



**Sistema WEB para predicción de rendimiento académico utilizando modelos de
Machine Learning para estudiantes de EGB elemental y media.**

Osorio Trávez, Carlos Fabián

Departamento de Ciencias de la Computación

Carrera de Ingeniería de Sistemas e Informática

Trabajo de titulación, previo a la obtención del título de Ingeniero en Sistemas e
Informática

Ing. Ñacato Caiza, German – Msc.

15 de agosto del 2022

Resultado de Verificación de Contenido



TESIS_OSORIO TRÁVEZ CARLOS FABIÁN.docx
Scanned on: 21:30 August 15, 2022 UTC



Identical Words	211
Words with Minor Changes	52
Paraphrased Words	139
Omitted Words	966

Certificación



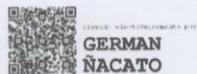
Departamento de Ciencias de la Computación

Carrera de Ingeniería de Sistemas e Informática

Certificación

Certifico que el trabajo de titulación: **“Sistema WEB para predicción de rendimiento académico utilizando modelos de Machine Learning para estudiantes de EGB elemental y media”** fue realizado por el señor **Osorio Trávez, Carlos Fabián**, el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizada en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Sangolquí, 15 de agosto del 2022.



.....
Ing. Nacato Caiza, German

C. C.: 1709217150

Responsabilidad de Autoría



Departamento de Ciencias de la Computación

Carrera de Ingeniería de Sistemas e Informática

Responsabilidad de Autoría

Yo, **Osorio Trávez, Carlos Fabián**, con cédula de ciudadanía N° 1715077473, declaro que el contenido, ideas y criterios del trabajo de titulación: **“Sistema WEB para predicción de rendimiento académico utilizando modelos de Machine Learning para estudiantes de EGB elemental y media”** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangolquí, 15 de agosto del 2022

Osorio Trávez, Carlos Fabián

C.C.: 1715077473

Autorización de Publicación



Departamento de Ciencias de la Computación

Carrera de Ingeniería de Sistemas e Informática

Autorización de Publicación

Yo **Osorio Trávez, Carlos Fabián**, con cédula de ciudadanía n° 1715077473, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: "**Sistema WEB para predicción de rendimiento académico utilizando modelos de Machine Learning para estudiantes de EGB elemental y media**" en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Sangolquí, 15 de agosto de 2022

Osorio Trávez, Carlos Fabián

C.C.: 1715077473

Dedicatoria

Dedico este trabajo a mi Padre Celestial con mucho amor, porque sin su amor incondicional, ayuda y cuidado no podría seguir cada día.

A mis padres Carlos y Yolanda, quienes son el pilar fundamental para ser la persona que soy, por ese apoyo incondicional, lo cual me ayuda a seguir adelante.

A mi esposa Mayra, mi hijo Carlitos y mi familia que ha sido mi motivación para no rendirme.

Agradecimiento

Agradezco a Dios por su bendición, darme fortaleza y sabiduría para seguir adelante.

A mi familia por ser un estímulo constante.

También agradecer mi Universidad ESPE, mis profesores del Departamento de Ciencias de la Computación, a mi director de tesis German Ñacato por su paciencia, tiempo y dedicación.

Tabla de Contenidos

Contenido

Resultado de Verificación de Contenido	2
Certificación	3
Responsabilidad de Autoría	4
Autorización de Publicación	5
Dedicatoria	6
Agradecimiento	7
Tabla de Contenidos	8
Resumen	16
Abstract.....	17
CAPITULO I.....	18
Introducción del Proyecto de Investigación.....	18
Introducción	18
Identificación del Problema	20
Justificación	21
Objetivos	24
Objetivo General.....	24

Objetivos Específicos	24
Alcance del proyecto	24
Factibilidad.....	25
Factibilidad Técnica.....	25
Factibilidad Tecnológica	26
Requerimiento de Hardware	26
Requerimiento de Software.....	27
Factibilidad Económica	27
Factibilidad Operativa.....	29
Factibilidad Operacional.....	29
Factibilidad Legal.....	29
Factibilidad Ética	30
CAPITULO II.....	31
Marco Teórico.....	31
Modelo Educativo Constructivista	31
El Aprendizaje Centrado en la Persona-Colectivo.....	32
Identificación de la Muestra.....	32
Proceso de Minería de Datos.....	34

Metodología KDD	34
Comprensión Del Dominio De Estudio Y Establecimiento De Objetivos	36
Selección De Datos	36
Data Training	37
Data Testing	38
Método De Retención.....	38
Validación Cruzada	39
Limpieza y Preprocesamiento de Datos	42
Selección De La Tarea Apropriada De Minería De Datos	42
Elección del Algoritmo de Minería de Datos	43
Aplicación del Algoritmo	43
Interpretación De Patrones Minados.....	43
Aplicación	44
Modelos de Machine Learning	44
Modelos Supervisados	44
Modelos no Supervisados	45
Python.....	46
Anaconda.....	46

Ingeniería de Software	47
Software.....	48
Ingeniería De Software.....	48
Pasos en el Proceso De Análisis	49
Ciclo De Vida Del Software	49
Angular JS	50
Microservicios	51
Microservicios ASP NET CORE.....	52
CAPITULO III.....	55
CONSTRUCCIÓN	55
Selección de la Muestra	55
Comprensión del Dominio de Estudio y Establecimientos de Objetivos.....	55
Levantamiento de Proceso Módulo de Predicción Académica	55
Determinación de Objetivos	59
Selección de los Datos	60
Division Training Dataset y Test Dataset.....	60
Interpretación y Evaluación	63
Prueba del Modelo	72

Arquitectura de la Solución	73
UX Diseño Hipermedial	74
Pruebas Y Despliegue Del Microservicios.....	85
Construcción Servicio Rest predicción	87
Construcción aplicación web Angular. - Predictor	90
CAPITULO IV	93
Evaluación	93
CAPITULO V	98
Conclusiones	98
Recomendaciones.....	99
Bibliografía	100

Índice de Tablas

Tabla 1 <i>Proyección de costos</i>	27
Tabla 2 <i>Equipo</i>	29
Tabla 3. <i>Ejemplo Validación Cruzada</i>	40
Tabla 4 <i>Ejemplo Validación Cruzada</i>	41
Tabla 5 <i>Cuadro análisis de dominio proceso evolución académica</i>	56
Tabla 6 <i>Tabla de Funcionalidad</i>	75
Tabla 7 <i>Métodos Api Rest</i>	84
Tabla 8 <i>Archivo Docker</i>	88
Tabla 9 <i>Archivo Requeriments</i>	89
Tabla 10 <i>Evaluación 1</i>	94
Tabla 11 <i>Evaluación 2</i>	94
Tabla 12 <i>Evaluación 3</i>	95
Tabla 13 <i>Evaluación 4</i>	96

Índice de Figuras

Figura 1 <i>Selección de la Muestra</i>	33
Figura 2 <i>Proceso Metodología KDD</i>	35
Figura 3 <i>Data Training</i>	37
Figura 4 <i>Division de DataSet Holdout Method</i>	39
Figura 5 <i>Fase SDLC</i>	50
Figura 6 <i>API Rest</i>	52
Figura 7 <i>Arquitectura de microservicios de Microsoft</i>	53
Figura 8 <i>Microservicios</i>	54
Figura 9 <i>Carga de archivo en Jupyter</i>	61
Figura 10 <i>División de Dataset Training y Dataset Testing</i>	61
Figura 11 <i>Training Dataset procesamiento y transformación.</i>	62
Figura 12 <i>Regresión Lineal</i>	64
Figura 14 <i>Regresión Lineal, modelo</i>	65
Figura 15 <i>Regresión Lineal, prediccion.</i>	66
Figura 16 <i>Regresión Lineal, calculo de error.</i>	67
Figura 17 <i>Codigo Regresión Lineal Multiple</i>	69
Figura 18 <i>Entrada y Salida Regresión Lineal Multiple.</i>	70

Figura 19 <i>Código Rergresion Lineal Multiple</i>	70
Figura 20 <i>Regresión Lineal Multiple, Variables Predictoras</i>	71
Figura 21 <i>Prueba de porcentaje de error</i>	72
Figura 22 <i>Arquitectura</i>	73
Figura 23 <i>Fases del sistema</i>	78
Figura 24 <i>Modelo Relacional</i>	79
Figura 25 <i>Patrón MVC API REST</i>	81
Figura 26 <i>Estructura proyecto API-REST</i>	82
Figura 27 <i>Ejemplo de salida metodo HTTP POST Microservicio</i>	83
Figura 28 <i>Postman</i>	86
Figura 29 <i>Despliegue de microservicio en IIS Local</i>	86
Figura 30 <i>Servicio Python predicción rendimiento</i>	87
Figura 31 <i>Despliegue servicio python en Docker</i>	88
Figura 32 <i>Contenedor Docker Servicio Python</i>	90
Figura 33 <i>Web Angular</i>	91

Resumen

El proyecto de investigación tiene como objetivo predecir de forma temprana el rendimiento a los estudiantes que según sus evaluaciones académicas presenten problemas en la asimilación de conocimientos impartidos en clase, utilizando un modelo de Machine Learning y de esta forma dotar de información valiosa al área de psicología y al área docente para que en forma coordinada y temprana se puedan ejecutar acciones que permitan al alumno cumplir sus metas académicas. Para determinar los procesos se realizó una visita al centro educativo y entrevistas con las autoridades, establecer los procesos necesarios para la predicción del rendimiento académico, además se analizó el modelo de Machine Learning y la metodología de minería de datos que se utilizó en la implementación del sistema demo.

Por consiguiente, la muestra con la cual se va a desarrollar en este proyecto corresponde a los alumnos de cualquier género que estén cursando el tercer año de educación básica y cuyas materias asignadas comprendan inglés y matemáticas.

Es por ese motivo que este proyecto presenta un demo y se desarrolló conjugando el uso de tecnologías Web y modelo Machine Learning.

La aplicación o sistema web emitirá una predicción de rendimiento académico del estudiante generando una alerta cuando el resultado no esté calificado en aprobar la materia por tales fallas en el conocimiento.

Palabras clave: rendimiento académico, machine learning, metodología KDD.

Abstract

The objective of the research project is to predict early the performance of students who, according to their academic evaluations, present problems in the assimilation of knowledge taught in class, using a Machine Learning model and in this way provide valuable information to the area of psychology. and the teaching area so that actions can be carried out in a coordinated and early manner that allow the student to meet their academic goals. To determine the processes, a visit to the educational center and interviews with the authorities were carried out, the necessary processes for the prediction of academic performance were established, in addition, the Machine Learning model and the data mining methodology used in the implementation of the study were analyzed demo system.

Therefore, the sample with which this project will be developed corresponds to students of any gender who are in the third year of basic education and whose assigned subjects include English and mathematics.

It is for this reason that this project presents a demo and was developed by combining the use of Web technologies and the Machine Learning model.

The application or web system will issue a prediction of the student's academic performance, generating an alert when the result is not qualified to pass the subject due to such knowledge failures.

Key words: academic performance, machine learning, methodology KDD.

CAPITULO I

Introducción del Proyecto de Investigación

Introducción

La Unidad Educativa los HELECHOS fundada en el año 2015 bajos los valores de integridad, respeto, desarrollo integral del conocimiento físico y académico. Desde sus inicios ha puesto al servicio de la comunidad una oferta educativa estrechamente ligada a los cambios sociales; ubicada en el sector del Valle de los Chillos, Isla Pinta y Av. General Rumiñahui, inicialmente cuenta con los siguientes niveles de educación desde Guardería a Décimo de Básica, en cada nivel se pone un énfasis especial en el desarrollo y aprendizaje de cada materia.

En su comienzo muchas mejoras se han implementado a nivel de procesos educativos, las cuales van alineadas con el objetivo a largo plazo de la institución de convertirse en un referente educativo en el medio, que innova tanto modelos educativos como herramientas tecnológicas que permitan apoyar a las diferentes áreas para que los alumnos puedan desarrollar sus habilidades de manera integral y a su vez alcanzar sus metas.

La institución alinea el cumplimiento de sus metas a la aplicación de la metodología constructivista en la cual los estudiantes ponen en práctica los conocimientos adquiridos al final de cada módulo, ya sea mediante la ejecución de algún proyecto integrador o la toma de alguna prueba académica que ponga de manifiesto el nivel de asimilación del conocimiento que cada uno de los estudiantes que ha adquirido durante el desarrollo de los temas en el módulo académico.

Por otro lado, el modelo académico de enseñanza está dividido en dos quimestres, en cada uno de los quimestres al estudiante se le evalúa tres veces, esto implica que durante el quimestre en cuestión se pueden consignar tres notas asociadas a: tareas, exposiciones, lecciones orales / escritas, pruebas y/o examen.

Durante los últimos años la institución ha tomado la iniciativa de profundizar el uso de las TI con el único objetivo de dotar de información relevante a las diferentes áreas que componen la institución, para que éstas puedan apalancar su accionar en datos que coadyuven al cumplimiento de los objetivos institucionales.

Dentro de este contexto se ha decidido dar el paso inicial con la construcción de un sistema que haga uso de las herramientas de Machine Learning. En el transcurso de las últimas décadas, se han encontrado soluciones en esta amplia tarea de la Inteligencia Artificial que es definido como una disciplina en el campo a través de algoritmos que dota a los ordenadores la capacidad de identificar patrones en datos masivos y elaborar predicciones. (Hawkins, 2004)

Según González. A. Machine Learning es una disciplina científica del ámbito de la Inteligencia Artificial que crea sistemas que aprenden automáticamente. Aprender en este contexto quiere decir identificar patrones complejos en millones de datos a través de un algoritmo que revisa esos datos y es capaz de predecir comportamientos futuros de forma automática, también en este contexto, implica que estos sistemas se mejoran de forma autónoma con el tiempo, sin intervención humana en virtud de la calidad de los datos que reciben como entrada. (González , 2021)

Ahora bien, teniendo definiciones que dieron origen al Machine Learning, se puede enfocar a lo que sucede con él en la actualidad, éste sigue siendo muy importante en campos como la informática y el sector empresarial donde su principal función es la de manejar los grandes volúmenes de información que generan todos los días dichos campos.

Identificación del Problema

Durante el ciclo de permanencia de un alumno en la institución desde que inicia su vida educativa, está sujeto a adherirse a una metodología y modelo educativo aplicados en la institución, esta metodología denominada constructivista conlleva la necesidad de poner en práctica los conocimientos adquiridos al final de cada módulo, ya sea mediante la ejecución de un proyecto integrador o por medio de un examen que va a ponderar de forma numérica el nivel de asimilación sobre los temas académicos impartidos en clase, los cuales conllevan una evaluación mensual de los contenidos impartidos en las materias de matemáticas e inglés. Mientras el alumno va escalando desde los niveles inferiores se hace necesario direccionar los conocimientos adquiridos de cada materia hacia temas y conceptos más complejos, para ello es necesario que el alumno haya asimilado de forma exitosa la información impartida en clases en los periodos precedentes.

Es por ello que se hace difícil determinar cuándo un alumno ha llegado con vacíos en las áreas antes mencionadas a los niveles superiores, ocasionando un verdadero inconveniente a las áreas docente y psicológica del plantel, que al momento de darse estos problemas en los niveles superiores se hace más difícil aplicar correctivos y llenar vacíos que permitan al estudiante explotar todo tu potencial intelectual y lograr sus metas. Según información proporcionada por la directora de la institución muchos de los alumnos que llegan a los niveles superiores lo hacen con vacíos en las áreas antes mencionadas, ocasionando con

esto deserción escolar, incumplimiento de los objetivos institucionales, inconvenientes entre las diferentes áreas y baja en los ingresos de la institución.

