CLASIFICADOR PARA LA DETECCIÓN DE CÁNCER DE SENO

Andrea Fabiana Villamizar Ruiz - 2161352
Anngy Nathalia Gómez Ávila - 2161343
María Fernanda Vera Negrón - 2161326

Inteligencia artificial

Francy Liliana Camacho Urrea

Universidad Industrial de Santander

2.019

0.1 Motivación para el desarrollo del proyecto

El cáncer de mama es el cáncer más común en las mujeres, tanto en el plano mundial como en la región de las Américas. En muchos países desarrollados, la mortalidad por este cáncer ha mostrado una tendencia significativa al descenso durante los últimos veinte años que se ha atribuido a los avances en el tratamiento y a la implementación de programas organizados de tamizaje.

En Colombia y en otros países de la región, el cáncer de mama empieza a perfilarse como un problema de salud pública, con un incremento en la incidencia y en la mortalidad y con una razón incidencia/mortalidad desfavorable, lo que se ha atribuido principalmente a problemas en el desempeño de los servicios de salud.

Datos sobre el cáncer de seno en Colombia.
Foto: EL TIEMPO

Uno de los factores críticos en la atención del cáncer es la oportunidad con la que se instauran los tratamientos. En cáncer de mama las demoras se han asociado con una menor supervivencia: un tiempo superior a tres meses entre la aparición de síntomas y el inicio del tratamiento disminuye la supervivencia global en 12 por ciento. Cuando el cáncer de mama se detecta tempranamente, se establece un diagnóstico adecuado, y se dispone de tratamiento para actuar de manera inmediata y efectiva, incrementando las posibilidades de curación. Las muertes por cáncer de seno representan el 15% de las muertes por cáncer, un estudio realizado en el año 2.015 muestra que 570.000 son las las muertes en el mundo a causa del cáncer de seno de las cuales 2.055 se presentan en Colombia.

El objetivo principal del proyecto "Clasificador para la detección de cáncer de seno" es determinar de manera rápida si un tumor detectado es benigno o maligno, teniendo como base características específicas de dicha masa y de está manera establecer un diagnóstico y tratamiento adecuado.

0.2 Tema principal de inteligencia artificial abordado

Machine Learning

En un tipo de técnicas de Inteligencia Artificial donde las computadoras aprenden a hacer algo sin ser programadas para ello. Es un conjunto de métodos capaces de detectar automáticamente patrones en los datos. Los tipos de algoritmos de aprendizaje automático difieren en su enfoque, el tipo de datos que ingresan y generan, y el tipo de tarea o problema que están destinados a resolver.Según esto pueden ser: Aprendizaje supervisado, aprendizaje no supervisado, aprendizaje de refuerzo, aprendizaje de características, entre otros; basándandos en la temática del proyecto, nos centraremos en aprendizaje supervisado.

Supervised learning(prendizaje supervisado): Son problemas que ya hemos resuelto, pero que seguirán surgiendo en un futuro. La idea principal consiste en que las computadoras aprendan de una multitud de ejemplos, y a partir de ahí se pueda realizar el resto de cálculos necesarios para no tener que ingresar nuevamente información. Ejemplos: reconocimiento de voz, detección de spam, reconocimiento de escritura, entre otros.

Para llevar a cabo el proyecto se tuvo en cuenta que el problema planteado es posible solucionarlo con *algoritmos de machine learning de clasificación con aprendizaje supervisado* debido a que estos algoritmos se usan cuando el resultado deseado es una etiqueta discreta, en otras palabras, son útiles cuando la respuesta al problema cae dentro de un conjunto finito de resultados posibles, el modelo entrenado es para predecir cualquiera de las dos clases objetivos, verdadero o falso, por ejemplo, se le conoce como clasificación binaria. Los métodos de clasificación que se implementaron en el transcurso del proyecto son: Regresión Logistica, Random Forest, KNN, SVM, y adicionalmente se aplicó Redes neuronales profundas DNN (deep learning).

Para llevar a cabo los modelos de clasificación anteriormente nombrados se implementó la librería Scikit-learn (sklearn) para aprendizaje de máquina de software libre en el lenguaje de programación Python, incluye varios algoritmos de clasificación, regresión y análisis de grupos.

- Regresión logística:
  Es un Algoritmo Supervisado y se utiliza para clasificación, requiere tamaños de muestra bastante grande, da resultados muy eficientes y no requiere demasiados recursos computacionales que hacen que sea asequibles ejecutar la producción. Adicional a esto, se debe tener en cuenta que no asume ningún error en la variable dependiente esto quiere decir, que es necesario eliminar las instancias mal clasificadas de los datos de entrenamiento; cabe

destacar, que es un algoritmo lineal, con una transformación no lineal en la salida.

- Random Forest:
  Es un método versátil de machine learning capaz de realizar tareas tanto de regresión como de clasificación. Maneja grandes cantidades de datos, tiene un método efectivo para estimar datos faltantes y mantiene la precisión cuando falta ua gran porción de datos, No predice más allá del rango en los datos de entrenamiento, y sobreajusta los conjuntos de datos que son particularmente ruidosos.

- KNN (K Nearest Neighbours):
  Es uno de los algoritmos de clasificación más simples, aunque a pesar e su simplicidad puede superar los clasificadores más potentes. Este algoritmo consiste en seleccionar un valor de K. Al momento del análisis los K datos más cercanos al valor que se desea predecir será la solución, acá lo importante es seleccionar un valor de K acorde a los datos para tener una mayor precisión en la predicción. Cabe destacar que es computacionalmente costoso, porque el algoritmo almacena todos los datos de entrenamiento.

