Trabajo Fin de Grado
Grado en Ingeniería de Tecnologías Industriales

Desarrollo y aplicación de técnicas de Machine Learning para la predicción de contagios por Covid-19

Autor: Cristina Artacho Gómez
Tutor: Alicia Robles Velasco

Dpto. Organización Industrial y Gestión de Empresas II
Escuela Técnica Superior de Ingeniería

Sevilla, 2021

Trabajo Fin de Grado

Grado en Ingeniería de Tecnologías Industriales

Desarrollo y aplicación de técnicas de Machine Learning para la predicción de contagios por Covid-19

Autor:

Cristina Artacho Gómez

Tutor:

Alicia Robles Velasco

Colaborador docente invitado

Dpto. Organización Industrial y Gestión de Empresas II

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2021

Trabajo Fin de Grado: Desarrollo y aplicación de técnicas de Machine Learning para la predicción de contagios por Covid-19

Autor: Cristina Artacho Gómez
Tutor: Alicia Robles Velasco

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2021

El Secretario del Tribunal

AGRADECIMIENTOS

Este trabajo ha sido escrito y motivado en el contexto de una pandemia mundial. Sin olvidar a todos los afectados por las graves consecuencias de esta enfermedad, me gustaría dedicar este trabajo a las siguientes personas:

En primer lugar, a mi tutora Alicia por su inestimable paciencia, ayuda y ánimo durante estos meses, fundamentales para la realización de este trabajo.

También a mis amigos, especialmente a los que me han acompañado en el estudio durante los años de carrera que finalizo con este trabajo. Estoy segura, de que a pesar de las dificultades, siempre recordaremos con cariño esta etapa.

Por último, a mis padres y a mi hermana Amaia. Sin su constante apoyo, consejo y amor durante todos los años de mi vida, nada de esto habría sido posible.

Cristina Artacho,

Sevilla 2021

RESUMEN

En este trabajo se han aplicado técnicas de Machine Learning para la predicción de contagios de Covid-19 en España. Esta predicción se ha hecho a nivel diario y por provincias en un periodo determinado de tiempo, mediante el uso de las técnicas Regresión Lineal Múltiple y Regresión Polinómica, de segundo y tercer grado. El sistema desarrollado permite predecir el número de contagios para el día siguiente al calculado y el número de contagios con siete días de posterioridad, aunque con las modificaciones necesarias, este número de días podría modificarse dentro de un rango determinado. Se han utilizado distintas variables de entrada, como las temperaturas máximas y mínimas registradas, la población, densidad y extensión de la provincia, o el número de pasajeros que han llegado por medio aéreo a una provincia determinada, con el objeto de cuantificar cuáles de estas variables han sido más determinantes para que se haya producido un número concreto de contagios. Gracias al lenguaje de programación Python y a las técnicas Machine Learning se han podido hacer cálculos con un gran número de datos. El objetivo de este trabajo es ayudar a aquellas empresas que necesiten conocer la situación epidemiológica de forma local en los próximos días o semanas, para planificar su stock de manera efectiva.

ABSTRACT

The aim of this Project is to demonstrate mathematical relationship between some variables and the daily number of Covid-19 infections in every province of Spain, in a certain period of time during the first wave of the Covid-19 pandemic. These variables are the density, population and extension of the province, their daily highest and lowest recorded temperature, and the number of air passengers who have arrived to the province during the month of study. The developed system is able to predict the number of infections for the day after the calculated one, and also the number of infections seven days later. Some Machine Learning techniques have been used to this end, such as Multiple Linear Regression, and Polinomial Regression of second and third degree. This has been implemented in Python with the Anaconda distribution. The final objective of this Project is to help those companies that need to know the epidemiological situation locally in the coming days or weeks, to plan their stock effectively.

ÍNDICE

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

NOTACIÓN

‘y’	Datos reales
‘$\hat{y}$’	Datos estimados
‘n’	Número total de muestras de datos
‘$\bar{y}$’	Valor medio de los valores de y
‘$\bar{y}'$’	Valor medio de los valores de y'
‘IA’	Incidencia acumulada
‘σ’	Desviación estándar
‘r’	Coeficiente de variación de Pearson
‘$\bar{X}$’	Valor medio de la variable X
‘β'	Coeficiente de ajuste en regresión
‘num_casos_ant’	Número de casos positivos del día anterior al calculado
‘num_casos_7’	Número de casos positivos con siete días de anterioridad al día calculado
‘Tmax_10’	Temperatura máxima registrada en la provincia con 10 días de anterioridad al día que se notifica el número de positivos
‘Tmin_10’	Temperatura mínima registrada en la provincia con 10 días de anterioridad al día que se notifica el número de positivos
‘AENA’	Número de pasajeros llegados a la provincia el mes de los cálculos