En consecuencia, el problema principal que afronta la institución es la falta de recurso humano para poder procesar grandes volúmenes de información y analizar el rendimiento de cada estudiante y, ante la presencia de alguna alerta generar de forma temprana acciones que permitan reducir el elevado índice de alumnos que llegan a los niveles superiores con vacíos académicos.

Por consiguiente, se hace necesario hacer uso de las herramientas de Machine Learning para generar un modelo que permita procesar información y emitir alertas tempranas a las respectivas áreas.

Este sistema pretende predecir el rendimiento académico de los estudiantes utilizando un modelo de Machine Learning basado en las notas de evaluación que obtienen los alumnos en las materias de inglés y matemáticas de la Unidad Educativa los HELECHOS.

Justificación

Alineada a la mejora continua de sus procesos, la institución ha creado un proceso denominado “Evolución Académica” cuyo objetivo principal es lograr la coordinación entre las áreas docente y psicológica para evaluar de forma coordinada el rendimiento de los alumnos y de ser necesario generar acciones que permitan cerrar los vacíos existentes en cada área de conocimiento. Este nuevo proceso se ha encontrado durante su primer año de implementación con un gran volumen de información física que necesita ser procesada y que va creciendo en virtud de cada nuevo periodo académico.

Este proceso manual que actualmente realiza “evolución académica”, tiene como consecuencia que se desconozcan y se dejen a un lado los nuevos casos generados en los niveles inferiores e iniciales que se convertirán en deuda y trabajo acumulado e impedirá que la “evolución académica” genere valor para la institución educativa

Ante lo anteriormente expuesto se puede asegurar que el uso del Machine Learning en el campo de la educación está tomando cada vez más auge, apoyando de forma significativa en cada uno de los procesos que conforman la cadena de valor educativa.

Finalmente, en este proyecto se va a construir un sistema Web que implemente un modelo de Machine Learning para predecir de forma temprana el rendimiento a los estudiantes que según sus evaluaciones académicas presenten problemas en la asimilación de conocimientos impartidos en clase. De esta forma se dotó de información valiosa al área de psicológica y al área docente para que en forma coordinada y temprana se puedan ejecutar acciones que permitan al alumno cumplir sus metas académicas.

Considerando la cantidad de alumnos a evaluar, se debe crear un sistema web para predecir el rendimiento de las materias Inglés y Matemáticas de los estudiantes, esto ya antes mencionado se necesitará un tiempo de 6 meses para construir el sistema requerido.

Para efectos de este proyecto se ha establecido que la población muestral corresponde a todos los alumnos de todos los niveles de la Unidad Educativa los Helechos. A partir de esta definición se selecciona un subconjunto representativo de la población muestral que se denomina Muestra Aleatoria Simple.

Por consiguiente, la muestra con la cual se va a desarrollar en este proyecto corresponde a los alumnos de cualquier género que estén cursando el tercer año de educación básica y cuyas materias asignadas comprendan inglés y matemáticas.

En este contexto el sistema va a ser creado y desarrollado conjugando el uso de tecnologías Web y modelo Machine Learning.

La aplicación o sistema web emitirá una predicción de rendimiento académico del estudiante generando una alerta cuando el resultado no esté calificado en aprobar la materia por tales fallas en el conocimiento.

Objetivos

Objetivo General

Construir un sistema Web mediante modelos de Machine Learning para predecir el rendimiento académico en función a las notas correspondientes a pruebas parciales, tareas y exámenes de evaluación que obtienen los alumnos en las materias de inglés y matemáticas de la Unidad Educativa los HELECHOS.

Objetivos Específicos

- Investigar modelos de Machine Learning utilizando Python.
- Profundizar el uso de la metodología KDD
- Diseñar y construir una aplicación Web que integre un modelo de Machine Learning dentro de su funcionalidad.
- Evaluar los resultados obtenidos de la predicción de notas en el sistema demo.

Alcance del proyecto

El proyecto contará con un sistema web que haga uso de un modelo de Machine Learning para predecir el rendimiento académico basado en las notas de evaluación que obtienen los alumnos en las materias de inglés y matemáticas durante los dos quimestres del periodo escolar.

En este contexto el sistema va a hacer creado y dividido usando:

- ✓ El modelo de base de datos SQL que apalanquen el funcionamiento del sistema Web,
- ✓ Data Training y Data Sets,

- ✓ El módulo de Machine Learning: Este módulo se va a construir usando Python como lenguaje de programación,
- ✓ Middleware: Se va a construir servicios REST que serán consumidos desde la aplicación Web,
- ✓ Aplicación Web.

Para efectos del Data Training se hará uso de Data Sets en extensión .csv con una estructura inicial compuesta por los siguientes campos: Nombres y Apellidos, Curso, Año, Materia, Profesor, Dificultad (Baja- Media- Intermedia- Alta), Predicción u Observación de los resultados, siendo esta susceptible a modificación en virtud de las necesidades del proyecto durante el proceso de limpieza de datos.

Por otro lado, el sistema web contará con la integración de tecnologías Angular, Python y Bases de Datos SQL, que permitan realizar una experiencia única al usuario final.

Factibilidad

Factibilidad es la disponibilidad de realizar las tareas, objetivos o metas; que se va a cumplir en un proyecto independientemente del tema que sea planteado, tomando en cuenta los recursos con los que se cuenta para su realización.

Factibilidad Técnica

En la parte técnica se hace indispensable adquirir las siguientes habilidades para este proyecto:

- Machine Learning
- Bases de datos

- Modelos de machine learning
- Programación Web
- Python
- Java o Net Core microservicios

Factibilidad Tecnológica

Será necesario tener los siguientes recursos tecnológicos para el desarrollo del proyecto:

- Sistema Operativo Windows 10 o superior
- Software utilitario Office 2019
- Visual Studio Code
- Python
- Motores de Base de datos SQL Server
- Anaconda
- Conexión a Internet

Requerimiento de Hardware

Los requerimientos mínimos para el desarrollo de este sistema son los siguientes:

- Computar Intel Core I7 con sistemas operativo Windows 10 o superior
- Impresora Epson Multifuncional
- Flash memory 32gb

Requerimiento de Software

Los requerimientos mínimos para el desarrollo de este sistema son los siguientes:

- Intelligent ID
- SQL server express
- Visor Studio Code
- Postman

Factibilidad Económica

Para la ejecución del proyecto de investigación se necesita un presupuesto aproximado de \$5963 dólares. Al ser un proyecto de titulación, dicho valor será asumido en su totalidad por el estudiante.

Tabla 1

Proyección de costos

PRESUPUESTO			
CANTIDAD	DESCRIPCIÓN	UNITARIO	TOTAL
	Hardware	V.	V.

	Computar Intel core I7 con sistemas			
1	operativo Windows 10 o superior	\$2000	\$2000	
1	Impresora Epson Multifuncional	\$350	\$350	
1	Flash Memory de 32gb	\$30	30\$	
1	Disco Duro de 1tb	115\$	115\$	
Talento Humano				
	Horas de Investigación, Horas de			
96	Desarrollo, Horas de Pruebas.	\$8	\$768	
60	Horas extras	\$5	\$300	
Otros				
6 meses	Internet	\$30	\$180	
COSTOS VARIABLES- Transporte				
				\$8
Mensual	Gasolina	\$80	0	
				\$1
	Resma de Hojas- Tinta- Cartuchos	\$120	20	
				\$2
10	Cd imprevistos	\$2	0	

			\$2
1	Imprevistos	\$2000	000
			Total: 5.963\$

Nota: Se detallan los siguientes rubros o gastos que intervienen en el proceso de la elaboración de este proyecto.

Factibilidad Operativa

Se cuenta con el siguiente equipo para la realización del proyecto:

Tabla 2

Equipo

Estudiante	Carlos Fabián Osorio Trávez
Beneficiario	Unidad educativa los Helechos

Factibilidad Operacional

Dado el escenario económico actual producto de la pandemia del COVID 2019 donde muchos alumnos han desertado de las instituciones y algunos han pasado al sector público, el presente proyecto se presentará como una demo es decir, se quedará en ambiente de desarrollo y se evaluará con el auspiciante los resultados obtenidos dejando la posibilidad abierta para que más adelante se implemente en un ambiente de producción.

Factibilidad Legal

El proyecto utilizara tecnologías de código abierto y no infringe ninguna norma local municipal o estatal por lo tanto es factible y se aplicaran los siguientes estándares y normas:

- ✓ Python Software Foundation
- ✓ Microsoft Net Core 3.1
- ✓ OSEMN process data science
- ✓ Desing Thinking
- ✓ Web W3C
- ✓ IEEE P2830™

Factibilidad Ética

Mediante la carta de auspicio, la institución se compromete a proporcionar toda la información necesaria para la ejecución del proyecto, consecuentemente es viable y los avances periódicos, así como los resultados finales, se los entregara a la directora de la institución.

CAPITULO II

Marco Teórico

Modelo Educativo Constructivista

Según Caisa. F. menciona que el modelo del constructivismo o perspectiva radical que concibe la enseñanza como una actividad crítica y al docente como un profesional autónomo que investiga reflexionando sobre su práctica, para el constructivismo aprender es arriesgarse a errar (ir de un lado a otro), muchos de los errores cometidos en situaciones didácticas deben considerarse como momentos creativos. (Caisa Tipan, 2016)

De acuerdo al constructivismo, en su dimensión pedagógica, concibe el aprendizaje como resultado de un proceso de construcción personal-colectiva de los nuevos conocimientos, actitudes y vida, a partir de los ya existentes y en cooperación con los compañeros y el facilitador. En ese sentido se opone al aprendizaje receptivo o pasivo que considera a la persona y los grupos como pizarras en blanco o bóvedas, donde la principal función de la enseñanza es vaciar o depositar conocimientos. (Caisa Tipan, 2016). De modo que, el aprendizaje se basa en la absorción de nuevos conocimientos brindados por otras personas o por aprendizaje propio como la vida cotidiana.

Para entender el aprendizaje, fue necesario de la contribución de varias propuestas para dar paso a la formulación de una Metodología Constructivista. Entre una de las propuestas más significativas se mencionan la siguiente:

El Aprendizaje Centrado en la Persona-Colectivo

Según el artículo de prensa de El Sembrador menciona que:

La persona-colectivo interviene en el proceso de aprendizaje con todas sus capacidades, emociones, habilidades, sentimientos y motivaciones; por lo tanto, los contenidos del proceso pedagógico no deben limitarse sólo al aprendizaje de hechos y conceptos (contenido conceptual), sino que es necesario atender en la misma medida a los procedimientos (contenido procedimental), las actitudes, los valores y las normas (contenido actitudinal), si se quiere una adaptación activa de la persona o grupos a nuevas situaciones sociales. Así mismo, hay que considerar sus propios estilos, ritmos y estrategias de aprendizaje. (EL SEMBRADOR, 2017, pág. 9). Por lo tanto, la educación se entiende en la capacidad del ser humano para entender su entorno en un grupo social.

Identificación de la Muestra

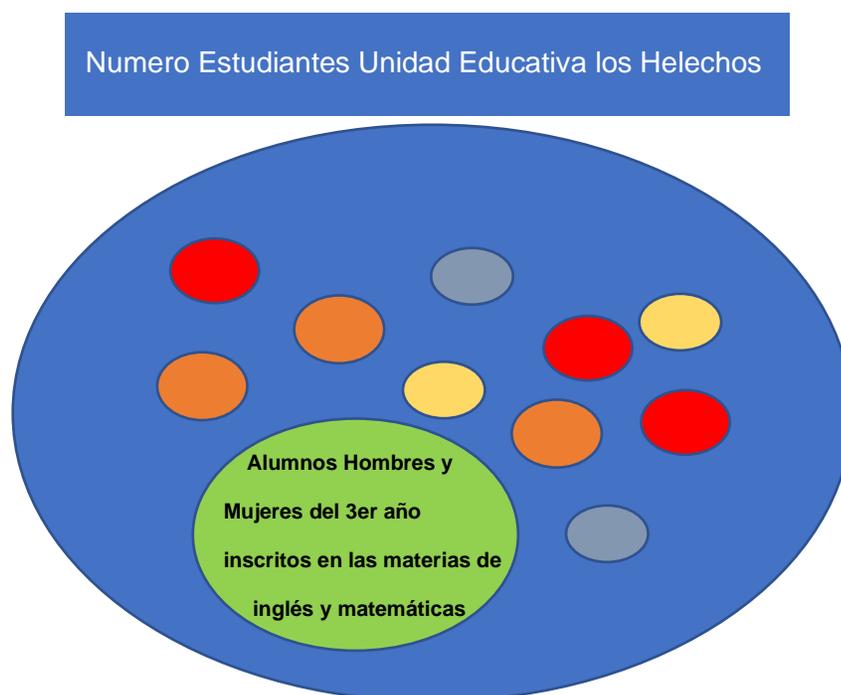
La muestra es un subconjunto de la población que debe ser representativa, a lo que se denomina muestra aleatoria simple, lo cual permitan facilitar el estudio que se desea realizar. Esta muestra aleatoria simple consiste en dar a todos los elementos de la población la misma posibilidad de ser parte de la muestra que se va a utilizar para ejecutar el estudio, lo que garantiza la representatividad de los datos.

Dado este antecedente se puede inferir que el punto inicial por el cual debe partir cualquier solución de machine learning son los datos sobre los cuales se van a evaluar los diferentes modelos predictivos y de aquellas evaluaciones seleccionar el que mejor se acople a dar solución al problema de negocio que se desea modelar. En virtud de lo anteriormente expuesto se procede a delimitar la muestra bajo la cual se va a desarrollar el proyecto.

Para efectos de este proyecto se ha determinado, en conjunto con las autoridades de la institución educativa que; de la población estudiantil con la que cuenta la institución se procederá a extraer una muestra aleatoria simple correspondiente a los alumnos de todos los géneros del tercer año de educación básica de la unidad educativa los HELECHOS que se encuentren inscritos en las materias de matemáticas e inglés. Dicha selección se muestra graficada en la Figura 1.

Figura 1

Selección de la Muestra



Nota: La figura 1 muestra la selección de la muestra.

Proceso de Minería de Datos

Como ha sido el caso en muchos otros paradigmas computacionales, la extracción de conocimiento de grandes repositorios de datos (es decir, minería de datos) comenzó como proyectos experimentales de prueba y error. Muchos profesionales han visto el problema desde la perspectiva de tratar de caracterizar lo que funciona y lo que no funciona. Durante bastante tiempo, los proyectos de minería de datos se llevaron a cabo como esfuerzos experimentales bastante artísticos. Sin embargo, para llevar a cabo un análisis de minería de datos metódicamente, se necesitaba desarrollar y seguir un proceso estandarizado. Con base en las mejores prácticas, los investigadores y profesionales de la minería de datos han propuesto varios procesos (flujos de trabajo en forma de metodologías sencillas paso a paso) para maximizar la probabilidad de éxito en la realización de proyectos de minería de datos. Estos esfuerzos han llevado a varios procesos estandarizados, uno de los cuales será parte del estudio del proyecto.

