



UDEC
UNIVERSIDAD DE
CUNDINAMARCA

Regresión Lineal

Predecir la nota final con base en horas de estudio y horas de sueño.

Juan Diego Ortiz Baquero

Gestor del conocimiento

Machine Learning [Conceptos]

Edison Gustavo Canon Varela

Universidad de Cundinamarca – Extensión Chía

Facultad de Ingeniería

Ingeniería de Sistemas y Computación

I. Resumen

Este informe presenta la implementación de una regresión lineal múltiple aplicada a un conjunto de datos sintéticos que simulan el rendimiento académico en función de dos variables independientes: horas de estudio y horas de sueño. El modelo se integra en una aplicación web desarrollada con Flask, utilizando plantillas Jinja2 y estilos CSS con Bootstrap para una visualización clara y estructurada. La solución incluye una pestaña “Regresión lineal” con dos submenús: conceptos teóricos y ejercicio práctico. El trabajo se gestionó colaborativamente en GitHub mediante la rama A4_RegresionLineal, con commits ordenados, Pull Request y merge a la rama principal. Palabras clave: regresión lineal, Flask, aprendizaje supervisado, GitHub, APA.

Palabras clave: aprendizaje supervisado, Flask, Jinja2, GitHub, APA.

II. Introducción

El aprendizaje supervisado permite entrenar modelos a partir de datos etiquetados para realizar predicciones con valor operativo. En esta actividad se implementa un modelo de regresión lineal múltiple para predecir la nota final de un estudiante en función de sus hábitos de estudio y sueño. El modelo se despliega en una aplicación web que permite visualizar los datos de entrenamiento, la línea de regresión, ingresar nuevos valores y obtener predicciones en tiempo real.

III. Objetivos

- 3.1. Implementar un modelo de regresión lineal con dos variables independientes.
- 3.2. Integrar el modelo en una aplicación Flask con visualización estructurada.

3.3. Gestionar el proyecto en GitHub mediante la rama A4_RegresionLineal.

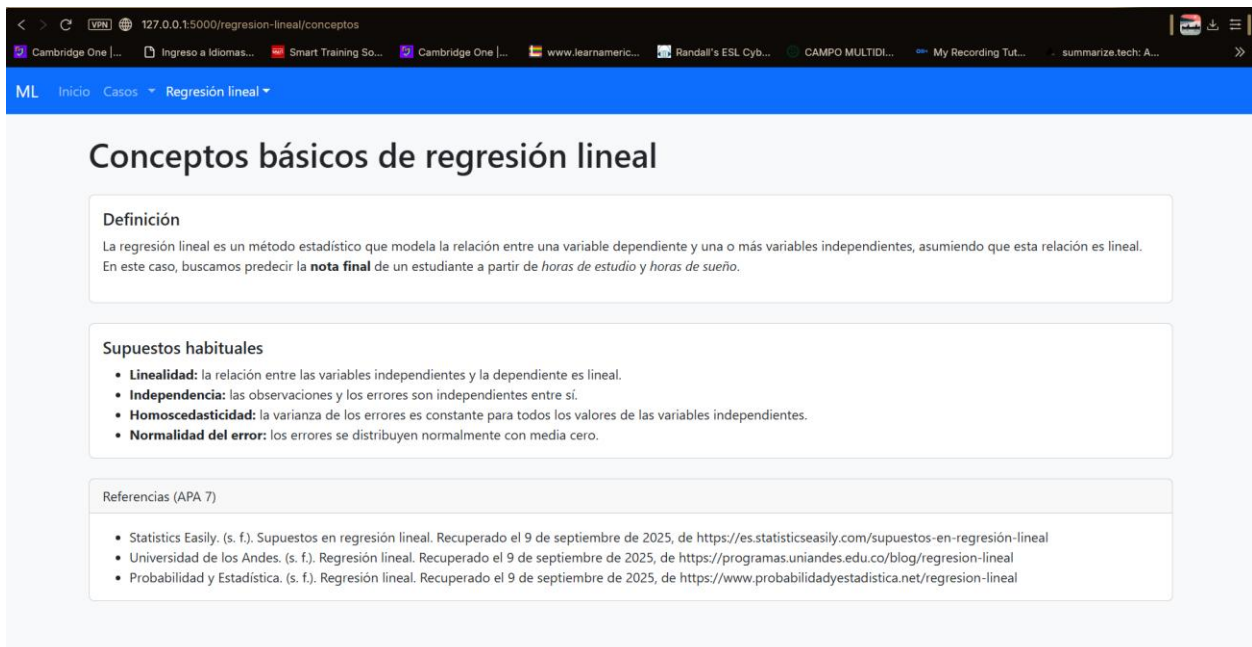
3.4. Documentar el proceso con capturas, código y referencias en formato APA.

