UNIVERSIDAD TÉCNICA DE MACHALA

FACULTAD DE INGENIERÍA CIVIL

MAESTRÍA EN SOFTWARE

DESARROLLO DE UNA APLICACIÓN PARA EL DIAGNÓSTICO DE DIABETES TIPO 2 UTILIZANDO APRENDIZAJE MÁQUINA

SEGUNDO RUPERTO CISNEROS VACA

TUTORA: Ing. Mazón Olivo Bertha, PhD

COTUTOR: Ing. Iván Ramírez, PhD

MACHALA, 2025

PENSAMIENTO

"El aprendizaje automático es el estudio sistemático de algoritmos y sistemas con el fin de mejorar su conocimiento o desempeño a través de la experiencia. De igual forma, ha sido definido como el campo científico que le confiere a máquinas la habilidad de aprender a través de la programación, el proceso de aprendizaje tiene por lo general cuatro tipos diferentes de algoritmos, no supervisado, supervisado, semi supervisado y aprendizaje de refuerzo. Este explora el estudio y construcción de algoritmos que puedan no sólo aprender, sino igualmente hacer predicciones sobre los datos"

Flach, Peter. 2012

DEDICATORIA

A mi amada esposa, Mónica,

Eres mi compañera de vida, mi amiga, mi confidente y mi inspiración. Gracias por tu amor, tu apoyo y tu paciencia. Sin ti, este logro no habría sido posible.

Recuerdo el día en que nos conocimos, como si fuera ayer. Nos sentamos en una heladería y hablamos durante horas. Inmediatamente supe que eras la mujer de mi vida.

A lo largo de los años, has estado a mi lado en las buenas y en las malas. Me has apoyado en mis sueños, me has consolado en mis derrotas y me has amado incondicionalmente.

Eres la mujer más fuerte, inteligente y hermosa que conozco. Me haces un hombre mejor.

A mis queridos hijos, Jair y Gabriel,

Ustedes son mi razón de ser. Me inspiran a ser un mejor hombre y un mejor padre. Gracias por su amor, su alegría y su energía.

Recuerdo el día en que nacieron, como si fuera ayer. Fue el momento más feliz de mi vida.

Verlos crecer y desarrollarse me ha llenado de orgullo. Son hombres inteligentes, cariñosos y con un gran corazón.

Me siento muy afortunado de ser su padre.

Esta tesis es para ustedes. Es un testimonio de mi amor por ustedes y de mi compromiso con su futuro.

AGRADECIMIENTO

En primer lugar, quiero expresar mi más sincero agradecimiento a mi esposa, Mónica Herrera, por su amor, apoyo y comprensión incondicionales. Sin su apoyo, este logro no habría sido posible.

También quiero agradecer a mis hijos, Jair y Gabriel, por su amor, aliento y sacrificios. Han sido mi mayor inspiración a lo largo de mi vida.

Agradezco a mi Mami, Ana y a Mami Isa, por su guía, apoyo y paciencia. Su orientación ha sido invaluable para mí.

Agradezco a la Universidad Técnica de Machala por brindarme la oportunidad de estudiar y crecer como profesional.

Finalmente, quiero agradecer a todos los que me han apoyado de alguna manera en el proceso de investigación y redacción de esta tesis. Su apoyo ha sido invaluable para mí.

Con profundo agradecimiento.

RESPONSABILIDAD DE AUTORÍA

Yo, Segundo Ruperto Cisneros Vaca, con C.C 1713229563, declaro que el trabajo de titulación "Desarrollo de una aplicación para el diagnóstico de Diabetes Tipo 2 utilizando aprendizaje máquina", en opción al título de Magister en Maestría En Software, es original y auténtico; cuyo contenido: conceptos, definiciones, datos empíricos, criterios, comentarios y resultados son de mi exclusiva responsabilidad.

SEGUNDO RUPERTO CISNEROS VACA

C.C. 1713229563

Machala, 17 de marzo de 2025

REPORTE DE SIMILITUD

CERTIFICADO DE ANÁLISIS
magister

Trabajo Titulación-Maestría en Software-Ruperto Cisneros-2025 -compilatio

9% Textos sospechosos

5% Similitudes
< 1% similitudes entre comillas
0% entre las fuentes mencionadas
4% Idiomas no reconocidos

Nombre del documento: Trabajo Titulación-Maestría en Software-Ruperto Cisneros-2025 -compilatio.pdf
ID del documento: 7f850c334748bf0b2ba804fd257afc4d24ba267d
Tamaño del documento original: 1,6 MB
Autores: []

Depositante: BERTHA EUGENIA MAZON OLIVO
Fecha de depósito: 18/1/2025
Tipo de carga: interface
fecha de fin de análisis: 18/1/2025

Número de palabras: 21.372
Número de caracteres: 144.831

Ubicación de las similitudes en el documento:

Fuentes principales detectadas

Nº	Descripciones	Similitudes	Ubicaciones	Datos adicionales
1	**la.mathworks.com** | Deep Learning - MATLAB & Simulink https://la.mathworks.com/discovery/deep-learning.html **5 fuentes similares**	2%		Palabras idénticas: 2% (423 palabras)
2	**aprendeia.com** | Aprendizaje no Supervisado - Aprende IA https://aprendeia.com/aprendizaje-no-supervisado-machine-learning/ **2 fuentes similares**	< 1%		Palabras idénticas: < 1% (145 palabras)
3	**www.redalyc.org** https://www.redalyc.org/pdf/142/14239902.pdf **3 fuentes similares**	< 1%		Palabras idénticas: < 1% (87 palabras)
4	**www.datasource.ai** | Comprensión de la Matriz de Confusión y Cómo Implementarl... https://www.datasource.ai/es/data-science-articles/comprension-de-la-matriz-de-confusion-y-co... **4 fuentes similares**	< 1%		Palabras idénticas: < 1% (79 palabras)
5	**ichi.pro** | Comprender la matriz de confusión y cómo implementarla en Python https://ichi.pro/es/comprender-la-matriz-de-confusion-y-como-implementarla-en-python-545031...	< 1%		Palabras idénticas: < 1% (59 palabras)

Fuentes con similitudes fortuitas

Nº	Descripciones	Similitudes	Ubicaciones	Datos adicionales
1	**1library.co** | Los supuestos conceptuales - La praxeología: perspectivas metodológicas https://1library.co/article/los-supuestos-conceptuales-la-praxeología-perspectivas-metodológicas...	< 1%		Palabras idénticas: < 1% (32 palabras)
2	**rstudio-pubs-static.s3.amazonaws.com** | Parte I Trabajo Final-Modelado Analítico ... https://rstudio-pubs-static.s3.amazonaws.com/1075094_3a08ae067ac74bd696009718bbe67735....	< 1%		Palabras idénticas: < 1% (36 palabras)
3	**repositorio.espe.edu.ec** https://repositorio.espe.edu.ec/bitstream/21000/20332/5/T-ESPE-038714.pdf.txt	< 1%		Palabras idénticas: < 1% (13 palabras)
4	**aleph.org.mx** | qué son las lipoproteínas y cuál es su función https://aleph.org.mx/que-son-las-lipoproteinas-y-cual-es-su-funcion	< 1%		Palabras idénticas: < 1% (17 palabras)
5	**Documento de otro usuario** #b51550 El documento proviene de otro grupo	< 1%		Palabras idénticas: < 1% (10 palabras)

Fuente mencionada (sin similitudes detectadas)

Estas fuentes han sido citadas en el documento sin encontrar similitudes.

1 https://github.com/srcisneros/DiabetesTipoII

CERTIFICACIÓN DEL TUTOR

Yo, Bertha Mazón Olivo, PhD, con C.C 0603100512, tutora del trabajo de titulación “Desarrollo de una aplicación para el diagnóstico de Diabetes Tipo 2 utilizando aprendizaje máquina”, en opción al título de Magister en Software, ha sido revisado según los procedimientos científicos, técnicos, metodológicos y administrativos establecidos por el Centro de Postgrado de la Universidad Técnica de Machala (UTMACH), razón por la cual doy fe de los méritos suficientes para que sea presentado a evaluación.

Ing. BERTHA MAZÓN OLIVO, PhD

Tutora

Machala, 17 de marzo de 2025

CESIÓN DE DERECHOS DE AUTOR

Yo, Segundo Ruperto Cisneros Vaca, con C.C. 1713229563; autor del trabajo de titulación "**DESARROLLO DE UNA APLICACIÓN PARA EL DIAGNÓSTICO DE DIABETES TIPO 2 UTILIZANDO APRENDIZAJE MÁQUINA**, en opción al título de Magíster En Software III, declaro bajo juramento que:

- El trabajo aquí descrito es de mi autoría, que no ha sido presentado previamente para ningún grado o calificación profesional. En consecuencia, asumo la responsabilidad de la originalidad del mismo y el cuidado al remitirse a las fuentes bibliográficas respectivas para fundamentar el contenido expuesto, asumiendo la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.
- Cedo a la Universidad Técnica de Machala de forma o exclusiva con referencia a la obra en formato digital los derechos de:
  a) Incorporar la mencionada obra en el repositorio Institucional para su democratización a nivel mundial, respetando lo establecido por la Licencia Creative Commons Atribution-No Comercial – Compartir igual 4.0 Internacional (CC BY NCSA 4,0); la Ley de Propiedad Intelectual del Estado Ecuatoriano y el Reglamento Institucional.
  b) Adecuarla a cualquier formato o tecnología de uso en INTERNET, así como correspondiéndome como Autora la responsabilidad de velar por dichas adaptaciones con la finalidad de que no se desnaturalice el contenido o sentido de la misma.

SEGUNDO RUPERTO CISNEROS VACA

CC. 1713229563

Machala, 17 de marzo de 2025

RESUMEN

La diabetes es una enfermedad crónica de larga duración que representa un desafío significativo para la salud pública a nivel mundial. La Asociación Americana de Diabetes (ADA) la clasifica como un problema creciente debido a su asociación con complicaciones graves, que afectan la calidad y esperanza de vida de los pacientes. En 2022, se estimó que 463 millones de personas padecían diabetes tipo 2, una cifra alarmante que subraya la urgencia de implementar estrategias innovadoras para su manejo y diagnóstico. Esta enfermedad aumenta considerablemente el riesgo de complicaciones crónicas, como enfermedades cardiovasculares, accidentes cerebrovasculares, ceguera y amputaciones, lo que genera un impacto negativo en la salud y calidad de vida de las personas.

En este contexto, el aprendizaje automático se presenta como una herramienta poderosa en el diagnóstico de enfermedades. Diversos estudios han demostrado su eficacia no solo en la detección de diabetes, sino también en enfermedades cardiovasculares, cáncer de mama y Covid-19, entre otras. Este trabajo se enfoca en el desarrollo de una aplicación web para el diagnóstico de diabetes tipo 2 mediante el uso de aprendizaje automático, guiado por la metodología CRISP-DM para la creación y evaluación de modelos de clasificación. El desarrollo de la aplicación web se realizó siguiendo las prácticas de la metodología XP. Los datos utilizados provienen de 961 observaciones de pacientes atendidos en la Clínica de Salud y Bienestar Postural, Quito - Ecuador, entre julio y diciembre de 2023, en un rango de edad de 30 a 60 años. De los modelos evaluados, Random Forest destacó como el más preciso, alcanzando un 99.46% de accuracy y un AUC(Área bajo la curva) de 99.98%. El principal aporte de esta investigación es proporcionar una herramienta basada en inteligencia artificial que facilite a médicos y pacientes el diagnóstico temprano de la diabetes tipo 2. Esta solución no solo reduce los costos asociados al diagnóstico, sino que mejora su accesibilidad y brindan apoyo específicamente a quienes residen en zonas rurales.

Palabras Clave: aprendizaje máquina, algoritmos de aprendizaje supervisado, Random Forest, diagnóstico de diabetes tipo 2.

ABSTRACT

Diabetes is a chronic, long-term disease that poses a significant challenge to global public health. The American Diabetes Association (ADA) classifies it as a growing problem due to its association with severe complications that affect patients' quality of life and life expectancy. In 2022, an estimated 463 million people worldwide were diagnosed with type 2 diabetes—a staggering figure that highlights the urgency of implementing innovative strategies for its management and diagnosis. This disease significantly increases the risk of chronic complications such as cardiovascular diseases, strokes, blindness, and amputations, generating a negative impact on people's health and quality of life. In this context, machine learning is presented as a powerful tool in the diagnosis of diseases. Various studies have demonstrated its effectiveness not only in detecting diabetes but also in diagnosing cardiovascular diseases, breast cancer, COVID-19, and more. This research focuses on developing a web application for the diagnosis of type 2 diabetes using machine learning, guided by the CRISP-DM methodology for the creation and evaluation of classification models. The web application development followed best practices from the XP methodology. The data used was collected from 961 patients' observations at the Postural Health and Wellness Clinic in Quito- Ecuador, between July and December 2023, within an age range of 30 to 60 years. Among the evaluated models, Random Forest stood out as the most accurate, achieving a 99.46% accuracy and an AUC of 99.98%. The main contribution of this research is to provide an artificial-intelligence-based tool that facilitates healthcare professionals and patients the early diagnose of type 2 diabetes. This solution not only reduces costs associated with diagnostics, but it improves accessibility, and provides support specifically to those who live in rural areas.

Keywords: machine learning, supervised learning algorithms, Random Forest, type 2 diabetes diagnosis.

ÍNDICE GENERAL

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

ABREVIATURAS Y SÍMBOLOS

CACES - Consejo de Aseguramiento de la Calidad de la Educación Superior.

LOES - Ley Orgánica de Educación Superior.

SNES - Sistema Nacional de Educación Superior.

MISS - Sistema de Información de Gestión.

CES - Consejo de Educación Superior.

IES - Instituciones de Educación Superior.

TIC - Tecnologías de la Información y Comunicación.

SES - Sistema de Educación Superior.

IES - Instituciones de Educación Superior.

DSS - Sistemas de Soporte de Decisión.

ETL - Extracción, transformación y carga.

ISO - Organización Internacional de Normalización.

BI - Inteligencia de Negocios.

VUEJS - Framework de JavaScript de código abierto, para frontend.

NODEJS - Framework de JavaScript de código abierto, para backend.

MYSQL - Gestor de bases de datos relacional.

JWT - Json Web Token.

API - Interfaz de Programación de Aplicaciones

SQL - Structured Query Language

BDA - Big Data Analytics

INTRODUCCIÓN

Actualmente, la diabetes se encuentra entre las enfermedades con mayores tasas de mortalidad en Ecuador. De acuerdo con las encuestas más recientes del INEC, se conoce como diabetes mellitus (DM) tipo 1 y es más común en infantes, niños y adolescentes con dificultades para producir insulina. Es importante tener en cuenta que la obesidad, la alimentación inadecuada y la diabetes gestacional son las principales causas de la diabetes tipo 2 en personas mayores de 40 años [1].

La diabetes es una condición de salud crónica en la cual la capacidad del cuerpo para producir insulina, lo que provoca problemas de salud graves como enfermedades cardíacas, enfermedades renales, pérdida de la visión, etc. La diabetes mellitus tipo 2 es un tipo de trastorno metabólico caracterizado por hiperglucemia (nivel alto de azúcar en la sangre) en presencia de resistencia a la insulina y falta relativa de insulina [2]. El aumento de casos de prediabetes es un problema mundial que probablemente generará mayores demandas de atención médica en el futuro cercano. A pesar de esto, la prediabetes no causará síntomas ya que eventualmente no envía señal alguna, paulatinamente conduce a la diabetes [2].

Según [3], con el fin de evitar o posponer estas complicaciones, la diabetes mellitus debe diagnosticarse temprano. Para [2], el diagnóstico de DM Tipo 2, se basa en un conjunto de pruebas, que incluyen un análisis de sangre para medir los niveles de glucosa y hemoglobina A1c. Estas pruebas pueden ser costosas, difíciles y pueden tardar varios días en completarse. Sin estar explícitamente programados para hacerlo, las computadoras pueden aprender de datos a través de la rama de la inteligencia artificial (IA) conocida como aprendizaje automático (ML: Machine Learning) [3]. ML se ha utilizado con éxito en una amplia gama de aplicaciones médicas, como el diagnóstico de distintos tipos de enfermedades. El desarrollo de una aplicación que utilice ML para el diagnóstico de la diabetes mellitus tipo 2 podría mejorar el diagnóstico temprano de la enfermedad [4]. Una aplicación precisa y fácil de usar podría ayudar a reducir la cantidad de personas que desarrollan complicaciones graves por la Diabetes Mellitus Tipo 2.

La diabetes se considera la segunda causa de mortalidad en Ecuador, siendo la primera causa de mortalidad en mujeres. Hasta noviembre de 2018, se han registrado 34597 casos de diabetes, según el Ministerio de Salud Pública [5].

El desarrollo de una aplicación que utiliza el aprendizaje máquina para el diagnóstico de diabetes tipo 2 es un tema de investigación relevante e importante porque nos permite confirmar un diagnóstico temprano; la diabetes tipo 2 es una enfermedad crónica que puede ser asintomática en sus etapas iniciales. Un diagnóstico temprano y preciso puede permitir intervenciones rápidas y cambios en el estilo de vida que pueden prevenir complicaciones a largo plazo.

Además, confirmaría la validez del análisis porque la aplicación podría considerar un diagnóstico más eficiente en comparación con los métodos convencionales, permitiendo una atención médica más acertada.

Un método computarizado y basado en el aprendizaje máquina podría reducir los costos asociados con diagnósticos repetidos y pruebas de laboratorio extensivas, así como la identificación del tratamiento según las particularidades únicas de cada paciente, la eficacia de las intervenciones médicas, y los datos recopilados por la aplicación podría contribuir a la investigación médica.

La creación de una aplicación de este tipo fomenta los avances en la inteligencia artificial, el aprendizaje máquina y fomenta el uso de tecnologías procedente en el sector de la salud.

Es importante destacar que una aplicación de diagnóstico de enfermedades, especialmente en el ámbito de la salud, debe desarrollarse en cooperación con profesionales médicos para garantizar la exactitud y seguridad.

En este contexto, y sabiendo que, en Ecuador, un problema que enfrenta el sector público a diario es la diabetes ya sea por una combinación de factores genéticos y ambientales, los cambios en el estilo de vida, incluidos los cambios en los hábitos alimenticios, pueden aumentar la obesidad y la inactividad física, dos factores importantes de riesgo para la diabetes tipo 2.

Por las razones antes expuestas, se formula el siguiente problema de investigación: ¿Cómo desarrollar una aplicación web para el diagnóstico de diabetes tipo 2 utilizando aprendizaje máquina?

Actualmente, no hay un enfoque sistemático para la detección temprana de la diabetes. Esto puede ser el resultado de programas de salud pública inexistentes, recursos insuficientes o una falta de conciencia sobre la relevancia de la detección temprana. En ciertos casos, los pacientes pueden no recibir un diagnóstico debido a la falta de atención médica disponible.

El objeto de estudio es la detección temprana de la diabetes mellitus tipo 2.

Las causas que delimitan el problema se citan a continuación:

- El diagnóstico de la diabetes tipo 2 a menudo implica múltiples pruebas y la interpretación de datos clínicos complejos.
- La ineficiencia en el proceso, conduce a diagnósticos tardíos y complicaciones mayores en el tratamiento de la enfermedad de diabetes en los pacientes.
- Esta enfermedad puede ser asintomática en sus etapas iniciales, lo que conduce a un diagnóstico tardío.
- Una aplicación sensible y específica podría contribuir al diagnóstico temprano y así mejorar las perspectivas de tratamiento.

El objetivo general de esta investigación es: Desarrollar una aplicación web con Angular que implemente un modelo de aprendizaje automático para el diagnóstico de la diabetes tipo 2 como asistencia de médicos especialistas.

Los objetivos específicos de este estudio son:

- Realizar una búsqueda bibliográfica de las técnicas de aprendizaje automático para diagnosticar Diabetes tipo 2 mediante la metodología de revisión sistemática de literatura.
- Elaborar una base de datos para diagnóstico de Diabetes con el apoyo de la Clínica de Salud y Bienestar Postural en edades comprendidas entre 30 y 60 años en el periodo julio – diciembre 2023.
- Aplicar algoritmos de redes neuronales artificiales para el diagnóstico de Diabetes tipo 2.
- Desarrollo de aplicación web que utilice el modelo de red neuronal más óptimo para el diagnóstico de diabetes tipo 2.

El campo de acción de este trabajo es el desarrollo de una aplicación web para el diagnóstico de diabetes tipo 2 utilizando aprendizaje máquina.

Para el diagnóstico de diabetes mellitus tipo 2, se debe considerar los siguientes datos: sexo, edad, peso y talla. Algunos exámenes médicos como: Glucosa en ayunas, Hemoglobina A1c, Colesterol, Triglicéridos, entre otros.

La hipótesis de este trabajo es: si se desarrolla una aplicación web que integre algoritmo de aprendizaje automático que apoye al médico especialista en el diagnóstico temprano de Diabetes Mellitus tipo 2, se logra más de un 75% de satisfacción.

Tabla 1

VARIABLES Y DIMENSIONAMIENTO

Variable	Definición	Categorías	Indicadores	Técnicas
Independiente: Desarrollo de una aplicación web para el diagnóstico de diabetes tipo 2 utilizando aprendizaje máquina.	La aplicación web para el diagnóstico de diabetes tipo 2 utilizando aprendizaje máquina es una aplicación que utiliza algoritmos de aprendizaje máquina para diagnosticar la diabetes tipo 2. La aplicación recopila datos del usuario, como la edad, sexo, el peso, la altura, los niveles de glucosa en sangre y otros factores de riesgo y utiliza estos datos para generar una probabilidad de que el usuario tenga diabetes.	Metodología del desarrollo de algoritmos de Machine Learning y el aplicativo web para el diagnóstico de diabetes tipo 2.	• Interpretación de los datos. • Fiabilidad de la obtención de datos. • Manejo de errores. • Desarrollo • Compilación. • Despliegue.	• Reuniones y entrevistas. • Observación • Creación de la data. • Manejo error de pruebas. • Técnicas de análisis. • Plataforma de compilación. • Plataforma de producción.
Dependiente: Apoyo al médico especialista en el diagnóstico temprano de Diabetes Mellitus tipo 2	Los algoritmos de ML pueden ser entrenados para interpretar los resultados de los análisis de sangre, lo que puede mejora de la precisión del diagnóstico de la Diabetes. Nivel de satisfacción de la aplicación web	Modelo de evaluación de algoritmos de ML para la precisión del diagnóstico de diabetes. Evaluación de la aplicación web	**Métricas** para evaluar modelos de clasificación: • Precisión • Sensibilidad • ROC-AUC CSAT(Customer Satisfaction Score)	• Pruebas • Plataforma de compilación. Encuesta de satisfacción a médicos especialistas en diagnóstico de Diabetes.

En la Tabla 1, se plantea las variables independiente y dependiente de este trabajo.

El diseño metodológico de este trabajo de investigación se plantea en base a las directrices expuestas en y consta de:

Un enfoque cuantitativo con el fin de evaluar y así comprender frecuencias, patrones, correlaciones, respuestas e hipótesis mediante análisis estadístico para expresar de manera gráfica y/o numérica los resultados encontrados objeto de la presente investigación.

Un alcance, correlacional dado que se requiere determinar si con una sintomatología en específico un paciente puede tener diabetes o ser propenso a ello, lo cual se logra a través del análisis de la relación o relaciones que existen entre las variables que intervienen en la enfermedad a través de procedimientos estadísticos.

Un diseño de investigación cuasi-experimental, dado que la muestra elegida para este estudio no es aleatoria; por el contrario, se trata de datos correspondientes a diagnósticos médicos de la enfermedad de diabetes mellitus tipo 2, realizados en orden de llegada de los pacientes a la Clínica

de Salud y Bienestar Postural, en edades comprendidas entre 30 y 60 años, en el periodo julio – diciembre 2023.

La metodología utilizada para el desarrollo de los algoritmos de Machine Learning que se utilizaron para detección de diabetes mellitus tipo 2 es CRISP-DM debido que es una metodología estándar para proyectos de minería de datos, ciencia de datos y Machine Learning. Se estructura en seis fases: Comprensión del negocio, Comprensión de los datos, Preparación de los datos, Modelado, Evaluación y Despliegue. Su enfoque iterativo y flexible permite abordar problemas desde la definición de objetivos hasta la implementación de resultados. Es aplicable en distintas industrias y facilita una ejecución ordenada del análisis de datos [6].