'x_1'	Variable que representa a 'num_casos_ant' o 'num_casos_7', dependiendo del modelo
'x_2'	Variable que representa la densidad de la provincia
'x_3'	Variable que representa la población de la provincia
'x_4'	Variable que representa la extensión de la provincia
'x_5'	Variable que representa la Temperatura máxima diaria registrada en la provincia con 10 días de antelación
'x_6'	Variable que representa la Temperatura mínima diaria registrada en la provincia con 10 días de antelación
'x_7'	Variable que representa el número de pasajeros que han llegado mensualmente a la provincia

INTRODUCCIÓN

Los coronavirus (CoV) son una extensa familia de virus que pueden causar diversas enfermedades, tanto en animales como en humanos. Estas enfermedades son de distinta gravedad, desde el resfriado común hasta otras más graves como el síndrome respiratorio agudo severo (SRAS). Los síntomas que pueden ocasionar estas infecciones son fiebre, tos, disnea y dificultad para respirar. En casos más graves se puede dar neumonía e insuficiencia renal, lo cual puede llegar a causar la muerte.

El 31 de diciembre de 2019 el gobierno de España fue informado por parte de China de la existencia de un grupo de casos de neumonía de causa desconocida. El 7 de enero identificaron ese brote como un nuevo tipo de coronavirus al que llamaron SARS-Cov-2. Actualmente esta enfermedad es una pandemia (declarada por la OMS el 11 de marzo de 2020) que afecta a gran parte de los países de todo el mundo.
Los síntomas más habituales de esta enfermedad son la fiebre, tos seca y cansancio. Otros síntomas menos comunes son congestión nasal, dolor de cabeza, diarrea, perdida de gusto u olfato y erupciones cutáneas [1].

Aproximadamente el 80% de los pacientes de esta enfermedad se recuperan sin tratamiento hospitalario. El 15% necesitan ingreso hospitalario, y el 5% cuidados intensivos. Las personas con más probabilidad de presentar un cuadro grave son personas mayores de 60 años o que sufren de patologías previas, como por ejemplo hipertensión arterial, problemas cardiacos o pulmonares, diabetes o cáncer. No obstante, se han detectado numerosos casos de personas jóvenes y sin patologías previas que han caído gravemente enfermas por la Covid-19.

Algunas de las personas que han superado esta enfermedad, siguen teniendo síntomas a largo plazo. Parece que la Covid-19 puede dejar secuelas a largo plazo, pero todavía no hay suficiente evidencia científica sobre esto por el poco tiempo que el virus lleva entre nosotros.

Esta enfermedad se contrae principalmente tras haber estado en contacto con algún infectado por el virus. La forma más común de transmisión es a través de las pequeñas gotas que pueden salir de la nariz o boca de una persona infectada tras toser, hablar, o estornudar. También es posible contagiarse si tras tocar alguna superficie en la cual hayan caído gotas, el sujeto se lleva las manos a la boca, nariz u ojos. Esto puede evitarse si las manos son lavadas con agua y jabón o con algún tipo de desinfectante a base de alcohol.

El Covid-19 también se puede contagiar a través de aerosoles. Los aerosoles son partículas diminutas de solidos o líquidos en el aire. Cuando una persona infectada tose, estornuda o habla a un volumen elevado, puede expulsar aerosoles, que pueden contagiar a otras personas sanas, a pesar del minúsculo tamaño de los mismos. Esto es especialmente probable en espacios cerrados, sin circulación de aire.

Para evitar la transmisión de estas gotas y de los aerosoles, una de las medidas que se recomienda a la población es la distancia social entre individuos. Por precaución se recomienda el uso de mascarillas en la población para evitar la transmisión del virus, ya que éstas filtran las pequeñas partículas que pueden ser contagiosas.

A pesar de que algunos enfermos de la Covid-19 presentan síntomas muy leves, estos también pueden contagiar a través de las gotas que expulsen al hablar o toser. Esto supone un problema para la expansión del virus ya que muchos enfermos no son detectados y pueden propagar el virus sin saberlo.

Desde la OMS se recomienda a las personas que hayan estado en contacto con un positivo hacer cuarentena durante 14 días. La cuarentena es una separación física de las personas enfermas respecto de las personas sanas.

Las personas que dan positivo en alguna prueba de detección de virus deben estar aislados al menos 10 días, y añadir 3 días desde que no presentan síntomas. En el caso de ser asintomático, la persona debe estar en aislamiento 10 días desde el día en el que se hizo la prueba con resultado positivo.