IV. Investigación – Conceptos Básicos de Regresión Lineal

La regresión lineal es un método estadístico que permite modelar la relación entre una variable dependiente y una o más variables independientes. Su objetivo es encontrar una función lineal que minimice el error entre las predicciones y los valores reales.

Supuestos del modelo:

- **Linealidad:** la relación entre variables es lineal.
- **Independencia:** los errores son independientes entre sí.
- **Homocedasticidad:** la varianza de los errores es constante.
- **Normalidad del error:** los errores siguen una distribución normal.



The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000/regresion-lineal/conceptos'. The browser tabs include 'Cambridge One', 'Ingreso a Idiomas...', 'Smart Training So...', 'Cambridge One', 'www.learnameric...', 'Randall's ESL Cyb...', 'CAMPO MULTIDI...', 'My Recording Tut...', and 'summarize.tech: A...'. The page has a blue header with 'ML Inicio Casos Regresión lineal'. The main content area is titled 'Conceptos básicos de regresión lineal' and contains three sections: 'Definición', 'Supuestos habituales', and 'Referencias (APA 7)'. The 'Definición' section states that linear regression is a statistical method that models the relationship between a dependent variable and one or more independent variables, assuming a linear relationship. The 'Supuestos habituales' section lists four assumptions: Linealidad (the relationship is linear), Independencia (observations and errors are independent), Homocedasticidad (the variance of errors is constant), and Normalidad del error (errors are normally distributed with a mean of zero). The 'Referencias (APA 7)' section lists three references: Statistics Easily, Universidad de los Andes, and Probabilidad y Estadística, all dated September 9, 2025.

Definición

La regresión lineal es un método estadístico que modela la relación entre una variable dependiente y una o más variables independientes, asumiendo que esta relación es lineal. En este caso, buscamos predecir la **nota final** de un estudiante a partir de *horas de estudio* y *horas de sueño*.

Supuestos habituales

- **Linealidad:** la relación entre las variables independientes y la dependiente es lineal.
- **Independencia:** las observaciones y los errores son independientes entre sí.
- **Homocedasticidad:** la varianza de los errores es constante para todos los valores de las variables independientes.
- **Normalidad del error:** los errores se distribuyen normalmente con media cero.

Referencias (APA 7)

- Statistics Easily. (s. f.). Supuestos en regresión lineal. Recuperado el 9 de septiembre de 2025, de <https://es.statisticseasily.com/supuestos-en-regresion-lineal>
- Universidad de los Andes. (s. f.). Regresión lineal. Recuperado el 9 de septiembre de 2025, de <https://programas.uniandes.edu.co/blog/regresion-lineal>
- Probabilidad y Estadística. (s. f.). Regresión lineal. Recuperado el 9 de septiembre de 2025, de <https://www.probabilidadyestadistica.net/regresion-lineal>

Imagen 1

Página web con los conceptos básicos de regresión lineal y referencias en formato APA 7.