La metodología utilizada para el desarrollo de la aplicación web es La Programación Extrema (XP), debido a que es una metodología ágil de desarrollo de software enfocada en mejorar la productividad y la calidad del código mediante ciclos de desarrollo cortos y entregas frecuentes. Sus prácticas clave incluyen la programación en parejas, desarrollo guiado por pruebas, integración continua, refactorización constante y la participación activa del cliente. XP promueve la adaptabilidad a cambios en los requisitos, la simplicidad en el diseño y la comunicación constante en el equipo. Al priorizar la retroalimentación rápida y la colaboración, busca entregar software funcional de alta calidad de forma eficiente [7].

La estructura de este trabajo de titulación se organiza de manera sistemática en varias secciones clave para abordar la investigación de forma integral. La Introducción establece el contexto, los objetivos y la importancia del estudio. El Capítulo 1 se presenta el Marco Teórico con los antecedentes referenciales, históricos, conceptuales y antecedentes contextuales, de la diabetes tipo 2, tecnologías como Machine Learning, Deep Learning y metodologías CRISP-DM y XP, fundamentales para el desarrollo de este proyecto. En el Capítulo 2 se presenta la Metodología de Investigación, se detallan el enfoque, tipo de estudio, métodos teóricos y empíricos, junto con las herramientas utilizadas. En el Capítulo 3 se aborda el Desarrollo y Resultados; donde se describe la aplicación técnica de los modelos de Machine Learning y el desarrollo de la aplicación web. Finalmente, en el Capítulo 4 se presenta la Evaluación y Discusión de Resultados, se analizan los modelos de Machine Learning implementados, se aplican pruebas estadísticas e hipótesis y se

comparan los resultados obtenidos con trabajos relacionados, culminando con las Conclusiones y Recomendaciones que sintetizan los hallazgos y aportes del estudio.

CAPÍTULO I. MARCO TEORICO

En este apartado se describen los antecedentes referenciales, conceptuales, históricos y contextuales que caracterizan al problema de estudio e inducen al lector a entender la temática del trabajo.

1.1 Antecedentes Referenciales

Los antecedentes referenciales están basados en la búsqueda de trabajos académicos de investigación tales como artículos científicos, libros, capítulos de libros, artículos de congresos entre otros. Los mismos que a través de criterios de búsqueda, con la finalidad de emplear aquellos que estén relacionados con esta investigación, como resultado de ello se ha realizado una comparación con la parte experimental del presente trabajo investigativo.

Considerando aquello es importante mencionar que la revisión sistemática de literatura también forma parte de este trabajo investigativo, ya que dentro de la ingeniería de software esta metodología es recomendable para identificar, evaluar y combinar la evidencia de estudios primarios usando un método riguroso.

1.1.1 Preguntas de Investigación

Para guiar el desarrollo de una aplicación web para el diagnóstico de la diabetes tipo 2 utilizando aprendizaje automático, se aplicó las siguientes preguntas de investigación:

- P1. ¿Cuáles son los factores de riesgo de la diabetes tipo 2 que se pueden utilizar para desarrollar un modelo de aprendizaje automático preciso?
- P2. ¿Qué algoritmo de aprendizaje automático es el más adecuado para el diagnóstico de la diabetes tipo 2?
- P3. ¿Cómo seleccionar y serializar el modelo de aprendizaje automático más preciso para el diagnóstico de diabetes tipo 2?
- P4. ¿Cómo desarrollar la aplicación web que interactúe mediante una API con el modelo de aprendizaje automático que se ha seleccionado para hacer el diagnóstico de diabetes tipo 2?

1.1.2 Palabras clave y cadenas de búsqueda

En el argumento del desarrollo de una aplicación web para el diagnóstico de la diabetes tipo 2 utilizando aprendizaje automático, las siguientes son algunas palabras clave relevantes:

Diabetes tipo 2, Aprendizaje automático, Diagnóstico, Factores de riesgo, Algoritmos de aprendizaje automático, Precisión, Confianza, Accesibilidad.

En el desarrollo de una aplicación web para el diagnóstico de la diabetes tipo 2 utilizando aprendizaje automático, las siguientes son algunas cadenas de búsqueda relevantes:

- "diabetes tipo 2" OR "diabetes mellitus tipo 2"
- "predictive modeling" OR "modelado predictivo"

1.1.3 Criterios de Inclusión y Exclusión

Los criterios de inclusión para la selección de artículos académicos y de investigación que se han considerado para este trabajo son:

- Estudios primarios
- Trabajos de los últimos cinco años de publicación.
- Investigaciones relacionadas con los objetivos de este trabajo.
- Estudios relacionados con algoritmo machine learning aplicados en la predicción de diabetes.
- Trabajos de desarrollo de aplicaciones web que consumen APIs de machine learning.

Los criterios de exclusión son:

- Estudios secundarios
- Artículos que no pertenecen a revistas indexadas.
- Trabajos que no están directamente relacionado con el tema u objetivos de esta investigación.
- Trabajo redundante o ninguna contribución significativa a la investigación.
- Estudios cortos menores a cuatro páginas.
- Trabajos duplicados o redundantes

1.1.4 Resultados de la Búsqueda

Con la finalidad de dar respuesta a las preguntas de investigación planteadas para el desarrollo de este trabajo, se realizó la búsqueda bibliográfica en diferentes bases de datos como se describe a continuación: Semantic Scholar, Google Academics, Redalyc, WorldWideScience.org, Springer Link y Core.
En los cuales encontramos los siguientes resultados, además es necesario mencionar que la temática a buscar es aplicación para el diagnóstico de diabetes tipo 2 con aprendizaje máquina.

Tabla 2
RESUMEN DE PAPERS EN DISTINTOS BUSCADORES

AÑO / BUSCADOR	2019	2020	2021	2022	2023	2024	TOTAL
Semantic Scholar	50	70	40	90	110	5	365
Google Academic	7	14	27	19	21	0	88
Redalyc	75	83	92	90	28	1	369
Springer Link	0	0	0	2	4	0	6
Scopus	67	68	80	85	90	29	419

Elaborado por: Ruperto Cisneros.

En la Tabla 2, se presenta la cantidad de trabajos encontrados por base de datos académica. Luego de aplicar los criterios de inclusión y exclusión antes expuestos, se seleccionó 88 papers que se procedió a realizar un análisis más exhaustivo.

RESUMEN DE REFERENCIAS BIBLIOGRAFICAS

Autor	Año	Buscador	DOI/Enlace
J. F. C. Andrade et al.	2019	Google Academic	https://doi.org/10.26820/recimundo/3.1.enero.2019.815-831
R. B. Ruiz y J. D. Velásquez	2023	Redalyc	https://doi.org/10.1016/j.rmclc.2022.12.001
J. M. Vegas Valle	2019	Scopus	https://dialnet.unirioja.es/servlet/tesis?codigo=260844
O. Iparraguirre-Villanueva et al.	2023	Semantic Scholar	https://doi.org/10.3390/diagnostics13142383

D. A. Ordóñez Barrios y E. R. Vizcarra Infantes	2018	Google Academic	https://doi.org/10.19083/tesis/624417
M. Rodríguez-Leyton y M. Charris	2019	Redalyc	https://www.scielo.org/es/
S. Chatterjee y A. S. Hadi	2006	Springer Link	https://www.wiley.com/en-us
R. Genuer y J.-M. Poggi	2020	Scopus	https://link.springer.com/
C. E. C. Figueroa y H. G. Chávez	2021	Semantic Scholar	https://ieeexplore.ieee.org/
L. Viveros-Rosas et al.	2023	Google Academic	https://www.sciencedirect.com/

Elaborado por: Ruperto Cisneros.

1.2 Antecedentes Históricos

1.2.1 Enfermedades Crónicas

El término "enfermedades no transmisibles" se refiere a condiciones médicas persistentes o enfermedades que no son contagiosas. La apoplejía, los ataques al corazón, la diabetes, el cáncer, el asma y la depresión son ejemplos comunes. La aparición de factores de riesgo metabólicos y de enfermedad en algunas de las otorrinolaringologías está precedida por conductas poco saludables. El sobrepeso y la obesidad, la presión arterial elevada, los niveles elevados de glucosa en la sangre y los niveles no óptimos de colesterol en la sangre son factores de riesgo asociados con enfermedades crónicas. La mayoría de estos factores de riesgo se consideran modificables a través de cambios en el comportamiento o medicamentos [8].

1.2.2. Diabetes tipo 2

De acuerdo con los registros más antiguos, lo que actualmente conocemos como diabetes mellitus (DM) ha sido un problema médico durante miles de años. Los primeros registros sobre el conocimiento de esta afección se encuentran en el papiro de Ebers, que se escribió en el noveno año del reinado de Amenofis I, aproximadamente en 1535 a. C.

Areteo de Capadocia (s. II d. C.) fue el primero en mencionar esta idea, ya que describió los síntomas, su evolución y su efecto fatal. Este personaje creía que la diabetes era "la emisión de la carne hacia la orina", notando la pérdida de peso de algunas personas [9].

El griego Claudio Galeno (s. II d. C.) planteó la idea de que la diabetes se debía un agotamiento de los riñones, una idea que se mantuvo durante siglos. El famoso médico suizo "Paracelso" (1493-1541) extrajo "sal" de la orina de pacientes diabéticos. Aunque es cierto que siglos antes se degustaba la orina de los diabéticos como parte de su método de diagnóstico. Willis hizo un gran aporte al entendimiento de la diabetes y al manejo de los pacientes que padecen esta enfermedad, dado que establece una prueba diagnóstica, sugirió dietas específicas que fueran más allá de ser hipocalóricas como tratamiento [9].

Appolinaire Bouchardat (1806-1886) aconsejaba a sus pacientes hacer ejercicio al decirles: "Se ganará el pan con el sudor de su frente", lo que demostraba que el ejercicio mejoraba la glucosuria. Recomendó a sus pacientes que probaran su propia orina todos los días.

Arnoldo Cantani (1837-1893) también hizo hincapié en la importancia de limitar la glucosuria mediante restricciones dietéticas. El paciente podía comer tanto como quisiera mientras no apareciera glucosuria, según Cantani [9].

FM Allen inventó el uso de la subnutrición en el tratamiento del diabético, estableciendo un régimen dietético riguroso con días de ayuno.

En los primeros años del siglo XX, se confirmó claramente el componente inflamatorio de la destrucción de células beta en pacientes jóvenes que fallaron poco después de la presentación inicial de DM1. Quizás en un futuro cercano se logre el objetivo que se ha buscado durante siglos: curar de manera efectiva al paciente diabético [9].

En la Figura 1, se presenta una línea de tiempo respecto a los personajes y aportaciones relevantes de la enfermedad de Diabetes.

Figura 1. Personajes y aportaciones relevantes

1.2.3. Análisis Predictivo

El análisis predictivo es un conjunto de tecnologías de inteligencia artificial que descubren patrones y relaciones en grandes cantidades de datos y los utilizan para predecir comportamiento y eventos.

El análisis predictivo se diferencia de otras tecnologías de inteligencia de negocios porque utiliza eventos pasados para predecir el futuro [8]. Además de métodos para evaluar la calidad de las predicciones en la práctica, Predictive Analytics incluye modelos estadísticos y otros métodos empíricos destinados a hacer predicciones empíricas, en contraste con las predicciones basadas exclusivamente en teoría [10]. Es fundamental comprender las limitaciones de los análisis predictivos. Primero, sin un conjunto de datos que sirva de entrenamiento y de un tamaño y calidad adecuados, no se puede avanzar. En segundo lugar, es esencial tener una definición clara del concepto que se predice, así como ejemplos históricos de lo que se predice [11].

1.2.4. Aprendizaje de Maquina (Machine Learning)

Para explicar cómo el aprendizaje automático es crucial para el desarrollo global, es necesario remontarse a sus orígenes y principalmente comprender su origen debido a que esta herramienta proviene de la IA. En la Figura 2, se presenta una línea de tiempo relacionada con los hitos importantes relacionados con el diagnóstico automático de diabetes.

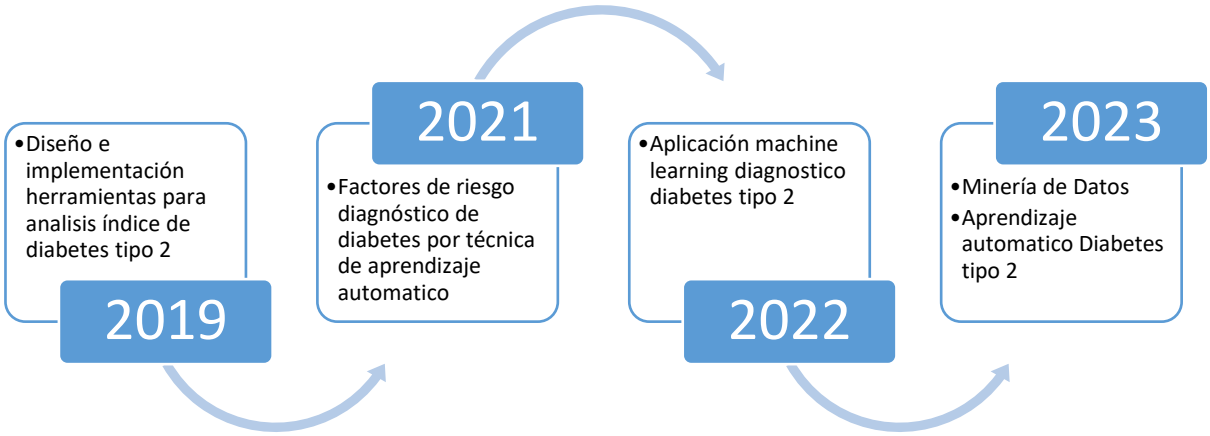

Figura 2. Línea de tiempo de los hitos más importantes relacionados con la Diabetes.

Es necesario remontarse al pasado, más precisamente al año 1943, cuando el matemático Walter Pitts y el neurofisiólogo Warren McCulloch presentaron su trabajo enfocado en lo que actualmente se conoce como inteligencia artificial. En su teoría, proponían analizar el cerebro como un organismo computacional y crear computadoras que funcionen igual o mejor que nuestra red neuronal [12].

Fue así como la humanidad comenzó a comprender y preocuparse por cuán inteligente podría llegar a ser una máquina y cuáles serían algunas de las consecuencias. En 1950, el científico informático, matemático, filósofo y deportista Alan Mathison Turing, logró crear el "Test de Turing" para determinar cuán inteligente era una computadora.

A finales de 1952, Arthur Samuel, un profesor e informático teórico, presentó el primer programa de computación capaz de aprender. Este software tenía la capacidad de jugar con mujeres y almacenaba información y estilos de juego, lo que le permitía mejorar su respuesta según el nivel del juego, lo que lo hacía cada vez mejor a medida que jugaba [13]. La cadena de logros en menos de dos décadas auguraba una trayectoria prometida en este campo, lo que llevó a que Martin Minsky, John McCarthy y otro grupo de profesionales dieran el nombre de "Artificial Inteligencie" en 1956 durante una conferencia científica en Darthmouth.

En 1979, algunos estudiantes de ingeniería de la Universidad de Stanford lograron construir un robot llamado "Coche de Stanford" que podía moverse por una habitación sorteando obstáculos. El instrumento principal de la inteligencia artificial que originó el aprendizaje automático, un algoritmo capaz de reconocer patrones [13].

En los años 80, se produjo una revolución en el procesamiento de datos gracias a la creación de modelos y sistemas expertos que fueron bien recibidos por las empresas.

Por lo tanto, modelos de software como el desarrollado por Gerald Dejong en 1981, llamado "Explanación Base Learning" (EBL), se desarrollaron rápidamente y constituyeron una variante de aprendizaje automático que incluía cuatro variables o pautas a tener en cuenta:

- Un dominio que debía reflejar el objetivo específico de la búsqueda.
- Un área de hipótesis que incluye todas las soluciones potenciales.
- Ejemplos de entrenamiento, que consistían en diferentes datos y soluciones que ya se habían encontrado.
- Los criterios de operatividad permiten identificar el tipo de dominio a tratar.

Este tipo de modelos comenzaron a ser apreciados a nivel industrial porque no solo permitían trabajar sobre las variables ingresadas, sino que también permitían almacenar nuevas para la formulación de sus propias variables. En 1985, el profesor e informático Terry Sejnowski descubrió con su programa "NetTalk" al contribuir a la evolución de esta disciplina, creando un algoritmo capaz de aprender la pronunciación de palabras a nivel escolar.

En 2008, Microsoft lanzó la interpretación beta del programa Azure Machine Learning, una aplicación que ofrece a los usuarios un servicio en la nube que les permite almacenar aplicaciones directamente en el centro de procesamiento de Microsoft. Después de tres años, IBM también revolucionó el aprendizaje automático al introducir su computadora Watson, la cual se destacó en el concurso Jeopardy. Este sistema venció a sus competidores humanos al responder con precisión, aunque tuvo problemas en algunas categorías que no le brindaban las pistas necesarias para que el sistema pudiera responder correctamente.

Al año siguiente, los amigos Jeff Dean de Google y Andrew N, profesor de la Universidad de Stanford, se unieron para liderar un proyecto innovador llamado Google Brain. Este proyecto consistía en una red neuronal que Google utilizaba para detectar patrones en imágenes y vídeos [14].

Al ser utilizado en sistemas de seguridad y militares, este tipo de Red Neuronal Profunda (RNP) se convirtió en una de las mejores maravillas del aprendizaje automático [14].

De esta manera, se comenzó a consolidar la excelente reputación de la inteligencia artificial, lo que llevó a Elon Musk, Sam Altman y otros a establecer Open AI en 2015. Esta organización sin fines de lucro recibió mil millones de dólares para iniciar y promover investigaciones. que permitieran el progreso de la inteligencia artificial en beneficio de la humanidad.

Se puede concentrar en lo que está sucediendo actualmente con el machine learning después de analizar los principales eventos e investigaciones que lo originaron, ya que no solo sigue siendo importante para la comunidad en materia de informática o el sector empresarial, donde su ocupación principal es tratar los grandes volúmenes de información que componen diariamente dichos campos [15].

Algunos campos de aplicación incluyen educación, medicina, finanzas, construcción y robótica, y la incorporación del aprendizaje automático puede ser muy beneficiosa en el futuro.

Según [15], la diabetes es una enfermedad cada vez más común que se asocia con peores pronósticos y mayor comorbilidad en pacientes con enfermedad cardiovascular.

En [14], la diabetes mellitus es una enfermedad crónica en aumento que afecta a 1 de cada 11 adultos y tiene graves complicaciones para los pacientes y los sistemas de salud.

Como se indica en [16], la investigación utilizó una serie de algoritmos basados en la base de datos, incluidos: La diabetes mellitus (DM) es un tipo de trastorno metabólico que produce hiperglucemia crónica, que suele ser causada por una secreción inadecuada de insulina:

- KNN con un 79.64% F1
- Red Neuronal Artificial con un 74.07% F1
- Un algoritmo compuesto por los tres algoritmos 80.32% F1.
- Naive Bayesian con un 79.67% F1

1.3 Antecedentes Conceptuales

1.3.1 Que es Diabetes

La Diabetes Mellitus (DM) abarca un grupo de trastornos metabólicos caracterizados por una hiperglucemia significativa, que surge debido a alteraciones en la secreción o la función de la insulina, o por la combinación de ambos mecanismos.

En el caso específico de la Diabetes Mellitus tipo 2, se define como un síndrome heterogéneo, influenciado tanto por factores genéticos como ambientales. Su principal rasgo distintivo es la elevación crónica de la glucosa en sangre, lo que ocasiona una disminución progresiva en la secreción y acción de la insulina, derivando en complicaciones agudas y crónicas. Esta patología se considera una de las principales causas de impacto negativo en la calidad de vida a nivel mundial, convirtiéndose en un problema significativo de salud pública.

De acuerdo con la Federación Internacional de Diabetes, en un informe del año 2015 se estimó que aproximadamente 415 millones de personas entre los 20 y 79 años padecían diabetes a nivel global, mientras que 193 millones adicionales permanecían sin diagnóstico. Se proyecta que para el año 2040 esta cifra podría alcanzar los 642 millones de adultos.

El mismo informe señala que en Ecuador la prevalencia de diabetes en adultos de 20 a 79 años es del 8,5 %. A nivel histórico, el número de personas diagnosticadas con diabetes ha mostrado un aumento alarmante: a principios del siglo XXI se registraban alrededor de 159 millones de casos, cifra que ascendió a entre 225 y 230 millones para el año 2010. Se espera que en 2025 este número alcanza los 380 millones y que para 2030 el incremento sea especialmente notable en personas de 45 a 64 años.

1.3.1.1. Diagnóstico

Para el diagnóstico de la Diabetes Mellitus tipo 2 (DM2) puede realizarse mediante cualquiera de los siguientes criterios:

1. Glucosa en ayunas: Una medida de glucemia en sangre venosa igual o superior a 126 mg/dl, que debe confirmarse con una prueba adicional en otro momento.
2. Prueba de tolerancia oral a la glucosa (PTOG): Una medida de glucemia en sangre venosa igual o mayor a 200 mg/dl, dos horas después de ingerir una carga de 75 g de glucosa. Glucosa casual: La presencia de síntomas clínicos de diabetes junto con una medición casual de glucosa en sangre venosa igual o mayor a 200 mg/dl. Los síntomas más comunes asociados a la diabetes incluyen un aumento del apetito, poliuria (micción frecuente), polidipsia (sed excesiva) y pérdida de peso inexplicable.
3. Una hemoglobina glicosilada A1c (HbA1c)* mayor o igual a 6,5 %[17].

1.3.1.2. Concepto Diabetes Mellitus tipo 2.

La diabetes mellitus es un trastorno endócrino y metabólico que se caracteriza por una hiperglicemia crónica causada por cambios en el metabolismo de los hidratos de carbono, proteínas y lípidos. Además, también puede ser causado por fallas multiorgánicas como la resistencia a la insulina en el tejido adiposo y muscular, junto con el deterioro progresivo de las células pancreáticas beta, que son las encargadas de segregar la insulina [18].

1.3.1.3. Epidemiología.

En el año 2014, alrededor del 14% de la población mundial tenía diabetes mellitus, de la cual el 90% era diabetes mellitus tipo 2. Según un estudio publicado en diciembre de 2018, se ha observado que en Ecuador la diabetes mellitus tipo 2 ha aumentó significativamente la mortalidad, causando un total de 4895 fallecimientos a causa de esta enfermedad solo en el año 2020 [19].

Según datos publicados en Update, se estima que para el año 2035 habrá 592 millones de personas con diabetes mellitus tipo 2 en todo el mundo, lo que representa un aumento del 55 % y un gasto del 8,2% en los sistemas sanitarios, lo que la convierte en un verdadero problema de salud [20].

Según la Federación Internacional de Diabetes, la prevalencia mundial de diabetes mellitus es del 8,8% en personas de 20 a 79 años, mientras que en América Central y América del Sur es del 8 %. Esto es significativo debido a las características de la población y los factores de riesgo presentes [21]. En Ecuador, la diabetes mellitus ocupó la segunda causa de mortalidad en el año 2010, y su prevalencia fue del 6%, con cifras mayores en aquellas provincias cercanas al Océano Pacífico tales como Guayas, Los Ríos y Manabí [22].

De acuerdo con el INEC, la tasa de diabetes en el país es del 2.7 % en la población general de 10 a 59 años, con un aumento notable al 10.3 % en el tercer decenio de vida, al 12.3 % para los mayores de 60 años. y al 15,2 % para los de 60 a 64 años. En las provincias de la costa y la zona insular, las tasas son más altas en mujeres.

De esta manera, se convierte en la segunda causa de muerte después de las enfermedades isquémicas del corazón. Es importante tener en cuenta que el 14 de noviembre es el "Día Mundial de la Diabetes", en el que se llevan a cabo campañas para promover la salud y prevenir la enfermedad y concientizar a la población sobre la enfermedad.

