cho, Lee et al., 2016), en el cual se mencionaba la cantidad de observaciones necesarias para poder entrenar adecuadamente a un algoritmo de análisis de imágenes médicas. El articulo utilizaba una muestra de 6000 observaciones, las cuales son muy cercanas al número de observaciones en nuestra base de datos.

4 algoritmos implementados con sus debidas justificaciones y el modelo seleccionado

Se escogieron los 4 algoritmos en base a la optimización de los hiperparámetros, ya que en todos se usa un modelo de optimización. Los algoritmos seleccionados son: Random Forest Classifier, Support Vector Machines, Regresión Logística, y Redes Neuronales Artificiales (ANN).

Se eligió el Random Forest Classifier por su capacidad para trabajar con grandes cantidades de datos, y por ser capaz de trabajar con datos continuos y categóricos. Este es el modelo de ensamble, que se eligió por su capacidad de combinar múltiples árboles de decisión para mejorar el modelo. Los valores de los otros hiperparámetros se seleccionan mediante la validación cruzada. Es el mejor modelo en todas las métricas de evaluación por lo que ha sido el algoritmo seleccionado.

Support Vector Machines es el segundo algoritmo, que también es capaz de trabajar con grandes conjuntos de datos y además puede trabajar con datos no linealmente separables.

Los valores de los hiperparámetros se optimizan utilizando una búsqueda en cuadrícula para encontrar la mejor combinación que maximice la precisión del modelo.

La Regresión Logística es un algoritmo que se eligió por su capacidad de manejar muchos datos y porque es fácil de interpretar. Los hiperparámetros incluyen la regularización; y, los otros hiperparámetros se obtienen mediante la validación cruzada.

El algoritmo de Redes Neuronales Artificiales (ANN) es el nuevo modelo aplicado, que es capaz de modelar relaciones complejas en los datos y también es capaz de trabajar con grandes cantidades de estos. Este algoritmo usa capas ocultas que permiten aprender características más complejas de los datos. Uno de los hiperparámetros es justamente el número de capas ocultas, y hay otros como la tasa de aprendizaje.

Matriz de confusión de cada algoritmo y curva ROC. Tabla resumida de las métricas de evaluación con sus algoritmos. Explicación y selección de la mejor métrica de evaluación junto con su algoritmo.

Para evaluar cada uno de los algoritmos se ha calculado distintas métricas de evaluación, entre ellas la matriz de confusión y la curva ROC para el set de entrenamiento, validación, y de prueba. Tanto la matriz de confusión como la curva ROC para cada algoritmo y para cada uno de los sets de la división de la base de datos se muestran en la sección de anexos.

A continuación, se muestra una tabla con las distintas métricas de evaluación: exactitud, precisión, sensibilidad, F1 score, AUC, especificidad y ROC, para cada uno de los algoritmos y además para cada set de entrenamiento, validación y prueba:

	Random Forest Classifier	Support Vector Machines	Regresión Logística	Redes Neuronales Artificiales (ANN)
Set de entrenamiento	Exactitud: 1.0 Precision: 1.0 Sensibilidad: 1.0 F1 Score: 1.0 AUC: 1.0 Especificidad: 1.0 ROC: 1.0	Exactitud: 0.776 Precision: 0.7442 Sensibilidad: 0.826 F1 Score: 0.783 AUC: 0.7771 Especificidad: 0.7282 ROC: 0.7770617572156198	Exactitud: 0.7338 Precision: 0.7097 Sensibilidad: 0.7712 F1 Score: 0.7392 AUC: 0.7346 Especificidad: 0.6979 ROC: 0.7345723684210527	Exactitud: 0.8148 Precision: 0.7768 Sensibilidad: 0.8719 F1 Score: 0.8216 AUC: 0.816 Especificidad: 0.7601 ROC: 0.8160052348613468
Set de validación	Exactitud: 0.9892 Precision: 0.9797 Sensibilidad: 1.0 F1 Score: 0.9897 AUC: 0.9887 Especificidad: 0.9774 ROC: 0.9887096774193548	Exactitud: 0.7693 Precision: 0.7676 Sensibilidad: 0.7994 F1 Score: 0.7832 AUC: 0.768 Especificidad: 0.7366 ROC: 0.7679831272047262	Exactitud: 0.7173 Precision: 0.7259 Sensibilidad: 0.7352 F1 Score: 0.7305 AUC: 0.7165 Especificidad: 0.6978 ROC: 0.7165136639891199	Exactitud: 0.5211 Precision: 0.5211 Sensibilidad: 1.0 F1 Score: 0.6852 AUC: 0.5 Especificidad: 0.0 ROC: 0.5