Referencias:

- Statistics Easily. (s. f.). Supuestos en regresión lineal. Recuperado el 9 de septiembre de 2025, de <https://es.statisticseasily.com/supuestos-en-regresión-lineal>
- Universidad de los Andes. (s. f.). Regresión lineal. Recuperado el 9 de septiembre de 2025, de <https://programas.uniandes.edu.co/blog/regresion-lineal>
- Probabilidad y Estadística. (s. f.). Regresión lineal. Recuperado el 9 de septiembre de 2025, de <https://www.probabilidadyestadistica.net/regresion-lineal>

V. Desarrollo – Entrenamientos y Predicción

El script `regression_model.py` realiza las siguientes tareas:

- **Carga de datos:** lee el archivo `notas.csv` o genera datos sintéticos si no existe.

```
El dataset contiene 50 registros, con las variables predictoras 'Horas de estudio' y 'Horas de sueño', y la variable objetivo 'Nota final'. El modelo de regresión lineal múltiple entrenado obtuvo un  $R^2$  de 0.493, lo que indica que explica aproximadamente el 49.3% de la variabilidad de la nota final. El intercepto es 3.7823, con coeficientes de 0.1066 para 'Horas de estudio' y 0.0591 para 'Horas de sueño'.
```

```
Flujo general del modelo:
```

1. Carga de datos: se lee el dataset desde CSV o se genera sintético si no existe.
2. Entrenamiento: se ajusta un modelo de regresión lineal múltiple con las variables definidas.
3. Predicción: se reciben nuevos valores de las variables independientes y se calcula la variable objetivo.

```
Ejemplo de predicción con 6h estudio y 7h sueño: 4.84
```

Figura 1. Evidencia de carga del dataset `notas.csv` en `regression_model.py`.

- **Entrenamiento:** ajusta un modelo de regresión lineal múltiple con `LinearRegression` de `scikit-learn`.

Modelo entrenado para predecir la **nota final** en función de las *horas de estudio* y *horas de sueño*.
Entrenado con 50 registros.
 R^2 del ajuste: 0.493.

Explicación del dataset

El dataset contiene 50 registros, con las variables predictoras 'Horas de estudio' y 'Horas de sueño', y la variable objetivo 'Nota final'. El modelo de regresión lineal múltiple entrenado obtuvo un R^2 de 0.493, lo que indica que explica aproximadamente el 49.3% de la variabilidad de la nota final. El intercepto es 3.7823, con coeficientes de 0.1066 para 'Horas de estudio' y 0.0591 para 'Horas de sueño'.

Figura 2. Resultados del entrenamiento del modelo de regresión lineal múltiple.

- **Predicción:** expone la función `predict()` que recibe horas de estudio y sueño, aplica penalización por falta de sueño y retorna la nota final estimada.

Ingresar datos nuevos

Horas de estudio

6,0

Horas de sueño

7,0

Predecir

Predicción de nota final: 4.84

Ecuación estimada

$$\text{Nota final} = 3.7823 + (0.1066) \cdot \text{Horas de estudio} + (0.0591) \cdot \text{Horas de sueño}$$

Figura 3. Ejemplo de predicción de nota final en la interfaz Flask.

- **Visualización:** genera gráficos con matplotlib y seaborn mostrando la línea de regresión y los datos de entrenamiento.