Los costos humanos y económicos se pueden reducir mediante un diagnóstico precoz, un control efectivo, la adherencia al tratamiento y la prevención. En el país, según el informe más reciente anunciado en 2014, el Instituto Nacional de Estadística y Censos (INEC), la diabetes mellitus se encuentra como la segunda causa de mortalidad en todo el mundo, siendo la primera causa de mortalidad en las mujeres y la tercera en los hombres [23].

La Organización Mundial de la Salud (OMS) calcula que, de los 56 millones de muertes registradas en el año 2012, el 68 % (38 millones) fueron causadas por enfermedades no contagiosas, de las cuales, dos tercios (28 millones) ocurrieron en países con ingresos bajos y medios. Las principales se debían a enfermedades cardiovasculares, diabetes, cáncer y enfermedades pulmonares crónicas, que podrían evitarse si se trabajara de manera multisectorial en promoción y prevención de la salud.

En los últimos 5 a 10 años, se ha observado un alarmante aumento en la prevalencia de la diabetes mellitus tipo 2 en personas cada vez más jóvenes. Anteriormente, la mayoría de los adultos y ancianos de 40 años eran diagnosticados con esta enfermedad. identificarse, así como un problema global complicado por múltiples factores ambientales y genéticos [19].

1.3.2 Machine Learning

Este aprendizaje de máquinas o también llamado aprendizaje automatizado o aprendizaje automático, constituye una de las ramas de la inteligencia artificial que permite implementar técnicas para que las computadoras aprendan. Esto se logra, a través de la búsqueda de algoritmos y heurísticas que trasladan muestras de datos en programas de computadora, sin necesidad que estos sean totalmente escritos.

1.3.2.1. Tipos de Aprendizaje de Machine Learning

Existen tres tipos de Machine Learning: El Machine Learning Supervisado, el Machine Learning no Supervisado y el Aprendizaje por Refuerzo

- En el Machine Learning Supervisado se necesita la intervención humana para que pueda realizarse, porque de alguna manera los algoritmos "aprenden" en base a los datos introducidos, clasificados y etiquetados por una persona. Los tipos de datos que se introducen en el algoritmo, pueden ser por Clasificación, que es cuando un objeto es clasificado dentro de diversas clases. O por Regresión, que es cuando se intenta predecir un valor numérico.
- En el Machine Learning No Supervisado, en cambio, no hay intervención para su ejecución, ya que los algoritmos aprenden en base a patrones o relaciones que se generan en base a datos introducidos como datos de entrada sin etiquetar. Los tipos de algoritmos que se generan en el Machine Learning, son: Por Clustering, donde los datos de salida se clasifican en grupos. Y por Asociación, en donde el conjunto de datos permite descubrir reglas a seguir.
- En el Aprendizaje por Refuerzo aprende a tomar decisiones mediante prueba y error, recibiendo recompensas o castigos

El Machine Learning busca imitar con el uso de algoritmos la inteligencia humana. El Deep Learning, por ejemplo, hace posible simular una red de neuronas, reproduciendo la estructura biológica presente en el ser humano [24].

1.3.2.1.1. Aprendizaje Supervisado para Clasificación

Un algoritmo de aprendizaje automático supervisado (a diferencia de un algoritmo de aprendizaje automático no supervisado) se basa en datos de entrada etiquetados (datos de entrenamiento) para deducir una función que produzca una salida adecuada cuando se le den nuevos datos sin etiquetar. El objetivo de un algoritmo de aprendizaje supervisado es obtener una clasificación utilizando lo aprendido de ejemplos de entrenamiento. En ejemplos de prueba, esta clasificación se puede utilizar para hacer predicciones. En [11], el término "aprendizaje máquina" o "aprendizaje máquina" surgió en el contexto del aprendizaje supervisado y hace referencia a la detección automática de patrones significativos en datos. Una característica de la aplicación del aprendizaje automático es que se puede utilizar en situaciones en las que los patrones han identificar son de

alta complejidad, lo que hace que un ser humano no tenga la habilidad suficiente para aprender de esto de manera específica [8].

Los datos de entrenamiento están compuestos por pares de elementos, generalmente representados como vectores: una parte corresponde a los datos de entrada y la otra a los resultados esperados. La salida de la función puede ser un valor numérico, como ocurre en los problemas de regresión, o una etiqueta que representa una clase (como en los de clasificación).

A diferencia del aprendizaje supervisado que intenta aprender una función que permita hacer predicciones dados algunos datos nuevos sin etiquetar, el aprendizaje no supervisado intenta aprender la estructura básica de los datos para darnos más información sobre los datos.

La idea central es el adquirir la capacidad de predecir a qué categoría pertenece una observación dado un conjunto de sus rasgos, cuantitativos y/o cualitativos[25].

1.3.2.1.2. Algoritmos de Clasificación en Machine Learning – Aprendizaje Supervisado

Existen varios algoritmos de Machine Learning para problemas de aprendizaje supervisado, específicamente para clasificación, como:

- **a)** Decision Tree Clasifier
- **b)** Logistic Regression Classifier
- **c)** Random Forest Classifier
- **d)** K Nearest Neighbours (KNN)
- **e)** Naive Bayes Classifier
- **f)** SVN Classifier
- **g)** XGBoost Classifier

A continuación, se describen brevemente cada uno de los algoritmos de Machine Learning para clasificación:

a) **Decision Tree Classifier.** El algoritmo **Decision Tree** organiza los datos en un árbol de decisiones donde cada nodo representa una pregunta o condición sobre una característica y las ramas indican el resultado de esa condición. Es fácil de interpretar y funciona dividiendo el conjunto de datos en subconjuntos más pequeños de forma recursiva (divide y vencerás). Utiliza medidas como **Gini** o **entropía** para determinar la "pureza" de los nodos y dividir los datos. Es útil para problemas de clasificación binaria o multiclase, pero puede

sobreajustarse si el árbol es muy profundo, por lo que requiere poda o limitación de la profundidad[26].

b) **Logistic Regression Classifier.** La Regresión Logística es un modelo lineal utilizado para la clasificación binaria. Aunque su nombre sugiere regresión, se basa en la función sigmoide (o logística) que convierte los valores de salida en probabilidades entre 0 y 1. Si la probabilidad es superior a un umbral (usualmente 0.5), se asigna a una clase. Es sencillo, eficiente en datasets grandes y funciona bien cuando las clases son linealmente separables. Sin embargo, tiene limitaciones con datos no lineales, aunque se puede extender a problemas multiclase mediante enfoques como One-vs-Rest (OvR)[27].

c) **Random Forest Classifier.** El Random Forest combina múltiples árboles de decisión para mejorar la precisión y reducir el sobreajuste. Utiliza el enfoque de bagging, donde se generan subconjuntos de datos aleatorios y se construyen árboles independientes. Cada árbol realiza una predicción y el modelo final vota por la clase mayoritaria (para clasificación). La aleatoriedad en la selección de características y muestras incrementa su robustez. Es eficiente en datasets grandes y maneja datos faltantes y ruido, aunque puede ser más lento al aumentar el número de árboles[28].

d) **K Nearest Neighbours (KNN).** El algoritmo KNN es un clasificador basado en la proximidad. No construye un modelo explícito, sino que almacena los datos de entrenamiento y clasifica una muestra nueva en función de sus K vecinos más cercanos, usando métricas de distancia como Euclidiana. La clase más común entre los vecinos determina la predicción. Es sencillo y efectivo para datasets pequeños y bien distribuidos, pero puede ser ineficiente en grandes volúmenes de datos. El rendimiento depende del valor de K y de la normalización de las características[29].

e) **Naive Bayes Classifier.** El Naive Bayes es un clasificador probabilístico basado en el Teorema de Bayes con el supuesto "naive" de independencia entre las características. Es especialmente efectivo en textos y aplicaciones como filtrado de spam y clasificación de documentos. Se calculan las probabilidades a priori y la clase con mayor probabilidad a posteriori se asigna como predicción. Es rápido, escalable y eficiente en datasets grandes, aunque su simplicidad puede limitar la precisión si las características son altamente correlacionadas[30].

f) **SVM Classifier (Support Vector Machine).** El SVM es un algoritmo de clasificación que busca encontrar un hiperplano óptimo que separe las clases en un espacio de características, maximizando la distancia (margen) entre los puntos más cercanos de cada clase (vectores de soporte). Funciona muy bien con datos linealmente separables y, mediante el uso de kernels (como RBF o polinomial), puede manejar problemas no lineales. Es robusto y preciso con datos pequeños y medianos, aunque su complejidad computacional puede ser alta en datasets grandes[20].

g) **XGBoost Classifier.** El XGBoost (eXtreme Gradient Boosting) es una técnica avanzada de boosting que combina múltiples árboles de decisión débiles en un modelo robusto. Se centra en ajustar los errores de árboles anteriores a través de la optimización del gradiente. Es una técnica eficiente, lo que lo hace ideal para competiciones de ciencia de datos. Maneja datos faltantes, reduce el sobreajuste mediante regularización y funciona bien en problemas grandes y complejos, tanto de clasificación como de regresión[31].

1.3.2.1.3. Algoritmo Random Forest Classifier

El algoritmo Random Forest es un método de aprendizaje supervisado que construye múltiples árboles de decisión a partir de un conjunto de datos de entrenamiento. Los resultados de estos árboles se combinan para formar un modelo único, más sólido y confiable que los resultados individuales de cada árbol.[28]. La construcción de cada árbol se realiza en dos etapas: 1. Se generan múltiples árboles de decisión utilizando el conjunto de datos, donde cada árbol contiene un subconjunto aleatorio de variables m (predictores) de forma que $m < M$ (donde M = total de predictores).

Cada árbol creado por el algoritmo Random Forest incluye un conjunto de observaciones seleccionadas de manera aleatoria mediante el método bootstrap, una técnica estadística que permite extraer muestras de una población, donde una misma observación puede aparecer en más de una muestra. Las observaciones que no son utilizadas en los árboles se denominan "no estimadas"(también conocidas como “out of the bag”) se utilizan para validar el modelo[32]. Las salidas generadas por todos los árboles se integran en un resultado final Y(conocido como ensamblaje), utilizando una regla específica. Por lo general, se emplea el promedio cuando las salidas de los árboles son numéricas, o el conteo de votos cuando las salidas son categóricas. En la Figura 3 se ilustra el funcionamiento del algoritmo Random Forest.

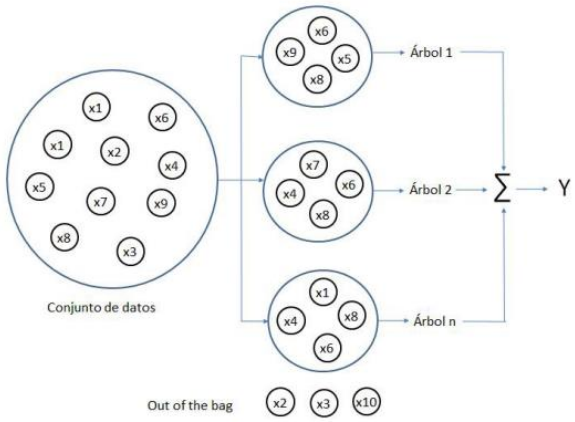

Figura 3: Algoritmo Random Forest
Fuente: Espinosa-Zúñiga Javier Jesús

Las principales características del algoritmo Random Forest [32] son:

- **a.** Se pueden emplear tanto para clasificación como para predicción: en el caso de clasificación, cada árbol emite un "voto" por una clase, y el modelo asigna la clase con mayor cantidad de votos. Esto implica que cada nueva observación se evalúa en todos los árboles, y se le asigna la clase más votada. En el caso de predicción, el resultado final del modelo se obtiene promediando las salidas de todos los árboles.
- **b.** El modelo es relativamente sencillo de entrenar en comparación con técnicas más complejas, pero ofrece un rendimiento similar.
- **c.** Presenta un desempeño altamente eficiente y se posiciona como una de las técnicas más precisas para manejar bases de datos grandes.
- **d.** Puede procesar cientos de predictores sin excluir ninguno, además de identificar cuáles son los más relevantes, lo que lo hace útil para la reducción de dimensionalidad.
- **e.** Conserva una alta precisión incluso cuando se enfrenta a grandes proporciones de datos faltantes.

1.3.2.2. Aprendizaje No Supervisado

Los algoritmos de Aprendizaje no Supervisado manejan datos sin entrenamiento previo, es una función que hace su trabajo con los datos a su disposición. En cierto modo, se deja a su suerte para que resuelva las cosas a su antojo.

Los algoritmos no supervisados funcionan con datos no etiquetados. Su propósito es la exploración. Si el Aprendizaje Supervisado funciona bajo reglas claramente definidas, el Aprendizaje no

Supervisado funciona bajo condiciones en las que los resultados son desconocidos y, por lo tanto, es necesario definirlos en el proceso[6].

Los algoritmos de Aprendizaje no Supervisado están acostumbrados:

- Explorar la estructura de la información y detectar patrones distintos,
- extraer ideas valiosas,
- aplicarla en su funcionamiento con el fin de aumentar la eficacia del proceso de toma de decisiones.

En otras palabras, describe la información, pasa por el grueso de la misma e identifica lo que realmente es.

1.3.3 Deep Learning

Deep Learning es un subcampo deMachine Learning que utiliza redes neuronales para enseñar a las computadoras a hacer lo que resulta natural para las personas: aprender a partir de ejemplos. Con Deep Learning, se aprende a realizar tareas de clasificación o regresión basados en datos de imágenes, texto o sonido. Los modelos de Deep Learning pueden alcanzar una precisión excepcional que, con frecuencia, supera el propio rendimiento humano[33].

1.3.3.1. Cómo funciona Deep Learning

Los modelos de Deep Learning están inspirados en redes neuronales, como el cerebro humano, (Figura 4) está formada de neuronas o nódulos conectados en una estructura en capas que relacionan las entradas con las salidas deseadas. Las neuronas situadas entre las capas de entrada y salida de una red neuronal se denominan capas ocultas. El término “Deep” (profundo) normalmente alude a la cantidad de capas ocultas de la red neuronal. Los modelos de Deep Learning pueden tener cientos o incluso miles de capas ocultas[33].

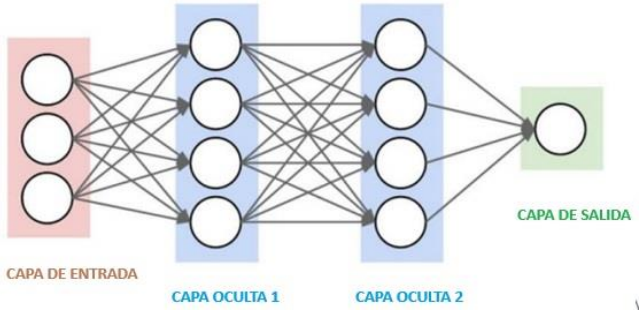

Figura 4: Estructura de la red Neuronal
Fuente: Bustos Gaibor, Samuel & Ramírez Castro, José.

Los modelos de Deep Learning se entrenan empleando grandes conjuntos de datos etiquetados. Suelen aprender directamente a partir de los datos, sin necesidad de extraer características manualmente. Si bien la teoría de una red neuronal artificial se elaboró por primera vez en 1958, Deep Learning requiere una potencia de cálculo significativa que no fue posible hasta la década de los 2000. En la actualidad, los recursos informáticos permiten crear y entrenar redes con cientos de conexiones y neuronas[33].

Las GPU de alto rendimiento cuentan con una arquitectura paralela que resulta eficiente para Deep Learning. En combinación con clusters o cálculo en la nube, esto permite reducir el tiempo de entrenamiento de una red de Deep Learning de semanas a horas, o incluso minutos.

1.3.3.2. Tipos de modelos de Deep Learning

Las redes neuronales convolucionales (CNN), las redes neuronales recurrentes (RNN) y los modelos de transformadores son tres tipos de modelos de Deep Learning.

CNN: Una red CNN convoluciona características aprendidas con datos de entrada y emplea capas convolucionales en 2D. Esta arquitectura resulta adecuada para procesar datos en 2D, como imágenes; CNN extrae características directamente de las imágenes. Las características relevantes se aprenden mientras la red se entrena con un conjunto de imágenes. Este proceso automatizado de extracción de características contribuye significativamente a la precisión de los modelos de Deep Learning en tareas de clasificación de imágenes. Además, las CNN se pueden utilizar para clasificar otros tipos de datos, como texto y series temporales[34].

RNN: Una red neuronal recurrente (RNN) es una arquitectura de red de Deep Learning que realiza predicciones sobre series temporales o datos secuenciales. Son especialmente eficaces para trabajar con datos secuenciales de longitud variable y resolver problemas de clasificación de señales naturales, procesamiento del lenguaje y análisis de vídeos. La red de memoria a corto-largo plazo (LSTM) es un tipo especial de RNN que es más eficiente en aprender dependencias a largo plazo que una RNN simple[34].

Transformadores: Los transformadores están creados para monitorear las conexiones entre los elementos de una secuencia. Utilizan un mecanismo de autoatención para identificar dependencias globales entre la entrada y la salida. Son comúnmente utilizados en el procesamiento del lenguaje natural y son la base de los modelos de lenguaje de gran escala (LLM) como BERT y ChatGPT[35].

1.3.3.3. Metodología para desarrollo de proyecto de machine learning

De acuerdo con [10], la diabetes se encuentra entre las principales causas de mortalidad en Ecuador. La Organización Mundial de la Salud reconoce tres tipos principales: tipo I, tipo 2 y gestacional. Un problema importante es que se diagnostica con frecuencia cuando la enfermedad ya está avanzada, lo que dificulta el tratamiento.

Es fundamental realizar un diagnóstico a tiempo para evitar complicaciones graves (amputaciones, ataques cardíacos, daño ocular, etc.), gastos elevados (hospitales, personales y del Estado) y tiempo perdido. El aprendizaje automático, también conocido como ML, es una técnica utilizada para predecir el riesgo de desarrollar diabetes. Esto ayuda a prevenir enfermedades fatales y gastos innecesarios. Varios estudios tienen como objetivo predecir el diagnóstico de diabetes con ML.

Por ser una enfermedad crónica muy común y con graves complicaciones, la diabetes mellitus (DM) es señalada como una de las enfermedades más importantes en la salud pública a nivel global [36]. El estudio mencionado propone un método de análisis de componentes principales para encontrar los factores de riesgo más comunes en pacientes con diabetes tipo 2. Después de eso, se agregan estos factores de riesgo al algoritmo de clasificación J48, lo que mejora los resultados y obtiene una precisión del 86,9%.

Un sistema de diagnóstico electrónico para la diabetes tipo 2 que se base en el aprendizaje automático o el aprendizaje automático (ML) se propone para implementar en Internet de las cosas médicas (IoMT)[37]. Sin embargo, los modelos de ML son críticos por su incapacidad para interpretar el proceso interno de decisión, lo que dificulta su adopción. Este estudio examina el uso de tres modelos supervisados interpretables: Naive Bayes, Random Forest y Decision Trees J48. Los modelos fueron entrenados y probados con un conjunto de datos públicos de diabetes, y se analiza la precisión, sensibilidad y especificidad de cada uno para seleccionar el mejor. Con una selección adecuada de características, Naive Bayes funciona bien para la clasificación binaria.

Este estudio de revisión [12], examina los avances recientes en la detección automática de la retinopatía diabética utilizando imágenes de fondo de ojo. Se enfoca en modelos híbridos y redes neuronales convolucionales como técnicas de aprendizaje automático. Clasifica las diversas propuestas según su arquitectura y objetivos, ofreciendo un panorama completo de los avances en este campo desde 2015. Esto puede mejorar la comprensión y fomentar la investigación sobre el diagnóstico automatizado de esta complicación de la diabetes.

Se prevé un mayor desarrollo de aplicaciones móviles basadas en aprendizaje profundo para la predicción de enfermedades como la diabetes debido a la creciente demanda de servicios de inteligencia artificial en dispositivos móviles, como expresa [38]. Se han realizado varios estudios para predecir la diabetes mellitus utilizando algoritmos de aprendizaje automático profundo y algoritmos de aprendizaje automático, pero se han centrado principalmente en el modelo predictivo.

Como se indica [22], la diabetes no controlada puede causar fallas múltiples. El aprendizaje automático y la inteligencia artificial han permitido la detección temprana y el diagnóstico automatizado, que son más ventajosos que el diagnóstico manual. Muchas publicaciones recientes hablan sobre la detección, el diagnóstico y la autogestión de esta enfermedad a través de técnicas de aprendizaje automático e inteligencia artificial.

Millones de personas en todo el mundo sufren de diabetes tipo 2, una enfermedad crónica, y el diagnóstico temprano y el tratamiento adecuado son cruciales para controlar la enfermedad y prevenir complicaciones graves. Actualmente, la diabetes tipo 2 se diagnostica mediante una combinación de pruebas clínicas y de laboratorio [39].

Estudios previos han alcanzado un 90% [22] de precisión en el diagnostico de la diabetes tipo 2, presentan limitaciones como la falta de accesibilidad, al no estar implementados en aplicaciones web de fácil uso para médicos y pacientes; el uso de datasets genéricos, que no pueden representar adecuadamente a la población ecuatoriana; y la falta de claridad dificulta su interpretación por parte de especialistas. La propuesta de este trabajo se diferencia al utilizar datos recolectados de pacientes locales en la Clínica de Salud y Bienestar Postural en la ciudad de Quito lo que mejora el modelo, además de integrar una interfaz web accesible desarrollada con Angular y Flask, que permite a los médicos ingresar datos y obtener diagnóstico de manera intuitiva.

En un estudio [12], se explica cómo las exposiciones ambientales contribuyen a la causa de la diabetes. Sin embargo, debido a la diversidad de la enfermedad, la complejidad de las exposiciones y los desafíos analíticos, esta comprensión es insuficiente. La minería de datos y el aprendizaje automático son dos ejemplos de técnicas de inteligencia artificial que pueden abordar estas limitaciones. Los tipos de métodos y exposiciones analizadas no se han revisado a fondo, a pesar de que se utilizan cada vez más en la investigación de la etiología y la predicción de la diabetes.

Para familiarizarse con el tema del proyecto, se presentarán las definiciones de todos los términos y conceptos clave del ámbito de aprendizaje máquina. Los conceptos que se desarrollarán serán básicos, complementarios y específicos del contexto.

1.3.3.3.1. Metodología CRISP-DM

CRISP-DM (Cross-Industry Standard Process for Data Mining) es una metodología estructurada para proyectos de minería de datos, ciencia de datos y machine learning. CRISP-DM, se puede ver como la metodología por excelencia para proyectos enfocados en extraer valor de los datos. A lo largo de estos veinte años, la metodología CRISP-DM ha servido de inspiración para otros estándares como SEMMA de SAS o ASUM-DM de IBM, además de generar diversas variantes que amplían o adaptan CRISP-DM a sectores industriales o tipos de proyectos específicos.[40].
La metodología CRISP-DM se conceptualiza en 6 fases, tal como se muestra en la Figura 5:

1. **Comprensión del negocio**: Identifica los objetivos empresariales, evalúa la situación actual y desarrolla un plan alineado con las necesidades del negocio.
2. **Comprensión de los datos**: Explora, recopila y analiza datos para identificar problemas como valores faltantes o inconsistencias.
3. **Preparación de los datos**: Selecciona, limpia y transforma los datos para generar un dataset listo para modelado, incorporando técnicas como feature engineering.
4. **Modelado**: Construye modelos predictivos o descriptivos mediante algoritmos estadísticos o de machine learning, ajustando parámetros para optimizar resultados.
5. **Evaluación**: Analiza el desempeño del modelo con métricas específicas y verifica si cumple con los objetivos empresariales.
6. **Despliegue**: Implementa el modelo en producción, crea reportes o dashboards, y monitorea su desempeño.