Set de prueba	Exactitud: 0.9876 Precision: 0.9764 Sensibilidad: 1.0 F1 Score: 0.9881 AUC: 0.9874 Especificidad: 0.9747 ROC: 0.9873551106427818	Exactitud: 0.7745 Precision: 0.7681 Sensibilidad: 0.8006 F1 Score: 0.784 AUC: 0.7739 Especificidad: 0.7471 ROC: 0.7738532212314442	Exactitud: 0.7353 Precision: 0.7383 Sensibilidad: 0.7472 F1 Score: 0.7427 AUC: 0.735 Especificidad: 0.7229 ROC: 0.7350483946105351	Exactitud: 0.8074 Precision: 0.795 Sensibilidad: 0.8399 F1 Score: 0.8168 AUC: 0.8067 Especificidad: 0.7734 ROC: 0.8066624432141959

Como se puede ver, entre las distintas métricas de evaluación se tienen diferentes datos los cuales representan cada uno una cosa diferente. No obstante, se decidió seleccionar la sensibilidad como métrica de evaluación a guiarse ya que, al estar trabajando con temas médicos de salud en donde se trata con el cuerpo humano y la vida de las personas, es preferible que se vuelva más sensible el sistema ya que es super riesgoso, porque se prefiere predecir a alguien que, si tiene alta probabilidad de sufrir un ataque cerebral y luego no termine teniendo, a que sea al revés, esto se conoce como un falso positivo.

Conociendo esto, el mejor algoritmo termina siendo el de Random Forest Classifier, con un valor de 1, que es lo que se buscaba con una sensibilidad super alta. Y de la misma manera, el algoritmo menos deseado sería en este caso la de regresión logística.

Análisis de las variables predictivas más importantes para la predicción.

Anteriormente se analizó la correlación entre las variables predictivas para tener una idea de cuáles son las más importantes, sin embargo, esto fue mediante una relación lineal. Entonces se utilizó el método basado en random forest el cual tiene una propiedad de importancia de características. Primero se dividió a los datos en un set de entrenamiento, validación y prueba, luego se creó el modelo y se ajustó a los datos de entrenamiento. Seguido se obtuvo la importancia de características del modelo entrenado mediante la función “feature_importances_” y se les ordena a las características por su importancia en orden descendente, finalmente se les grafica en un histograma para observar mejor. Cabe recalcar que las variables predictivas más importantes son la edad (age), el nivel promedio de glucosa (avg_glucose_level) e índice de masa corporal (Bmi) como se puede ver en el anexo 2 (ver sección de anexos); estas variables tienen un ligero cambio que las anteriormente predichas porque este modelo logra comparar no solo mediante regresión lineal si no varios aspectos a la vez. Es importante mencionar que se puede utilizar esta información para reducir la dimensionalidad de ser necesario.

Costos de implementación del modelo.

Los costos de implementación para un modelo como este dependen de varios factores como lo son los costos de infraestructura informática, costos de desarrollo del modelo y los costos de la integración del modelo. Para implementar un modelo como este son necesarios servidores y una interfaz que facilite el uso de este por los empleados del hospital. Dependiendo del tamaño del hospital y la accesibilidad del modelo, este costo podría oscilar entre las decenas de miles a cientos de miles de dólares. Si asumimos que la obtención de los datos con los que trabajamos han sido obtenidos de pacientes, se debería tomar en cuenta este costo dentro del desarrollo del modelo. Este costo podría significar decenas de miles de dólares. El costo del desarrollo de un modelo como este también se debe tomar en cuenta. Por último, los costos de implementación del modelo a los sistemas ya existentes en el hospital, así como la capacitación del personal para su uso pueden representar un costo de decenas de miles de dólares. Si se trata de dar una cifra exacta, esta oscilaría entre los $120.000; sin embargo, se vuelve difícil por los costos ya que estos van a depender del tamaño del hospital, la cantidad de usuarios, la cantidad de pacientes, la infraestructura actual del hospital, entre otros.

La factibilidad de implementar este modelo en un hospital depende de la capacidad adquisitiva del hospital y el uso que se le daría al mismo. Las ventajas que este modelo podría traer a la institución adecuada, sería la asistencia en diagnóstico a los doctores para la detección y tratamiento temprano para pacientes que tienen altas posibilidades de sufrir un ataque cerebrovascular. Otra ventaja sería la implementación de infraestructura que permitirá implementar más modelos como este para distintas afecciones.

Las desventajas de la implementación de este modelo es la inversión alta que se debe realizar para implementar el mismo. Otra desventaja es que es un modelo muy específico, el cual se enfoca en un tipo de afección que no es demasiado común del que no todos los pacientes y especialistas podrían beneficiarse.