Metodología KDD

KDD (Descubrimiento de Conocimiento en bases de datos) es uno de los cuatro procesos estandarizados de minería de datos utilizados en la actualidad. Propuesto por Fayyad y sus colegas en 1996, involucra dentro de su proceso la ejecución de las siguientes tareas:

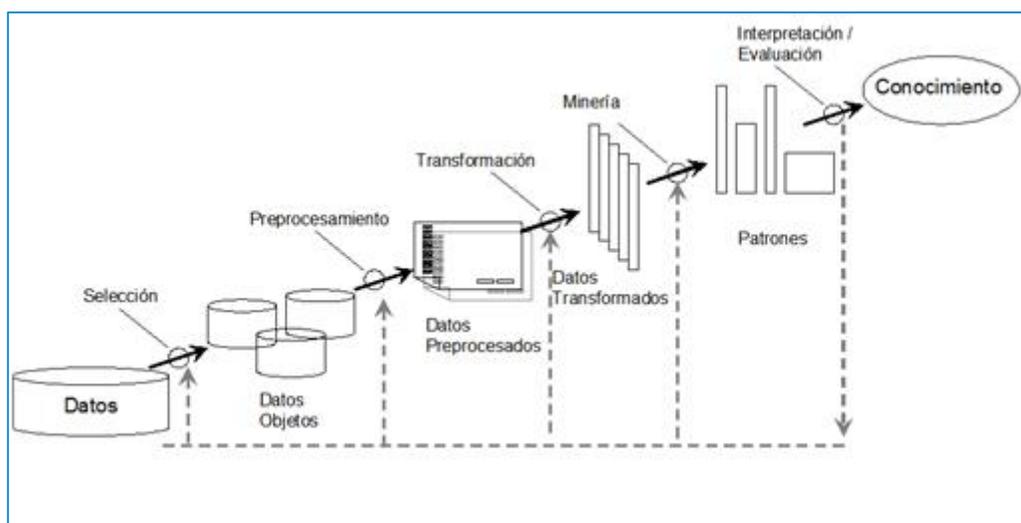
- Selección
- Pre procesamiento
- Transformación
- Minería de datos
- Evaluación e implantación.

Las tareas listadas anteriormente convierten a esta metodología en un proceso iterativo e interactivo de extremo a extremo con muchos pasos y muchos comentarios y réplicas para garantizar que las tareas se lleven a cabo correctamente. Los resultados se liberan en los siguientes pasos correctamente y de manera adecuada.

En la Figura 2 se describen de forma gráfica los pasos de procesamiento que forman parte de la metodología KDD incluyendo en estas flechas direccionales etiquetadas y el resultado de cada paso se muestra como una imagen gráfica que representa el artefacto que se entrega al finalizar la ejecución de cada tarea dentro del proceso.

Figura 2

Proceso Metodología KDD



Nota: Proceso Metodología KDD. Fuente: BInformatica (2014). Proceso Metodología KDD [Imagen]. Blogger. <http://introduccionbi.blogspot.com/2014/11/proceso-kdd.html>

A continuación, se describe de forma más detallada cada una de las tareas que se deben ejecutar dentro de la metodología KDD en función de lo expuesto en la Figura 2

Comprensión Del Dominio De Estudio Y Establecimiento De Objetivos

En este punto lo que se requiere es entender la problemática a la cual se debe hacer frente y entender su contexto con el objetivo de proponer soluciones viables y reales. En tal virtud es prioritario identificar las propias limitaciones y reglas del escenario en estudio para poder definir de forma más clara las metas a llegar.

Selección De Datos

Una vez definido los objetivos a alcanzar se deben seleccionar datos disponibles que permitan apalancar el estudio e integrarlos en uno solo, esto permite llegar a alcanzar los objetivos planteados por el análisis.

La selección de datos es la identificación de los diferentes repositorios de datos con los que se cuenta para poder realizar la investigación. Son los datos a los que se tiene acceso dentro de su organización y/o datos de la organización, así como los datos a los que tiene acceso a través de Internet o los datos que se pueden comprar u otros repositorios de datos. Por lo tanto, los datos sin procesar se pueden seleccionar, organizar y extraer para incluir solo aquellos que necesitará consultar y extraer para el objetivo específico que se tiene en mente.

El producto entregado por esta tarea dentro del proceso KDD se denomina DataSet de datos objetivo.

Los DataSets no son más que objetos bidimensionales que posee un conjunto de datos tabulados en filas y columnas que han facilitado el análisis de la información. Cada columna dentro de un DataSet corresponde a una variable y cada fila corresponde a un caso dentro del estudio que se haya abordado.

Los DataSet vienen representados en diferentes formatos como, por ejemplo:

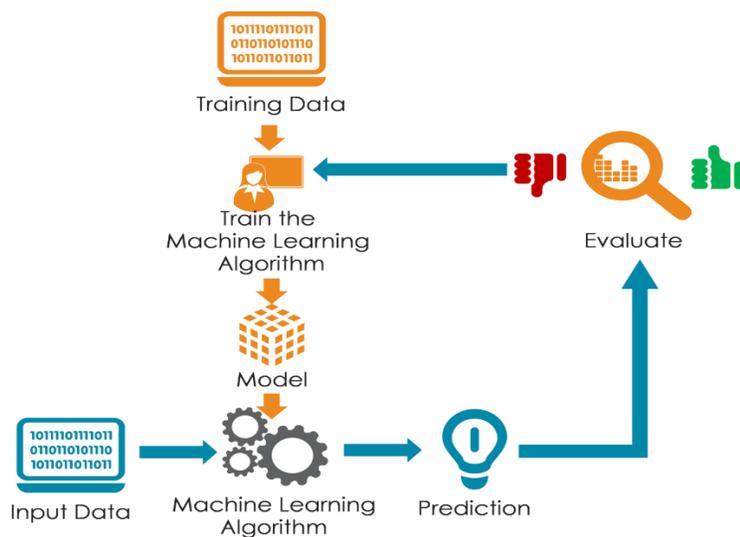
- TXT
- EXCEL
- CVS

Data Training

Para facilitar el análisis de la información obtenida, las buenas prácticas de Machine Learning sugieren la generación de un data training que no es más que extraer un número de registros del DataSet con el objeto de poder aplicar sobre este los diferentes modelos de Machine Learning, convirtiéndose en el conjunto de datos real que se utilizara para entrenar al modelo.

Figura 3

Data Training



Nota: Data Training Fuente: Around. T. (2018). Data Training [Imagen]. Medium.

<https://medium.com/@tekaround/train-validation-test-set-in-machine-learning-how-to-understand-6cdd98d4a764>

Data Testing

Son los datos utilizados para corroborar que el modelo generado a partir de los datos de entrenamiento realice una predicción de forma adecuada. La proporción utilizada para cada uno de estos elementos utilizados en el análisis.

Método De Retención

Conocido también como Holdout Method consiste en dedicar el 80% del conjunto de los datos a entrenar el algoritmo y el 20% de los datos a validarlo. Este genera un inconveniente dado que una vez que se ha seleccionado un modelo en específico y aplicado a los datos de validación no se puede determinar cuál sería el error que se podría cometer con datos nuevos, dado que no se puede tomar como referencia la medida de los datos de validación obtenidos, este se hace por descarte de un modelo en específico. Para solventar esto se agrega un tercer conjunto de datos denominado datos de prueba que solo servirán para probar el modelo final y hacer una estimación de su capacidad predictiva.

Normalmente el porcentaje en el que se dividirían el conjunto de los datos para este método serían los mostrados en la figura 4.

Figura 4

Division de DataSet Holdout Method



Nota: Muestra los datos del DataSet Holdout Method Fuente: Anónimo. (s.f.) Division de DataSet holdout Method [Imagen]. Interactivechaos.

<https://interactivechaos.com/es/manual/tutorial-de-machine-learning/metodo-de-retencion>

Validación Cruzada

Según Segovia. M. & Camacho. M. menciona que el procedimiento de validación es una técnica utilizada para evaluar los resultados de un análisis estadístico y garantizar que son independientes de la partición entre datos de entrenamiento y prueba. Consiste en repetir y calcular la media aritmética obtenida de las medidas de evaluación sobre diferentes particiones. Se utiliza en entornos en los cuales el objetivo principal es la predicción y se quiere estimar qué tan preciso es un modelo que se llevará a cabo a la práctica. (Segovia Vargas & Camacho Miñano, 2012)

Como saber que el desempeño real del modelo antes de aplicarlo, la solución es usar el Data Training para entrenar y evaluar al mismo tiempo lo que se denomina validación cruzada la cual consiste en dividir una tabla de datos en K partes iguales siempre dejando una parte de esta división para pruebas del modelo. Para ejemplificar esto se dividirá la data set en 6 partes

El concepto de validación cruzada indica que una vez se haya realizado la división de la tabla en K partes iguales se procede a entrenar el modelo con dichas partes dejando siempre una para prueba, en este ejemplo en particular el entrenamiento se daría según la tabla 3.

Tabla 3;Error! No se encuentra el origen de la referencia.

Ejemplo Validación Cruzada

ITERACION	K DE ENTRENAMIENTO	K DE TEST
1	k1 k2 k3 k4 k5	k6
2	k1 k2 k3 k4 k6	k5
3	k1 k2 k3 k5 k6	k4
4	k1 k2 k4 k5 k6	k3
4	k1 k3 k4 k5 k6	k2
5	k2 k3 k4 k5 k6	k1

Una vez que se haya realizado todas las interacciones de entrenamiento posibles se promedia el desempeño obtenido de cada interacción y se observa el desempeño esperado.

Tabla 4*Ejemplo Validación Cruzada*

ITERACION	K DE ENTRENAMIENTO	K DE TEST	VALOR DESEMPEÑO
1	k1 k2 k3 k4 k5	k6	VAL INTERACCION 1
2	k1 k2 k3 k4 k6	k5	VAL INTERACCION 2
3	k1 k2 k3 k5 k6	k4	VAL INTERACCION 3
4	k1 k2 k4 k5 k6	k3	VAL INTERACCION 4
4	k1 k3 k4 k5 k6	k2	VAL INTERACCION 5
5	k2 k3 k4 k5 k6	k1	VAL INTERACCION 6
		Rendimiento Esperado	SUMATORIA / 6 (VALOR DE K)

Limpieza y Preprocesamiento de Datos

En esta fase se procede a determinar la confiabilidad de los datos seleccionados realizando tareas de permitan dar fiabilidad a la información contenida en el DataSet, eliminando valores atípicos, imputando variables o atributos con datos faltantes o eliminar información que no es útil para el objetivo planteado.

La transformación de los datos, mejorar la calidad de estos los mismos que involucren reducción de dimensiones es decir disminuir la cantidad de variables del conjunto de datos o transformaciones donde se cambia variables numéricas a categóricas conocido también como discretización.

Por lo general, el conjunto de datos que alimenta la minería de datos tiene un formato plano. La forma plana significa que hay filas y columnas en una sola entidad. Columnas que representan variables. Filas que representan las muestras de datos preprocesados.

Selección De La Tarea Apropia De Minería De Datos

En esta fase se selecciona el paradigma apropiado de minería de datos que permitan alcanzar el objetivo planteado en la investigación ya sea clasificación, regresión, agrupación etc.

Los objetivos identificados durante el análisis del domino de la investigación pueden clasificarse en lo siguiente:

- **Predicción:** Es encontrar un modelo que sea utilizado para casos futuros o desconocidos.
- **Prescripción:** Es observar el comportamiento del modelo.

Elección del Algoritmo de Minería de Datos

Se procede a seleccionar la técnica o algoritmo, incluso más de uno para la búsqueda del patrón y obtener conocimiento, el meta aprendizaje se enfoca a explicar la razón por la que un algoritmo funciona mejor en determinada problemática y para cada técnica existen diversas posibilidades. Cada algoritmo tiene su propia esencia, es decir su forma de obtener los resultados por lo que se recomienda conocer las propiedades de los algoritmos a utilizar y ver cual se ajusta mejor a los datos.

Aplicación del Algoritmo

Una vez seleccionada las técnicas, el siguiente paso es aplicarlas a los datos seleccionados limpios y procesados, pudiendo darse el caso de que se ejecuten varias veces los algoritmos intentando ajustar los parámetros que optimicen los resultados, estos parámetros pueden variar de acuerdo al método seleccionado.

Interpretación De Patrones Minados

Una vez aplicados los algoritmos al conjunto de datos se evalúan los patrones que se generaron y el rendimiento que se obtuvo para verificar que se cumpla con las metas planteadas en las fases iniciales para realizar esta técnica evaluación se utiliza la validación cruzada el cual realiza una partición de los datos dividiéndose en entrenamiento que servirán para crear más del 80% de pruebas y será utilizados para determinar que el algoritmo realiza su trabajo de forma efectiva.

Aplicación

Si los resultados de la evaluación son satisfactorios, la última etapa es aplicar el modelo encontrado al contexto y comenzar a resolver las problemáticas y si por el contrario los resultados no son satisfactorios se debe volver a las etapas iniciales del flujo donde se realizarán ajustes analizando desde la etapa de selección de los datos hasta la de la evaluación.

Modelos de Machine Learning

“Según el artículo de Microsoft menciona que un modelo de aprendizaje automático es un archivo que se ha entrenado para reconocer determinados tipos de patrones. Puede entrenar un modelo con un conjunto de datos, y proporcionarle un algoritmo que puede usar para averiguar y obtener información de esos datos.

(Microsoft, 2022)

Una vez adiestrado el modelo, se puede usar para diferenciar la información nueva y realizar predicciones.

Los modelos de Machine Learning se pueden clasificar en dos categorías:

- Supervisados
- No Supervisados

Modelos Supervisados

Dependen de datos históricos para dar una respuesta en base a conocer los datos, utilizan este valor del pasado para entrenarse y construir un modelo que sea capaz de predecir un valor de una variable en el futuro.

Este tipo de modelos está ligado a la resolución de problemas de clasificación y regresión. Los algoritmos utilizados en este tipo de modelos son los siguientes:

- Regresión Lineal
- Árboles de decisión.
- Clasificación de Naïve Bayes.
- Regresión por mínimos cuadrados.
- Regresión Logística.
- Support Vector Machines (SVM).

Usos del modelo de regresión

- ✓ Pronosticar: Predecir la variable respuesta para valores futuros de las variables predictoras
- ✓ Descripción: Establece una ecuación que describa la relación entre la variable dependiente y las variables predictoras.
- ✓ Control: Controlar el comportamiento o variación de la variable de respuesta de acuerdo a los valores de las variables predictoras

Modelos no Supervisados

Prescinde de restricciones y no necesita tener datos del pasado para devolver una variable de salida, todas las variables del DataSet se utilizan como variables predictoras para construir una salida sin necesidad de datos previos.

Este tipo de modelos se usan para solventar problemas asociados al Clustering, coocurrencias, perfilado o Profiling.