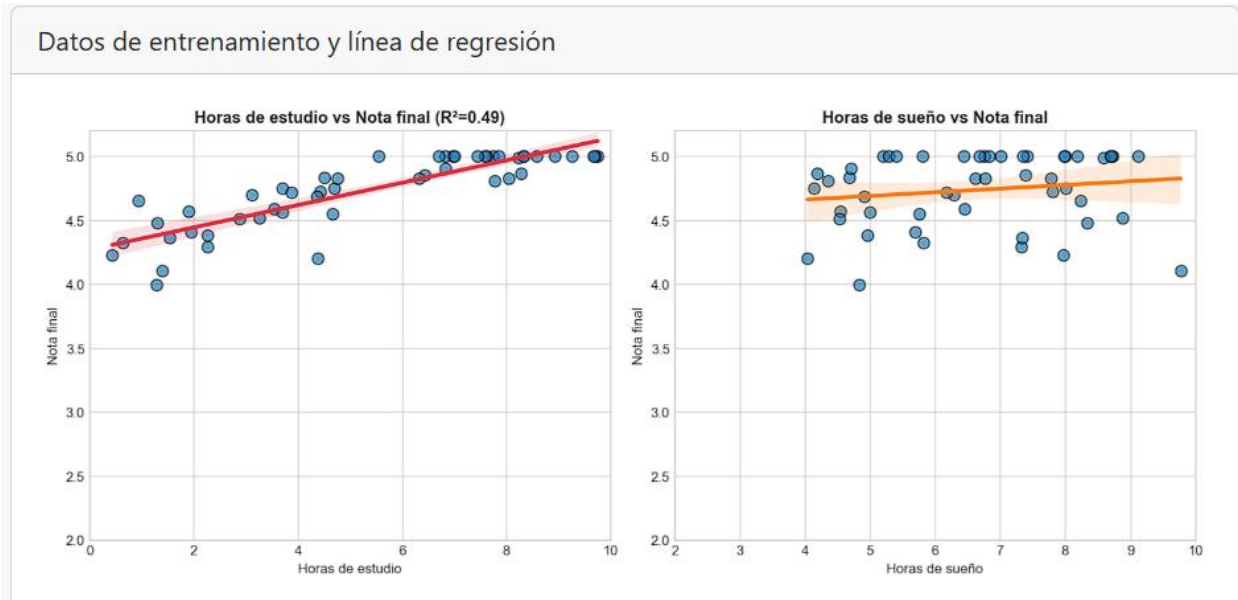


Figura 4. Visualización de datos de entrenamiento y línea de regresión.

- **Documentación:** incluye funciones que explican el modelo y el flujo general (`get_dataset_description_text()` y `get_workflow_description()`).

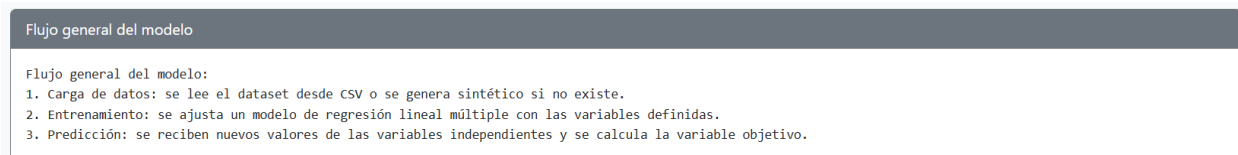


Figura 5. Descripción del flujo general del modelo en la interfaz web.

VI. Interfaz Web – Flask y Bootstrap

La aplicación incluye:

- Menú principal con pestaña “Regresión lineal”.



Figura 6. Menú principal de la aplicación con la pestaña “Regresión lineal”

- Submenú “Conceptos básicos” con teoría y referencias APA.



Figura 7. Vista del submenú “Conceptos básicos” con teoría y referencias APA.

- Submenú “Ejercicio práctico” con:
 - Explicación del dataset.

- Gráficos de entrenamiento con línea de regresión.
- Formulario para ingresar nuevos datos.
- Resultado de la predicción.
- Ecuación estimada del modelo.