Enlace al repositorio de GitHub:

https://github.com/ggvvmm/Proyecto-Analitica.git

Conclusiones

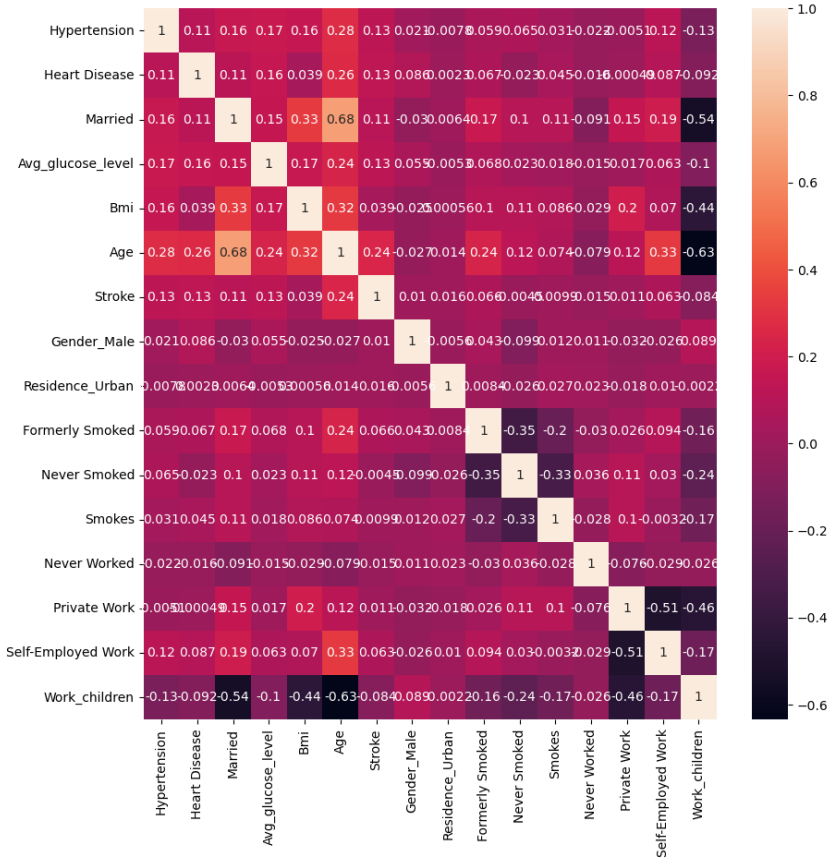
En conclusión, luego de generar un Exploratory Data Analysis (EDA), haber preprocesado y balanceado los datos; así como para entrenarlos, validarlos y probarlos con el sistema, se los aplica diferentes algoritmos para evaluar; hasta que finalmente se aplican métricas de evaluación para saber qué algoritmo es el mejor y de cuál guiarse, se afirma que el modelo es exitoso, teniendo un porcentaje de predicción del 98% de exactitud y 97% de precisión; esto teniendo en cuenta que se eligió el modelo utilizado con Random Forest Classifier ya que fue el mejor en distintas características, y pudiendo afirmar que las mejores variables predictivas que ayudan a la importancia de la variable de respuesta "Stroke" son: la edad (age), el nivel promedio de glucosa (avg_glucose_level) e índice de masa corporal (Bmi) como se puede ver en el anexo 2 (ver sección de anexos). Cabe recalcar que se prefirió que el modelo sea sensible, ya que, al trabajar con temas de salud, puede ser riesgoso y se prefiere minimizar los falsos negativos.

Se recomienda a las personas tener los conceptos claros de Machine Learning antes de empezar con el proyecto, ya que eso puede ahorrar mucho tiempo, así como centrarse en el preprocesamiento de la base de datos, porque eso es un punto clave para que sea un proyecto exitoso.