Los tipos de algoritmos utilizados en este modelo son los siguientes:

- Algoritmos de clustering
- Análisis de componentes principales
- Descomposición en valores singulares (singular value decomposition)
- Análisis de componentes principales (Independent Component Analysis)

Python

Python fue creado por Guido van Rossum, un programador holandés a finales de los 80 y principio de los 90 cuando se encontraba trabajando en el sistema operativo Amoeba. Primariamente se concibe para manejar excepciones y tener interfaces con Amoeba como sucesor del lenguaje ABC. (Yan & Yan, 2018)

“Python es un lenguaje de programación multiparadigma, esto significa que más que forzar a los programadores a adoptar un estilo particular de programación, permite varios estilos: programación orientada a objetos, programación imperativa y programación funcional”. (VELASCO & MAMARANDI, s.f)

Anaconda

“Anaconda es una suite de código abierto que combina una serie de aplicaciones, librerías, imágenes e interfaces interactivas y conceptos diseñados para el desarrollo de la ciencia de datos con Python”. (Bermeo Calle & Chicaiza Sangurima, 2020)

Es utilizado para realizar operaciones de ciencias de datos en el mundo real, ofrece aporte para los siguientes lenguajes de programación:

- R
- Python

- Octave
- Julia

Ingeniería de Software

Según Foster. E. menciona que es un conjunto de componentes interrelacionados e interdependientes que interactúan y funcionan como un todo para lograr objetivos específicos. Un sistema eficaz debe ser sinérgico. El sistema suele operar en un entorno externo a sí mismo. Un sistema también puede definirse como la combinación de personal, materiales, instalaciones y equipos que trabajan juntos para convertir entradas en salidas significativas y necesarias. (Foster, 2014)

Los siguientes son algunos principios fundamentales sobre los sistemas:

- Los componentes de un sistema están interrelacionados y son interdependientes.
- El sistema se suele considerar como un todo.
- Cada sistema tiene objetivos específicos.
- Debe haber algún tipo de entradas y salidas.
- Los procesos prescriben la transformación de insumos en productos.
- Los sistemas exhiben entropía, es decir, tendencia a desorganizarse.
- Los sistemas deben estar regulados (planificación, retroalimentación y control).
- Todo sistema tiene subsistemas.
- Los sistemas exhiben una tendencia a un estado final.

Software

Es la combinación de programa(s), base(s) de datos y documentación en una suite sistémica, y con el único propósito de resolver problemas específicos del sistema y cumplir objetivos predeterminados. El software agrega valor a los componentes de hardware de un sistema informático.

Ingeniería De Software

De acuerdo a Foster. E. la Ingeniería de Software es el proceso mediante el cual se investigan, planifican, modelan, desarrollan, implementan y gestionan los sistemas de software. También incluye la reingeniería de los sistemas existentes con miras a mejorar su rol, función y desempeño. el objetivo final es la provisión o mejora de las comodidades deseables y la mejora de la productividad dentro del dominio del problema relacionado. (Foster, 2014). Por lo tanto, son procesos que ayudan a tener una mejora continua en el proceso de desarrollo de software.

Transformación del sistema puede tomar varios caminos, algunos de los cuales pueden ser:

- Mejorar el funcionamiento interno del sistema.
- Modificación de entradas y salidas
- Modificación de las metas y objetivos del sistema
- Rediseñando el sistema
- Diseño y desarrollo de un nuevo sistema basado en problemas existentes

Pasos en el Proceso De Análisis

Antes de embarcarse en cualquier trabajo importante en Ingeniería de Software, se lleva a cabo un proceso de investigación y análisis. Este proceso puede resumirse en los siguientes pasos:

- Define el problema.
- Comprender el problema (sistema)
- Las interrelaciones e interdependencias
- Tener una imagen de las variables en el trabajo dentro del sistema
- Definir la extensión del sistema (problema).
- Identifica soluciones alternativas.
- Examinar soluciones alternativas.
- Elija la mejor alternativa.
- Seguir la alternativa elegida.
- Evaluar el impacto del sistema (nuevo/modificado).

Ciclo De Vida Del Software

Cada sistema de software tiene un ciclo de vida: un período durante el cual se investiga/concibe, diseña, desarrolla y sigue siendo aplicable o necesario. Se han propuesto varios modelos de ciclo de vida, como por ejemplo las siguientes:

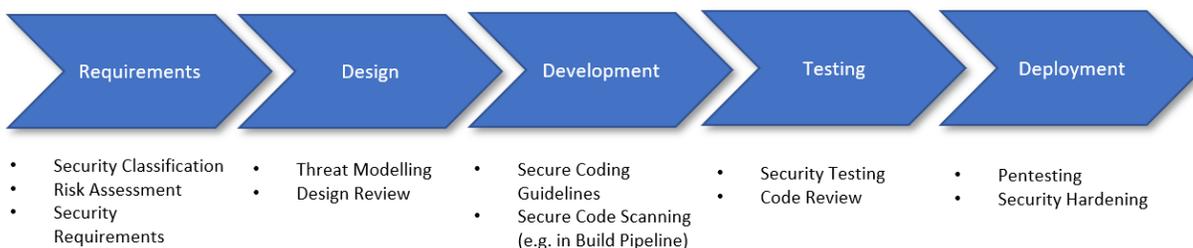
- Modelo de cascada
- Modelo de prototipo por fases
- Modelo de desarrollo iterativo
- Modelo de prototipo rápido

- Modelo de Transformación Formal
- Modelo basado en componentes
- Modelo de desarrollo ágil

Independientemente del modelo que se utilice, un sistema de software pasa por las cinco fases, como se muestra en la figura 18 (también se muestran los entregables relacionados). Estas fases constituyen el ciclo de vida de desarrollo de software (SDLC).

Figura 5

Fase SDLC



Nota. Fases del Ciclo de Vida del Software. Fuente: Anónimo. (2018). Secure SDLC - Secodis GmbH [Imagen]. secodis. <https://www.secodis.com/secure-sdlc2/?lang=en>

Angular JS

Angular es un proyecto de código abierto mantenido por Google y una comunidad de desarrolladores. La nueva plataforma Angular es muy diferente del marco heredado que pueda tener usado en el pasado. En colaboración con Microsoft, Google convirtió a TypeScript en el idioma predeterminado para Angular. TypeScript es un superconjunto de JavaScript que permite a los desarrolladores apuntar a navegadores heredados como Internet Explorer 11, al tiempo que les permite escribir código JavaScript moderno que funciona en navegadores

imperecederos como Chrome, Firefox y Edge. Las versiones heredadas de Angular, versiones en el rango 1.xx, se denominan AngularJS. La versión 2.0.0 y versiones superiores se denominan Angular. Donde AngularJS es un marco monolítico de JavaScript SPA, Angular es una plataforma que es capaz de apuntar a navegadores, marcos móviles híbridos, aplicaciones de escritorio y vistas renderizadas del lado del servidor. (Uluca, 2020)

Microservicios

Los microservicios son servicios liberables de forma independiente que se modelan en torno a un dominio empresarial. Un servicio encapsula la funcionalidad y la hace accesible a otros servicios a través de redes; usted construye un sistema más complejo a partir de estos componentes básicos. Un microservicio puede representar el inventario, otro la gestión de pedidos y otro el envío, pero juntos pueden constituir un sistema de comercio electrónico completo. Los microservicios son una opción de arquitectura que se enfoca en brindarle muchas opciones para resolver los problemas que podría enfrentar.

Son un tipo de arquitectura orientada al servicio, aunque se opina sobre cómo se deben trazar los límites del servicio, y en la que la implementación independiente es clave. Son agnósticos a la tecnología, que es una de las ventajas que ofrecen.

Desde el exterior, un solo microservicio se trata como una caja negra. Aloja la funcionalidad comercial en uno o más puntos finales de la red (por ejemplo, una cola o una API REST, como se muestra en la Figura 24), sobre los protocolos que sean más apropiados. Los consumidores, ya sean otros microservicios u otros tipos de programas, acceden a esta funcionalidad a través de estos puntos finales en red. Los detalles de implementación internos (como la tecnología en la que está escrito el servicio o la forma en que se almacenan los datos)

están completamente ocultos al mundo exterior. Esto significa que las arquitecturas de microservicios evitan el uso de bases de datos compartidas en la mayoría de las circunstancias; en cambio, cada microservicio encapsula su propia base de datos cuando es necesario. (Newman, 2021)

Figura 6

API Rest

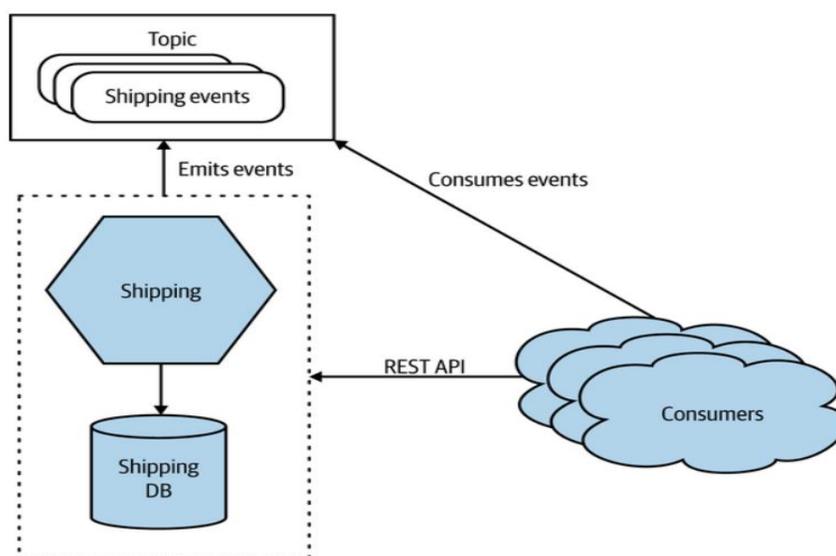


Figure 1-1. A microservice exposing its functionality over a REST API and a topic

Nota. Microservicios API REST. Fuente: Newman. S. (2021). API Rest [Imagen]. O'Reilly.

<https://www.oreilly.com/library/view/building-microservices-2nd/9781492034018/ch01.html>

Microservicios ASP NET CORE

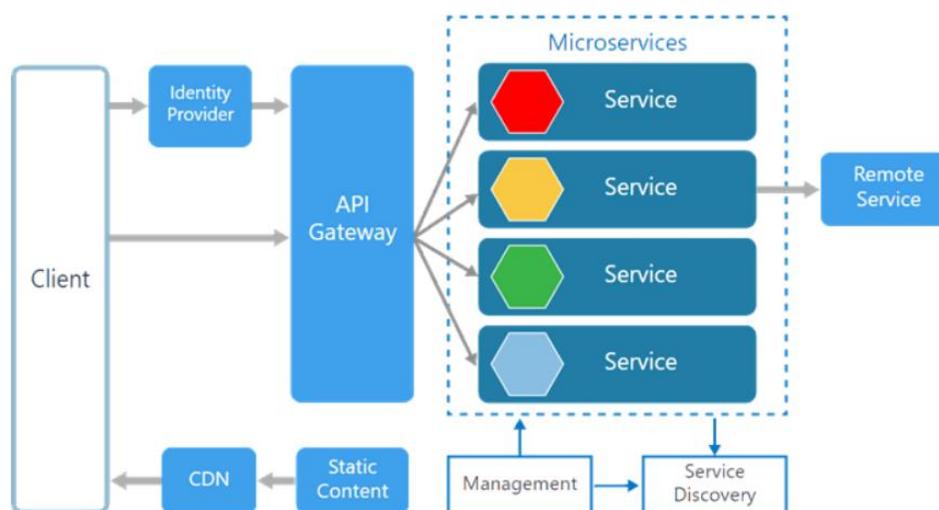
El término microservicios representa un estilo de desarrollo de software que ayuda a establecer mejores prácticas destinadas a aumentar la velocidad y la eficiencia del desarrollo

de un proyecto. Los microservicios aplican una cierta cantidad de principios y patrones arquitectónicos como arquitectura. Cada microservicio es independiente, pero, por otro lado, dependen unos de otros. Todos los microservicios en un proyecto de desarrollo se implementan en producción a su propio ritmo.

En la Figura 25 se muestra la arquitectura de microservicios de Microsoft

Figura 7

Arquitectura de microservicios de Microsoft



Nota. Arquitectura de microservicios de Microsoft. Fuente: Santimacnet (2017). Microservicios de Microsoft [Imagen]. Santimacnet. <https://santimacnet.wordpress.com/2017/01/20/azure-arquitecturas-para-microservicios-en-net/>

Figura 8*Microservicios*

Nota. Microservicios. Fuente: Escamilla. B. (2022). Microservicios [Imagen]. Morioh.

<https://morioh.com/p/e5a235c363c9>

El microservicio es un enfoque para crear pequeños servicios, cada uno de los cuales se ejecuta en su propio espacio y puede comunicarse a través de mensajes. Estos son servicios independientes que llaman directamente a su propia base de datos. (Escamilla, 2022)

CAPITULO III

CONSTRUCCIÓN

Selección de la Muestra

La unidad educativa los HELECHOS ha proporcionado una base de datos en archivos Excel de alumnos correspondientes al tercer año de educación básica, matriculados en las materias de inglés y matemáticas. La información correspondiente a las notas de cada alumno servirá como punto de partida para el análisis de la información y la selección de una muestra aleatoria que permita aplicar de forma acertada los diferentes modelos de machine learning.

Elección del Modelo de Machine Learning

En virtud de las bases de datos entregadas para este proyecto por parte de la institución educativa LOS HELECHOS, y dado que se pretende realizar una predicción de rendimiento escolar basado en datos históricos se ha seleccionado para este proyecto utilizar los modelos supervisados de machine learning.

Comprensión del Dominio de Estudio y Establecimientos de Objetivos

Levantamiento de Proceso Módulo de Predicción Académica

Se crea el mapa de procesos y subprocesos denominado “Evolución Académica” con el objetivo de determinar las tareas que contienen y así poder determinar con mayor exactitud el dominio del estudio.

Tabla 5*Cuadro análisis de dominio proceso evolución académica*

PROCESO	SUB PROCESO	ACTIVIDADES
Evolución Académica	Identificación Del Nivel Educativo A Analizar	<ul style="list-style-type: none"> • Listar los diferentes niveles que están disponibles para el análisis. • Seleccionar el nivel del listado sobre el cual se va a proceder a realizar el análisis. • Registrar el nivel seleccionado y proceder.
	Recopilación información académica de los estudiantes	<ul style="list-style-type: none"> • Seleccionar la información asociada al nivel que se identificó como objeto de análisis. • Listar los alumnos asociados al nivel objeto del análisis. • Registrar la información personal del alumno. • Identificar la información del representante legal.

	Selección de materia para el análisis	<ul style="list-style-type: none">• Identificar las materias disponibles para el análisis.• Seleccionar una materia bajo la cual se realizará el análisis
	Identificación de docente	<ul style="list-style-type: none">• Obtener información del personal docente disponible.• Identificar al docente en función de la materia de análisis seleccionada.
	Análisis de la información recopilada	<ul style="list-style-type: none">• Revisar las notas obtenidas por el alumno.• Identificar los periodos de menor rendimiento.• Identificar los temas del periodo de menor rendimiento.• Calcular el porcentaje de aprovechamiento obtenido por periodo.• Registrar el porcentaje de vacío académico

	Emisión de resultado de análisis	<ul style="list-style-type: none">• Preparación de información para elaboración de informe.• Generación de informe e impresión.• Registro de firmas de responsables del informe
	Sincronización departamento psicológico	<ul style="list-style-type: none">• Discutir con psicología el resultado del informe.• Identificar de puntos relevantes del informe.• Determinar áreas de conocimiento con mayor déficit.
	Identificación de acciones y correctivos	<ul style="list-style-type: none">• Generar plan de acción por el área psicológica.• Generar plan de acción por área de déficit.• Transmitir el plan de acción al docente.• Seguimiento del cumplimiento del plan de acción

	Divulgación de resultados a representantes	<ul style="list-style-type: none"> • Identificar la información de los representantes del alumno. • Emitir informe al representante vía mail de contacto
	Generación de Reportes	<ul style="list-style-type: none"> • Generar reporte de análisis y correctivos
	Registro histórico de resultados	<ul style="list-style-type: none"> • Registro de informe en bitácora educativa

Determinación de Objetivos

Dentro de la metodología KDD la determinación de los objetivos a investigar es fundamental, dado que en virtud de los mismos se puede alinear el análisis y esfuerzos a la consecución exitosa de los mismos. Durante las sesiones con la institución se pudo acordar el siguiente objetivo a cumplir:

- Predicción de rendimiento académico oportuno, tomando como insumo la base de datos de los alumnos de tercer año de educación básica de la INSTITUCION EDUCATIVA LOS HELECHOS.
- Contar con una aplicación web que permita la visualización de las predicciones.