- SVM (Support Vector Machines):
  Es un clasificador discriminatorio definido formalmennte por un hiperplano de separación, para datos linealmente separables, este algoritmo de clasficación funciona increíblemente bien, para los datos que son casi linealmente separables, SVM puede funcionar bien con el valor correcto de hiperplano, y para datos que no son separables linealmente, podemos proyectar los dato al especio donde es perfectamente o casi linealmente separables para obtener buenos resultados.

Arquitecturas de deep learning tales como las redes profundas neuronales(DNN) también fueron implementadas en este proyecto.

- Redes Redes neuronales profundas(deep learning):
  Una red neuronal profunda (DNN) es una red neuronal artificial (ANN) con múltiples capas entre las capas de entrada y salida. El DNN encuentra la manipulación matemática correcta para convertir la entrada en la salida, ya sea una relación linea o una relación no lineal. La red se mueve a través de las capas calculando la probabilidad de cada salida. Los DNN suelen ser redes de alimentación directa en las que los datos fluyen desde la capa de entrada a la capa de salida sin bucles. Al principio, el DNN crea un mapa de neuronas virtuales y asigna valores numéricos aleatorios, o "pesos", a las conexiones entre ellos. Los pesos y las entradas se multiplican y devuelven una salida entre 0 y 1. Si la red no reconoce con precisión un patrón particular, un algoritmo ajustaría los pesos. De esa manera, el

algoritmo puede hacer que ciertos parámetros sean más influyentes, hasta que determine la manipulación matemática correcta para procesar completamente los datos.

0.3 Funcionamiento y simulación del proyecto

Inicialmente se toma una muestra de líquido del seno del paciente, este procedimiento ambulatorio implica el uso de una aguja de pequeño calibre para tomar el líquido, conocido como aspiración con aguja fina (FNA), directamente de un bulto o masa del seno, que se detectó previamente mediante autoexamen y/o mamografía. El líquido de la FNA se coloca en un portaobjetos de vidrio y se tiñe para resaltar los núcleos de las células constituyentes. Una cámara de video montada en un microscopio transfiere una imagen de la FNA a una estación de trabajo. ahí se determinan los límites exactos de los núcleos. Para una imagen típica que contiene entre 10 y 40 núcleos, el proceso de análisis de la imagen tarda aproximadamente de dos a cinco minutos.

Se calculan diez características para cada núcleo:

- área
- radio
- perímetro
- simetría
- número de concavidades
- tamaño de las concavidades
- dimensión fractal(del limite)
- compacidad
- suavidad(variación local de los segmentos radiales
- textura(varianza de los niveles de gris en el interior del límite)

El valor medio(mean), el valor extremo (WORST) (es decir, el valor mayor o el peor: el tamaño más grande, la forma más irregular) y el error estándar (SE) de cada una de estas funciones celulares se calculan para cada imagen, lo que da como resultado un total de 30 características para cada muestra.

En el tratamiento de datos inicialmente se reemplazan los valores que no son númericos; los valores no númericos en este caso son los obtenidos en diagnostico 'M'(maligno) es reemplazado por 0 y 'B'(benigno) es reemplazado por 1, seguido a esto, se consulta la existencia de valores nulos en el dataset y se procede a

eliminarlos, de igual manera, se eliminan las columnas que no son útiles para el proyecto, es decir, características que no aportan información valiosa para el proceso de clasificación; las columnas 'id' y 'unnamed:32'. Seguidamente, se crea la matriz de correlaciones para observar los parámetros que están linealmente correlacionados. Al llevar a cabo el analisis de la matriz de correlaciones se observa que hay características relacionadas entre si,

-*radius_mean, perimeter_mean, y area_mean,* elegimos *radius_mean.*
-*radius_se, perimeter_se, y area_se,* elegimos *radius_se.*
-*radius_worst, perimeter_worst, y area_worst, elegimos radius_worst.*
-*radius_worst y radius_mean, elegimos radius_mean.*
-*concave points_mean, concavity_mean, y concave points_worst,* elegimos *concave points_mean*
-*texture_mean, y texture_worse,* elegimos *texture_mean.*

De las 14 variables anteriores conservamos solo 4 que son las siguientes: radius_mean, radius_se, concave points_mean y texture_mean.

Terminado el tratamiento de datos, se procede a realizar el entrenamiento y a implementar los siguientes métodos de clasificación: Logistic Regrassion, Random Forest, KNN, SVM. Para la evaluación de veracidad se utilizaron cross_val_score con diferente generador de validación: "KFold" validación cruzada de K iteraciones y "LeaveOneOut" validación cruzada dejando uno fuera; F1score y accuracy score.

```
Matriz de confusión
[[40  1]
 [ 0 73]]
tn:  40  fp:  1  fn:  0  tp:  73
Precisión
F1 score = 0.9932
Accuracy Score= 0.9912
Cross validation score 1: 0.9771
Cross validation score 2:0.9754
```

Figure 1: Resultados obtenidos al implementar el método Logistic Regression.

```
Matriz de confusión
[[40  1]
 [ 0 73]]
tn:  40  fp:  1  fn:  0  tp:  73
Precisión
F1 score = 0.9932
Accuracy Score= 0.9912
Cross validation score 1: 0.9737
Cross validation score 2:0.9754
```

Figure 2: Resultados obtenidos al implementar el método SVM.