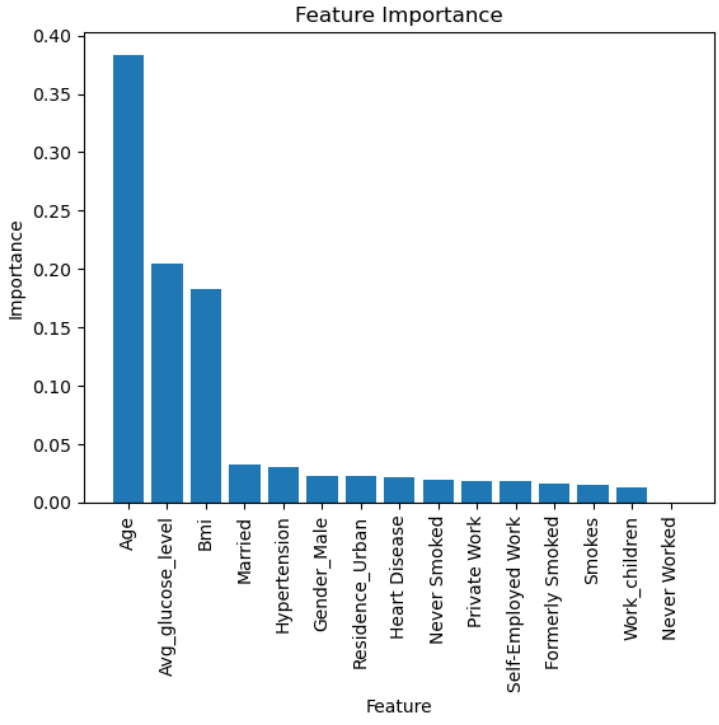
Entre las limitaciones del proyecto, encontramos que no se pudo escoger las variables predictivas para el proyecto, siendo esto que, si se hacía un estudio previo, se podría haber determinado mejores variables y que tengan un aporte más relevante a la variable de respuesta.

ANEXOS

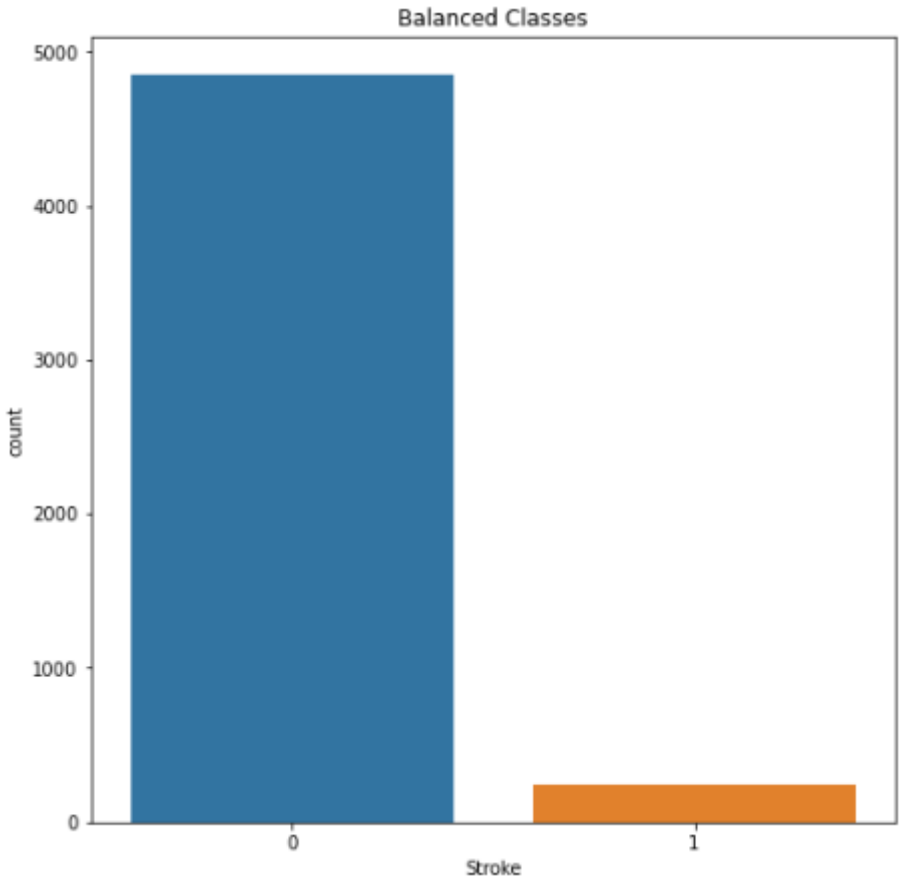
Anexo 1. Tabla de correlación entre las variables predictivas y la variable de respuesta



Anexo 2. Histograma con las variables predictivas más importantes



Anexo 3. Histograma con clases de la variable respuesta stroke



Anexo 4. Random Forest Classifier – Set de entrenamiento – Matriz de confusión

```
Set de entrenameinto:
Confusion Matrix (Accuracy 1.0000)

       Prediction
Actual    0    1
     0 2976    0
     1    0 2850
```