Selección de los Datos

Tal como se explicó en el ítem de objetivos, la base de datos provista por la institución para este proyecto corresponde a los alumnos de tercer año de educación básica de la unidad educativa, sobre estos datos se procederá con las siguientes fases de la metodología KDD, apoyados con el uso de la herramienta Jupyter - Notebook de Python.

Division Training Dataset y Test Dataset.

En esta fase se toma como referente la base de datos entregada por la institución y se procede a generar dos conjuntos de Dataset denominados training and testing, que básicamente en el caso del Dataset training va a permitir procesar, transformar y evaluar los diferentes modelos de machine Learning con el fin de obtener un modelo que permita alcanzar el objetivo planteado en las primeras fases de la metodología.

En lo referente al Dataset testing, este permite una vez identificado el modelo que realizará la mejor predicción sobre los datos del training, comprobar su eficacia utilizando para esto en un conjunto de datos totalmente diferente a los que se usaron para el entrenamiento del modelo. La proporción sobre la cual se determinó el porcentaje de datos que se utilizara para generar el Dataset Training y el Dataset Testing corresponde a las mejores prácticas de Machine Learning en la cual se describe el uso de una proporción de 80/20, es decir un 80% para el aprendizaje y un 20% para el test utilizando una técnica de validación cruzada utilizando librería de aprendizaje estadístico de Python sklearn como se muestra a continuación:

Figura 9

Carga de archivo en Jupyter

```

In [1]: import pandas as pd
import numpy as np
mainpath="/Users/ARMANDO/Documents/PREDICTOR/ TESIS - MACHINE-LEARNING/machine-learning/notas"
filename="/og_notas_gb.xlsx"
notas_data=pd.read_excel(mainpath + "/" + filename,"Hoja de Calificaciones",header=1)
#notasxls.head()

In [2]: #notasxls.shape

In [3]: notas_data.shape
Out[3]: (2016, 18)

In [4]: notas_data.head()

```

Nota: Archivo Jupyter.

Figura 10

División de Dataset Training y Dataset Testing

```

In [7]: # generar conjunto de entrenamiento y de testing METODO VALIDACION CRUZADA

In [8]: from sklearn.model_selection import train_test_split

In [9]: train, test=train_test_split(notas_data,test_size=0.2)
|

In [10]: train.shape
Out[10]: (1612, 18)

In [11]: test.shape
Out[11]: (404, 18)

```

Nota: Método de Validación Cruzada.

Como se puede observar en las imágenes siguientes se procedió a realizar la división de los datos en dos Dataset usando validación cruzada implementada en la librería Sklearn y utilizando para este particular una proporcionalidad del 80/20.

Limpieza y Transformación de los Datos

Dado que en este punto ya se tiene bien definidos con conjuntos de datos que se utilizara para la construcción del modelo de predicción y los que se utilizara para validación de dicho modelo se procede a realizar el procesamiento y transformación de la información, previo a que la misma sea sometida a los diferentes modelos de predicción de Machine Learning, para esto se vuelve a apoyar del uso de Jupyter.

Figura 11

Training Dataset procesamiento y transformación.

```
In [4]: notas_data.head()
```

```
Out[4]:
```

	CODIGO	NOMBRE	PARCIAL1	PARCIAL2	PARCIAL3	PPARCIALT1	LECCION1	LECCION2	LECCION3	PARCIALLECCION	EJERCICIO1	PEJERCICIO	TARE
0	1	Aguas Reyes Alejandro André	49	35	68	50.66	53	79	53	61.66	55	55	
1	2	Arroba Buenaño Santiago Xavier	26	10	12	16.00	37	29	10	25.33	55	55	
2	3	carranza Santacruz David Alejandro	48	15	61	41.33	26	64	21	37.00	27	27	
3	4	Carrillo López Jorge Washinton	35	35	45	38.33	89	100	41	76.66	100	100	
4	5	Castro Taco Gina Tatiana	45	46	97	62.66	74	100	74	82.66	100	100	

Nota: Procesamiento de la Data.

Como resultado de este análisis se puede identificar que la base de datos entregada por la institución, contiene datos completos correspondientes a los nombres de los alumnos, tipo de evaluación, notas de cada evaluación y promedio de cada conjunto de notas.

A partir de este análisis se procede a identificar las variables predictoras sobre las cuales se va a evaluar los diferentes modelos de Machine Learning.

Una vez realizado el análisis de los datos, en conjunto con la institución educativa se llega a la conclusión que la mejor forma de abordar este proyecto es tomar como variables predictivas las correspondientes a los promedios de cada bloque que comprende las siguientes clasificaciones:

- Promedios de tareas
- Promedios de lecciones
- Promedios de deberes
- Promedios de ejercicios
- Promedio de prueba

Interpretación y Evaluación

Para este punto se procede a someter el conjunto de Dataset training a los diferentes modelos de predicción a evaluar los resultados e interpretarlos y en función de esa interpretación se determinará si el modelo aplicado es el candidato idóneo para poder exponer dicho modelo en un servicio web que pueda ser de consumo por un aplicativo.

El orden en el cual se va a proceder a evaluar los diferentes modelos de aprendizaje automático será el siguiente:

- *Regresión Lineal*
- *Regresión Lineal Múltiple*
- *Regresión Logística*
- *Clustering y Clasificación*
- *Árboles y Bosques Aleatorios*

- *Máquinas de Soporte*

Cabe indicar que si en una de las evaluaciones los resultados son favorables se dará por sentado que este es el modelo definitivo que será implementado en el proyecto, dejando por fuera la evaluación de los modelos que le siguen, a este acuerdo se llegó dado que el tiempo para ejecutar el proyecto es corto.

Una vez encontrado el modelo que despliegue un resultado exitoso y cumpla con el objetivo, no es necesario el evaluar más modelos e interpretar sus resultados.

- **Evaluación de Regresión Lineal**

Nuevamente, se apoya en el uso de Jupyter (pandas, numpy y sklearn) para someter el Dataset Training al modelo de regresión lineal como se muestra a continuación.

Figura 12

Regresión Lineal

```
In [7]: # generar conjunto de entrenamiento y de testing METODO VALIDACION CRUZADA
In [8]: from sklearn.model_selection import train_test_split
In [9]: train, test=train_test_split(notas_data,test_size=0.2)
In [10]: train.shape
Out[10]: (1612, 18)
In [11]: test.shape
Out[11]: (404, 18)
In [12]: dataframetraining=pd.DataFrame(train)
```

Nota: Evaluación de Regresión Lineal.

Figura 13

Regresión Lineal

```
In [13]: import statsmodels.formula.api as smf
```

```
In [14]: linearmodelo=smf.ols(formula="NOTABLOQUE~PPARCIALT1", data=dataframetraining).fit()  
linearmodelo.params
```

```
Out[14]: Intercept      29.676737  
PPARCIALT1      0.772570  
dtype: float64
```

Nota: Caso Practico Regresión Lineal.

Donde:

NOTABLOQUE: Corresponde a la variable a ser predecida.

PPARCIALT1: Corresponde a los valores de los totales parciales de las tareas asignadas a un alumno.

Data: Corresponde al conjunto de datos de entrenamiento

El modelo lineal predictivo = $29.883903 + 0.769137 * PPARCIALT1$

Figura 14

Regresión Lineal, modelo.

```
In [18]: linearmodelo.summary()
```

Out[18]: OLS Regression Results

Dep. Variable:	NOTABLOQUE	R-squared:	0.574			
Model:	OLS	Adj. R-squared:	0.573			
Method:	Least Squares	F-statistic:	2165			
Date:	Tue, 19 Jul 2022	Prob (F-statistic):	3.09e-300			
Time:	10:57:53	Log-Likelihood:	-5828.2			
No. Observations:	1612	AIC:	1.166e+04			
Df Residuals:	1610	BIC:	1.167e+04			
Df Model:	1					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
Intercept	29.6767	0.736	40.311	0.000	28.233	31.121
PPARCIALT1	0.7726	0.017	46.530	0.000	0.740	0.805
Omnibus:	127.751	Durbin-Watson:	2.028			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	120.948			
Skew:	-0.610	Prob(JB):	5.45e-27			
Kurtosis:	2.440	Cond. No.	146.			

Nota: Modelo de Regresión Lineal.

Mientras más elevado sea el valor del BIC, mejor será la predicción.

Figura 15

Regresión Lineal, predicción.

```
In [19]: dataframetraining.index.duplicated()
```

Out[19]: array([False, False, False, ..., False, False, False])

```
In [20]: prediccion_rendimiento=linearmodelo.predict(pd.DataFrame(dataframetraining["PPARCIALT1"]))
```

```
In [21]: prediccion_rendimiento
```

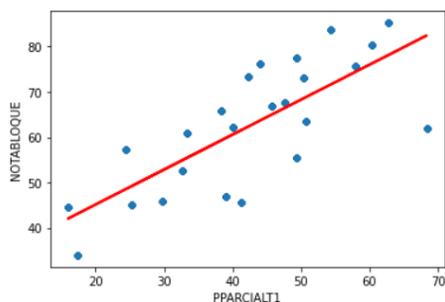
Out[21]:

1178	61.607054
355	48.473364
1006	64.952282
930	68.560183
311	76.285883
...	...
1154	61.607054
330	68.560183
683	49.245934
1029	60.579536
851	49.245934

Length: 1612, dtype: float64

```
In [22]: import matplotlib.pyplot as plt
%matplotlib inline
dataframetraining.plot(kind="scatter", x="PPARCIALT1", y="NOTABLOQUE")
plt.plot(pd.DataFrame(dataframetraining["PPARCIALT1"]),prediccion_rendimiento, c="r" , linewidth=2)
```

```
Out[22]: [<matplotlib.lines.Line2D at 0x1dc745d7eb0>]
```



Nota: Predicción y resultado.

Se puede observar que la regresión lineal es la técnica que mejor se acopla al cumplimiento del objetivo.

Cálculo de error estándar residual

Figura 16

Regresión Lineal, calculo de error.

```
In [23]: dataframetraining["rend_pred"] = 28.199216 + 0.728121*dataframetraining["PPARCIALT1"]
```

```
In [24]: dataframetraining["RSE"] = (dataframetraining["NOTABLOQUE"] - dataframetraining["rend_pred"])**2
```

```
In [25]: SSD=sum(dataframetraining["RSE"])
SSD
```

```
Out[25]: 149126.33935023504
```

```
In [26]: RSE=np.sqrt(SSD/(len(dataframetraining)-2))
RSE
```

```
Out[26]: 9.624191160665939
```

```
In [27]: parcial_media=np.mean(dataframetraining["NOTABLOQUE"])
parcial_media
```

```
Out[27]: 62.30570719602978
```

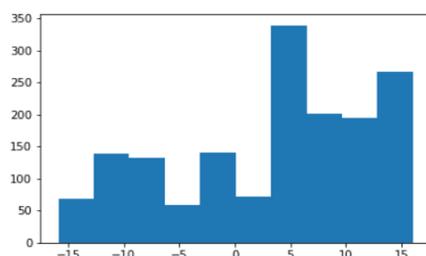
```
In [28]: error=RSE/parcial_media
error
```

```
Out[28]: 0.15446724856818908
```

```
In [29]: plt.hist((dataframetraining["NOTABLOQUE"] - dataframetraining["rend_pred"]))
```

```
In [29]: plt.hist((dataframetraining["NOTABLOQUE"] - dataframetraining["rend_pred"]))
```

```
Out[29]: (array([ 68., 139., 132.,  58., 141.,  72., 339., 201., 195., 267.]),
array([-15.89172393, -12.70020554, -9.50868714, -6.31716875,
        -3.12565036,  0.06586804,  3.25738643,  6.44890482,
         9.64042321, 12.83194161, 16.02346   ]),
<BarContainer object of 10 artists>)
```



Nota: Calculo de Error.

El resultado obtenido al aplicar la regresión lineal simple no son los deseados, motivo por el cual se procede a evaluar los datos con el siguiente modelo.

- **Regresión lineal múltiple.**

Es un modelo lineal que involucra más de una variable predictora y en lugar de tener una sola pendiente para una variable predictora, va a tener diferentes para cada una de las variables predictoras.

Para esto se hará uso Jupyter en conjunto con la librería sklearn de Python.

Este análisis se puede abordar desde dos enfoques diferentes: Ascendente y Descendente

- ✓ *Ascendente*: Cuando se parte de una sola variable predictora y se va aumentando las variables predictoras, una por una calcula su pvalor, cuando este resultado sea muy alto implica que esa variable predictora no aporta nada al modelo predictivo.
- ✓ *Descendente*: Cuando se parte de todas las variables predictoras y se va eliminando una por una, en virtud de cuan grande sea su pvalor, va calculado hasta obtener un modelo que aporte al modelo predictivo.

Para este particular se opta por usar el enfoque Descendente

Figura 17

Código Regresión Lineal Múltiple

```
In [30]: from sklearn.feature_selection import RFE
        from sklearn.svm import SVR

In [31]: feature_cols=["PPARCIALT1", "PARCALLECCION", "PEJERCICIO", "PTAREAS"]

In [32]: X=dataframetraining[feature_cols]
        Y=dataframetraining["NOTABLOQUE"]

In [33]: estimator=SVR(kernel="linear")
        selector=RFE(estimator,2,step=1)
        selector=selector.fit(X,Y)
```

Nota: Regresión lineal múltiple.

Donde:

El numero 2 corresponde al número de variables predictoras que deseo que contenga mi modelo.

Step=1 indica que se vaya evaluando una por una las variables predictoras en función de su feature_cols= variables predictoras candidatas

X = conjunto de variables predictoras

Y= variable a ser predecida

Estimator=SVR(kernel="linear") = indico que el tipo de modelo que deseo crear es lineal.

Figura 18

Entrada y Salida Regresión Lineal Múltiple.

```
In [34]: selector.support_  
Out[34]: array([ True,  True, False, False])
```

Nota: Regresión lineal múltiple.

True, significa que una variable predictora fue seleccionada para el modelo y false que una variable predictora no fue seleccionada.

Figura 19

Codigo Rerregresion Lineal Múltiple

```
In [35]: selector.ranking_  
Out[35]: array([1, 1, 2, 3])
```

Las variables seleccionadas se ordenan con rango 1 a partir de aquí las subsiguientes se ordenan en orden decreciente en función de su significatividad con respecto al modelo, es decir que una variable de rango dos (2), va a ser más significativa que una con rango (3) y así sucesivamente.