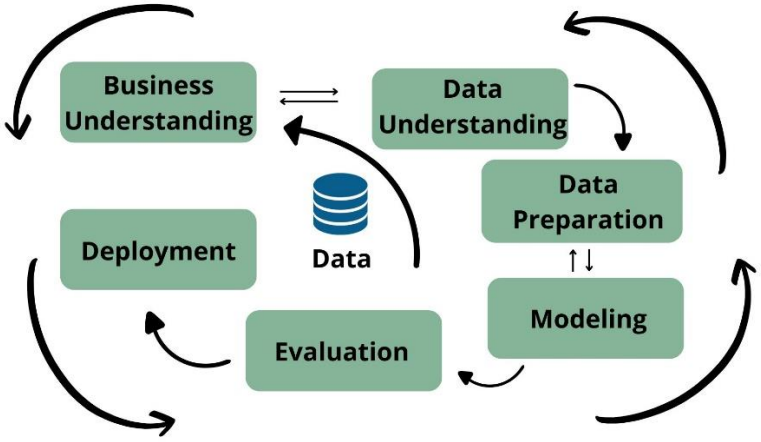

Figura 5: Metodología CRISP-DM
Fuente: López, Miguel Ángel 2019

1.4 Antecedentes Contextuales

La aplicación de esta investigación se llevará a cabo en la Clínica de Salud y Bienestar Postural, está dentro del Sistema del Ministerio de Salud Pública de Ecuador y ofrece atención especializada a pacientes de diversas especialidades y subespecialidades médicas, tanto preventivas, ambulatorias, de recuperación y rehabilitación. Además, brinda atención a pacientes con enfermedades agudas y crónicas.

Cuenta con un equipo médico profesional y experimentado, así como administrativo, colaboradores y técnicos con experiencia, lo que permite satisfacer de cierta manera las necesidades de la población.

Para llevar a cabo esta investigación, se elegirá la Clínica de Salud y Bienestar Postural se encuentra en la ciudad de Quito provincia de Pichincha en el sector de Carcelén.

El personal médico que va a colaborar es: Dra. Mónica Jaqueline Herrera Zúñiga, con código 14153.

Durante la consulta médica, da atención a pacientes que acuden por control general, es en estos caos donde se envía exámenes para evaluar si existe alteración a nivel de su glucosa, los pacientes retornan con resultados donde realiza la evaluación y diagnóstico de pacientes con diabetes tipo 2.

Los exámenes necesarios para el estudio son:

Glucosa, colesterol, triglicéridos, índice de masa corporal, antecedentes patológicos personales y antecedentes patológicos familiares.

Se espera ingresar en la base de datos que ayudará a desarrollar la aplicación web para respaldar el diagnóstico de diabetes mellitus tipo 2 y la ratificación del diagnóstico de DM mediante el apoyo

del diagnóstico del personal médico correspondiente y el respaldo de los resultados de los exámenes de laboratorio.

1.4.1 Establecimiento de requerimientos

El prototipo de este trabajo de investigación va a consistir en el manejo de algoritmo de Machine Learning de clasificación enfocado en el diagnóstico de diabetes mellitus tipo 2, donde se ha recogido los datos de edad comprendidos entre 30 y 60 años de edad, en el periodo de julio a diciembre 2023.

1.4.2 Desarrollos de aplicaciones web que consumen APIs de machine learning

Para comprender el proceso de desarrollo de aplicaciones para diagnóstico médico de diabetes, se revisó algunos trabajos relacionados como los que se describen a continuación:

1.4.2.1. Desarrollo de un Modelo Predictivo para la Diabetes Mellitus de Tipo 2 usando Data Clínica y Genética

Según [20], estudios genéticos recientes proporcionan pruebas convincentes de que la ubicación de un gen y el polimorfismo de un grupo de moléculas están directamente relacionados con la probabilidad de desarrollar diabetes mellitus de tipo 2. Se propone crear un modelo predictivo de diabetes mellitus tipo 2 utilizando datos genéticos y clínicos bajo esta premisa. Se empleó un método estadístico de muestreo de 961 participantes para desarrollar el modelo; 673 fueron sujetos aleatorios y 288 tenían diabetes mellitus de tipo 2. Los pacientes aleatorios tenían una edad promedio de 64.1 años y los pacientes diagnosticados con diabetes tenían una edad promedio de 64.5 años.

Los 684 participantes recibieron una muestra de 499 grupos de moléculas relacionadas con la diabetes mellitus de tipo 2. Para mejorar el resultado, se tomó en cuenta otros factores que pueden obstaculizar el desarrollo de la enfermedad, como el índice de masa corporal y el conteo de triglicéridos. Para todos los algoritmos probados, se establece una relación de clasificación incorrecta de diabetes mellitus tipo 2 de 27,98 ±2,76. Se utiliza un modelo de ajuste multiescenario para mejorar los resultados de la clasificación, seleccionando factores de riesgo para realizar un análisis de reconocimiento de patrón que permita agrupaciones más reducidas según características comunes. Se crearon tres grupos basados en esto: el primero incluía factores relacionados con la obesidad (índice de masa corporal, peso y relación entre cintura y cadera), el segundo tenía valores más altos de glucosa, resistencia a la insulina, colesterol y hemoglobina, y el tercero tenía valores

más altos de índice de masa corporal, peso y relación entre cintura y cadera. A través del uso de modelos estadísticos de decisión de árbol, se descubrió que la prevalencia de la diabetes mellitus tipo 2 en hombres y mujeres en los grupos 2 y 3 era similar, pero menor, en el grupo 1. A continuación, se muestra un cuadro de dispersión entre la relación de triglicéridos, índice de masa corporal e incidencia de la diabetes mellitus tipo 2:

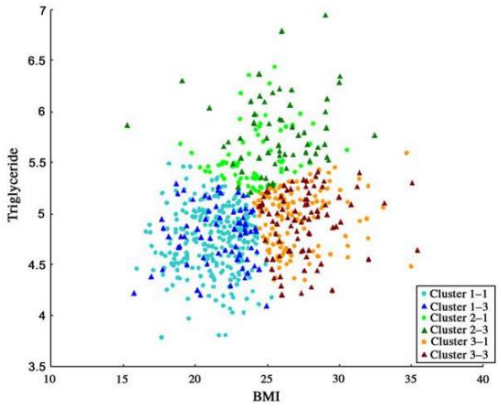

Figura 6: Dispersión entre triglicéridos, índice de masa corporal e incidencia de diabetes mellitus tipo 2
Fuente: Akmal et al. (2011)

En los clusters 1-3, 2-3 y 3-3 se distingue a los pacientes con la enfermedad y observamos una tendencia hacia un alto índice de masa corporal y triglicéridos en estos grupos (Figura 6).
Se descubrirán factores de riesgo para desarrollar la enfermedad como resultado del análisis estadístico y de clasificación de factores:

- Información clínica: talla, índice de masa corporal, circunferencia de la cintura, presión sanguínea, colesterol, conteo de triglicéridos y nivel de glucosa.
- Historia familiar y personal: hipertensión, diabetes mellitus de tipo 2, enfermedades cerebrovasculares y otras enfermedades vasculares
- Polimorfismo de grupos de moléculas (SNP): se identificaron 499 SNPs en 87 genes candidatos asociados a la diabetes mellitus tipo 2.

Para crear un modelo predictivo ideal, se utilizó inicialmente el método estadístico de chi cuadrado. Sin embargo, debido a la distribución desigual de los tipos de genotipificación (dentro de los 499 SNPs, hubo un 70-75% de genotipos dominantes, un 5-15% receptores y el resto eran heterocigóticos), se obtuvieron ratios de error muy altos. (entre 28.24 y 52.69%) debido a la

distribución desigual de los tipos de genotipificación. Los modelos creados utilizando árboles de decisión y métodos de aprendizaje automático no mostraron grandes variaciones en función de los mismos factores. Dado que el análisis concluye que las tasas de error fueron más altas de lo esperado, se recomienda utilizar muestras mayores y métodos de análisis mejores para integrar la genotipificación para cuantificar objetivamente los SNPs de alto riesgo.

1.4.2.2. Predicción de Diabetes con Algoritmos de Aprendizaje Supervisado de Redes Neuronales Artificiales

Los autores [3], sugieren usar redes neuronales multicapas en lugar de técnicas convencionales para hacer predicciones de diabetes exitosas. Esto se debe a que hay algoritmos de redes neuronales que han sido probados para diagnosticar tuberculosis, neumonía, cáncer al pulmón y varones cardíacos.

Se tomaron muestras de 250 pacientes con diabetes, con 27 variables numéricas incluyendo la presión sanguínea, la acidez de la orina, la creatinina y otras.

Los autores describen el aprendizaje supervisado como un método para determinar la relación entre la entrada y la salida mediante el uso de un conjunto de datos, aprendiendo la salida a partir de las señales de error emitidas durante el diagnóstico y avanzando hasta encontrar el valor mínimo de error permitido. Los siguientes algoritmos, que están disponibles en las herramientas de redes neuronales de MATLAB, se utilizaron:

- Método del gradiente conjugado de Fletcher-Powell (a)
- Método del gradiente conjugado de Polak-Ribiére (b)
- Método del gradiente conjugado a escala. (c)

Se utilizó un análisis de regresión para determinar la relación entre los resultados obtenidos y los deseados; se encontró un coeficiente de estimación R, siendo 1 el resultado ideal.

El siguiente bosquejo se deriva del examen de regresión realizado. En la Figura 7, encontramos el coeficiente de especificación R correspondiente a cada algoritmo, representado por la línea punteada:

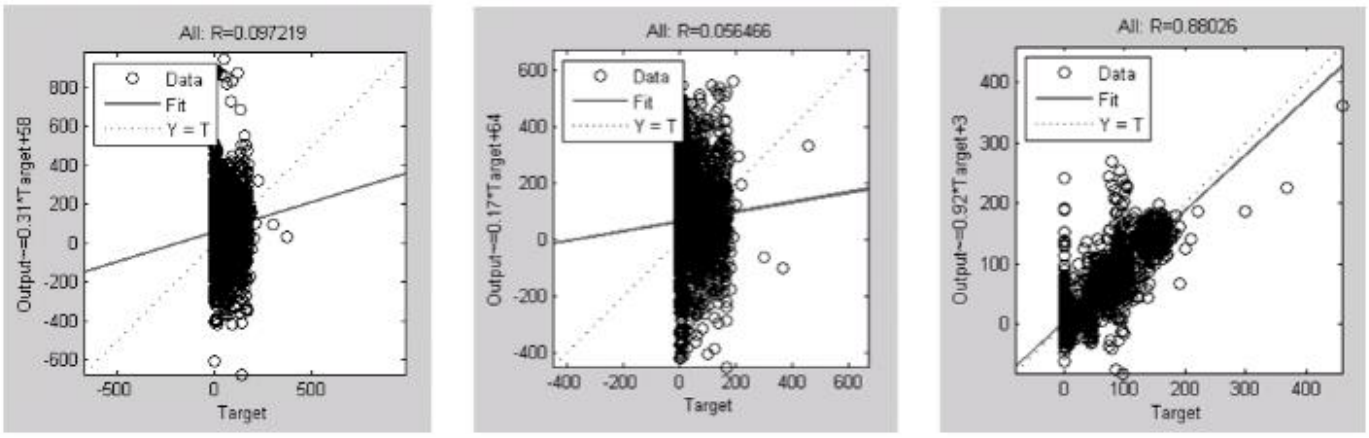

Figura 7: Dispersión de resultados de los algoritmos frente al coeficiente de correlación R

Fuente: Akmal et al. (2011)

El algoritmo de gradiente conjugado una escala se encuentra casi sobre la línea de referencia, lo que indica un resultado cercano entre lo esperado y lo predicho. El objetivo del estudio es descubrir el algoritmo que funciona mejor con los datos recopilados. Las conclusiones del coeficiente R son las siguientes (Tabla 3):

Tabla 3
COEFICIENTE DE CORRELACIÓN POR ALGORITMO

Algorithm	Epochs	Correlation Coefficient R
Fletcher-Powell Conjugate Gradient	**39**	**0.097219**
Polak-Ribiére Conjugate Gradient	**52**	**0.056466**
Scaled Conjugate Gradient	**108**	**0.88026**

Fuente: Akmal et al. (2011)

Identificando que, en cuanto a la predicción de diabetes, el algoritmo de gradiente conjugado una escala es lejanamente el de mejor desempeño en comparación con los algoritmos de Polak-Ribiére y Fletcher-Powell.

CAPÍTULO II. METODOLOGÍA DE INVESTIGACIÓN

En este capítulo se describe el marco metodológico utilizado en el presente trabajo investigativo, tiene como objetivo desarrollar una aplicación para detectar el diagnóstico de Diabetes Mellitus tipo 2 empleando aprendizaje automático.

2.1 Paradigma o enfoque y alcance

El paradigma empleado es cuantitativo porque se va a trabajar con datos numéricos ya que, para el desarrollo de esta aplicación, se consideró los exámenes de laboratorio, enviado por el personal médico que labora en esta institución médica.

El paradigma cuantitativo posee una concepción global positivista, hipotético-deductiva, particularista, objetiva, orientada a los resultados y propia de las ciencias naturales. En contraste, al paradigma cualitativo que postula una concepción global fenomenológica, inductiva, estructuralista, subjetiva, orientada al proceso y propia de la antropología social [16].

2.2 Tipo de investigación y alcance

El estudio se enmarca en un diseño de investigación cuasi-experimental, porque se ha trabajado con un conjunto de datos reales de pacientes, que han sido o no diagnosticados con diabetes desde edades comprendidas entre 30 y 60 años en el periodo julio – diciembre 2023.

Para llevar a cabo este estudio, se recopiló información de los exámenes de laboratorio clínico, los mismos resultados tomados de la Clínica de Salud y Bienestar Postural, de los cuales 961 pacientes fueron diagnosticados con diabetes y 42 no presentan un diagnóstico de diabetes.

El alcance de esta investigación es Correlacional porque se analiza las relaciones existentes entre las variables que influyen en la predicción de diabetes, tales como:

1. **Género:** hace referencia al sexo masculino y femenina.
2. **Edad:** años que tiene el paciente.
3. **IMC (Índice de masa corporal):** Es un número que se calcula con base en el peso y la estatura de la persona.
4. **Glucosa:** Mide la cantidad de un azúcar en una muestra de sangre.

5. **HbA1c (hemoglobina glicocilada):** Es un examen de sangre para la detección de diabetes tipo 2 y prediabetes.
6. **Colesterol:** Determinar el riesgo de acumulación de depósitos de grasa en las arterias
7. **Triglicéridos:** Mide la cantidad de una grasa que hay en la sangre llamada triglicérido.
8. **HDL (grasa buena):** Este parámetro llamado colesterol de lipoproteínas de alta densidad es conocido como el colesterol "bueno" en el torrente sanguíneo.
9. **Urea:** mide la cantidad de nitrógeno en la sangre que proviene de un producto de desecho, se produce en el hígado y se excreta del cuerpo en la orina.
10. **LDL:** también conocido como el colesterol "malo", es la principal causa de obstrucciones en las arterias
11. **VLDL:** Este parámetro mide el colesterol malo de lipoproteínas de muy baja densidad, se produce en el hígado y se libera en el torrente sanguíneo para suministrar a los tejidos del cuerpo un cierto tipo de grasa.
12. **Creatinina:** Es una forma de medir el funcionamiento de los riñones al momento de filtrar los desechos de la sangre.

Estas variables se han dividido en 2 grupos:

- Variables independientes o predictoras: Género, Edad, IMC, Glucosa, Colesterol, Triglicéridos, HDL, Urea, LDL, VLDL, Creatinina, Hemoglobina Glicocilada.
- Variable dependiente o target: Diabetes (1 tiene diabetes, 0 no tiene diabetes)

2.3 Población y muestra

En este caso se trató con dos tipos de población:

a) Población para la recolección de datos que servirá para entrenar los modelos de clasificación de machine learning.

   La población a investigar está constituida por aproximadamente 961 pacientes atendidos en el lapso de julio a diciembre del año 2023.

   Es así que, al tratarse del desarrollo de una aplicación, empleando aprendizaje máquina se consideró a toda la población, de manera que se pueda dar un diagnóstico asertivo.

b) Población para evaluar el sistema (grupo de 16 médicos que atienden en la clínica) no hay un proceso de selección de muestra, se aplica una encuesta donde se recaba la información necesaria para este estudio.

2.4 METODOS TEORICOS

Los métodos empleados en este trabajo investigativo son:

Histórico lógico: Este método permite realizar una recopilación de información sobre la evolución de la diabetes, análisis históricos, casos de estudio donde se encontró patrones que se asemeje o sirva para predecir hechos actuales producto de esta investigación[41].

Análisis de datos: Este método sirve para examinar, organizar, interpretar y extraer información significativa a partir de los datos recolectados. Su principal objetivo es identificar patrones, tendencias y relaciones entre las variables del estudio, lo que permite validar hipótesis, responder preguntas de investigación y alcanzar los objetivos propuestos[41].

Síntesis: Este método es útil para integrar y combinar información proveniente de diferentes fuentes o partes del estudio con el fin de obtener una visión más clara y completa del tema o problema en cuestión. Este proceso implica la organización y consolidación de los resultados o datos relevantes, permitiendo establecer conexiones, identificar patrones comunes y generar nuevas ideas o conclusiones a partir de la información recopilada. En este trabajo, el método de síntesis se utilizó para, partiendo de los resultados de análisis, realizar la interpretación de los resultados más relevantes [17].

CRISP-DM (*Cross-Industry Standard Process for Data Mining*) es una metodología estructurada para el desarrollo de proyectos de minería de datos, ciencia de datos y machine learning. Se compone de seis fases principales, que permiten abordar un problema de manera iterativa, flexible y orientada a resultados. CRISP-DM se caracteriza por ser iterativa, ya que permite regresar a fases previas cuando es necesario, y flexible, adaptándose a las particularidades de cada proyecto. Su estructura clara facilita una ejecución eficiente, aplicable en diferentes industrias y tipos de análisis de datos [18]. A continuación, se describen las fases de esta metodología:

- La primera fase es la Comprensión del negocio, donde se identifican los objetivos desde la perspectiva empresarial, se evalúa la situación actual, se formulan los objetivos del proyecto y se desarrolla un plan de trabajo inicial. Aquí, la clave es entender las necesidades del negocio para alinearlas con el proceso de análisis de datos.
- La segunda fase es la Comprensión de los datos, cuyo propósito es explorar la información disponible. Se recopilan los datos iniciales, se realizan análisis descriptivos y se identifican problemas como valores faltantes, outliers o inconsistencias. Esta etapa permite evaluar si los datos son adecuados para cumplir los objetivos definidos.
- La siguiente fase es la Preparación de los datos, donde se seleccionan, limpian y transforman los datos para crear un dataset final listo para el modelado. Esto implica resolver problemas de calidad, integrar diversas fuentes de datos y generar nuevas variables (feature engineering) que mejoren la capacidad predictiva de los modelos.
- En la cuarta fase, Modelado, se aplican técnicas estadísticas o de machine learning para construir modelos predictivos o descriptivos. Se seleccionan los algoritmos más adecuados, se ajustan sus parámetros y se evalúan sus resultados iniciales, buscando siempre el equilibrio entre desempeño y complejidad del modelo.
- La Evaluación es la quinta fase, donde se analizan los resultados obtenidos por el modelo. Aquí se utilizan métricas específicas como precisión, recall o AUC-ROC para determinar su desempeño. Además, se verifica si los resultados cumplen con los objetivos del negocio y se decide si el modelo es suficientemente robusto o requiere ajustes adicionales.
- Finalmente, la fase de Despliegue se enfoca en llevar los resultados al entorno productivo. Esto puede incluir la implementación del modelo en un sistema automatizado, la creación de reportes o dashboards y el monitoreo continuo del desempeño. La documentación y entrega del proceso completan esta última etapa.

La programación extrema (XP), La programación extrema (XP) es una metodología ágil de gestión de proyectos que se centra en la velocidad y la simplicidad con ciclos de desarrollo cortos. Si bien tiene una estructura rígida, el resultado de estos sprints altamente centrados y las integraciones continuas buscan dar como resultado un producto de mayor calidad [19].

La metodología XP (Figura 8) es un conjunto de técnicas que dan agilidad y flexibilidad en la gestión de proyectos. También es conocida como Programación Extrema (Extreme Programming)

y se centra crear un producto según los requisitos exactos del cliente. De ahí, que le involucre al máximo durante el método de gestión del desarrollo del producto[42].

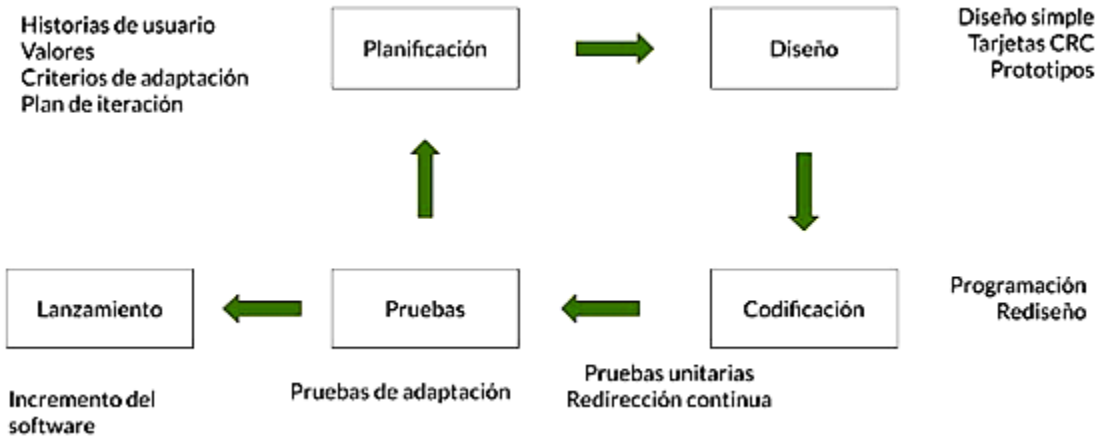

Figura 8: Fases Metodología XP o Programación Extrema
Fuente: Calvo Diego. (2018)

3.5 Métodos empíricos y materiales utilizados

2.5.1. Encuesta

Se empleó una encuesta dirigida a los médicos especialistas en diagnóstico de diabetes. Esta encuesta se elaboró basando en el método de CSAT (Customer Satisfaction Score) que es utilizado para medir la satisfacción de los clientes con respecto a un producto, servicio o experiencia específica. Generalmente, se obtiene a través de encuestas breves que preguntan a los clientes qué tan satisfechos están con una interacción reciente, utilizando una escala de puntuación (por ejemplo, de 1 a 5). La pregunta típica puede ser: "¿Qué tan satisfecho está con nuestro producto/servicio?". El CSAT se calcula tomando el promedio de las respuestas o el porcentaje de clientes que dieron calificaciones altas (por ejemplo, respuestas de 4 o 5 en una escala de 1 a 5). Este indicador es valioso porque proporciona una visión directa de la percepción del cliente sobre la calidad y el rendimiento, permitiendo a las empresas identificar áreas de mejora y medir el impacto de cambios o mejoras en sus productos o servicios.

2.6. Herramientas

En este estudio de investigación como ya se ha mencionado, se pretende detectar el diagnóstico de diabetes tipo 2, para lo cual, esta aplicación se divide en tres partes:

- Google Colaboratory y Python para el desarrollo de modelos de Machine Learning, junto con las librerías o paquetes:
    - **Scikit-learn**: Una de las librerías más populares para machine learning. Proporciona una amplia variedad de algoritmos de clasificación, regresión, clustering y reducción de dimensiones. Además, incluye herramientas para preprocesamiento de datos, selección de características y evaluación de modelos.
    - **TensorFlow (Keras)**: Aunque es principalmente para deep learning, Keras, que ahora es parte de TensorFlow, también ofrece herramientas para modelos más simples de machine learning.
    - **Matplotlib**: Crear gráficos básicos y personalizarlos
    - **Pandas**: Manipulación y análisis de datos estructurados (como archivos CSV).
    - **Numpy:** Manejo eficiente de matrices y arreglos multidimensionales.
    - **Imbalanced-learn (imblearn):** Resolver problemas de desbalanceo en los datos.
    - **Seaborn**: Visualización estadística avanzada y estética.
    - **sklearn.model_selection:** Divide los datos en entrenamiento y prueba.
    - **sklearn.preprocessing:** Transformar datos para mejorar la eficiencia de los modelos.
    - **sklearn.neighbors:** Implementar modelos basados en vecinos cercanos.
- Desarrollo de la aplicación web:
    - Angular para el frontend
    - Lenguaje Python para el backend con el IDE Visual Code

CAPÍTULO III. DESARROLLO DE LA PROPUESTA Y RESULTADOS

El presente capítulo se detalla el proceso para realizar la aplicación web y la detección de diabetes tipo 2 aplicando algoritmos de machine learning. Se utilizó la metodología CRISP-DM para guiar el proceso de desarrollo de los algoritmos de Machine Learning para clasificación (Diagnóstico positivo y diagnóstico negativo de diabetes) y, la metodología ágil XP para gestionar el ciclo de vida del desarrollo de la aplicación web.