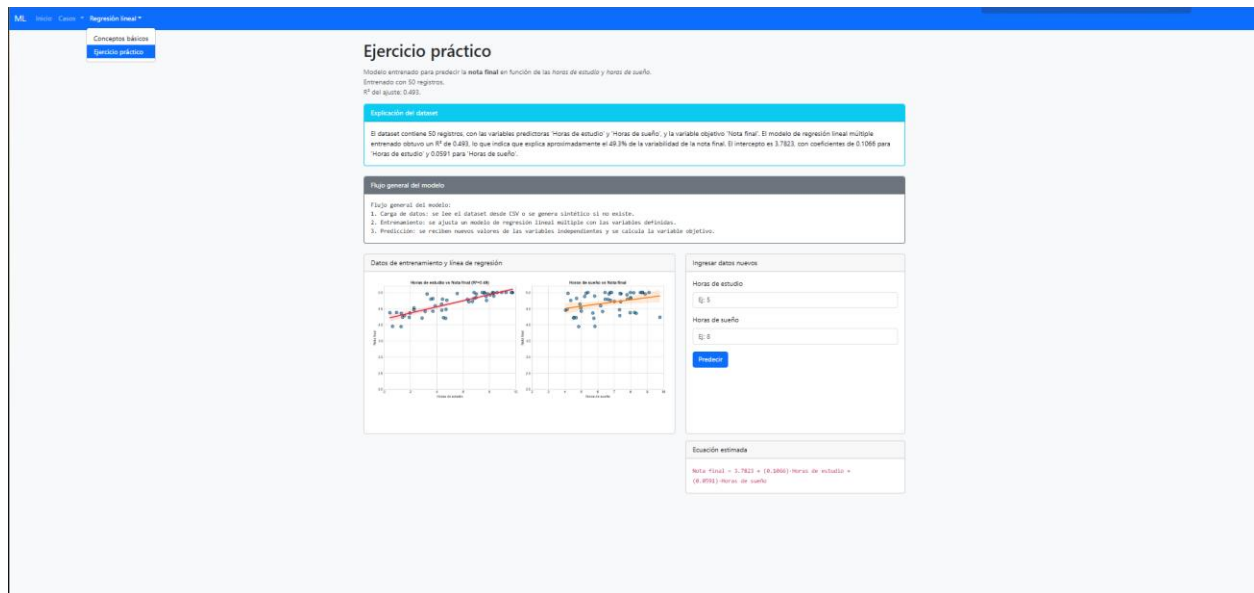


Figura 8. Vista del submenú “Ejercicio Práctico” con el ejercicio.

Todo el diseño se realizó con Bootstrap 5 para garantizar una presentación clara, responsiva y profesional.

VII. GitHub – Flujo de Trabajo

El proyecto se gestionó en GitHub con la siguiente estructura:

- Rama de desarrollo: A4_RegresionLineal.