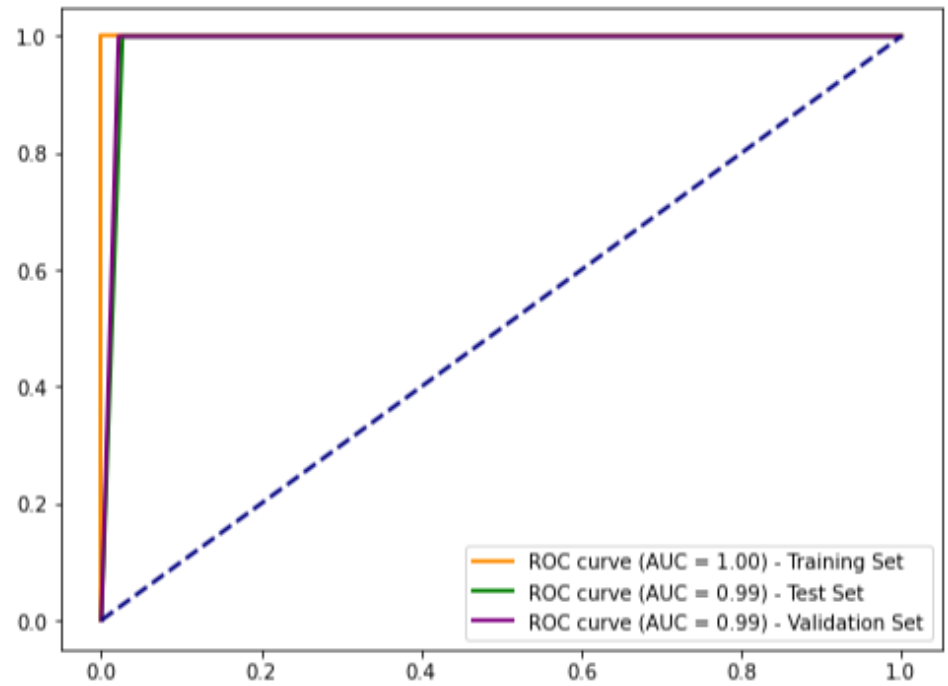
Anexo 5. Random Forest Classifier – Set de validación – Matriz de confusión

```
Set de validación:
Confusion Matrix (Accuracy 0.9871)

       Prediction
Actual   0   1
     0 924  25
     1   0 993
```

Anexo 6. Random Forest Classifier – Set de prueba – Matriz de confusión

```
Set de prueba:
Confusion Matrix (Accuracy 0.9897)

       Prediction
Actual    0    1
     0  910   20
     1    0 1012
```

Anexo 7. Random Forest Classifier – Curvas ROC



Anexo 8. Support Vector Machines – Set de entrenamiento – Matriz de confusión

```
Set de entrenameinto:
Confusion Matrix (Accuracy 0.7662)

       Prediction
Actual    0    1
     0 2124  852
     1  510 2340
```

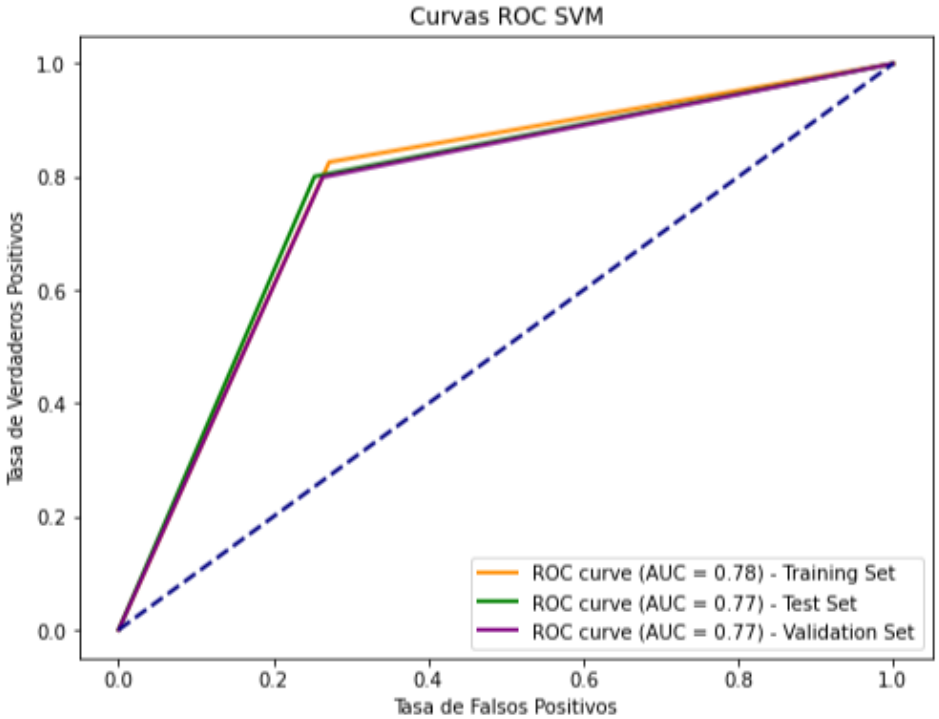
Anexo 9. Support Vector Machines – Set de validación – Matriz de confusión

```
Set de validación:
Confusion Matrix (Accuracy 0.7703)

       Prediction
Actual   0   1
     0 688 261
     1 185 808
```