3.1 Metodología CRISP- DM

La metodología CRISP-DM utilizada para estructurar y desarrollar los modelos predictivos de machine learning, en la Tabla 4 indica las fases que se utiliza.

Tabla 4
ETAPAS DE LA METODOLOGÍA CRISP-DM

Fase	**Descripción Aplicada al Proyecto**	**Elementos en el Documento**
1. Comprensión del Negocio	Identificación del problema: mejorar el diagnóstico temprano de diabetes tipo 2 mediante una aplicación de aprendizaje automático.	Objetivo general y específico; descripción del problema en el contexto de la Clínica de Salud y Bienestar Postural.
2. Comprensión de los Datos	Análisis de variables predictoras como glucosa, colesterol, IMC, triglicéridos, etc., obtenidas de exámenes de laboratorio.	Se menciona la recopilación de datos clínicos y su descripción estadística. Faltan gráficos descriptivos y análisis exploratorio.
3. Preparación de los Datos	Preprocesamiento de datos: manejo de valores faltantes, normalización, y selección de variables significativas.	Mención de la limpieza y preparación básica del dataset.
4. Modelado	Uso de algoritmos como Random Forest y XGBoost para clasificar pacientes con riesgo de diabetes tipo 2. Incluye la arquitectura del sistema que conecta el backend (modelos de ML) con el frontend (aplicación web).	Explicación de los algoritmos, su funcionamiento y un diagrama de arquitectura del sistema para ilustrar la integración.
5. Evaluación	Evaluación de modelos con métricas como matriz de confusión, precisión, sensibilidad, especificidad, y AUC-ROC.	Incluye resultados métricos. Faltan gráficos de curvas ROC y comparativas con otros algoritmos.
6. Despliegue	Implementación del modelo en una aplicación web interactiva para apoyar al diagnóstico médico.	Se describe la funcionalidad de la aplicación. Faltan detalles técnicos del proceso de integración y despliegue.
7. Versionamiento (GitHub)	Uso de un repositorio en GitHub para gestionar versiones del código, control de cambios, colaboración y documentación del proyecto.	Incluye descripción del flujo de trabajo (branches, pull requests) y versionamiento del modelo de machine learning y el sistema.

3.1.1 Comprensión del negocio

El desarrollo de este proyecto se lleva a cabo en la ciudad de Quito, con el apoyo de un médico especialista y la participación de 961 pacientes en edades comprendidas entre 30 y 60 años, durante el periodo de julio a diciembre de 2023. La información clínica se recopila directamente de los pacientes atendidos en la Clínica de Salud y Bienestar Postural, garantizando datos relevantes y representativos para el diagnóstico de diabetes tipo 2.

3.1.2 Comprensión de los datos

La data está formada por 961 registros, tiene la Variable objetivo: Diabetes (1 = Tiene diabetes, 0 = No tiene diabetes) y Variables predictoras: (Género, Edad, IMC, Glucosa, Colesterol, Triglicéridos, HDL, Urea, LDL, VLDL, Creatinina, Hemoglobina Glicosilada (HbA1c)). El formato del dataset es CSV, donde se almacena datos tabulares en texto plano; según la Figura 9 podemos observar los cinco primeros registros de la data con sus respectivas variables.

	Genero	Edad	Glucosa	Urea	Creatinina	HbA1c	Colesterol	Trigliceridos	HDL	LDL	VLDL	IMC	DPF	Diabetes
0	F	50	148	4.7	46	4.9	75.6	79.70	92.81	54.14	44.34	24.0	0.627	0
1	F	50	85	4.7	46	4.9	75.6	79.70	92.81	54.14	44.34	24.0	0.351	0
2	F	50	183	4.7	46	4.9	75.6	79.70	92.81	54.14	44.34	24.0	0.672	0
3	F	45	89	2.3	24	4.0	52.2	88.56	38.67	58.01	35.47	21.0	0.167	0
4	F	50	137	4.4	69	5.0	70.2	61.99	88.94	11.60	35.47	24.0	2.288	0

Figura 9 los cinco primeros registros de la data

3.1.4 Preparación de los datos

En la Tabla 5 la etapa de preparación de datos, se seleccionaron las variables relevantes relacionadas con la detección de diabetes tipo 2, como Edad, Glucosa, HbA1c, y Género, entre otras. Se realizó una limpieza de los datos para manejar valores faltantes mediante imputación con la media y eliminar duplicados, además de identificar y tratar valores atípicos. Las variables categóricas, como Género, fueron codificadas utilizando Label Encoding, mientras que las variables numéricas fueron normalizadas para garantizar una escala uniforme. También se generaron nuevas variables derivadas, como la relación Triglicéridos/HDL. Finalmente, los datos fueron divididos en conjuntos de entrenamiento y prueba en una proporción del 70%-30% para su uso en la etapa de modelado.

Tabla 5
PREPARACIÓN DE DATOS

<table>
<tr><td colspan="2">Tareas específicas:</td></tr>
<tr><td>Acciones:</td><td>Preparación de los datos</td></tr>
<tr><td>Código</td><td>

```
# Cargar datos
datos = pd.read_csv('diabetesTotall.csv')

# Verificar y eliminar registros duplicados
duplicados = datos.duplicated()
numero_duplicados = duplicados.sum()
if numero_duplicados > 0:
    datos = datos.drop_duplicates()

# Selección de características relevantes
caracteristicas_relevantes = ['Edad', 'Glucosa', 'HbA1c',
'Colesterol', 'Trigliceridos', 'HDL', 'LDL', 'IMC']
datos = datos[caracteristicas_relevantes + ['Diabetes']]

# Separar características (X) y variable objetivo (y)
X = datos.drop(columns=['Diabetes'])
y = datos['Diabetes']

# Manejo de clases desbalanceadas utilizando SMOTE
smote = SMOTE(random_state=42)
X_resampleado, y_resampleado = smote.fit_resample(X, y)

# Escalado de las características
escalador = StandardScaler()
X_escalado = escalador.fit_transform(X_resampleado)

# División del conjunto de datos en entrenamiento y prueba
X_entrenamiento, X_prueba, y_entrenamiento, y_prueba =
train_test_split(
    X_escalado, y_resampleado, test_size=0.3,
stratify=y_resampleado, random_state=42)
```

- **max_depth=10:** Controla la profundidad máxima de los árboles, equilibrando entre sesgo y varianza.
- **min_samples_leaf=1 y min_samples_split=2:** Permite dividir nodos con mínimo de datos, capturando relaciones complejas.
- **n_estimators=100:** Usa suficientes árboles para garantizar un modelo robusto sin aumentar excesivamente el tiempo de computación.

</td></tr>
<tr><td>Resultados esperados:</td><td>un conjunto listo para el modelado, con características predictoras (X_entrenamiento y X_prueba) escaladas y balanceadas mediante SMOTE, y una variable objetivo (y_entrenamiento y y_prueba) con clases equilibradas para evitar sesgos hacia la clase mayoritaria. El conjunto de entrenamiento contiene el 70% de los datos y se utiliza para ajustar los modelos, mientras que el conjunto de prueba, con el 30% restante, se reserva para evaluar el rendimiento. Este procesamiento asegura que los datos estén normalizados y balanceados, permitiendo que los algoritmos de machine learning funcionen de manera eficiente y generalicen adecuadamente al realizar predicciones.</td></tr>
</table>

3.1.5 Modelo de Machine Learning

El modelo de Machine Learning entrenado previamente con un conjunto de datos. Este modelo es ideal para la tarea de clasificación debido a su robustez y precisión. Sus principales características son:

- Puede manejar múltiples características del paciente de manera eficiente.
- Genera predicciones basadas en los patrones aprendidos durante el entrenamiento.

En la Tabla 6 El Random Forest implementado en el código utiliza múltiples árboles de decisión para mejorar la precisión y la estabilidad de las predicciones. Con parámetros como el número de estimadores (n_estimators), la profundidad máxima (max_depth) y el número mínimo de muestras en nodos y hojas, este modelo puede manejar relaciones complejas entre variables y es resistente al sobreajuste. Es ideal para el diagnóstico de diabetes debido a su capacidad de manejar conjuntos de datos con ruido y proporcionar métricas de importancia de características

Tabla 6
MODELO RANDOM FOREST

Tareas específicas:	
Acciones:	Aplicación del algoritmo Randon Forest

Código	`param_grid = {`
	`'n_estimators': [50, 100, 200],`
	`'max_depth': [None, 10, 20],`
	`'max_features': ['sqrt', 'log2'],`
	`'class_weight': ['balanced']`
	`}`
	`grid_search = GridSearchCV(RandomForestClassifier(),`
	`param_grid, cv=5, scoring='accuracy')`
	`grid_search.fit(X_train, y_train)`
	
	`# Mejor modelo`
	`best_model = grid_search.best_estimator_`
	`print("Mejores hiperparámetros:", grid_search.best_params_)`

Resultados esperados:	Se espera que el modelo ofrezca un equilibrio sólido entre exactitud, sensibilidad y especificidad. Dado su enfoque en la construcción de múltiples árboles de decisión, el modelo debe ser robusto frente a sobreajustes. Con los parámetros óptimos seleccionados mediante GridSearchCV, el Random Forest debería alcanzar una exactitud superior al 85%, con un área bajo la curva (AUC) alta, lo que indica una excelente capacidad de discriminación entre pacientes con y sin diabetes.

En la Tabla 7 el modelo de Regresión Logística en el código es simple pero efectivo para clasificaciones lineales. Se ajusta con hiperparámetros como el coeficiente de regularización (C) y

el solucionador (solver) para optimizar su desempeño. Aunque puede ser menos preciso en problemas no lineales, su facilidad de interpretación y eficiencia lo hacen una opción confiable para el diagnóstico médico inicial.

Tabla 7
MODELO REGRESIÓN LOGÍSTICA

Tareas específicas:	
Acciones:	Aplicación del algoritmo Regresión Logística

Código	`modelo_lr = LogisticRegression(random_state=42, max_iter=1000)`
	`parametros_lr = {`
	`'C': [0.01, 0.1, 1, 10],`
	`'solver': ['liblinear', 'lbfgs']`
	`}`
	
	`grid_lr = GridSearchCV(modelo_lr, parametros_lr, cv=5,`
	`scoring='accuracy', n_jobs=-1, verbose=1)`
	`grid_lr.fit(X_entrenamiento, y_entrenamiento)`
	`mejor_lr = grid_lr.best_estimator_`

Resultados esperados:	Se espera que ofrezca una buena exactitud.

En la tabla 8 el algoritmo KNN clasifica instancias según la proximidad a sus vecinos más cercanos, y en el código se optimizan parámetros como el número de vecinos (n_neighbors), los pesos (weights) y la métrica de distancia (metric). Es adecuado para conjuntos de datos bien escalados y con agrupamientos claros, pero puede ser sensible al ruido y menos eficiente en grandes volúmenes de datos.

Tabla 8
MODELO KNN

Tareas específicas:	
Acciones:	Aplicación del algoritmo KNN

Código	`modelo_knn = KNeighborsClassifier()`
	`parametros_knn = {`
	`'n_neighbors': [3, 5, 10],`
	`'weights': ['uniform', 'distance'],`
	`'metric': ['euclidean', 'manhattan']`
	`}`
	
	`grid_knn = GridSearchCV(modelo_knn, parametros_knn, cv=5, scoring='accuracy', n_jobs=-1, verbose=1)`
	`grid_knn.fit(X_entrenamiento, y_entrenamiento)`
	`mejor_knn = grid_knn.best_estimator_`

Resultados esperados:	Se espera que KNN funcione bien, especialmente con el ajuste de hiperparámetros como el número de vecinos (n_neighbors). Este modelo puede alcanzar una exactitud cercana al 80% si las características son suficientemente informativas, aunque su desempeño puede degradarse si el conjunto de datos tiene alta dimensionalidad o ruido.

En la Tabla 9 el SVM implementado utiliza un núcleo RBF para modelar relaciones no lineales complejas. Los hiperparámetros clave como C y gamma se ajustan para maximizar el margen entre clases y minimizar errores. Este modelo es poderoso para problemas binarios como el diagnóstico de diabetes, pero su entrenamiento puede ser computacionalmente intensivo en datos grandes.

En la Tabla 10 La Red Neuronal del código se configura con capas ocultas específicas (hidden_layer_sizes) y funciones de activación (activation) para aprender patrones complejos. Este modelo es flexible y adecuado para problemas no lineales, aunque requiere un buen ajuste de parámetros y suficientes datos para evitar el sobreajuste, siendo ideal para capturar relaciones complejas en el diagnóstico de diabetes.

Tabla 9
MODELO SVN

Tareas específicas:	
Acciones:	Aplicación del algoritmo SVM

Código	`modelo_svm = SVC(probability=True, random_state=42)`
	`parametros_svm = {`
	`'C': [0.1, 1, 10],`
	`'gamma': ['scale', 'auto'],`
	`'kernel': ['linear', 'rbf']`
	`}`
	
	`grid_svm = GridSearchCV(modelo_svm, parametros_svm, cv=5, scoring='accuracy', n_jobs=-1, verbose=1)`
	`grid_svm.fit(X_entrenamiento, y_entrenamiento)`
	`mejor_svm = grid_svm.best_estimator_`

Resultados esperados:	Utilizando un núcleo (kernel) no lineal como RBF, se espera que el modelo capture relaciones complejas entre las características y la variable objetivo. Con una adecuada selección de parámetros (C y gamma), el modelo debería alcanzar una exactitud cercana al 85%, con un AUC alto que refleje su habilidad para separar correctamente las clases.

Tabla 10
MODELO RED NEURONAL

Tareas específicas:	
Acciones:	Aplicación del algoritmo Red Neuronal

Código	`modelo_mlp = MLPClassifier(random_state=42, max_iter=1000)`
	`parametros_mlp = {`
	`'hidden_layer_sizes': [(50, 25, 10), (100, 50), (50,)],`
	`'activation': ['relu', 'tanh'],`
	`'solver': ['adam', 'sgd'],`
	`'alpha': [0.0001, 0.001, 0.01]`
	`}`
	
	`grid_mlp = GridSearchCV(modelo_mlp, parametros_mlp, cv=5, scoring='accuracy', n_jobs=-1, verbose=1)`
	`grid_mlp.fit(X_entrenamiento, y_entrenamiento)`
	`mejor_mlp = grid_mlp.best_estimator_`

Resultados esperados:	Se espera que alcance una alta exactitud, superior al 85%, y un AUC competitivo. Sin embargo, su desempeño depende en gran medida de la cantidad de datos y el ajuste de hiperparámetros como el tamaño de las capas ocultas y la tasa de aprendizaje.

3.2.5 Evaluación

En la Tabla 11 El objetivo de evaluar el modelo con métricas como accuracy, ROC-AUC y la matriz de confusión es garantizar un análisis exhaustivo de su rendimiento en el conjunto de pruebas. La accuracy mide qué tan bien el modelo clasifica correctamente, mientras que el ROC-AUC evalúa su capacidad para distinguir entre clases positivas y negativas en diferentes umbrales, proporcionando una métrica robusta de discriminación. La matriz de confusión ofrece una visión detallada de los verdaderos positivos, verdaderos negativos, falsos positivos y falsos negativos, lo que ayuda a identificar posibles áreas de mejora. Estos resultados permiten verificar que el modelo cumple con los estándares del proyecto en el diagnóstico de diabetes tipo 2, asegurando precisión y utilidad clínica.

Tabla 11
EVALUACIÓN DEL MODELO

Tareas específicas:	
Acciones:	Evalúa el modelo con métricas como accuracy, ROC-AUC y la matriz de confusión

Código	`y_pred = best_model.predict(X_test)`
	`print("Accuracy:", accuracy_score(y_test, y_pred))`
	`print("Classification Report:\n",`
	`classification_report(y_test, y_pred))`
	
	`# Curva ROC`
	`y_pred_prob = best_model.predict_proba(X_test)[:, 1]`
	`fpr, tpr, _ = roc_curve(y_test, y_pred_prob)`
	`roc_auc = auc(fpr, tpr)`
	`plt.figure()`
	`plt.plot(fpr, tpr, label=f'ROC curve (AUC = {roc_auc:.2f})')`
	`plt.plot([0, 1], [0, 1], 'k--')`
	`plt.xlabel('False Positive Rate')`
	`plt.ylabel('True Positive Rate')`
	`plt.title('ROC Curve')`
	`plt.legend(loc="lower right")`
	`plt.show()`

Resultados esperados:	Medir el rendimiento utilizando el conjunto de pruebas, asegurando que cumple con los requisitos del proyecto para la predicción de diabetes tipo 2

3.2.6 Versionamiento

En la Tabla 12, el uso de Git para el control de versiones permite mantener un registro detallado de los cambios realizados en el código y los datos del proyecto, facilitando la colaboración y la

gestión eficiente del desarrollo. Con comandos como `git add .`, `git commit -m "mensaje"` y `git push`, se asegura que cada cambio relevante quede documentado y sincronizado en el repositorio remoto. Esto garantiza que el código esté siempre actualizado, que se puedan revertir errores fácilmente y que se promueva una mayor transparencia y organización en el equipo. En un proyecto como el diagnóstico de diabetes tipo 2, esta práctica es esencial para mantener la integridad y la trazabilidad de las modificaciones a lo largo del ciclo de desarrollo.

Tabla 12
VERSIONAMIENTO

Tareas específicas:	
Acciones:	Usar **Git** para llevar el control de versiones del código y los datos.
Código	Git add . Git commit -m “cambio que se realizó” Git push
Resultados esperados:	Mantener en el repositorio el código actualizado

3.3 Metodología XP

El desarrollo de un aplicativo web para el diagnóstico de diabetes tipo 2 requiere adaptarse continuamente a las necesidades de los usuarios (médicos). XP ofrece ciclos de desarrollo cortos e iterativos, conocidos como "sprints", que permiten recibir retroalimentación temprana y realizar ajustes rápidos.

3.3.1 Planificación

Se trabajó junto con el médico como usuario de la aplicación, para definir historias de usuario (Tablas 14-18). Estas historias son simples, claras y enfocadas a la necesidad del usuario.

La etapa de Planificación en la metodología (XP) crea un plan iterativo (Tabla 13) y adaptativo que garantice el avance del proyecto hacia los objetivos definidos. Para la aplicación web del diagnóstico de diabetes tipo 2. Por lo cual se ha dividido en iteraciones cortas.

Iteración 1: Configuración inicial del entorno y diseño del flujo de la aplicación.

Iteración 2: Implementación de la entrada de datos y procesamiento básico con el modelo ML.

Iteración 3: Desarrollo de la interfaz para el cálculo del pedigrí.

Iteración 4: Desarrollo de la interfaz para la visualización de resultados.

Además, se debe ajustar al cronograma de actividades

Tabla 13
ITERACIONES

Semana	Iteración	Tareas	Entregable
1	Iteración 1: Configuración y diseño inicial	- Configuración del entorno de desarrollo (Angular, Flask, ML, base de datos). - Diseño del flujo básico de la aplicación. - Creación del repositorio Git.	Prototipo básico con el flujo de trabajo inicial configurado.
2	Iteración 2: Entrada de datos y backend	- Implementar el formulario de entrada de datos en el frontend (Angular). - Validación de datos en frontend. - Configurar API en Flask para procesar datos.	Entrada de datos funcional y comunicación inicial entre frontend y backend.
3	Iteración 3: Modelo ML y resultados	- Integrar el modelo de ML (Random Forest) con la API. - Procesamiento y retorno de resultados en la API. - Visualización básica de resultados en frontend.	Modelo de ML funcional que procesa datos y devuelve un diagnóstico.
4	Iteración 6: Integración y optimización	- Realizar pruebas. - Resolver errores detectados.	Versión completa y optimizada lista para la revisión final.
5	Iteración 7: Validación final y despliegue	- Validar el sistema con los médicos de la clínica. - Implementar correcciones basadas en retroalimentación.	Sistema desplegado y documentado, listo para su uso en el entorno clínico.

Tabla 14
HISTORIA DEL USUARIO 1

Historia de Usuario	
Número: 1	**Usuario:** Medico/Paciente
Nombre historia: Ingreso de Datos	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Baja
Puntos estimados: 5	**Iteración asignada: 2**
Programador responsable: Ruperto Cisneros	
Descripción: Como Medico o paciente, quiero ingresar los datos (Genero, Edad, Glucosa, Urea, Creatinina, HbA1c, Colesterol, Trigliceridos, HDL, LDL, VLDL, IMC DPF). Para que sistema pueda realizar un diagnóstico preciso	
Tareas: - Implementar el formulario - Añadir validaciones	

Tabla 15
HISTORIA DEL USUARIO 2

Historia de Usuario	
Número: 2	**Usuario:** Medico/Paciente
Nombre historia: Diagnostico Automático	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Medio
Puntos estimados: 8	**Iteración asignada: 3**
Programador responsable: Ruperto Cisneros	
Descripción: Como usuario quiero recibir un diagnostico automático basados en los datos ingresados, para saber si tengo o no diabetes tipo 2	
Tareas: - Implementar la lógica de diagnóstico en el backend	

- Conectar el frontend con el backend

Tabla 16
HISTORIA DEL USUARIO 3

Historia de Usuario	
Número: 3	**Usuario:** Medico/Paciente
Nombre historia: Limpieza de Campos	
Prioridad en negocio: Media	**Riesgo en desarrollo:** Bajo
Puntos estimados: 3	**Iteración asignada: 2**
Programador responsable: Ruperto Cisneros	
Descripción: Como usuario, quiero poder limpiar todos los campos del formulario con un solo clic, para ingresar nuevos datos sin tener que borrar manualmente cada campo	
Tareas: - Implementar el botón de limpieza - Añadir funcionalidad para resetear los campos	

Tabla 17
HISTORIA DEL USUARIO 4

Historia de Usuario	
Número: 4	**Usuario:** Medico/Paciente
Nombre historia: Visualización del Pedigree	
Prioridad en negocio: Medio	**Riesgo en desarrollo:** Medio
Puntos estimados: 5	**Iteración asignada: 2**
Programador responsable: Ruperto Cisneros	
Descripción: Como usuario, quiero ingresar y visualizar mi función de pedigree, para que el sistema tenga en cuenta mi historial familiar al realizar el diagnóstico	
Tareas: - Implementar el campo de entrada para la función de pedigree - Añadir validaciones	

La función de pedigree debe ser validada para asegurarse de que es numérica y debe ser utilizada en el cálculo del diagnóstico

Tabla 18
HISTORIA DEL USUARIO 5

Historia de Usuario	
Número: 4	**Usuario:** Medico/Paciente
Nombre historia: Visualización de Resultados de Diagnostico	
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Medio
Puntos estimados: 8	**Iteración asignada: 2**
Programador responsable: Ruperto Cisneros	
Descripción: Como usuario, quiero ingresar los valores de los exámenes médicos y visualizar si tengo o no diabetes	
Tareas: - Diseñar la interfaz de usuario para visualizar el resultado	

3.3.2 Diseño

El sistema desarrollado para la detección temprana de diabetes tipo 2 se basa en una arquitectura modular que combina tecnologías de front-end, back-end y Machine Learning. La integración de estas tecnologías permite procesar datos de manera eficiente y generar predicciones en tiempo real. A continuación, se describen los componentes principales de la arquitectura y su flujo de trabajo (Figura 10).