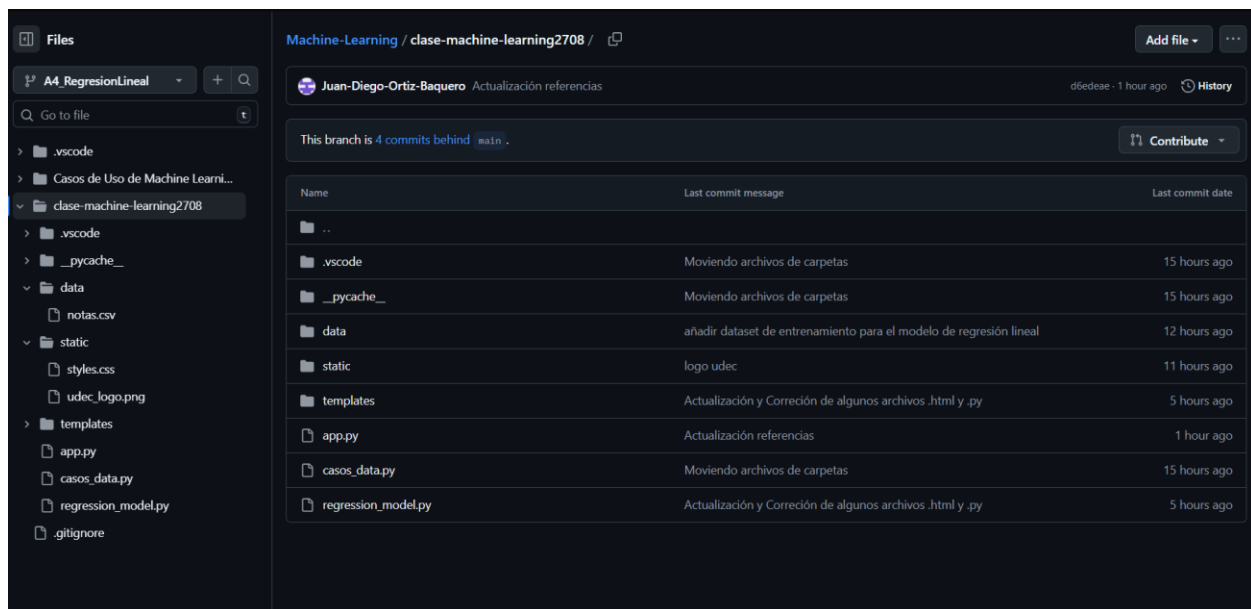


Figura 6 Captura de pantalla del repositorio en GitHub, rama A4_RegresionLineal, mostrando la estructura de archivos y carpetas del proyecto.

- Commits descriptivos y ordenados.

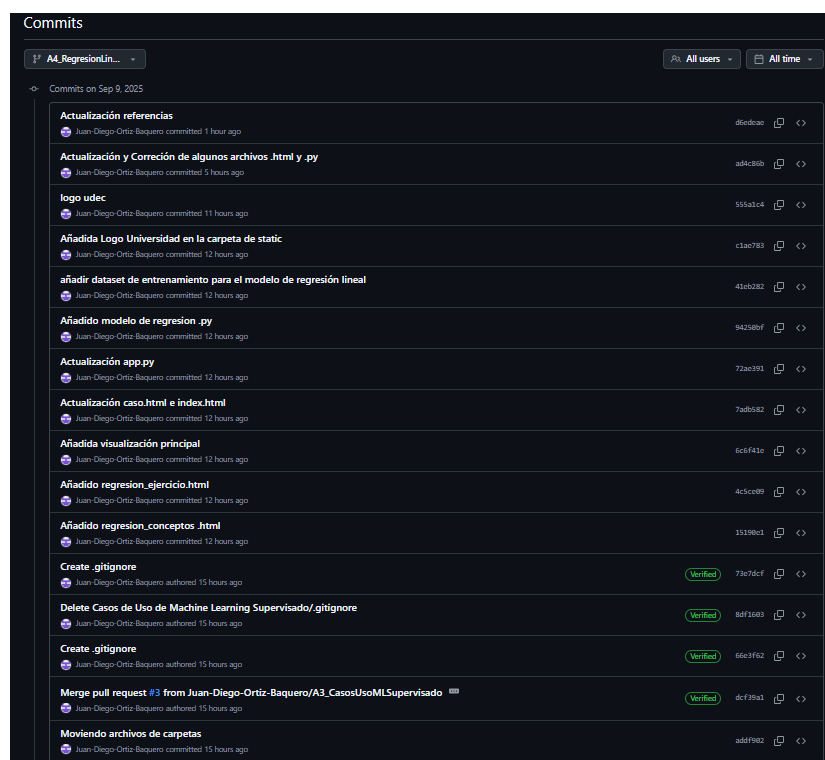


Figura 7 Captura de pantalla de la sección *Commits* en GitHub, mostrando un historial de cambios con mensajes descriptivos y ordenados realizados en la rama de desarrollo, incluyendo la verificación de autoría (*Verified*)

- Pull Request abierto desde la rama de actividad.