Anexo 10. Support Vector Machines – Set de prueba – Matriz de confusión

```
Set de prueba:
Confusion Matrix (Accuracy 0.7848)

       Prediction
Actual   0   1
     0 679 251
     1 167 845
```

Anexo 11. Support Vector Machines – Curvas ROC



Anexo 12. Regresión Logística – Set de entrenamiento – Matriz de confusión

```
Set de entrenameinto:
Confusion Matrix (Accuracy 0.7278)

       Prediction
Actual    0    1
     0 2059  917
     1  669 2181
```

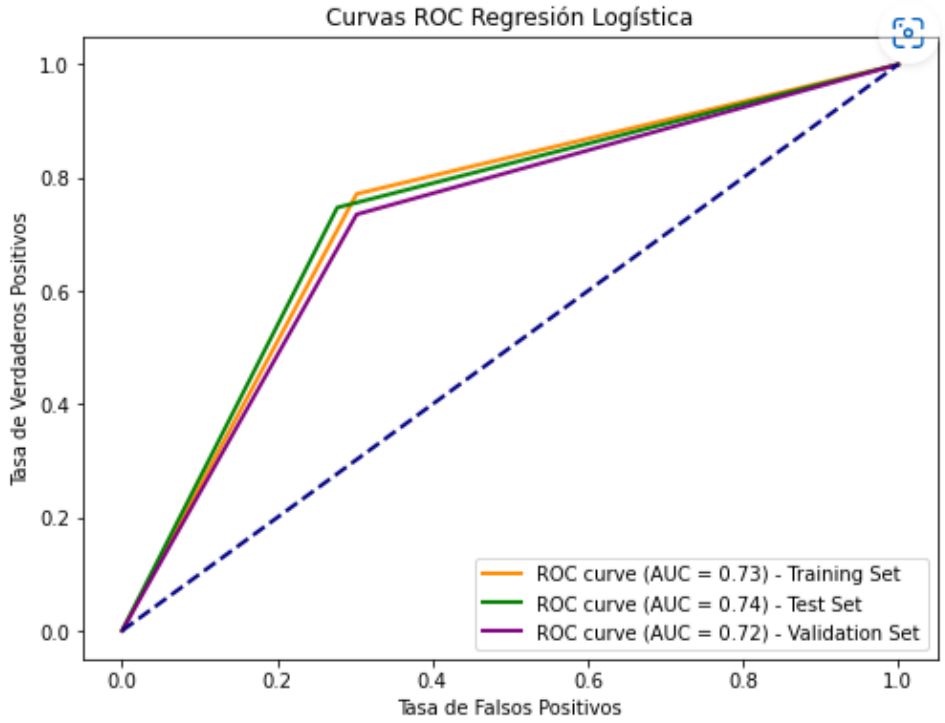
Anexo 13. Regresión Logística – Set de validación – Matriz de confusión

```
Set de validación:
Confusion Matrix (Accuracy 0.7400)

       Prediction
Actual   0   1
     0 667 282
     1 223 770
```

Anexo 14. Regresión Logística – Set de prueba – Matriz de confusión

```
Set de prueba:
Confusion Matrix (Accuracy 0.7451)

       Prediction
Actual   0   1
     0 652 278
     1 217 795
```

Anexo 15. Regresión Logística – Curvas ROC



Anexo 16. Redes Neuronales Artificiales (ANN) – Set de entrenamiento – Matriz de confusión

```
Set de entrenameinto:
Confusion Matrix (Accuracy 0.8304)

       Prediction
Actual    0    1
     0 2306  670
     1  318 2532
```