De análisis anterior se determina que las variables predictoras a utilizar en la regresión lineal múltiple son PPARCIALT1 y PARCIALLECCION

Figura 20

Regresión Lineal Múltiple, Variables Predictoras

```

In [36]: from sklearn.linear_model import LinearRegression

#Determinacion de las variables predictoras

In [37]: X_PREDICTORAS=X[["PPARCIALT1","PARCIALLECCION"]]

In [38]: lm=LinearRegression()
lm.fit(X_PREDICTORAS,Y)
Out[38]: LinearRegression()

In [39]: lm.intercept_
Out[39]: 22.077339421919874

In [40]: lm.coef_
Out[40]: array([0.1360487 , 0.57745595])

In [41]: lm.score(X_PREDICTORAS,Y)
Out[41]: 0.8449446473635476

In [41]: lm.score(X_PREDICTORAS,Y)
Out[41]: 0.8449446473635476

In [42]: prediccion_rendimiento_multi=lm.predict(pd.DataFrame(dataframetraining[["PPARCIALT1","PARCIALLECCION"]]))

In [43]: prediccion_rendimiento_multi
Out[43]: array([78.33465944, 79.13007663, 60.23996054, ..., 60.87230386,
71.55985891, 68.3307276 ])

In [44]: SSD=(sum(dataframetraining["NOTABLOQUE"] - prediccion_rendimiento_multi)**2)
SSD
Out[44]: 7.901198514950976e-22

In [45]: RSE2=np.sqrt(SSD/(len(dataframetraining)-2-1))

In [46]: RSE2
Out[46]: 7.007586457630192e-13

In [47]: error_multilineal=RSE2/parcial_media
error_multilineal
Out[47]: 1.1215917790361989e-14

In [48]: porcentaje= error_multilineal * 100

In [49]: porcentaje
Out[49]: 1.121591779036199e-12

```

Nota: Regresión lineal múltiple porcentaje de error.

Como se puede observar el resultado de este análisis indica que el error de este modelo es del 1.12×10^{-12} , lo cual indica que este es el modelo que se debe aplicar para el entrenamiento. Con este resultado y basados en los acuerdos sobre evaluación de modelos, se descartan de este análisis aquellos que se encuentran listados luego del modelo lineal múltiple.

Prueba del Modelo

Como se había indicado en el inicio, se procedió a partir los datos en dos Datasets, uno correspondiente a los datos para el entrenamiento y otros para probar el modelo, en esta fase se realiza las pruebas del modelo lineal múltiple aplicando el mismo sobre el data set de pruebas lo cual da como resultado lo siguiente:

Figura 21

Prueba de porcentaje de error

```

In [102]: prueba_rendimiento_multi=lm.predict(pd.DataFrame(dataframetesting[["PPARCIALT1", "PARCALLECCION"]]))

In [103]: SSDPRUEBA=(sum(dataframetesting["NOTABLOQUE"] - prueba_rendimiento_multi)**2)
SSDPRUEBA
Out[103]: 10914.472791783788

In [104]: RSEPREUBA=np.sqrt(SSDPRUEBA/(len(dataframetesting)-2-1))
RSEPREUBA
Out[104]: 5.217100405192985

In [106]: error_multilineal_prueba=RSEPREUBA/parcial_media
error_multilineal_prueba
Out[106]: 0.0836021671158547

In [108]: porcentaje_error_prueba= error_multilineal_prueba * 100

In [109]: porcentaje_error_prueba
Out[109]: 8.36021671158547

```

Nota: Código para determinar el porcentaje de error.

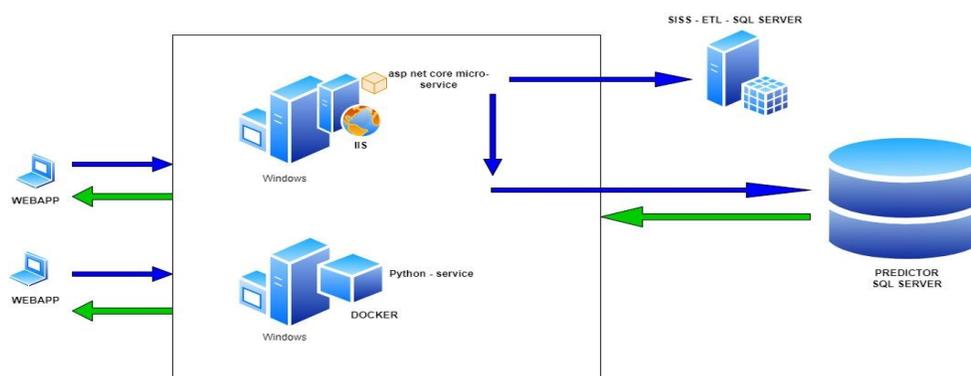
Las pruebas arrojan un resultado del 8.36% de error lo cual, una vez acordado con los representantes de la institución es un resultado aceptable para un punto de partida del proyecto.

Arquitectura de la Solución

Para el sistema predictor se plantea la siguiente arquitectura:

Figura 22

Arquitectura



Nota: Arquitectura de la aplicación.

El sistema consta de los siguientes componentes:

- Front End en angular.
- Back end un micro servicio en aspnet core usando EF 6, que será el encargado de realizar los CRUD necesarios sobre la base PREDICTOR (SQL – SERVER).

- El micro – servicio al mismo tiempo es el encargado de invocar la ejecución de un Job que ejecuta un ETL que carga la información de las notas de los estudiantes al repositorio predictor.
- Un servicio en Python Dockerizado, que contiene el algoritmo del modelo de Machine Learning (Regresión lineal múltiple) que será utilizado para predecir el rendimiento en virtud de dos variables predictoras tal cual como se identificó en el análisis previo.
- Un ETL SQL, que es el encargado de cargar la información de los estudiantes y sus notas y registrarlos en la base de datos predictor.
- Un job sql agent que está a espera de ser ejecutado por intermedio de un procedimiento almacenado.
- Una base de datos SQL server que será la encargada de almacenar la información necesaria para dar forma.

UX Diseño Hipermedial

✓ Fase de investigación

Para esta parte previa al desarrollo, se apalanca las actividades en el análisis preliminar realizado para el entendimiento del dominio del negocio dentro de la metodología KDD. El resultado de este análisis corresponde al punto de partida para la elaboración.

Del UX asociado al software con el fin de cumplir las expectativas de los usuarios y alcanzar el objetivo planteado.

✓ Fase de organización

Para esta esta fase, tomando como insumo el análisis del negocio realizado se procede a definir las funcionalidades a implementar, los flujos de navegación enfocados en el usuario y otros aspectos que apalanquen el cumplimiento del objetivo establecido.

De lo anteriormente expuesto se desprende el siguiente backlog de funcionalidad.

Tabla 6

Tabla de Funcionalidad

FUNCIONALIDAD	ROL
Permitir el logueo del usuario mediante una clave y contraseña.	USUARIO
Emitir un mensaje de error cuando el usuario y clave no coincidan con las residentes en el base de datos.	USUARIO
Permitir al administrador, agregar, dar de baja a un usuario, modificar la información del usuario.	ADMINISTRADOR
Mostrar opciones dentro del sistema en función del tipo de rol que accede.	USUARIO / ADMINISTRADOR
Al momento de ingresar mostrar en algún lugar el ROL y nombre correspondiente al usuario que accede.	USUARIO / ADMINISTRADOR
Mostrar una opción para cerrar la sesión del usuario.	USUARIO / ADMINISTRADOR
Construir la aplicación usando WEBAPI.	SISTEMA

Manejar logs de invocación de servicio a nivel backend.	SISTEMA
---	---------

Disponer de un menú amigable donde se muestren las opciones
Administrador.

Dar a lugar a la visualización de las opciones para administrar
usuarios

Procesar

USUARIO /

Dar a lugar a la invocación del Job que ejecuta la carga de
registros académicos desde un Excel hacia la base de datos

ADMINISTRADOR

Predecir y Ejecutar el modelo predictor

Reporte: Muestra un reporte básico de la ejecución de la
predicción.

Mostrar una opción que permita al usuario identificar a que curso
pertenece la predicción a realizar

USUARIO

Mostrar un campo que permita al usuario registrar un mail al cual
enviar una síntesis de la predicción

USUARIO

Mostrar campos para ingresar los valores de promedio total tareas
y promedio total lecciones que servirán para la ejecución del
modelo predictor

USUARIO

Registrar la fecha de la ejecución de la predicción

SISTEMA

Disponer de una opción que permita al usuario ingresar la información de los temas asociados a los promedios totales ingresados previo a predicción.

USUARIO

Registrar el nombre, el curso, la materia y el valor de predicción por alumno

SISTEMA

Realizar la carga masiva de información de los registros estudiantiles previo a la predicción

SISTEMA

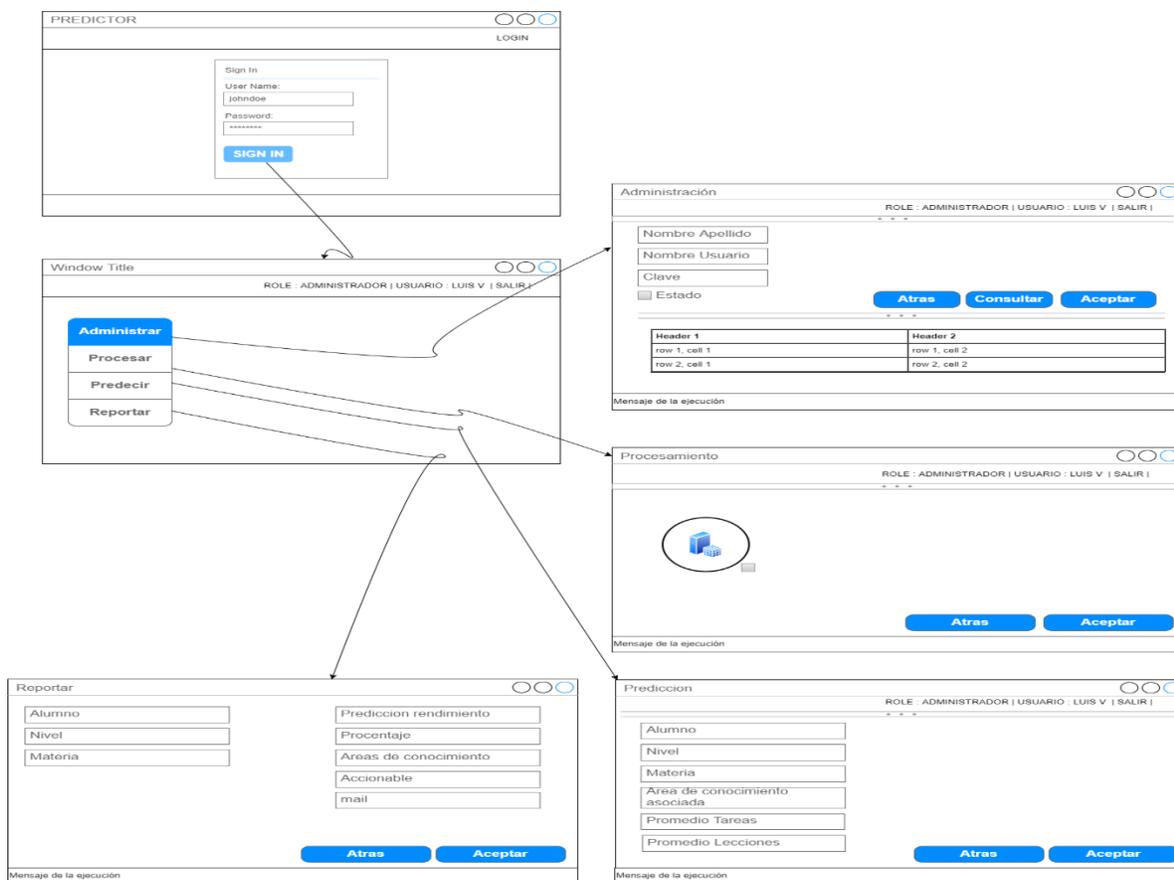
Disponer de un medio que permita ejecutar un modelo de regresión lineal múltiple usando dos variables predictoras.

SISTEMA

✓ **Fase Prototipado**

Figura 23

Fases del sistema



Nota: Fases del sistema.

✓ **Fase de Pruebas**

En esta fase ya se hace necesario instalar las siguientes herramientas para proceder con la construcción del aplicativo tanto a nivel de back-end, front-end, base de datos y análisis de datos.

- SQL-SERVER
- JUPYTER

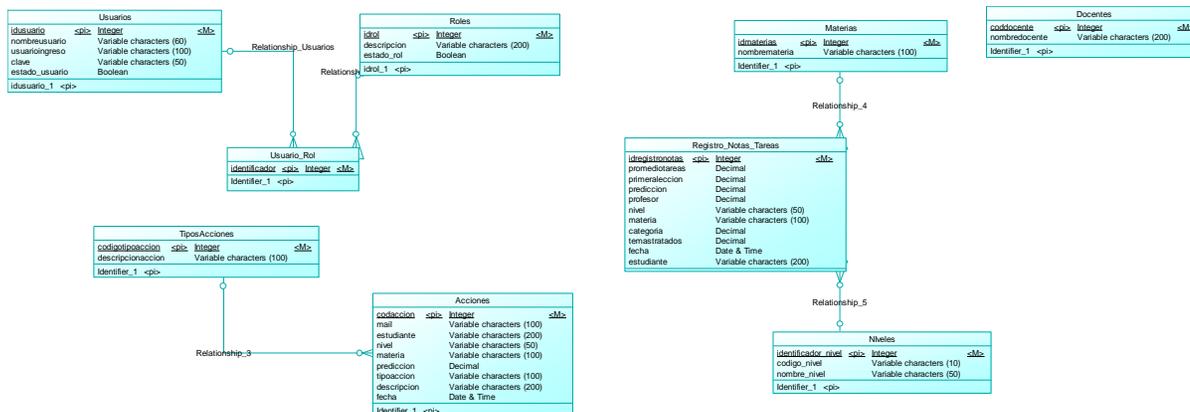
- Visual Estudio 2022
- ANACONDA
- PYTHON
- ANGULAR
- DOCKER

Una vez instaladas las herramientas de desarrollo se procede a construir cada uno de los componentes que darán forma al sistema y se procede a describir el ámbito de acción de cada uno de ellos.

SQL-SERVER .- Este será el motor de base de datos, que albergara dentro de sí una base en modelo entidad relación donde se podrá almacenar la información asociada a usuarios, roles, perfiles, notas académicas, predicción, y acciones tomadas en función de los valores de predicción de cada estudiante.

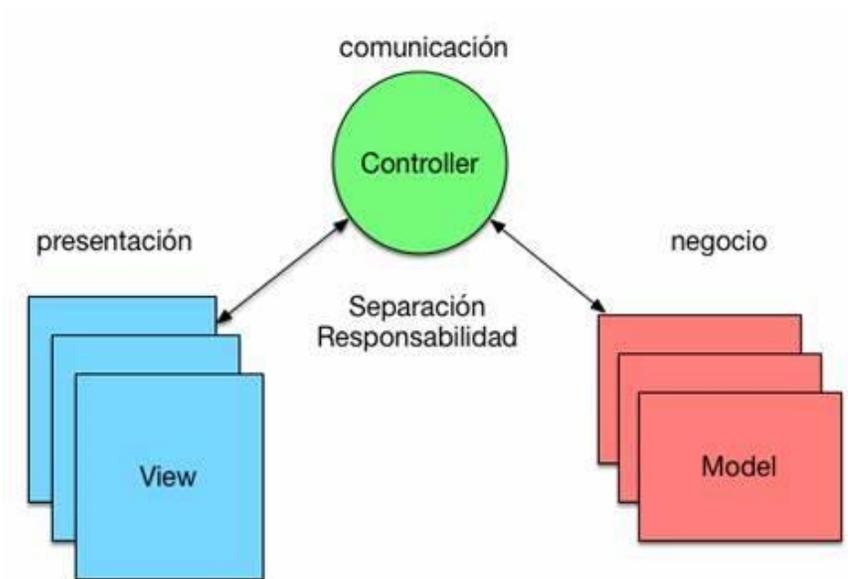
Figura 24

Modelo Relacional



Nota: Modelo Entidad Relación.