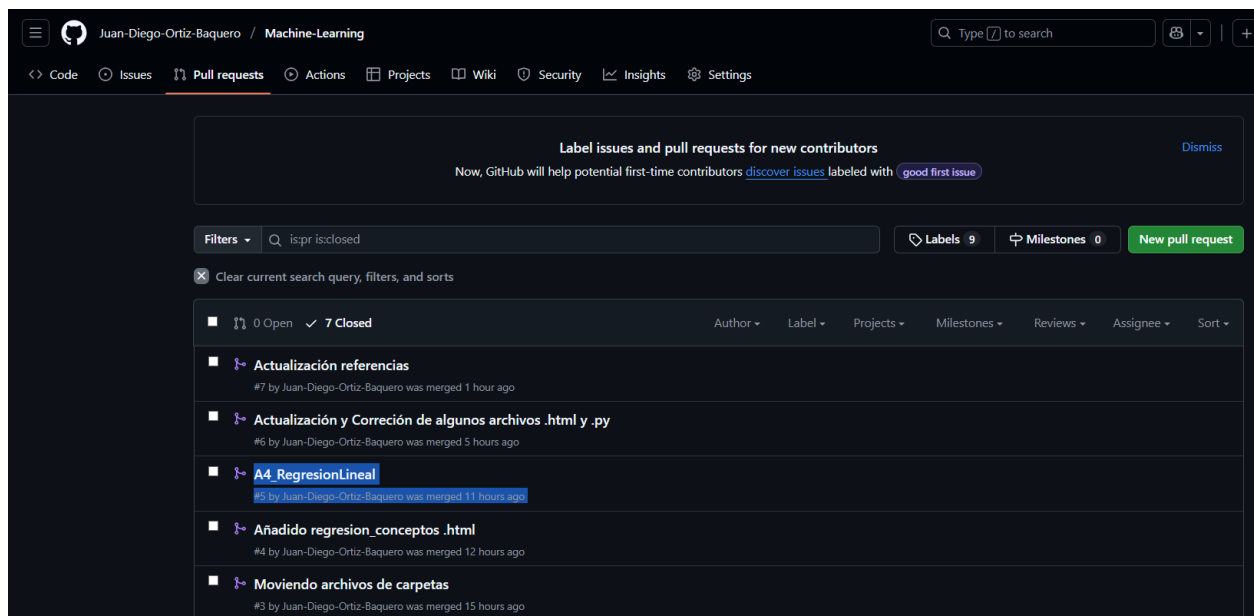


Figura 8 Captura de pantalla de la sección *Pull requests* en GitHub, mostrando la solicitud de integración (*Pull Request*) abierta desde la rama de actividad *A4_RegresionLineal* hacia la rama principal (*main*), con el listado de cambios y actualizaciones recientes.

- Merge realizado a la rama principal (*main*).