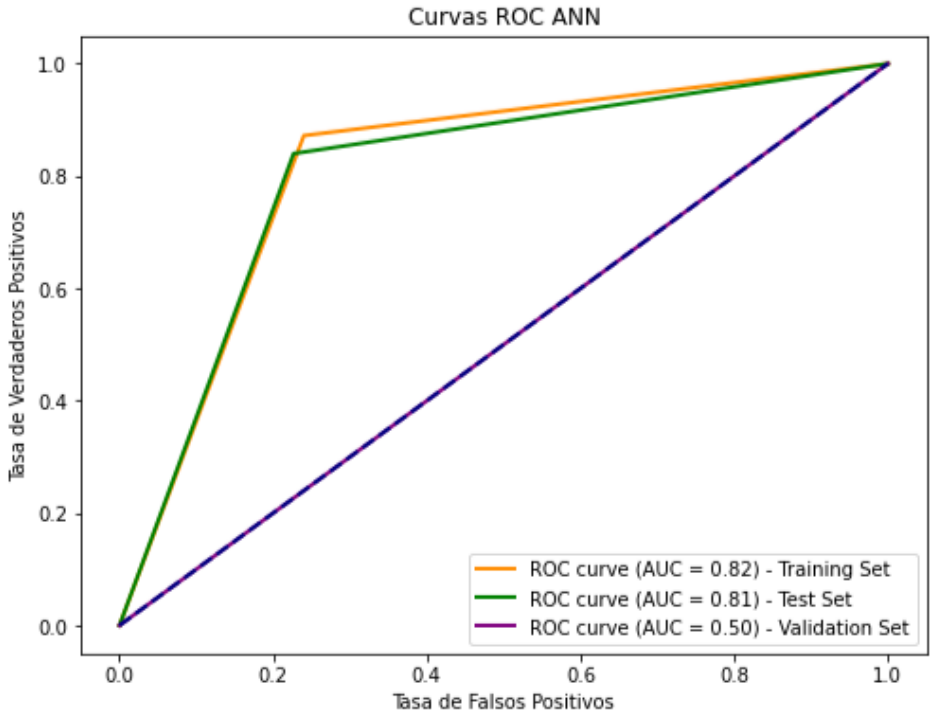
Anexo 17. Redes Neuronales Artificiales (ANN) – Set de validación – Matriz de confusión

```
Set de validación:
Confusion Matrix (Accuracy 0.8234)

       Prediction
Actual   0   1
     0 726 223
     1 120 873
```

Anexo 18. Redes Neuronales Artificiales (ANN) – Set de prueba – Matriz de confusión

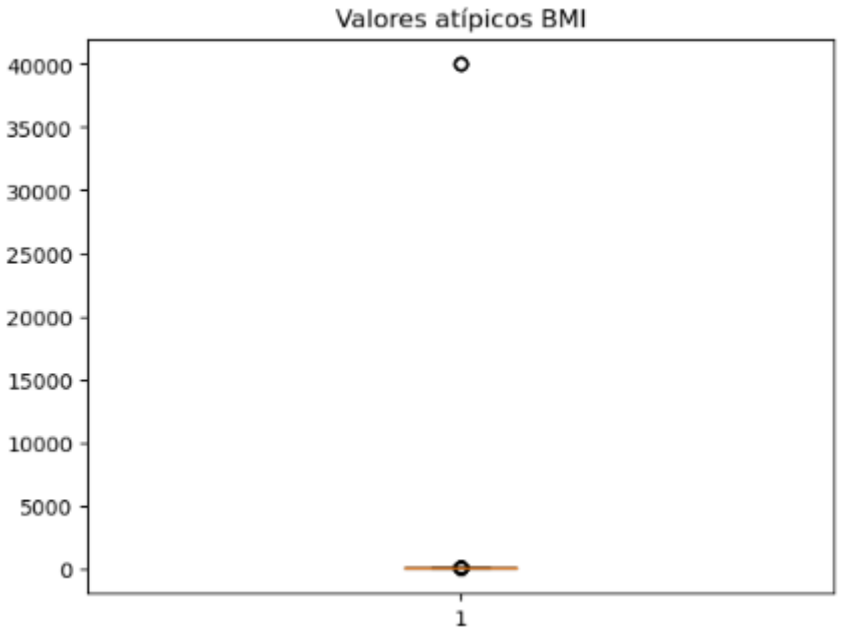
```
Set de prueba:
Confusion Matrix (Accuracy 0.5999)

       Prediction
Actual   0   1
     0 216 714
     1  63 949
```