La imagen precedente da el punto de partida para la definición de la forma en la cual se va a construir los microservicios que van a interactuar con el modelo de base de datos, para este particular, se optó por utilizar visual estudio en su versión community edition, a la vez se usara c# como lenguaje de programación y un patrón de diseño MVN, Model – View – Controller para la construcción de los API-REST que se dispondrán para el consumo de front end.

Figura 25*Patrón MVC API REST*

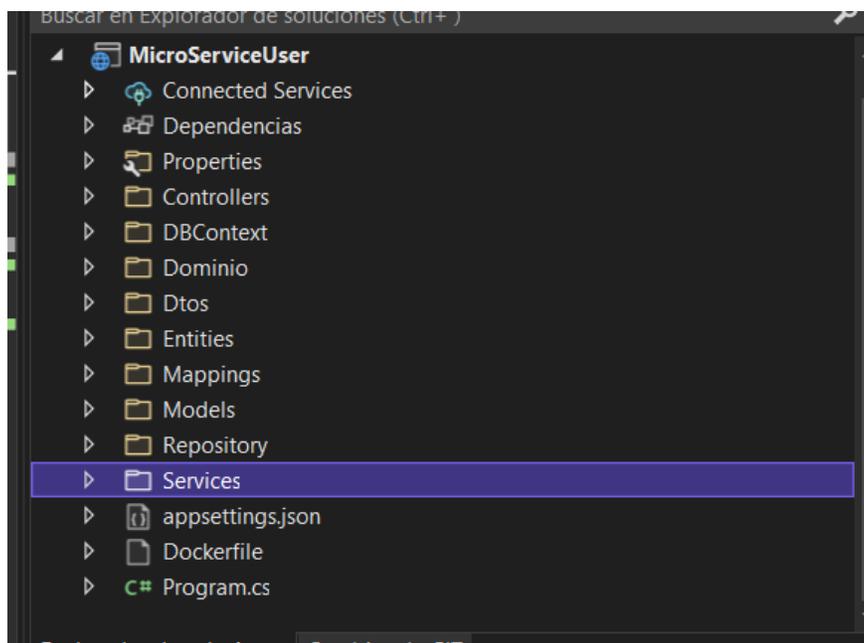
Nota. Patrón MVC API REST. Fuente: Álvarez. C. (2016). Patrón MVC API REST [Imagen].

Arquitecturajava. <https://www.arquitecturajava.com/patron-mvc-arquitectura-cliente-vs-servidor/>

Dato que la visión de los microservicios es que deben actuar sobre su propia base de datos se toma este concepto y se traslada a una sola API que expondrá los endpoint para que sean consumidos desde el front- end. La estructura de API-REST se muestra a continuación:

Figura 26

Estructura proyecto API-REST



Nota: Estructura proyecto API-REST.

Controllers. - Se encargan de exponer los métodos API REST para el consumo de Front-End mediante cadenas JSON, como por ejemplo:

```
Data:{"campo1","valor1","campo2","valor2"}
```

DBContext. - Se encarga de manejar el contexto de conexión a la base de datos SQL de la aplicación para este particular se hace uso de entity framework como ORM para exponer la base datos a la lógica del negocio.

Dominio. - En esta capa se declara las interfaces que van a moldear el negocio, manejar el microservicio y permitirá utilizar un concepto denominado inyección de dependencias que van aportar a tener un código más limpio, sostenible y eficiente.

Entities. - Entidades que están mapeadas hacia las tablas de la base de datos

Services. - Esta capa es la encargada de manejar la lógica de negocio de la aplicación, realizar validaciones de datos asociadas al negocio, para este particular solo servirá de puente entre los controladores y la capa repositorio mediante el uso de inyección de dependencias.

Repository. - Esta capa contiene la lógica de acceso a la base de datos, las consultas a las tablas mediante el uso de LINQ. La consulta se la realiza sobre las clases que se encuentran en la capa Entities.

Para la construcción de cada uno de endpoint en cada controlador se utilizaron métodos HTTP POST que permitan en cada invocación tener una respuesta a la misma en el siguiente formato.

Figura 27

Ejemplo de salida metodo HTTP POST Microservicio

```
{
  "tipoaccion": [
    {
      "codaccion": 1,
      "descripcionaccion": "REFUERZO"
    },
    {
      "codaccion": 2,
      "descripcionaccion": "CORRECTIVA"
    }
  ],
  "error": {
    "codigoError": "0",
    "mensajeError": "OK"
  }
}
```

Nota: Ejemplo de salida metodo HTTP POST Microservicio.

El siguiente es un listado de los end point programados para la aplicación.

Tabla 7

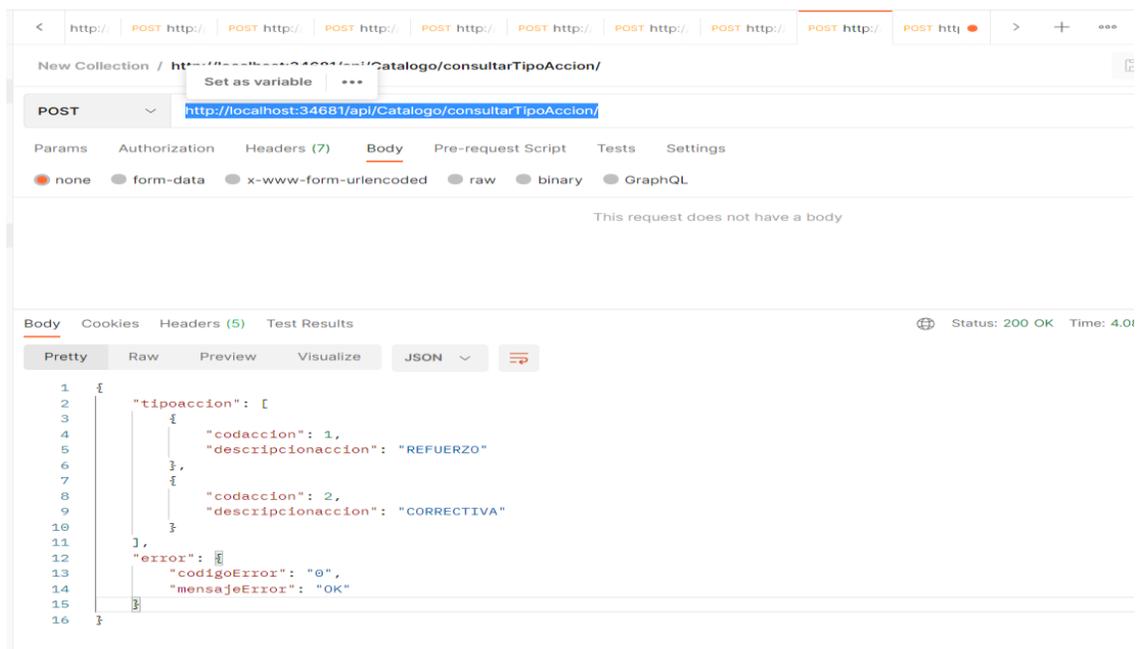
Métodos Api Rest

NOMBRE	URL	METOD O HTTP
CargarNotas	http://localhost:8081/api/Academic/cargarNotas/	HTTP- POST
CargarPromediosAlumnos	http://localhost:34681/api/Academic/cargarPromediosAlumnos/	HTTP- POST
Cargararchivoxls	http://localhost:34681/api/Academic/cargararchivoxls/	HTTP- POST
ConsultarMaterias	http://localhost:34681/api/Catalogo/consultarMaterias/	HTTP- POST
ConsultarNiveles	http://localhost:34681/api/Catalogo/consultarNiveles/	HTTP- POST
ConsultarTipoAccion	http://localhost:34681/api/Catalogo/consultarTipoAccion/	HTTP- POST
RegistrarPrediccion	http://localhost:34681/api/Prediccion/registrarPrediccion/	HTTP- POST
Reporte rendimient	http://localhost:34681/api/Prediccion/reporterendimiento/	HTTP- POST

RegistrarAcciones	http://localhost:34681/api/Prediccion/registrarAcciones/	HTTP- POST
ConsultarUsuario	http://localhost:34681/api/User/consultarUsuario/	HTTP- POST
Consultartodosusuarios	http://localhost:34681/api/User/consultartodosusuarios/	HTTP- POST
Modificarusuario	http://localhost:34681/api/User/modificarusuario/	HTTP- POST
RegistrarUsuario	http://localhost:34681/api/User/registrarUsuario/	HTTP- POST
Consultarroles	http://localhost:34681/api/User/consultarroles/	HTTP- POST

Pruebas Y Despliegue Del Microservicios.

Para la puesta en ejecución de los microservicios existen dos opciones para ejecutar puede ser directamente desde el IDE de desarrollo visual estudio que posee un servidor web interno denominado IIS Express o desplegarlo en Internet Information Services levantado en entorno de desarrollo. Para la prueba de los end point se utilizó postman.

Figura 28*Postman*

Nota: Prueba de Microservicios.

Figura 29*Despliegue de microservicio en IIS Local*

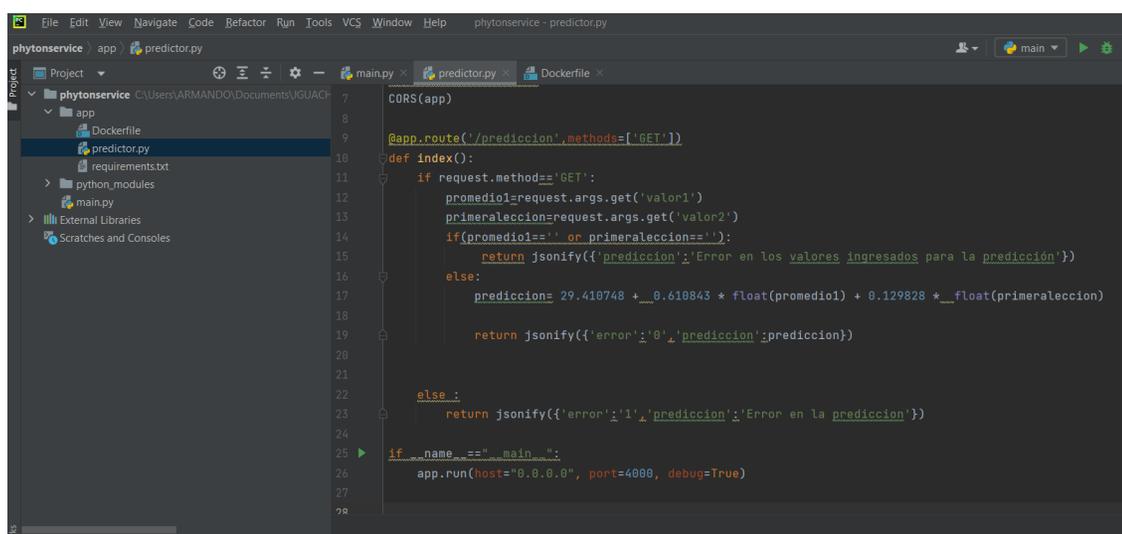
Nota: Despliegue de microservicio en IIS Local.

Construcción Servicio Rest predicción

Una vez realizado el análisis de datos usando anaconda, y su herramienta jupyter, se procede a trasladar el resultado de ese análisis a un servicio que será invocado desde front end. Para efectos de este desarrollo se utilizará Python como lenguaje de programación y PyCharm en su versión community como IDE de desarrollo.

Figura 30

Servicio Python predicción rendimiento



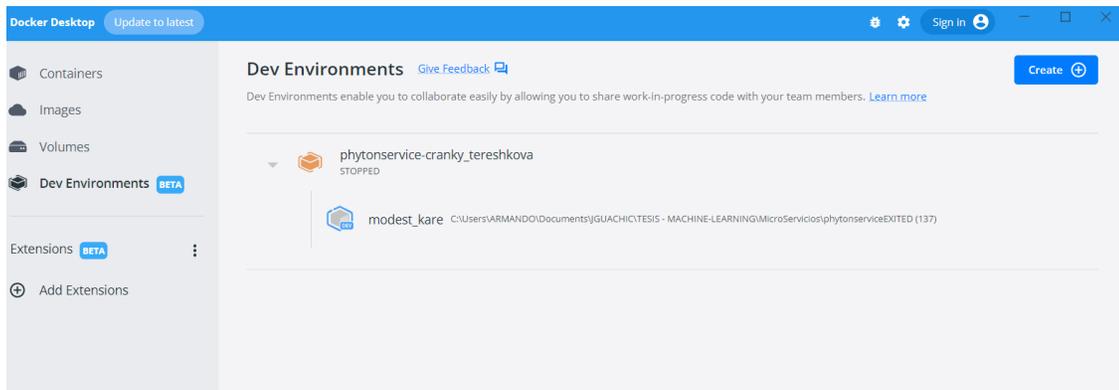
```
File Edit View Navigate Code Refactor Run Tools VCS Window Help phytoservice - predictor.py
phytoservice app predictor.py
Project
  phytoservice C:\Users\ARMANDO\Documents\JGUACH
    app
      Dockerfile
      predictor.py
      requirements.txt
    python_modules
    main.py
    External Libraries
    Scratches and Consoles
7 CORS(app)
8
9 @app.route('/prediccion', methods=['GET'])
10 def index():
11     if request.method == 'GET':
12         promedio1=request.args.get('valor1')
13         primeraLeccion=request.args.get('valor2')
14         if(promedio1==' ' or primeraLeccion==' '):
15             return jsonify({'prediccion':'Error en los valores ingresados para la prediccion'})
16         else:
17             prediccion= 29.410748 + __0.610843 * float(promedio1) + 0.129828 * __float(primeraleccion)
18             return jsonify({'error':'0','prediccion':prediccion})
19
20
21
22     else :
23         return jsonify({'error':'1','prediccion':'Error en la prediccion'})
24
25
26 if __name__ == "__main__":
27     app.run(host="0.0.0.0", port=4000, debug=True)
28
```

Nota: Servicio Python predicción rendimiento.

Este servicio una vez construido se procede a desplegarlo en un contenedor Docker con el propósito de que se pueda consumir desde la aplicación front end.

Figura 31

Despliegue servicio python en Docker



Nota: Despliegue servicio python en Docker.