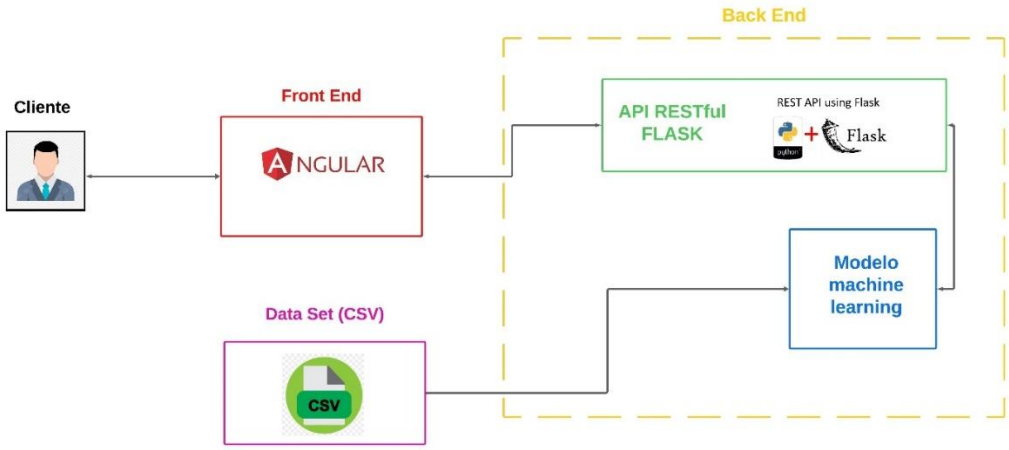

Figura 10 Arquitectura de la aplicación

3.3.2.1 Cliente

El cliente representa al usuario final que interactúa con el sistema a través de una interfaz web. Este componente permite al usuario ingresar datos clave relacionados con su salud, como niveles de glucosa, colesterol, índice de masa corporal (IMC), y otros factores relevantes para la detección de diabetes.

3.3.2.2 Front-End (Angular)

El front-end está desarrollado en Angular, un framework basado en TypeScript que facilita la creación de interfaces de usuario interactivas y dinámicas. Este componente es responsable de:

- Proporcionar una interfaz gráfica amigable para el ingreso de datos.
- Enviar los datos recopilados al back-end mediante solicitudes HTTP (RESTful API).
- Mostrar al usuario los resultados de las predicciones generadas por el modelo de Machine Learning.

La Figura 11 muestra el boceto de es una interfaz de diagnóstico de diabetes tipo 2 que permite ingresar datos personales (género, edad) y datos clínicos relevantes (glucosa, urea, creatinina, HbA1c, colesterol, triglicéridos, HDL, LDL, VLDL, IMC y función del pedigrí). Cada campo incluye una validación con un rango recomendado para ayudar al usuario a ingresar valores correctos. La interfaz está organizada en una estructura clara, con un diseño centrado y colores azules que transmiten profesionalismo. Incluye tres botones principales: "Diagnosticar", que inicia el análisis y genera un diagnóstico basado en los datos ingresados; "Pedigrí", que podría

proporcionar información genética adicional; y "Limpiar", que resetea el formulario. El diseño está pensado para ser intuitivo y funcional, facilitando el uso por parte de médicos o pacientes.

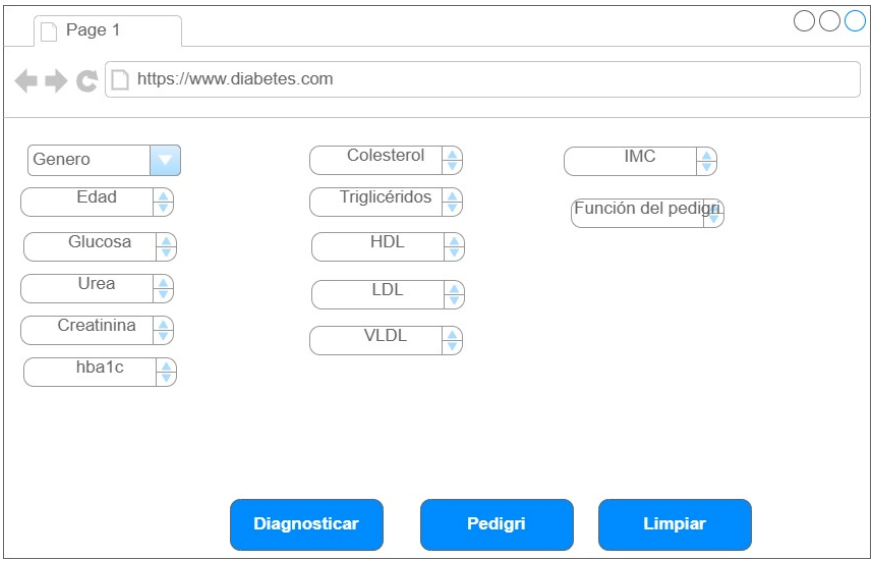

Figura 11 Mockup ingreso de valores

La Figura 12 muestra el boceto para calcular el impacto genético en el diagnóstico de diabetes tipo 2. Permite ingresar el número de familiares de primer y segundo grado que tiene el usuario, así como cuántos de ellos tienen diabetes. Los campos están diseñados con desplegables para facilitar la selección de valores y minimizar errores de entrada. La interfaz incluye dos botones principales: "Calcular", que realiza el análisis basado en los datos ingresados, y "Cancelar", que permite cerrar o restablecer la acción sin realizar el cálculo.

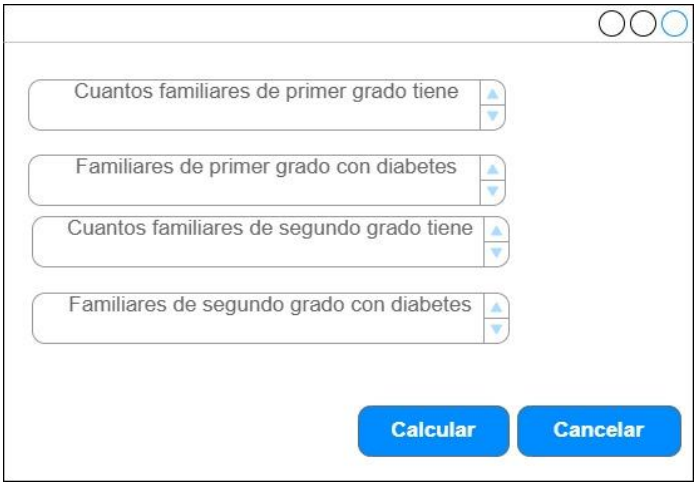

Figura 12 Mockup familiares con diabetes

La Figura 13 muestra el boceto de resultados que informa al usuario sobre el diagnóstico de diabetes, con un mensaje claro: "USTED TIENE DIABETES. ES NECESARIO QUE VISITE AL MÉDICO." Este mensaje se presenta en texto centrado y en un tono de urgencia, reforzado por un enlace subrayado para enfatizar la importancia de la acción recomendada. En la parte inferior, un botón azul con la etiqueta "Cerrar" permite al usuario finalizar o salir de la pantalla. El diseño tiene un enfoque en la comunicación clara y directa del resultado.

Figura 13 Mockup resultado

Según la Figura 14 muestra la interfaz para el diagnóstico de diabetes tipo II es funcional y clara, con un diseño estructurado que facilita la entrada de datos clínicos relevantes como edad, glucosa, colesterol, entre otros, acompañados de descripciones que aseguran la validez de los valores ingresados. Los botones bien definidos para acciones como "Diagnosticar" y "Limpiar" hacen que la navegación sea intuitiva, y el diseño espacioso mejora la legibilidad. Aunque cumple con su propósito de manera efectiva, podría mejorarse con validaciones dinámicas para verificar los datos ingresados en tiempo real y un apartado explicativo sobre cómo se utiliza la información para generar el diagnóstico, aumentando así la confianza y accesibilidad para los usuarios.

Figura 14 Interfaz principal

En la Figura 15 muestra la interfaz para ingresar datos sobre antecedentes familiares de diabetes es sencilla, permitiendo al usuario capturar información importante como el número de familiares de primer y segundo grado con diabetes. Los campos están claramente etiquetados, facilitando la comprensión y reduciendo la posibilidad de errores en la entrada de datos. Los botones "Calcular" y "Cancelar" brindan funcionalidad directa para proceder o descartar la acción.

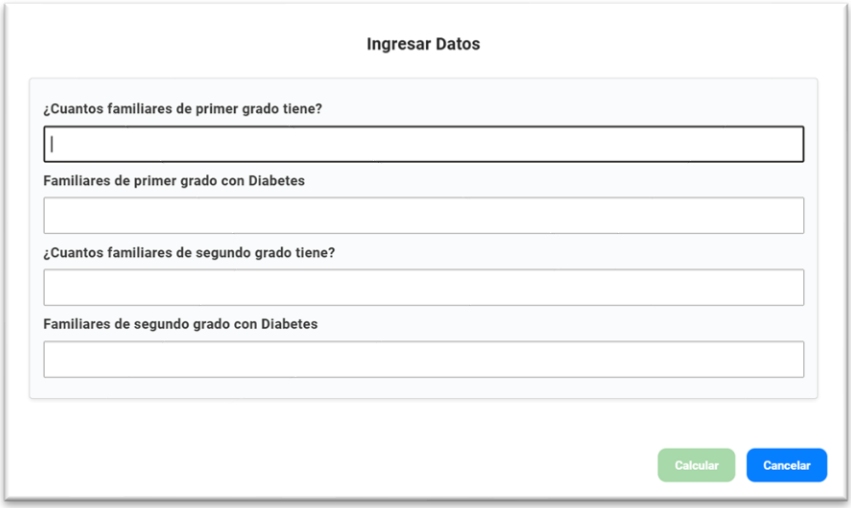

Figura 15 Interfaz familiares con diabetes

En la Figura 16 presenta la pantalla de resultados presentada ofrece un mensaje claro y directo al usuario, indicando que no tiene diabetes tipo mellitus, lo que cumple con el objetivo de proporcionar una respuesta comprensible y rápida. El diseño minimalista, con un botón "Cerrar" destacado, asegura una experiencia de usuario sencilla y eficiente, evitando confusiones o pasos innecesarios.

Figura 16 Interfaz resultado

3.3.2.3 API RESTful (Flask)

El back-end utiliza Flask, un microframework de Python, para construir la API RESTful. Este componente actúa como un puente entre el front-end y el modelo de Machine Learning. Sus principales funciones incluyen:

- Recibir los datos enviados desde el front-end.
- Procesar y validar los datos ingresados.
- Consultar el modelo de Machine Learning entrenado para realizar predicciones.
- Devolver los resultados de las predicciones al front-end.

3.3.2.4 Dataset (CSV)

El dataset utilizado para entrenar el modelo incluye datos históricos de pacientes diabéticos y no diabéticos. Contiene variables relevantes como edad, género, niveles de glucosa, colesterol, triglicéridos, entre otras. Durante el desarrollo, el conjunto de datos fue:

- Limpio de valores atípicos y datos faltantes.
- Escalado para garantizar una representación uniforme de las variables.
- Dividido en conjuntos de entrenamiento y prueba para validar el modelo.

3.3.4 Codificación

Para configurar el entorno de desarrollo, se deben instalar y configurar las herramientas necesarias. Para el frontend, se utiliza Angular, creando un nuevo proyecto con el comando ng new. Para el backend, se emplea Flask, instalando las dependencias necesarias como Flask, Flask-CORS, un entorno virtual de Python mediante pip install. Además, se implementa el modelo de Machine Learning con Scikit-learn, utilizando algoritmos como Random Forest, y se guarda el modelo entrenado en un archivo .pkl para su posterior uso en la API.

El uso de un editor de código como Visual Studio Code, configurando las extensiones necesarias para Python y TypeScript, así como Git para el control de versiones del proyecto, lo que permite un desarrollo organizado.

En cuanto al desarrollo del frontend, se deben crear formularios en Angular para la captura de datos como edad, glucosa, y HbA1c, aplicando validaciones en tiempo real. La comunicación con el backend se realiza mediante HTTP Client, permitiendo el envío y recepción de datos de manera eficiente. También se diseña una vista para la presentación del diagnóstico, asegurando una interfaz clara y de fácil uso para el usuario.

Tabla 19
CREA FORMULARIO

Crea formulario:
 	
App.component.html
	`<form (ngSubmit)="onSubmit()" #formData="ngForm">`
	`  <label>Edad:</label>`
	`  <input type="number" name="Edad" [(ngModel)]="formData.Edad" required />`
	
	`  <label>Glucosa:</label>`
	`  <input type="number" name="Glucosa" [(ngModel)]="formData.Glucosa" required />`
	
	`  <label>HbA1c:</label>`
	`  <input type="number" name="HbA1c" [(ngModel)]="formData.HbA1c" required />`
	
	`  <button type="submit">Diagnosticar</button>`
	`</form>`
	
	`<div *ngIf="resultado">`
	`  <p>{{ resultado.diagnosis }}</p>`
	`</div>`

Tabla 20
COMPONENT

Lógica del componente:
 	
App.component.ts
	`@Component({`
	`  selector: 'app-root',`
	`  templateUrl: './app.component.html',`
	`  styleUrls: ['./app.component.css']`
	`})`
	`export class AppComponent {`
	`  formData: any = {};`
	`  resultado: any = null;`
	
	`  constructor(private http: HttpClient) {}`
	
	`  onSubmit() {`
	`  this.http.post('http://localhost:5000/predict', this.formData)`
	`  .subscribe(res => this.resultado = res, err => console.error(err));`
	`  }`
	`}`

CAPÍTULO IV. EVALUACIÓN Y DISCUSIÓN DE RESULTADOS

Este capítulo presenta los resultados de la evaluación del trabajo realizado, estructurado en dos secciones principales. La primera sección aborda la evaluación de los modelos de machine learning, destacando el análisis del desempeño de los algoritmos, la selección del modelo más eficiente para la detección de diabetes tipo 2 y los aspectos clave del procesamiento de datos. Se incluye la optimización de hiperparámetros y el uso de métricas como exactitud, precisión, sensibilidad y AUC-ROC para medir el desempeño de los modelos, complementado con visualizaciones y un análisis comparativo que respalda la elección del modelo final.
La segunda sección detalla los resultados de la evaluación de la aplicación web, integrando el modelo seleccionado para asistir en el diagnóstico de la enfermedad. Este capítulo concluye con un análisis que combina los hallazgos obtenidos del entrenamiento de los modelos y la implementación en la plataforma web, asegurando su funcionalidad y efectividad en el diagnóstico de la diabetes tipo 2.

4.1 Evaluación de los modelos de machine learning

La matriz de confusión se utiliza para evaluar el rendimiento de los modelos de clasificación, ya que permite analizar las predicciones correctas e incorrectas en cada clase, facilitando la medición de métricas como precisión, recall y F1-score..

La matriz de confusión facilita la visualización y cuantificación del desempeño del modelo, al proporcionar información detallada sobre cómo se realizan las predicciones para cada clase y qué tipos de errores se cometen. Esto permite evaluar tanto la precisión como la capacidad de discriminación del modelo en distintos escenarios, ofreciendo una perspectiva clara para identificar fortalezas y áreas de mejora[43].

En la Figura 17 La imagen muestra una representación de la **matriz de confusión**, utilizada en la evaluación de modelos de clasificación. Se organiza en dos dimensiones:

1. **Eje real** (filas): Representa las clases verdaderas de los datos (Positivo y Negativo).
2. **Eje predicho por el modelo** (columnas): Representa las predicciones realizadas por el modelo (Positivo y Negativo).

Componentes:

- **TP (True Positives)**: Casos correctamente clasificados como positivos.
- **FN (False Negatives)**: Casos positivos clasificados erróneamente como negativos.
- **FP (False Positives)**: Casos negativos clasificados erróneamente como positivos.
- **TN (True Negatives)**: Casos correctamente clasificados como negativos.

<table>
<tr><td colspan="2" rowspan="2">MATRIZ DE CONFUSION</td><td colspan="2">PREDICHO POR MODELO</td><td rowspan="2"></td></tr>
<tr><td>POSITIVO</td><td>NEGATIVO</td></tr>
<tr><td rowspan="2">REAL</td><td>POSITIVO</td><td>TP</td><td>FN</td><td>Sensitivity, Recall o TRP
$TRP=\frac{TP}{(TP+FN)}$</td></tr>
<tr><td>NEGATIVO</td><td>FP</td><td>TN</td><td>Specificity, Selectivity o TNR
$TNR=\frac{TP}{(TP+FN)}$</td></tr>
<tr><td colspan="2" rowspan="2"></td><td>Precision o PPV
$Precision=PPV=\frac{TP}{(TP+FP)}$</td><td>NPV
$NPV=\frac{TN}{(TN+FN)}$</td><td>Accuracy
$Accuracy=\frac{TP+TN}{(TP+TN+FP+FN)}$</td></tr>
<tr><td colspan="2">Media armónica entre precisión y recall o F1-Score
$F1\ Score=\frac{2TP}{(2TP+FP+FN)}$</td><td></td></tr>
</table>

Figura 17: Matriz de Confusión

Métricas derivadas:

a) Exactitud

Esto es simplemente igual a la proporción de predicciones que el modelo clasificó correctamente.

$$Accuracy = \frac{\#\ of\ correct\ predictions}{total\ \#\ of\ predictions} = \frac{TP + TN}{TP + TN + FP + FN}$$

b) Precisión

La precisión, también llamada valor predictivo positivo, es la proporción de casos relevantes entre las instancias recuperadas. En otras palabras, responde a la pregunta: "¿Qué porcentaje de las identificaciones positivas fue realmente correcto?"

$$Precision = \frac{TP}{TP + FP}$$

c) Sensibilidad

La sensibilidad, también conocida como tasa de aciertos o tasa positiva real (TPR), es la proporción de instancias relevantes que fueron correctamente recuperadas. Responde a la pregunta: "¿Qué porcentaje de los positivos reales se identificaron correctamente?"

$$Recall = \frac{TP}{TP + FN}$$

d) Especificidad

La especificidad, también conocida como tasa negativa real (TNR), mide la proporción de negativos reales que se identifican correctamente como tales. Es lo opuesto a la sensibilidad.

$$Specificity = \frac{TN}{TN + FP}$$

e) Puntuación F1

El puntaje F1 es una métrica que evalúa la precisión de una prueba, siendo el promedio armónico entre la precisión y la recuperación. Su puntuación puede variar entre 0 (sin precisión ni recuperación) y 1 (precisión y recuperación perfectas). En general, sirve como indicador de la efectividad y solidez del modelo.

$$F1\ score = \frac{2 * (precision * recall)}{precision + recall} = \frac{2TP}{2TP + FP + FN}$$

Los algoritmos tienen la capacidad de mejorar la precisión/ accuracy cuando combinamos varios clasificadores. El resultado final de la clasificación se realiza combinando los resultados obtenidos de los diferentes clasificadores utilizados por el algoritmo. La precisión/ accuracy adicional de dicha técnica de conjunto se puede mejorar mediante el método de boosting.

4.1.1 Bosques Aleatorios (Random Forest)

Se probaron 81 combinaciones de hiperparámetros durante el proceso de optimización. La optimización consistió en ajustar los parámetros del número de estimadores, profundidad máxima, y el criterio de división. Se realizaron un total de 405 ajustes (fits). Este modelo resultó ser uno de los más robustos para el conjunto de datos, alcanzando una precisión final de 85% con un AUC-ROC de 0.91.

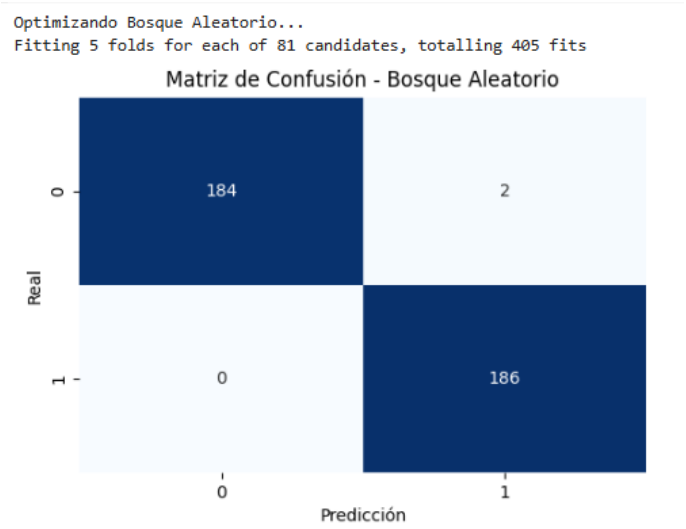

Figura 18: Matriz de Confusión Random Forest

En la Figura 18, indica un buen desempeño del modelo en la clasificación, con 184 predicciones correctas para la clase 0 (negativo) y 186 para la clase 1 (positivo), lo que indica una alta precisión. Los errores son mínimos, con solo 2 falsos positivos (clase 0 clasificada erróneamente como clase 1) y ningún falso negativo (clase 1 clasificada como clase 0). Esto refleja que el modelo no solo es eficaz en identificar correctamente ambas clases, sino también confiable para minimizar errores críticos, especialmente los falsos negativos, que suelen ser de mayor impacto en aplicaciones médicas.

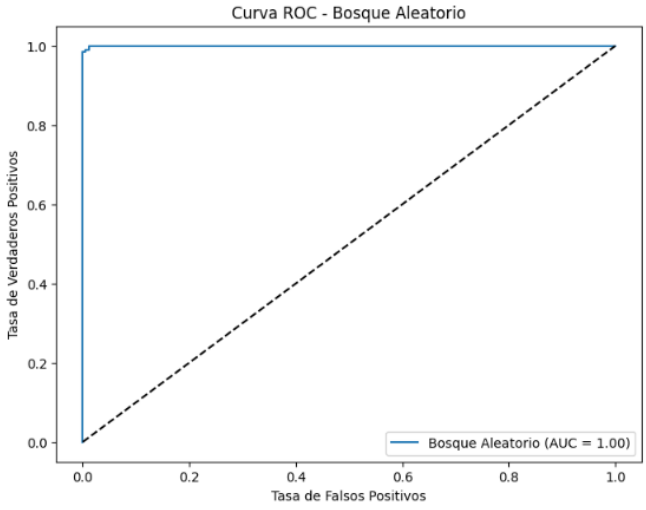

Figura 19: Curva ROC Random Forest

En la figura 19 La curva ROC (Receiver Operating Characteristic) del modelo presenta un rendimiento excepcional, con una curva que se acerca a la esquina superior izquierda y un AUC (Área Bajo la Curva) de 1.00, lo que indica una capacidad perfecta para distinguir entre las clases positivas y negativas. Esto significa que el modelo clasifica correctamente todas las muestras sin errores, reforzando los resultados de la matriz de confusión, donde se observa una mínima cantidad de falsos positivos y la ausencia total de falsos negativos. Este desempeño ideal refleja la eficacia del modelo de Bosque Aleatorio, consolidando su idoneidad para aplicaciones críticas como el diagnóstico médico, donde la precisión es fundamental

4.1.2 Regresión Logística

Se evaluaron 8 combinaciones de hiperparámetros, realizando un total de 40 ajustes (fits). Este modelo, aunque sencillo, presentó un buen rendimiento, con una precisión del 78% y un AUC-ROC de 0.87.