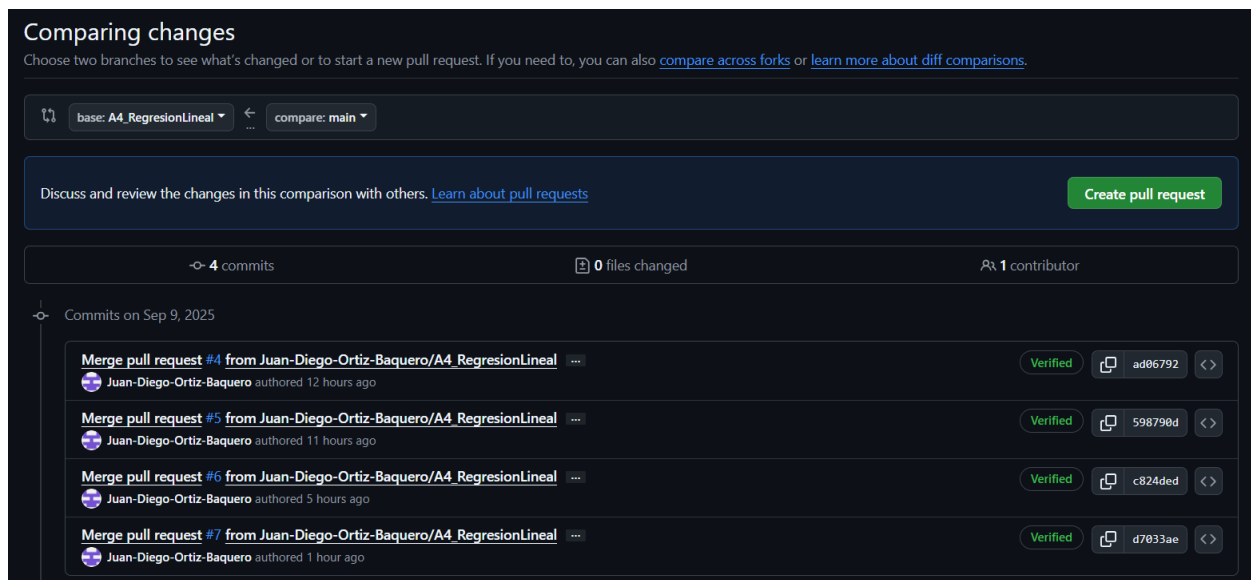


Figura 9 Captura de pantalla de la interfaz de GitHub mostrando la comparación y fusión (merge) de la rama A4_RegresionLineal hacia la rama principal (main)

- Evidencia del flujo: capturas del historial de commits, PR y estructura del repositorio.

VIII. Link repositorio github

<https://github.com/Juan-Diego-Ortiz-Baquero/Machine-Learning>

Referencias

- Statistics Easily. (s. f.). Supuestos en regresión lineal. Recuperado el 9 de septiembre de 2025, de <https://es.statisticseasily.com/supuestos-en-regresión-lineal>
- Universidad de los Andes. (s. f.). Regresión lineal. Recuperado el 9 de septiembre de 2025, de <https://programas.uniandes.edu.co/blog/regresion-lineal>

- Probabilidad y Estadística. (s. f.). Regresión lineal. Recuperado el 9 de septiembre de 2025, de <https://www.probabilidadyestadistica.net/regresion-lineal>
- Scikit-learn Developers. (2023). Regresión lineal. En Documentación de Scikit-learn (versión en español). Recuperado de https://scikit-learn.org/stable/modules/linear_model.html
- Universidad Nacional Autónoma de México. (2019). Introducción a la regresión lineal múltiple. En Curso de Estadística Aplicada. Recuperado de https://www.estadistica.unam.mx/curso_regresion_lineal
- DataCamp. (2024, 3 de mayo). Python Seaborn tutorial para principiantes: Empezar a visualizar datos. Recuperado el 9 de septiembre de 2025, de <https://www.datacamp.com/es/tutorial/seaborn-python-tutorial>
- GitHub Docs. (2024). *Acerca de las solicitudes de extracción (Pull Requests)*. Recuperado de <https://docs.github.com/es/pull-requests>
- GitHub Docs. (2024). *Acerca de las ramas*. Recuperado de <https://docs.github.com/es/get-started/using-git/about-branches>
- Pandas Development Team. (2024). *Pandas: análisis de datos en Python*. Recuperado de <https://pandas.pydata.org/docs/>
- Bootstrap Team. (2024). *Bootstrap 5: documentación oficial*. Recuperado de <https://getbootstrap.com/docs/5.3/getting-started/introduction/>

- Flask Documentation. (2024). *Flask: desarrollo web con Python*. Recuperado de <https://flask.palletsprojects.com/en/stable/>
- Hunter, J. D., et al. (2024). *Matplotlib: visualización con Python*. Recuperado de <https://matplotlib.org/stable/index.html>
- Interactive Chaos. (s.f.). *Tutorial de Seaborn*. Recuperado de <https://interactivechaos.com/es/manual/tutorial-de-seaborn/tutorial-de-seaborn>
- Amidi, A., & Amidi, S. (2018). *Hoja de referencia de Aprendizaje Supervisado* (Trad. J. P. Chavat, rev. F. Gonzalez-Herrera, A. Melgar-Lopez y F. Diaz). Stanford University. Recuperado de <https://web.stanford.edu/~shervine/l/es/teaching/cs-229/hoja-referencia-aprendizaje-supervisado>