Tabla 8

Archivo Docker

Contenido archivo Docker

FROM alpine:3.10

RUN apk add --no-cache python3-dev \

&& pip3 install --upgrade pip

WORKDIR /app

COPY ./app

RUN pip3 --no-cache-dir install -r requirements.txt

CMD ["python3" , "predictor.py"]

Tabla 9*Archivo Requeriments*

Contenido del archivo requirements
certifi==2022.6.15
charset-normalizer==2.1.0
click==8.1.3
colorama==0.4.5
Flask==2.1.3
Flask-Cors==3.0.10
idna==3.3
itsdangerous==2.1.2
Jinja2==3.1.2
MarkupSafe==2.1.1
requests==2.28.1
six==1.16.0
urllib3==1.26.11
Werkzeug==2.1.2

Para iniciar la ejecución del servicio se procede con el siguiente comando en el entorno de desarrollo.

```
docker run -it --publish 7000:4000 flaskapp
```

Figura 32*Contenedor Docker Servicio Python*

NAME	IMAGE	STATUS	PORT(S)	STARTED
gracious_bouman e11e3f6e6ad7	flaskapp	Exited	7000	
objective_banzai 225846b82d51	flaskapp	Exited	7000	
unruffled_hodgkin f3de0754a2d2	<none>	Exited (1)	7000	
competent_morse c4e2d68d22b5	<none>	Exited (1)	7000	
mystifying_heisenberg 5cfe4c909e76	<none>	Exited	7000	
optimistic_shannon ...	<none>	Exited	7000	

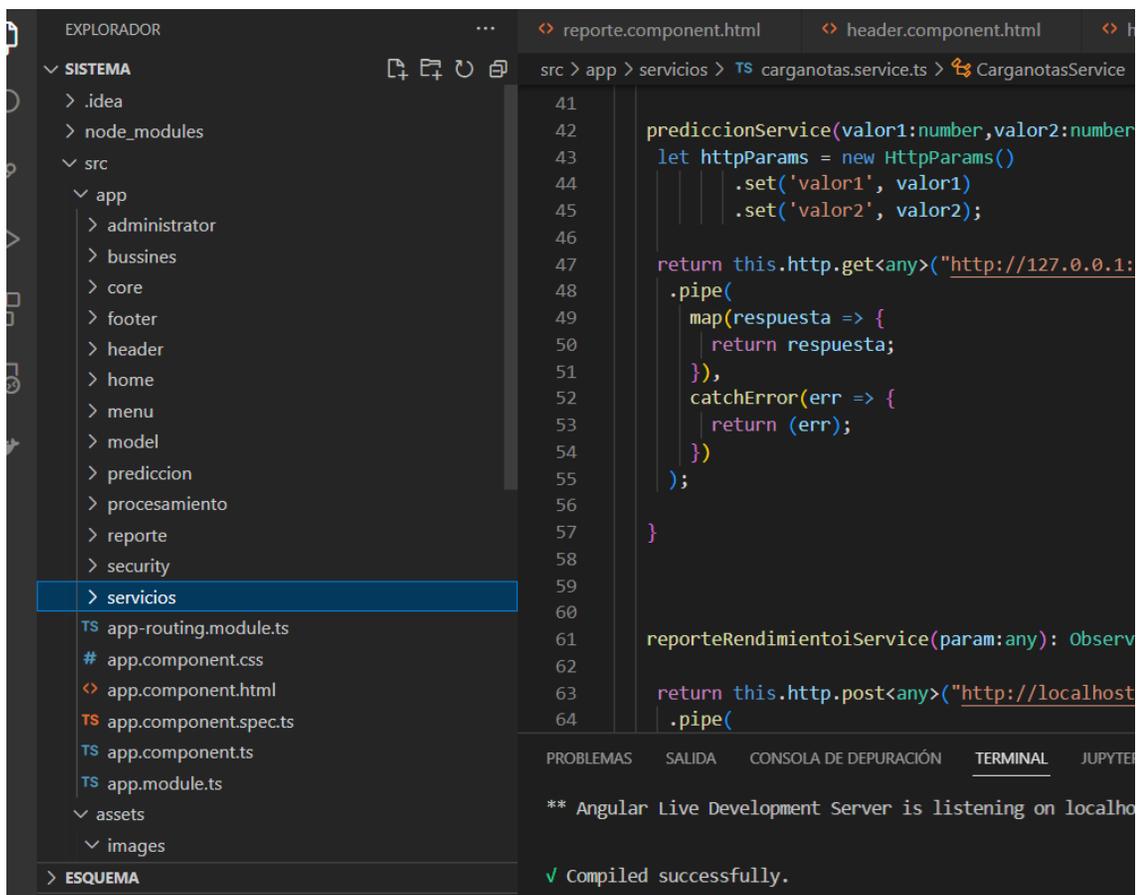
Nota: Contenedor Docker Servicio Python.

Construcción aplicación web Angular. - Predictor

Para la construcción del aplicativo web se utiliza como FRAMEWORK de desarrollo Angular en su versión 12. Angular es un Framework desarrollado por GOOGLE que permite desarrollar aplicaciones WEB más enriquecidas, que brindan una mejor experiencia de usuario. El patrón de diseño utilizado para construcción del aplicativo web es MVC.

Figura 33

Web Angular



The screenshot shows an IDE with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with a 'servicios' folder highlighted. The code editor shows the following TypeScript code:

```
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64
```

```
prediccionService(valor1:number,valor2:number  
  let httpParams = new HttpParams()  
  .set('valor1', valor1)  
  .set('valor2', valor2);  
  
  return this.http.get<any>("http://127.0.0.1:  
    .pipe(  
      map(respuesta => {  
        return respuesta;  
      }),  
      catchError(err => {  
        return (err);  
      })  
    );  
  }  
  
reporteRendimientoiService(param:any): Observ  
  return this.http.post<any>("http://localhos  
    .pipe(  
      map(respuesta => {  
        return respuesta;  
      }),  
      catchError(err => {  
        return (err);  
      })  
    );  
  }  
};
```

The terminal at the bottom shows the following output:

```
PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL JUPYTER  
** Angular Live Development Server is listening on localho  
✓ Compiled successfully.
```

Nota. Angular.

Para este desarrollo se tomaron las siguientes consideraciones dictadas por parte del usuario y las cuales dieron una base para la elección de Angular como framework para la construcción del front-end.

El aplicativo se lo debe construir como WEB -APP. Aplicaciones web que se muestran en una sola página.

Se debe hacer uso de herramientas que mejoren la experiencia de usuario. Se complementa el desarrollo con Bootstrap.

Utilizar librerías actualizadas que ayuden a agilizar el desarrollo y presentar un demo en forma más ágil. Se acopla el uso de nodejs al desarrollo.

Se debe presentar un menú de navegación, se usó el routing dentro de Angular para disponer de un ruteo entre las diferentes opciones de la aplicación.

CAPITULO IV

Evaluación

En este capítulo se procede a realizar la evaluación de los resultados obtenidos durante las pruebas del sistema con el fin de aportar elementos complementarios que refuercen lo obtenido durante las pruebas realizadas sobre el sistema.

En tal virtud una vez realizada las pruebas se puede indicar que el sistema predictor, cumple el objetivo planteado durante la fase inicial del análisis, el cual está orientado a ofrecer una visión temprana del porcentaje de rendimiento académico de un estudiante con el propósito de poder tomar medidas anticipadas que permitan al alumno cumplir con sus metas académicas.

Cabe mencionar que con este sistema se planta la semilla para una automatización integral del proceso de evolución académica, dado que al ser un demo está sujeto a modificaciones posteriores en cuando la institución se recupere de las pérdidas económicas derivadas de los últimos eventos político-sociales, por la que paso el país.

Se hace énfasis que si bien es cierto se da un paso muy importante con este sistema, mucho de los procedimientos todavía son manuales y sujetos a la mejora que las nuevas tecnologías pueden aportar.

A continuación, se muestra un extracto de los resultados que el sistema arroja versus los obtenidos por el alumno en periodo escolar 2021.

Tabla 10*Evaluación 1*

 Alumno: Aguas Reyes Alejandro André

 Predicción Rendimiento:

Codigo	1
Alumno	Aguas Reyes Alejandro André
Promedio tareas	50.66 ✓ <small>Valido</small>
Primera leccion	53 ✓ <small>Valido</small>
Posible rendimiento	68.666311

 Rendimiento Real 2021: 63.46

N	ESTUDIANTE	49	35	68	50.66	53	79	53	61.66	55	55.00	70	90	80.00	NOTA BLOQUE	70	63.46
1	Aguas Reyes Alejandro André				50.66												

 Porcentaje error: 5.20

Tabla 11*Evaluación 2*

 Alumno: carranza Santacruz David Alejandro

 Predicción Rendimiento:

Codigo	3
Alumno	carranza Santacruz David Alejandro
Promedio tareas	41.33 ✓ <small>Valido</small>
Primera leccion	26 ✓ <small>Valido</small>
Posible rendimiento	48.66819400000001
Categoria	RIESGO

Rendimiento Real 2021: 45.66

3	carranza Santacruz David Alejandro	48	15	6	41,33	26	64	21	37,00	27	27,00	80	75	77,00	46	45,66
---	------------------------------------	----	----	---	-------	----	----	----	-------	----	-------	----	----	-------	----	-------

Porcentaje error: 4.33

Tabla 12

Evaluación 3

Alumno: Castro Taco Gina Tatiana

Predicción Rendimiento:

Codigo	4
Alumno	Carrillo López Jorge Washinton
Promedio tareas	62.66 ✓ Valido
Primera leccion	74 ✓ Valido
Posible rendimiento	84.22040199999999
Categoria	ALTO

Rendimiento Real 2021: 85.26

S/ Castro Taco Gina Tatiana	45	46	97	62.66	74	100	74	82.66	100	100.00	81	90	85.00	96	85.26
-----------------------------	----	----	----	-------	----	-----	----	-------	-----	--------	----	----	-------	----	-------

Porcentaje error: 1.04

Tabla 13

Evaluación 4

Alumno: Castro Taco Gina Tatiana

Predicción Rendimiento:

Codigo	16
Alumno	Rivadeneir Montenegro Daniela
Promedio tareas	58 ✓ Valido
Primera leccion	68 ✓ Valido
Possible rendimiento	79.776376
Categoria	ALTO

Rendimiento Real 2021: 75.73

W	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U		
7	16	Rivadeneir Montenegro Daniela	63	50	61	58,00	68	100	62	76,66	77	77,00	81	85	83,00	84	75,73
8	17	Cristianal Pizarro Vizcaino	10	11	11	11,00	11	11	11	11,00	11	11,00	11	11	11,00	11	11,00

Porcentaje error: 4.04

De las evidencias mostradas en este capítulo se puede dar por entendido que la evaluación de nuestro sistema es positiva dentro del ámbito de estudio y del objetivo planteado para este proyecto.

En la mayoría de los casos el margen de error oscila entre uno y cinco puntos lo cual es un valor aceptable en virtud de la cantidad de datos proporcionados por la institución para este proyecto.

CAPITULO V

Conclusiones

- Si bien es cierto la evaluación del sistema proporciona evidencia que el margen de error oscila entre uno y cinco puntos, no es menos cierto que puede darse el caso que ante la presencia de nuevos escenarios este valor crezca debido a la que la cantidad de información histórica proporcionada por la institución con la que se entrenó y evaluó nuestro algoritmo fue pequeña en relación a proyectos de machine learning de gran envergadura. Sin embargo, cabe mencionar que los modelos de machine learning son perfectibles en el tiempo en función de cuanto más crezca la data con la que se puede entrenar nuestro modelo.
- Si bien es cierto el uso de dummy data sets es muy empleado durante el análisis de datos para poder completar la información con la cual se pueda entrenar a nuestro modelo, se debe aceptar que este no aportara valor al momento de entrenar a nuestro modelo; por tal motivo siempre se debe dar prioridad a realizar el análisis con la mayor cantidad de datos históricos posibles y cuando estos no estén disponibles proceder a usar dummy data sets.
- Anaconda y Python ofrecen una gran variedad de librerías que permiten a las personas que inician en machine learning realizar un análisis de datos muy apegado a la metodología elegida, en este caso KDD.
- EL patrón de diseño MVC Model – View – Controller, es una buena opción a la hora de construir API REST que expongan funcionalidad de negocio.
- Al momento de afrontar un proyecto de machine learning, se debe estar conscientes que más del 50% del tiempo será invertido en realizar el análisis de los datos históricos con los cuales se construirá la data set training y data set testing.

- Angular como framework de desarrollo de WEBAPP proporciona gran variedad de paquetes y librerías que permite dotar aplicaciones web de contenido enriquecido que se traduce en una mejor experiencia de usuario, que a la larga es la razón de ser del desarrollo del software.

Recomendaciones

Como recomendaciones se pueden listar las siguientes:

- Se recomienda dar continuidad al proyecto una vez que se haya superado la situación económica por la cual muchas empresas e instituciones educativas se han visto afectadas debido a los últimos acontecimientos político-sociales.
- Abordar la mejora continua del modelo predictivo mediante la aportación de más información y escenarios de entrenamiento, ampliando el conjunto de estudio.
- Realizar un análisis de costo beneficio a futuro sobre la posibilidad de desplegar dicho aplicativo en la nube con el propósito de monetizar la inversión realizada en el desarrollo.

Bibliografía

Foster, E. (2014). *Software Engineering: A Methodical Approach*. Apress.

Bermeo Calle, I. A., & Chicaiza Sangurima, J. A. (2020, agosto 31). *Repositorio ESPE*.

Repositorio ESPE: <http://repositorio.espe.edu.ec/bitstream/21000/23327/1/T-ESPEL-ENI-0448.pdf>

Bonaccorso, G. (2018). *Machine Learning Algorithms, 2nd Edition*. Packt Publishing.

Caisa Tipan, F. C. (2016, Octubre). *Repositorio UTC*. Repositorio Digital Universidad Técnica de Cotopaxi: <http://repositorio.utc.edu.ec/bitstream/27000/3784/7/PI-000127.pdf>

EBF. (2019, Septiembre 11). *EBF*. EBF : <https://ebf.com.es/blog/ventajas-y-desventajas-de-las-metodologias-agiles-y-su-aplicacion-en-el-trabajo/>

EL SEMBRADOR. (2017, Abril). EL CONSTRUCTIVISMO: COMO MODELO EDUCATIVO DE NUESTRO COLEGIO. p. 9.

Escamilla, B. (2022, Marzo). *Morio*. <https://morioh.com/p/e5a235c363c9>

García, N. (2020, Mayo). *Máxima Formación*. Máxima Formación :

<https://www.maximaformacion.es/blog-dat/que-son-los-arboles-de-decision-y-para-que-sirven/>

González , A. (2021). *cleverdata*. cleverdata: <https://cleverdata.io/que-es-machine-learning-big-data/>

GONZÁLEZ , G. S. (2022, Enero). *Machine Learning Studio*. Machine Learning Studio:

<https://mlstudio.jaol.net/principal-components-analysis-pca-better-explained/#:~:text=El%20an%C3%A1lisis%20de%20componentes%20principales%20>

28PCA%29%20es%20un,de%20manera%20efectiva%20en%20menos%20columnas%
20de%20caracter%C3%ADsticas.?msclkid=f8ba5408cfd

Hawkins. (2004). *aicor*. aicor: <https://www.aicor.com/machine-learning/>

Microsoft. (2022). ¿Qué es un modelo de aprendizaje automático? *¿Qué es un modelo de aprendizaje automático?*

Newman, S. (2021). *Building Microservices, 2nd Edition*. O'Reilly Media, Inc.

Petrov, I. (2018, Mayo 21). *TECHNOSIGHT*. TECHNOSIGHT:

<https://www.technosight.co.uk/post/the-10-algorithms-machine-learning-engineers-need-to-know>

Segovia Vargas, M. J., & Camacho Miñano, M. (2012). ¿Qué indicadores económico-financieros podrían condicionar la decisión del experto independiente sobre la supervivencia de una empresa en su fase preconcursal? Evidencia empírica en España. *13(32)*, 97-119.

Uluca, D. (2020). *Angular for Enterprise-Ready Web Applications, 2nd Edition*. Packt Publishing.

VELASCO, E., & MAMARANDI, J. (s.f). *Repositorio Espe*. Repositorio Espe:

<http://repositorio.espe.edu.ec/bitstream/21000/9296/1/AC-ESPEL-ENI-0334.pdf>

woodyweb. (2015, Agosto 24). *woodyweb*. woodyweb:

<https://woodyweb.wordpress.com/2015/08/24/metodologias-de-desarrollo/>

Yan, J., & Yan, Y. (2018). *Hands-On Data Science with Anaconda*. Packt Publishing.