Anexo 19. Redes Neuronales Artificiales (ANN) – Curvas ROC

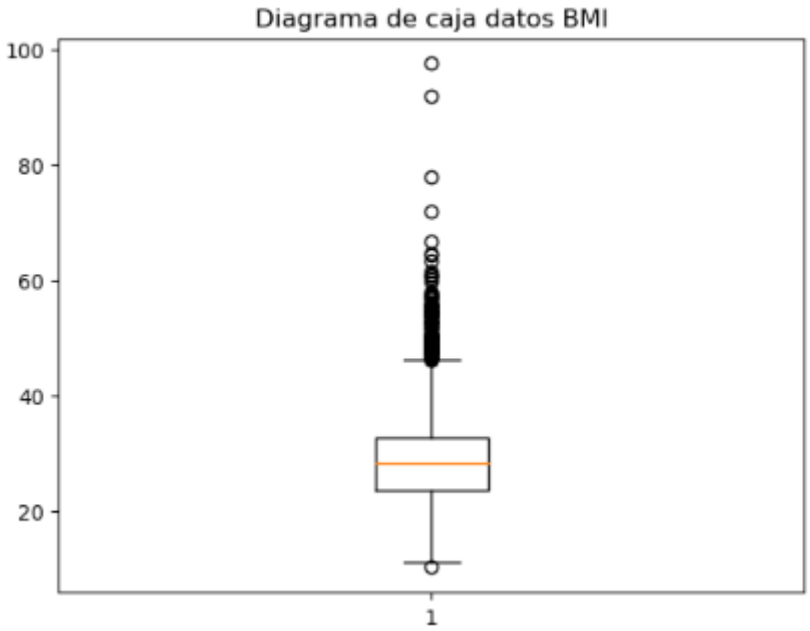


Anexo 20. Diagrama de caja variables “age” y “avg_glucose_level”

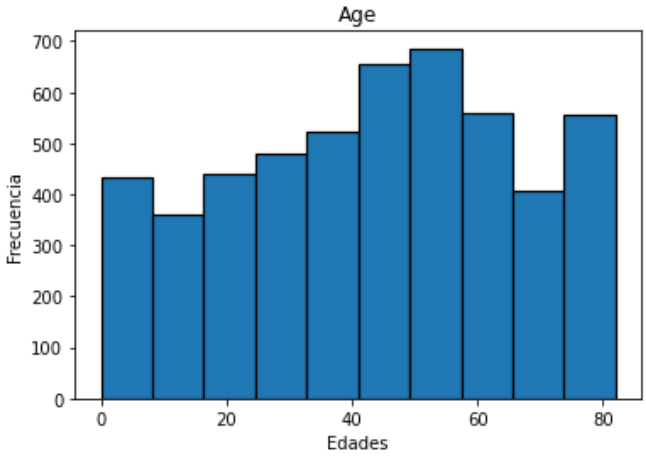
Anexo 21. Diagrama de caja variable “bmi” valores atípicos



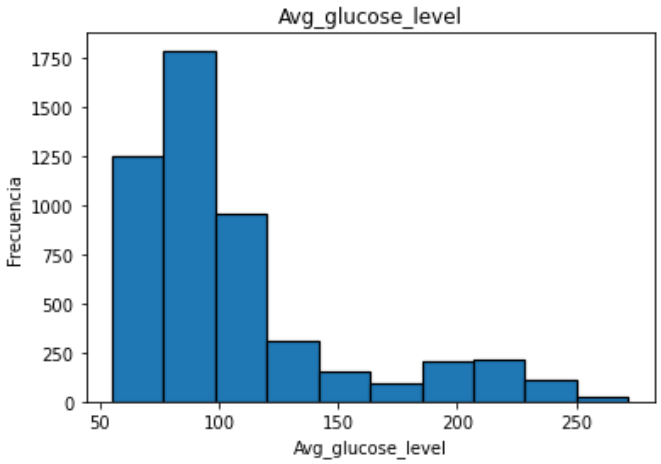
Anexo 22. Diagrama de caja variable “bmi” valores imputados



Anexo 23. Histograma para visualizar los datos de edades



Anexo 24. Histograma para visualizar los datos de niveles de glucosa promedio



Referencias

AprendeIA. (2019). Conjunto de datos desbalanceado. AprendeIA. Conjunto de datos desbalanceado - Aprende IA

Ariadna. (s/f). Algoritmos neuronales. Complex systems and AI. Algoritmos neuronales: sistemas complejos e inteligencia artificial (complex-systems-ai.com)

Badr, W. (2019). Having an Imbalanced Dataset? Here Is How You Can Fix It. Towards Data Science. Having an Imbalanced Dataset? Here Is How You Can Fix It. | by Will Badr | Towards Data Science

Cho, J. Lee, K. Et.al. (2016) How much data is needed to train a medical image deep learning system, to achieve necessary high accuracy.

Gutierrez-Garcia, J. O. (2021). Datos de Entrenamiento, Validación y Prueba: ¿Cómo crearlos y qué objetivos tienen? Machine Learning. YouTube. https://www.youtube.com/watch?v=vdYzm4xC7mc

Johnson, A. E., Stone, D. J., Celi, L. A., & Pollard, T. J. (2018). The MIMIC code repository: enabling reproducibility in critical care research. Journal of the American Medical Informatics Association. https://academic.oup.com/jamia/article/25/1/32/4259424

Kuo, M. H., Chang, F. L., & Su, M. H. (2018). The benefits and challenges of applying big data in health care: A systematic review. Journal of nursing research, 26(3), 174-182. https://journals.lww.com/jnrtwna/Fulltext/2017/12000/The_Lived_Experience_of_Gynecologic_Cancer.7.aspx