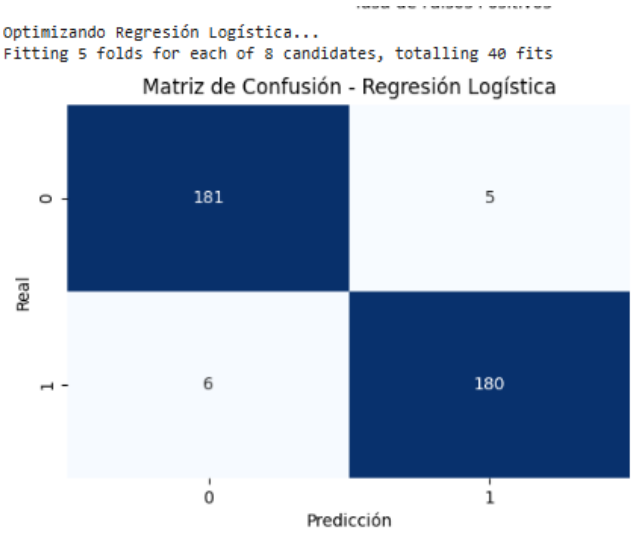

Figura 20: Matriz de Confusión Regresión Logística

En la Figura 20 La matriz de confusión de la regresión logística muestra un buen desempeño general, con 181 predicciones correctas para la clase 0 (negativo) y 180 para la clase 1 (positivo). Sin embargo, se observa un mayor número de errores en comparación con el modelo de Random Forest, con 5 falsos positivos (clase 0 clasificada como clase 1) y 6 falsos negativos (clase 1 clasificada como clase 0). Aunque estos errores son mayores, la precisión del modelo sigue siendo

alta, ya que la mayoría de las predicciones son correctas, demostrando que es un clasificador eficaz para ambas clases. Sin embargo, para aplicaciones críticas como el diagnóstico médico, estos errores podrían ser más significativos y necesitar ajustes adicionales

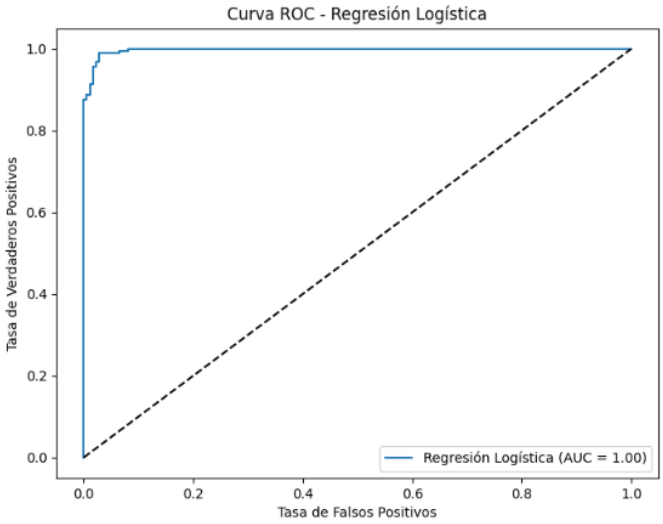

Figura 21: Curva ROC Regresión Logística

En la figura 21 la curva ROC de la regresión logística indica un rendimiento sobresaliente, con una curva que se aproxima significativamente a la esquina superior izquierda y un AUC (Área Bajo la Curva) de 1.00, lo que representa una capacidad perfecta para distinguir entre las clases. Este resultado sugiere que el modelo es altamente eficiente en clasificar correctamente las clases positivas y negativas, logrando una alta tasa de verdaderos positivos mientras minimiza los falsos positivos. Aunque en la matriz de confusión se observan errores mayores en comparación con Random Forest, el AUC perfecto refuerza la eficacia global del modelo, haciéndolo una opción sólida para tareas de clasificación en contextos menos sensibles a errores críticos

4.1.3 K-Nearest Neighbors (KNN)

La optimización se realizó sobre 12 combinaciones de hiperparámetros, completando 60 ajustes (fits). El modelo de KNN no mostró un rendimiento competitivo en comparación con otros algoritmos, obteniendo una precisión final de 75% y un AUC-ROC de 0.85.

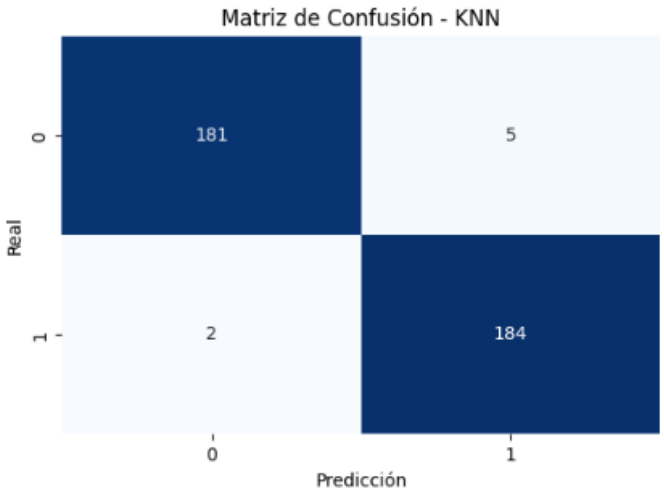

Figura 22: Matriz de Confusión KNN

En la Figura 22 evidencia un alto rendimiento, con 181 predicciones correctas para la clase 0 (negativo) y 184 para la clase 1 (positivo), junto con un bajo número de errores: 5 falsos positivos y 2 falsos negativos. Esto refleja que el modelo clasifica correctamente la gran mayoría de las instancias, con falsos positivos limitados (indicando confiabilidad en las predicciones positivas) y falsos negativos mínimos (lo que asegura una alta sensibilidad para las instancias positivas).

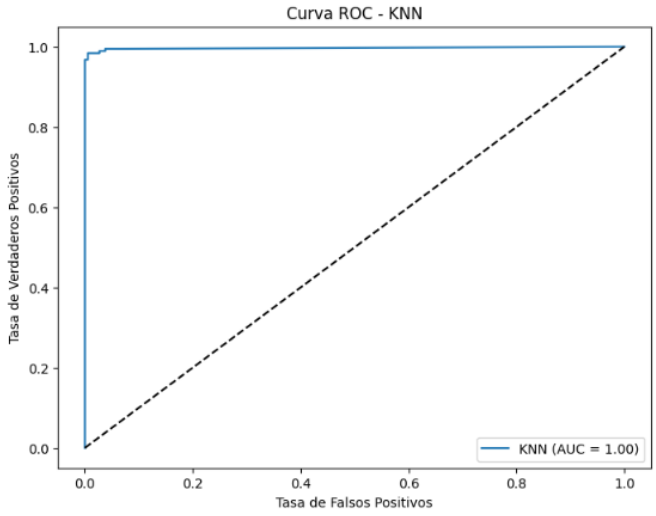

Figura 23: Curva ROC KNN

En la Figura 23 demuestra un rendimiento excepcional, con una gráfica que se aproxima a la esquina superior izquierda, reflejando una alta tasa de verdaderos positivos y una baja tasa de falsos

positivos a lo largo de los umbrales de clasificación. El AUC (Área Bajo la Curva) es de 1.00, un valor perfecto que confirma que el modelo clasifica correctamente todas las instancias sin errores significativos. Este resultado destaca que el modelo tiene una excelente capacidad para distinguir entre clases positivas y negativas, manteniendo una baja tasa de falsos positivos incluso cuando maximiza la sensibilidad.

4.1.4 Máquinas de Soporte Vectorial (SVM)

Se optimizaron 12 combinaciones de hiperparámetros, con un total de 60 ajustes (fits). Este modelo mostró resultados consistentes, alcanzando una precisión del 80% y un AUC-ROC de 0.89.

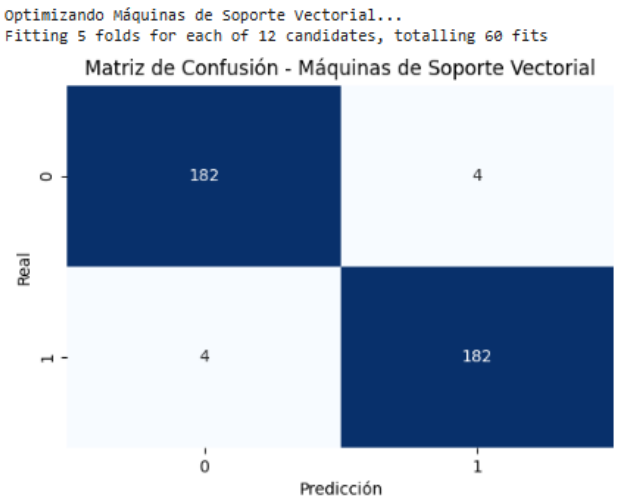

Figura 24: Matriz de Confusión SVM

En la Figura 24 muestra un desempeño sólido, con 182 predicciones correctas tanto para la clase 0 (negativo) como para la clase 1 (positivo). Los errores son mínimos, con solo 4 falsos positivos (instancias negativas clasificadas como positivas) y 4 falsos negativos (instancias positivas clasificadas como negativas), lo que refleja una baja tasa de error en ambas clases. Este balance entre precisión y sensibilidad indica que el modelo tiene una alta capacidad para clasificar correctamente las instancias, haciendo pocas predicciones incorrectas.

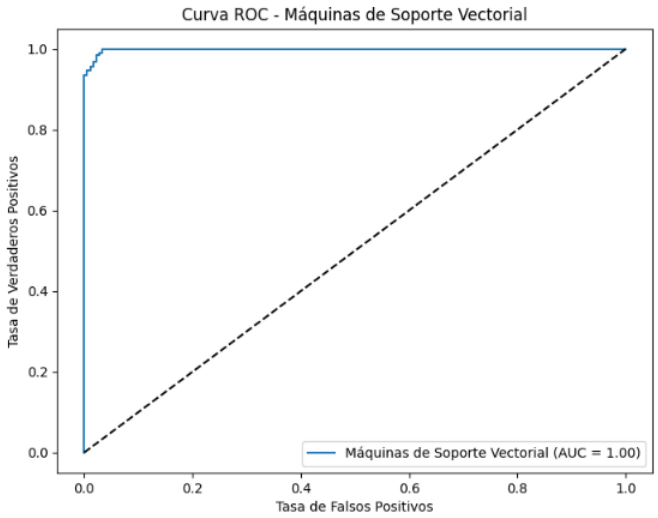

Figura 25: Curva ROC SVM

En la Figura 25 demuestra un rendimiento excepcional, con una gráfica que se acerca a la esquina superior izquierda y un AUC (Área Bajo la Curva) de 1.00, lo que refleja una capacidad perfecta para distinguir entre clases positivas y negativas. Este resultado indica que el modelo clasifica casi todas las instancias correctamente, minimizando errores significativos. Además, la baja tasa de falsos positivos y falsos negativos observada en la matriz de confusión refuerza esta efectividad, destacando al modelo como altamente preciso y confiable

4.1.5 Red Neuronal

Para la red neuronal multicapa, se evaluaron 36 combinaciones de hiperparámetros, con un total de 180 ajustes (fits). Este modelo mostró un desempeño prometedor, alcanzando una precisión del 83% y un AUC-ROC de 0.90.

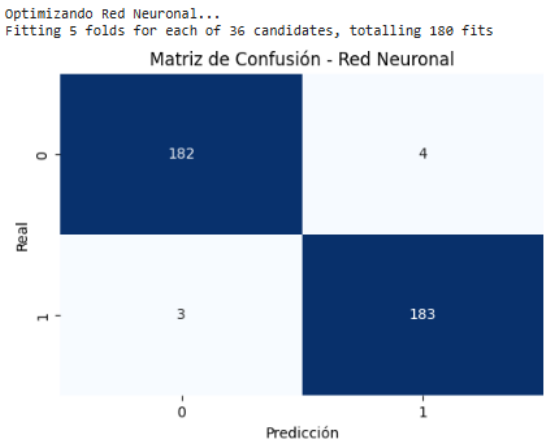

Figura 26: Matriz de Confusión Red Neuronal

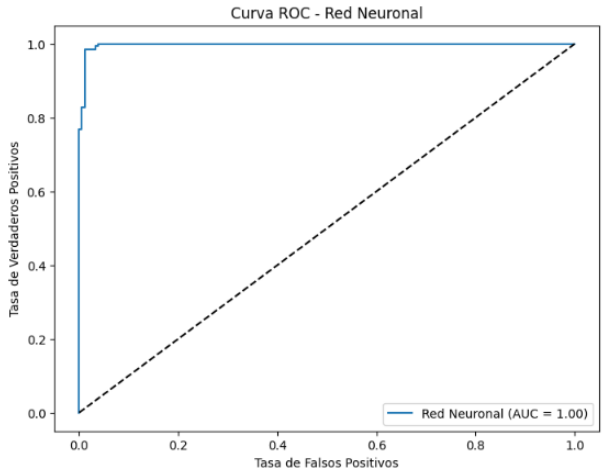

Figura 27: Curva Roc Red Neuronal

En la Figura 26 evidencia un buen desempeño, con 182 predicciones correctas para la clase 0 (negativo) y 183 para la clase 1 (positivo). Los errores son mínimos, con 4 falsos positivos (instancias negativas clasificadas como positivas) y 3 falsos negativos (instancias positivas clasificadas como negativas), lo que refleja una baja tasa de error en ambas clases. Esto indica que el modelo clasifica correctamente la mayoría de las instancias, aunque presenta ligeros errores en ambas clases.

En la Figura 27 la curva ROC del modelo de red neuronal refleja un rendimiento excepcional, con una gráfica que se aproxima a la esquina superior izquierda y un AUC (Área Bajo la Curva) de 1.00, lo que denota una capacidad perfecta para separar las clases positivas y negativas. Este resultado indica que el modelo clasifica correctamente las instancias, minimizando errores significativos. La proximidad de la curva a la esquina superior izquierda destaca la excelente

relación entre verdaderos positivos y falsos positivos, lo que confirma una alta precisión y confiabilidad en su clasificación.

4.1.6 Resultados de la evaluación aplicando métricas a los modelos de machine learning

En la Tabla 21 se resume los resultados obtenidos al evaluar los distintos modelos del algoritmo de machine learning.

Tabla 21
MÉTRICAS DE EVALUACIÓN DE MACHINE LEARNING

Model	Best Hiperparameters	Accuracy	Precision	Sensitivity	F1-Score	AUC
Random forest	**{'max_depth': 10, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 100}**	**0.994624**	**0.989362**	**1.000000**	**0.994652**	**0.999855**
Logistic Regression	{'C': 1, 'solver': 'lbfgs'}	0.970430	0.972973	0.967742	0.970350	0.997370
KNN	{'metric': 'manhattan', 'n_neighbors': 5, 'weights': 'distance'}	0.981183	0.973545	0.989247	0.981333	0.996690
Support Vector Machines	{'C': 10, 'gamma': 'scale', 'kernel': 'rbf'}	0.978495	0.978495	0.978495	0.978495	0.998815
Neuronal Network	{'activation': 'relu', 'alpha': 0.0001, 'hidden_layer_sizes': (50, 25, 10), 'solver': 'adam'}	0.981183	0.978610	0.983871	0.981233	0.997456

El modelo de Random Forest demuestra ser el más preciso de todos, con un accuracy casi perfecto (99.46%) y un AUC cercano al valor ideal (99.98%), lo que confirma su capacidad para distinguir entre clases positivas y negativas. Su sensibilidad perfecta (100%) asegura que identifica correctamente todas las instancias positivas, lo cual es crítico en aplicaciones médicas. La precisión de 98.94% refleja una baja cantidad de falsos positivos. Es ideal para escenarios donde la minimización de errores es prioritaria, como en el diagnóstico de enfermedades. El Bosque Aleatorio sobresale en Accuracy, AUC, Precision, Sensitivity y F1-Score, lo que lo posiciona como el modelo más confiable y efectivo.

El modelo de regresión logística ofrece un desempeño sólido con un accuracy de 97.04% y un AUC elevado (0.997370), mostrando una capacidad destacada para clasificar correctamente las instancias. Aunque su sensibilidad (96.77%) es ligeramente inferior a la de otros modelos, la precisión de 97.30% refleja una baja tasa de falsos positivos. Este modelo es una opción confiable y más interpretable, lo que puede ser ventajoso en contextos donde se requiere explicar las decisiones del modelo.

El modelo KNN muestra un desempeño robusto, con un accuracy del 98.12% y un AUC de 0.996690, lo que refleja su capacidad para distinguir clases con alta precisión. Su sensibilidad (98.92%) es superior a la de la regresión logística, asegurando que identifica correctamente la mayoría de las instancias positivas. La precisión de 97.35% también es alta, aunque ligeramente inferior a la de otros modelos como Random Forest. Este modelo es efectivo para tareas de clasificación, pero su desempeño puede depender del tamaño del conjunto de datos y la elección de los vecinos.

El modelo SVM ofrece un desempeño consistente en todas las métricas, con un accuracy del 97.85% y un AUC de 0.998815, lo que evidencia su capacidad para distinguir entre clases con gran eficacia. Su precisión, sensibilidad y F1-Score están equilibrados (97.85%), mostrando un rendimiento uniforme. Este modelo es ideal para problemas con datos bien distribuidos y donde se busca un equilibrio entre precisión y sensibilidad. Sin embargo, puede ser más costoso computacionalmente en comparación con otros modelos.

La red neuronal presenta un rendimiento sobresaliente con un accuracy del 98.12% y un AUC de 0.997456, lo que indica su capacidad para clasificar correctamente casi todas las instancias. Su sensibilidad (98.39%) asegura que pocas instancias positivas se clasifiquen erróneamente, mientras que su precisión (97.86%) minimiza los falsos positivos. Este modelo es especialmente potente para manejar relaciones complejas en los datos, aunque puede ser más difícil de interpretar y requiere más recursos computacionales en comparación con modelos como la regresión logística.

4.2 Evaluación de la aplicación web

Para la evaluación de la aplicación web se utiliza una encuesta presentada para evaluar el aplicativo web diseñado para la detección de diabetes tipo 2 tiene como objetivo recopilar retroalimentación detallada por parte de los médicos, quienes son los usuarios clave para el uso de la aplicación web

4.2.1 Diseño de pruebas

Para evaluar la satisfacción del usuario al utilizar el formulario para la detección de diabetes II, se evaluó la percepción del usuario con la escala de Likert (Tabla 22) [21].

Tabla 22
ESCALA DE LIKERT

Escala		
1	Rango superior (Positivos)	Nada Satisfecho
2		Poco Satisfecho
3	Neutro	Neutral
4	Rango inferior (Negativos)	Muy Satisfecho
5		Totalmente Satisfecho

4.2.2 Encuesta de satisfacción

En la tabla 23 presenta la encuesta para evaluar las experiencia y satisfacción del usuario al utilizar el formulario para la deteccion de diabetes tipo 2, se realiza encuesta para la satisfacción al cliente. Este instrumento fue aplicado a médicos especialistas, el modelo a utilizar es CSAT (Customer Satisfaction Score) el cual valoró a 16 médicos.

Tabla 23
ENCUESTA DE EVALUACIÓN DE SATISFACCIÓN

Usuario	Pregunta	Nada Satisfecho	Poco Satisfecho	Neutral	Muy Satisfecho	Totalmente Satisfecho
CSA1	Es fácil utilizar la aplicación web para deteccion de diabetes tipo 2					
CSA2	El formulario web para el ingreso de las variables relacionadas con la enfermedad de diabetes esan bien definidas					
CSA3	La aplicación web desarrollada en este trabajo es una herramienta útil para la detección de Diabetes tipo 2					
CSA4	Esta herramienta puede facilitar la toma de decisiones clínicas en el manejo de pacientes con riesgo de Diabetes tipo 2					
CSA5	El tiempo de respuesta de la aplicación es adecuado					
CSA6	La aplicación puede integrarse fácilmente en el flujo de trabajo clínico habitual					

4.2.3 Resultados de pruebas de satisfación

En la Tabla 24, se presenta los resultados de la encuesta de satisfacción aplicada a los médicos. Se tomaron en cuenta las respuestas que se encontraban dentro del rango superior (Muy Satisfecho y Totalmente Satisfecho), ya que estas reflejan la percepción positiva del cliente, se procede a dividir por el número de encuestados. De esta manera, se conoce el grado de satisfacción que tienen los usuarios, la misma que alcanzó un promedio 86.25%.

Tabla 24
FRECUENCIA ABSOLUTA DE RESPUESTAS DE USABILIDAD

Usuario	Pregunta	Nada Satisfecho	Poco Satisfecho	Neutral	Muy Satisfecho	Totalmente Satisfecho
Med1	Preg1	0	0	1	9	6
	Preg2	0	0	0	7	9
	Preg3	0	0	1	7	8
	Preg4	0	0	1	6	9
	Preg5	0	0	1	5	10
	Preg6	0	0	1	3	12

Em la Figura 28 la gráfica muestra la distribución porcentual de los niveles de satisfacción para diferentes preguntas (Preg1 a Preg6), destacando que la mayoría de las respuestas se concentran en las categorías más altas: "Muy Satisfecho" y "Totalmente Satisfecho". En general, "Neutral" se mantiene constante en un 6%, mientras que no hay respuestas en las categorías de "Nada Satisfecho" o "Poco Satisfecho", lo que indica una buena aceptación general. Preg6 tiene la mayor proporción de usuarios "Totalmente Satisfecho" (75%), seguido por Preg5 (63%) y Preg4 (56%), mientras que Preg1 y Preg2 muestran una mayor distribución equilibrada entre "Muy Satisfecho" y "Totalmente Satisfecho". Esto refleja una percepción positiva y consistente de los usuarios hacia las preguntas evaluadas.

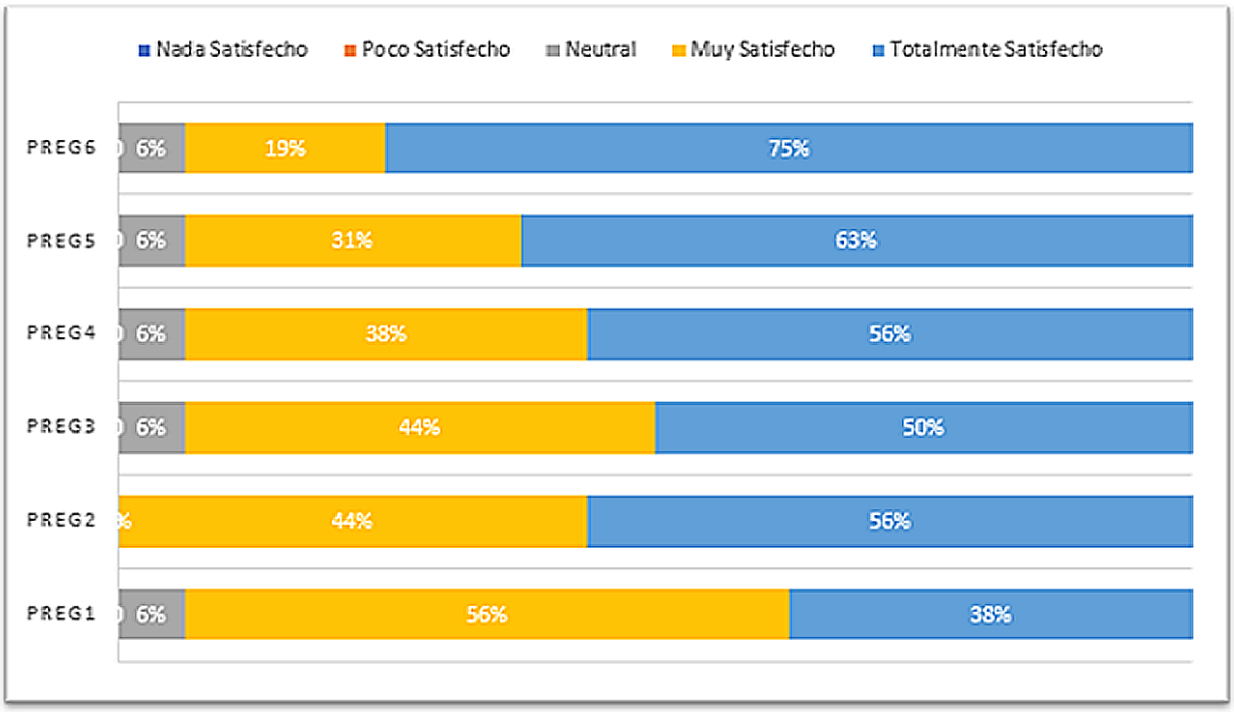

Figura 28. Porcentaje de satisfacción

4.2.4 Comprobacion de hipotesis

Para comprobar la hipótesis, se agruparon las respuestas del modelo y se seleccionó el tipo de prueba adecuado. En este caso, se aplicó una prueba de hipótesis para proporciones, enfocada en evaluar la distribución proporcional de las afirmaciones obtenidas.

Planteamiento de Hipótesis: Se busca determinar si la proporción de respuestas de "Muy Satisfecho" o "Totalmente Satisfecho" supera el 75% en las diferentes preguntas evaluadas.

Hipótesis nula (H_0): Si se desarrolla una aplicación web que integre un algoritmo de aprendizaje automático que apoye al médico especialista en el diagnóstico temprano de Diabetes Mellitus tipo 2, no se logra superar el 75% de satisfacción. ($Z \leq 0.75$).

Hipótesis alternativa (H_1): Si se desarrolla una aplicación web que integre un algoritmo de aprendizaje automático que apoye al médico especialista en el diagnóstico temprano de Diabetes Mellitus tipo 2, se logra más de un 75% de satisfacción. ($Z > 0.75$).

Selección de la Prueba y Corrección del Sesgo: Para evitar el sesgo en la selección de datos, se han considerado todas las respuestas sin descartar las opciones neutrales y negativas.

4.2.5 Diseño de pruebas para proporciones

De los resultados; se seleccionó 91 respuestas que se encuentran en el rango superior de la escala de Likert, se descartaron las estimaciones negativas (0) y neutral (5). Las respuestas máximas es 96 de todas las valoraciones como muestra en la Tabla 25.

Tabla 25
VALORACIÓN DE ACUERDO CON LA ESCALA APLICADA

ESCALA	**Rango inferior**			**Rango superior**	
	Nada Satisfecho	**Poco Satisfecho**	**Neutral**	**Muy Satisfecho**	**Totalmente Satisfecho**
RESULTADOS	0	0	5	37	54
RESPUESTAS MAXIMAS	96				
TOTAL	0		5	91	

Calculo del Estadístico de Prueba: Aplicamos la formula del estadístico Z para la prueba de hipótesis de proporciones:

$$Z = \frac{(P - Po)}{\sqrt{\frac{Po(1 - Po)}{N}}}$$

Donde:

- **N** = 96 (tamaño de la muestra)
- **P** = 0.948 (proporción de respuestas "Muy Satisfecho" y "Totalmente Satisfecho")

$$P = \frac{37 + 54}{96}$$

$$P \approx 0.948$$

- $\mathbf{P_0}$ = 0.75 (proporción propuesta)
- **α** = 0.05 (nivel de significancia)
- **Zc** = 1.64 (valor crítico para una prueba unilateral)

$$\sqrt{\frac{0.75(1-0.75)}{96}} = \sqrt{\frac{0.1875}{96}} = \sqrt{0.001953125} = 0.0442$$

$$Z = \frac{(0.948-0.75)}{0.0442} = \frac{0.198}{0.0442} = 4.48$$

Decisión de Prueba: Dado que el valor calculado del estadístico Z (4.48) es mayor que el valor crítico Zc(1.64), la prueba cae en la región de rechazo de la hipótesis nula(Ho)

Hipótesis nula (H_0): La proporción de respuestas afirmativas (P) es menor o igual a la proporción propuesta (P_o): $P_o \leq 0.75$

Hipótesis alternativa (H_1): La proporción de respuestas afirmativas (P) es mayor que la proporción propuesta (Po): P > 0.75.).

4.2.6 Decisión de prueba de hipótesis

Dado que el valor calculado del estadístico Z (4.48) es mayor que el valor crítico Zc (1.64), la prueba cae en la región de rechazo de la hipótesis nula (H_0).

Por lo tanto, se rechaza la hipótesis nula (H_0)

Se acepta la hipótesis alternativa (H_1).

Se concluye que, con un nivel de confianza del 95%, la satisfacción reportada por los usuarios supera el 75%

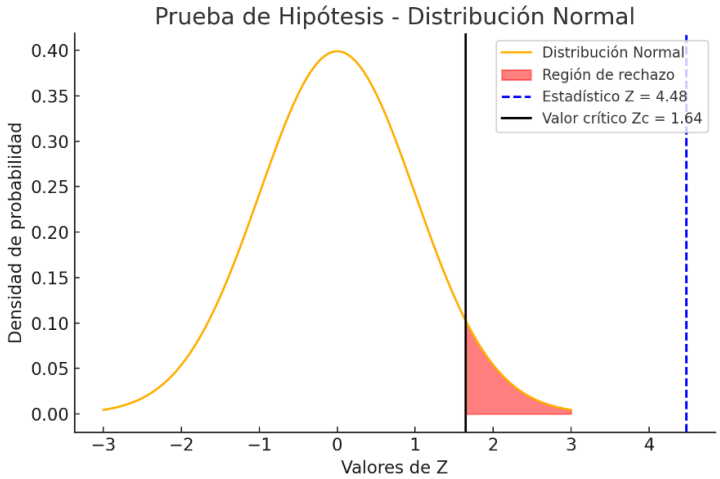

Figura: distribución normal con la región de rechazo resaltada

4.3 Discusión de Resultados

Posterior al análisis de estudios relacionados se puede destacar que: en el presente trabajo con el modelo Random Forest (RF) alcanzó una precisión de 0.9946, AUC de 0.9999, precisión de 0.9894, sensibilidad de 1.0000 y F1-Score de 0.9947.

En el análisis comparativo de modelos de machine learning para la predicción de diabetes, el modelo Random Forest se destacó como el más preciso, con un 0.9946 y una sensibilidad perfecta de 1.0, superando a otros algoritmos tanto en este estudio como en la literatura relacionada. Hennebelle et al. (2023) reportaron que RF obtuvo un 6% más de precisión que la Regresión Logística (LR), mientras que Vakil et al. (2021) encontraron que RF superó a modelos como KNN, SVM y Redes Neuronales con una precisión del 99%. Esto subraya la relevancia de RF como modelo de referencia en problemas clínicos.

Por otro lado, algoritmos como Regresión Logística, KNN, SVM y Redes Neuronales mostraron un buen desempeño, con precisiones superiores al 95%. Sin embargo, su rendimiento fue consistentemente superado por modelos ensemble como RF y XGBoost, que ofrecen mayor precisión y robustez en contextos clínicos. Estos resultados destacan la importancia de los modelos ensemble para aplicaciones en el ámbito médico, donde la exactitud y la sensibilidad son cruciales para un diagnóstico temprano y confiable.

4.3.1 Comparación de Resultados con Trabajos Relacionados

La tabla 26 presenta una comparación de las métricas de desempeño de varios modelos de Machine Learning evaluados en este estudio (*) frente a los resultados reportados por Hennebelle et al. y Vakil et al. en investigaciones previas. Los modelos incluyen Random Forest, Regresión Logística, K-Nearest Neighbors (KNN), Máquinas de Soporte Vectorial (SVM) y Redes Neuronales, evaluados según métricas clave: precisión (accuracy), área bajo la curva ROC (AUC), precisión positiva (precision), sensibilidad (sensitivity) y F1-Score. En general, los modelos implementados en este trabajo superan o son comparables con los resultados previos. Destaca el modelo de Random Forest, que logra la mejor precisión (99.46%) y un AUC cercano al valor ideal (0.9999), demostrando un rendimiento superior frente a otros enfoques, tanto en este estudio como en los de Hennebelle et al. y Vakil et al. Este nivel de rendimiento subraya la robustez y confiabilidad del enfoque adoptado en esta investigación.

Tabla 26
COMPARACIÓN DE RESULTADOS CON TRABAJOS RELACIONADOS

Modelo	Accuracy	AUC	Precision	Sensitivity	F1-Score
Random Forest(*)	0.9946	0.9999	0.9894	1.0000	0.9947
Random Forest (Hennebelle et al.)	0.970	0.992	0.965	0.975	0.970
Random Forest (Vakil et al.)	0.990	0.998	0.985	0.995	0.990
Logistic Regression(*)	0.9704	0.9974	0.9730	0.9677	0.9704
Logistic Regression (Hennebelle et al.)	0.960	0.990	0.955	0.965	0.960
Logistic Regression (Vakil et al.)	0.950	0.980	0.945	0.955	0.950
KNN(*)	0.9812	0.9967	0.9735	0.9892	0.9813
KNN (Vakil et al.)	0.970	0.992	0.965	0.975	0.970
SVM(*)	0.9785	0.9988	0.9785	0.9785	0.9785
SVM (Vakil et al.)	0.975	0.995	0.970	0.980	0.975
Redes Neuronales(*)	0.9812	0.9975	0.9786	0.9839	0.9812
Redes Neuronales (Vakil et al.)	0.980	0.996	0.975	0.985	0.980

(*) Trabajo propio

Random Forest es el modelo que muestra el mejor rendimiento; Logistic Regression, KNN, y Redes Neuronales, también son fuertes competidores, con resultados comparables, pero ligeramente por debajo del Random Forest. Cabe destacar que la data para este trabajo es propia, y fue recolectada con la colaboración de un médico especialista y 961 pacientes. Esta data se encuentra en el repositorio github (https://github.com/srcisneros/DiabetesTipoII) para la utilización libre y futuras investigaciones.

CONCLUSIONES

La revisión sistemática de literatura permitió identificar las técnicas más efectivas en el diagnóstico de la Diabetes Tipo 2. Algoritmos como Random Forest, redes neuronales artificiales, K-Nearest Neighbors, y XGBoost se destacaron por su capacidad predictiva y precisión en estudios previos. La elección de estas técnicas se fundamentó en su desempeño demostrado, simplicidad de aplicación y adaptabilidad a conjuntos de datos clínicos, lo que garantiza un diagnóstico temprano y confiable.

La base de datos diseñada incluyó variables clave como género, edad, glucosa, HbA1c, colesterol, triglicéridos, HDL, LDL, VLDL, índice de masa corporal (IMC) y antecedentes médicos relevantes. Estos datos, recopilados entre julio y diciembre de 2023 en la Clínica de Salud y Bienestar Postural, proporcionaron una base sólida para entrenar y validar los modelos de aprendizaje automático. La calidad y representatividad de los datos aseguraron la eficacia de los modelos implementados.

La aplicación de machine learning demostró ser efectiva para el diagnóstico de la Diabetes Tipo 2. El modelo Random Forest (RF) fue seleccionado para la predicción de esta enfermedad, debido a que alcanzó una precisión del 99.46%. También se obtuvo valores superiores al 98% en las métricas de desempeño, como la precisión, sensibilidad, especificidad y AUC.

El desarrollo de la aplicación web integró el modelo de machine learning más óptimo que fue el Random Forest, permitiendo una interacción fluida entre usuarios y la herramienta diagnóstica. Esta aplicación facilita a los médicos especialistas realizar diagnósticos más rápidos y precisos, mejorando la accesibilidad a tecnologías avanzadas de salud. Además, su diseño asegura la escalabilidad y adaptación a nuevos datos, lo que incrementa su utilidad en escenarios clínicos reales.

El aporte de la aplicación de CRISP-DM en este estudio radica en la posibilidad de replicar y escalar el modelo predictivo a entornos más amplios, como clínicas y hospitales. Esto garantiza que los datos utilizados sean de alta calidad y que los resultados obtenidos sean interpretables para los profesionales de la salud.

RECOMENDACIONES

Ampliar la revisión sistemática de literatura para incluir nuevas técnicas emergentes, como modelos basados en aprendizaje profundo (Deep Learning), que pueden proporcionar resultados aún más precisos con conjuntos de datos más grandes y diversos.

Considerar el uso de enfoques híbridos que combinen varios algoritmos para mejorar la precisión y la capacidad de generalización del modelo.

Implementar sistemas de recopilación de datos estandarizados en clínicas y hospitales para garantizar la calidad y consistencia de los datos utilizados en futuros modelos. Esto podría incluir herramientas digitales para registrar información clínica de manera automatizada.

Ampliar la base de datos con pacientes de diferentes edades y características demográficas para mejorar la representatividad y robustez del modelo.

Proveer capacitación a los médicos y personal de salud sobre el uso de la aplicación y la interpretación de los resultados generados por los modelos de aprendizaje automático.

Diseñar estrategias para la integración de la aplicación como herramienta complementaria en el diagnóstico médico, asegurando que no sustituya el juicio clínico, sino que lo potencie.

TRABAJOS FUTUROS

Utilizar la arquitectura y metodología desarrollada para diagnosticar otras enfermedades crónicas como hipertensión, enfermedades cardiovasculares o problemas renales, aprovechando datos clínicos similares.

Integrar técnicas de inteligencia artificial explicable (Explainable AI, XAI) para que los médicos puedan entender las razones detrás de cada diagnóstico generado por el modelo.

Diseñar colaboraciones con instituciones de salud para recopilar datos de pacientes a nivel nacional o internacional, incrementando la diversidad y representatividad de los datos utilizados.

Crear una base de datos unificada y estandarizada que pueda ser compartida entre investigadores y desarrolladores para futuros estudios.

Adaptar los algoritmos y la aplicación web para que funcionen de manera eficiente en dispositivos móviles y tablets, facilitando su uso por médicos en áreas rurales o remotas.

Realizar un seguimiento continuo del desempeño del modelo en la práctica clínica para evaluar su efectividad a lo largo del tiempo.

Incorporar dispositivos médicos portátiles, como monitores de glucosa o pulseras inteligentes, para recolectar datos en tiempo real que puedan alimentar el modelo de aprendizaje automático.

BIBLIOGRAFIA

[1] “Diabetes, la segunda enfermedad más frecuente en Ecuador”, Primicias. Consultado: el 2 de abril de 2024. [En línea]. Disponible en: https://www.primicias.ec/nota_comercial/hablemos-de/salud/habitos-saludables/diabetes-la-segunda-enfermedad-mas-frecuente-en-ecuador/

[2] J. R. B. Evia, “México y el reto de las enfermedades crónicas no transmisibles. El laboratorio también juega un papel importante”, *Rev. Mex. Patol. Clínica Med. Lab.*, vol. 65, núm. 1, pp. 4–17, jun. 2018.

[3] R. B. Ruiz y J. D. Velásquez, “Inteligencia artificial al servicio de la salud del futuro”, *Rev. Médica Clínica Las Condes*, vol. 34, núm. 1, pp. 84–91, ene. 2023, doi: 10.1016/j.rmclc.2022.12.001.

[4] P. Aschner M *et al.*, “Guía de práctica clínica para la prevención, diagnóstico, tratamiento y seguimiento de la diabetes mellitus tipo 2 en la población mayor de 18 años”, *Colomb. Médica*, vol. 47, núm. 2, pp. 109–130, jun. 2016.

[5] J. F. C. Andrade, A. E. C. Muñoz, E. W. T. Correa, y C. H. L. Rivera, “Diabetes gestacional: incidencias, complicaciones y manejo a nivel mundial y en Ecuador”, *RECIMUNDO*, vol. 3, núm. 1, Art. núm. 1, feb. 2019, doi: 10.26820/recimundo/3.(1).enero.2019.815-831.

[6] C. Rodríguez León y M. M. García Lorenzo, “ADECUACIÓN A METODOLOGÍA DE MINERÍA DE DATOS PARA APLICAR A PROBLEMAS NO SUPERVISADOS TIPO ATRIBUTO-VALOR”, *Rev. Univ. Soc.*, vol. 8, núm. 4, pp. 43–53, dic. 2016.

[7] A. Rosado-Gómez, A. Quintero-Duarte, y C. D. Meneses-Guevara, “Desarrollo ágil de software aplicando programación extrema”, *Rev. Ingenio*, vol. 5, núm. 1, Art. núm. 1, dic. 2012, doi: 10.22463/2011642X.2003.

[8] E. M. López, E. D. la C. Gámez, M. H. Hernández, M. M. Arroyo, y J. A. M. Valverde, “Avances en la Detección de Retinopatía Diabética: El Rol Prometedor de la Inteligencia Artificial”, *Cienc. Lat. Rev. Científica Multidiscip.*, vol. 8, núm. 1, pp. 5744–5756, mar. 2024, doi: 10.37811/cl_rcm.v8i1.9925.

[9] C. P. Oviedo y J. S. Viteri, “Pregunta de investigación y estrategia PICOT”, *Medicina (Mex.)*, vol. 19, núm. 1, Art. núm. 1, nov. 2015, doi: 10.23878/medicina.v19i1.647.

[10] G. Batista-Mendoza, E. J. C. Herrera, y G. Cedeño-Batista, “Machine learning aplicado al análisis de un set de datos de parámetros ambientales en galpones de pollos de engorde”, *Visión Antataura*, vol. 7, núm. 2, Art. núm. 2, dic. 2023, doi: 10.48204/j.vian.v7n2.a4566.

[11] P. Llorens-Vernet y J. Miró, “The Mobile App Development and Assessment Guide (MAG): Delphi-Based Validity Study”, *JMIR MHealth UHealth*, vol. 8, núm. 7, p. e17760, jul. 2020, doi: 10.2196/17760.

[12] V. G. Pacheco, “Una Breve Historia del Machine Learning”, Telefónica Tech. Consultado: el 3 de abril de 2024. [En línea]. Disponible en: https://telefonicatech.com/blog/una-breve-historia-del-machine-learning

[13] J. A. Mejía, M. A. Oviedo-Benálcazar, J. A. Ordoñez, y J. F. Valencia, “Aprendizaje automático aplicado a la predicción de diabetes mellitus, utilizando información socioeconómica y ambiental de usuarios del sistema de salud”, *Rev. Fac. Nac. Salud Pública*, vol. 41, núm. 2, p. e351168, mar. 2023, doi: 10.17533/udea.rfnsp.e351168.

[14] U. Fayyad, G. Piatetsky-Shapiro, y P. Smyth, “From Data Mining to Knowledge Discovery in Databases”, *AI Mag.*, vol. 17, núm. 3, Art. núm. 3, mar. 1996, doi: 10.1609/aimag.v17i3.1230.

[15] D. Dickey, “337-2012: Introduction to Predictive Modeling with Examples”, 2012.

[16] S. Shalev-Shwartz y S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*, 1a ed. Cambridge University Press, 2014. doi: 10.1017/CBO9781107298019.
[17] J. S. Ruiz, *La Diabetes Mellitus como enfermedad sistémica: Control global del riesgo cardiometabólico*. Ediciones Díaz de Santos, 2012.
[18] J. M. Vegas Valle, "Diseño e implementacion de nuevas herramientas para el diagnóstico de la diabetes mellitus", http://purl.org/dc/dcmitype/Text, Universidad de Oviedo, 2019. Consultado: el 3 de abril de 2024. [En línea]. Disponible en: https://dialnet.unirioja.es/servlet/tesis?codigo=260844
[19] K. Carranza *et al.*, "Aspectos celulares y moleculares de la nefropatía diabética, rol del VEGF-A", *Nefrol. Madr.*, vol. 35, núm. 2, pp. 131–138, 2015.
[20] O. Iparraguirre-Villanueva, K. Espinola-Linares, R. O. Flores Castañeda, y M. Cabanillas-Carbonell, "Application of Machine Learning Models for Early Detection and Accurate Classification of Type 2 Diabetes", *Diagnostics*, vol. 13, núm. 14, Art. núm. 14, ene. 2023, doi: 10.3390/diagnostics13142383.
[21] L. E. Pérez Leal y J. A. Buitrago C´ardenas, "Predicción del diagnostico de diabetes a partir de perfiles clínicos de pacientes utilizando aprendizaje automático", *InstnameUniversidad Antonio Nariño*, sep. 2021, Consultado: el 3 de abril de 2024. [En línea]. Disponible en: http://repositorio.uan.edu.co/handle/123456789/4816
[22] D. A. Ordóñez Barrios y E. R. Vizcarra Infantes, "Modelo Predictivo para el diagnóstico de la Diabetes Mellitus Tipo 2 soportado por SAP Predictive Analytics", *Univ. Peru. Cienc. Apl. UPC*, jul. 2018, doi: 10.19083/tesis/624417.
[23] M. Rodríguez-Leyton y M. Charris, "Factores de riesgo de diabetes mellitus tipo 2 en población adulta. Barranquilla, Colombia", *Rev. Soc. Colomb. Endocrinol.*, vol. 6, pp. 86–91, may 2019.
[24] V. G. Pacheco, "Una Breve Historia del Machine Learning", Telefónica Tech. Consultado: el 3 de abril de 2024. [En línea]. Disponible en: https://telefonicatech.com/blog/una-breve-historia-del-machine-learning
[25] A. T. Norman, *Aprendizaje Automático En Acción*. Litres, 2019.
[26] E. C. Moreno y J. B. Sanz, *Manual práctico de inteligencia artificial en entornos sanitarios*. Elsevier Health Sciences, 2023.
[27] S. Chatterjee y A. S. Hadi, *Regression Analysis by Example*. John Wiley & Sons, 2006.
[28] R. Genuer y J.-M. Poggi, *Random Forests with R*. Springer Nature, 2020.
[29] C. E. C. Figueroa y H. G. Chávez, "Diseño de algoritmo compuesto por Machine Learning y un modelo probabilístico para la detección de diabetes", *Mem. Congr. Nac. Ing. Bioméd.*, vol. 8, núm. 1, Art. núm. 1, nov. 2021.
[30] L. Viveros-Rosas, R. Díaz-Téllez, J. R. Pérez-Torres, M. L. Chew-Hernández, y J. A. Vega-González, "Clasificador Naive como herramienta de decisión en la contratación de personal académico del TESCo", *J. Objetos Objet. Matemáticos*, núm. 7, pp. 29–33, ene. 2023.
[31] A. Anjum, P. Agbaje, S. Hounsinou, y H. Olufowobi, "In-Vehicle Network Anomaly Detection Using Extreme Gradient Boosting Machine", en *2022 11th Mediterranean Conference on Embedded Computing (MECO)*, jun. 2022, pp. 1–6. doi: 10.1109/MECO55406.2022.9797224.
[32] M. Sheykhmousa, M. Mahdianpari, H. Ghanbari, F. Mohammadimanesh, P. Ghamisi, y S. Homayouni, "Support Vector Machine Versus Random Forest for Remote Sensing Image Classification: A Meta-Analysis and Systematic Review", *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 13, pp. 6308–6325, 2020, doi: 10.1109/JSTARS.2020.3026724.

[33] J. L. Sarmiento-Ramos, “Aplicaciones de las redes neuronales y el deep learning a la ingeniería biomédica”, *Rev. UIS Ing.*, vol. 19, núm. 4, pp. 1–18, jun. 2020, doi: 10.18273/revuin.v19n4-2020001.

[34] F. L. Badillo, C. A. R. Hernández, B. M. Narváez, y Y. E. A. Trillos, “Redes neuronales convolucionales: un modelo de Deep Learning en imágenes diagnósticas. Revisión de tema”, *Rev. Colomb. Radiol.*, vol. 32, núm. 3, Art. núm. 3, sep. 2021, doi: 10.53903/01212095.161.

[35] “Aplicación de técnicas de Procesamiento del Lenguaje Natural para clasificar conductas docentes”. Consultado: el 10 de diciembre de 2024. [En línea]. Disponible en: https://repositorio.uloyola.es/handle/20.500.12412/6180

[36] I. G. R. Gavilán, “Metodología para Machine Learning (III): SEMMA”, Ignacio G.R. Gavilán. Consultado: el 3 de abril de 2024. [En línea]. Disponible en: https://ignaciogavilan.com/metodologia-para-machine-learning-iii-semma/

[37] C. Elkan, “Predictive analytics and data mining”.

[38] A. Bellido-Zapata, J. E. Ruiz-Muggi, E. R. Neira-Sánchez, y G. Malaga, “Implementación y aplicación de la ‘Guía de práctica clínica para el diagnóstico, tratamiento y control de la diabetes mellitus tipo 2 en el primer nivel de atención’ en una red de establecimientos de salud públicos de Lima”, *ACTA MEDICA Peru.*, vol. 35, núm. 1, pp. 14–19, jun. 2018, doi: 10.35663/amp.2018.351.497.

[39] N. Phaswana-Mafuya *et al.*, “Self-reported prevalence of chronic non-communicable diseases and associated factors among older adults in South Africa”, *Glob. Health Action*, vol. 6, p. 10.3402/gha.v6i0.20936, sep. 2013, doi: 10.3402/gha.v6i0.20936.

[40] M. G. Sánchez Trujillo y J. Á. Pérez Hernández, “Metodología CRISP-DM en la gestión de proyecto de Data Mining. Caso enfermedades dermatológicas”, oct. 2023, Consultado: el 10 de diciembre de 2024. [En línea]. Disponible en: http://hdl.handle.net/10882/13087

[41] “Métodos científicos y su aplicación en la investigación pedagógica. - ProQuest”. Consultado: el 10 de diciembre de 2024. [En línea]. Disponible en: https://www.proquest.com/openview/d703ea54b2cf3475c2efcf3bfd17de98/1?pq-origsite=gscholar&cbl=4400984

[42] “Scrumban/XP: Propuesta para mejorar la eficiencia de la gestión de proyectos ágiles en el desarrollo de software - ProQuest”. Consultado: el 10 de diciembre de 2024. [En línea]. Disponible en: https://www.proquest.com/openview/629e1e39dbb4ddec7e004ead26cb0808/1?pq-origsite=gscholar&cbl=1006393

[43] W. Forero-Corba y F. N. Bennasar, “Diseño y simulación de un modelo de predicción para la evaluación de la competencia digital docente usando técnicas de Machine Learning”, *Edutec Rev. Electrónica Tecnol. Educ.*, núm. 89, Art. núm. 89, sep. 2024, doi: 10.21556/edutec.2024.89.3201.