Tras la exposición al virus los síntomas suelen aparecer a los 5 o 6 días de media, pero esta cifra puede variar entre 1 y 14 días [2] [3][4].

1.1. El Coronavirus SARS-CoV-19 en España

El primer caso de Covid-19 en España se detectó el 31 de enero de 2020 en la isla de la Gomera, en las islas Canarias. Después de tomar algunas medidas previas, como la suspensión de eventos de más de 1.000 personas o el cierre de centros educativos en algunas regiones, el 14 de marzo el gobierno de España aprobó el Real Decreto 463/2020, en el cual se declaro el estado de emergencia para contener la propagación de la enfermedad. En este Decreto se limitaba la libertad de circulación de las personas, con la excepción de algunas causas de fuerza mayor, afectando estas medidas a todo el territorio nacional.
Esta limitación, conocida como confinamiento total se prolongó hasta el 4 de mayo de 2020, cuando empezó la desescalada hacia la nueva normalidad. Esta desescalada consistió en 4 fases que finalizaron el 21 de junio de 2020 con lo que se conoce como nueva normalidad.
En ese punto de la pandemia la situación en España era de 246.272 casos confirmados y 28.323 fallecidos [5] [6].

1.1.1. Factores determinantes

Aunque este virus se ha propagado por todo el mundo, en algunas zonas ha incidido con más intensidad que en otras, y esto puede haber sido causado en mayor o menor medida por los siguientes factores:

- Condiciones climáticas: Ciertos virus respiratorios, como por ejemplo el de la gripe, se propagan con más intensidad en el invierno, por las bajas temperaturas. Aunque el Covid-19 se ha extendido por todo el mundo, incluso en zonas con ambientes cálidos, no se descarta la posibilidad de que las condiciones climáticas sean determinantes en la transmisibilidad del virus. Por ello, en este trabajo se van a utilizar los datos climatológicos diarios en cada una de las provincias españolas para intentar buscar una relación entre las distintas temperaturas y el número de infectados por Covid-19 [7].

- Densidad poblacional: Una de las medidas recomendadas a la población para minimizar la exposición al virus ha sido la conocida como distancia social. Es sabido que en las zonas de mayor densidad poblacional es más difícil conservar esta distancia por la forma de vida que se tiene en este tipo de ciudades, en las que el transporte público es requerido por un mayor número de usuarios ,y en las que hay más probabilidad de sufrir aglomeraciones. Por esto, otro de los datos que vamos a usar en la densidad de cada una de las provincias.

- Aeropuertos: Pese a que durante el estado de alarma el movimiento de personas a través de aeropuertos fue muy reducido, puede haber tenido algún impacto en el número de infectados, ya que no se realizó ningún control a los viajeros para conocer si eran portadores del virus. También es importante tener en cuenta los viajes realizados poco antes del estado de alarma.

- Sistema sanitario: El sistema sanitario español está descentralizado, esto se refleja en el número de camas de hospital o de cuidados intensivos (UCI) que existe por habitante en cada provincia y comunidad. Esto puede haberse visto reflejado en el número de fallecidos por esta enfermedad.

- Edad de su población: La edad y las patologías que suele tener asociadas son un factor de riesgo en esta enfermedad. Esto puede haber influido en el número de infectados, y especialmente de fallecidos por Covid-19.

- Ola: Las epidemias suelen sucederse en distintos ciclos o oleadas, con distintas fases de ascenso y caída en el número de infectados. Durante la epidemia de Covid-19, a fecha de febrero de 2021 se han registrado tres olas consecutivas distintas, en las que la fase de descenso de cada una de ellas se ha caracterizado por un número de medidas restrictivas a la población, incluyendo el confinamiento del conjunto de la población española en la primera ola. Los datos utilizados en este trabajo se refieren a esta primera ola.

1.1.2. Términos e indicadores

Algunos de los indicadores más utilizados en el seguimiento de la evolución de la pandemia son los siguientes:

- Incidencia: son los casos que se registran en una región determinada en un espacio de tiempo definido.

- Tasa de incidencia: Se utiliza para representar la velocidad de aparición de casos nuevos. En esta pandemia se ha utilizado más comúnmente el dato de incidencia acumulada en los últimos 14 días, para 100000 habitantes.

- Prevalencia: cuantifica qué proporción de una determinada población esta infectada en un momento concreto.

- Letalidad: muestra la gravedad de la enfermedad, es la proporción de casos que resultan en fallecimientos.

- Periodo de incubación: periodo de tiempo que transcurre desde la infección hasta la aparición de síntomas.

- Periodo de latencia: periodo de tiempo que transcurre desde la infección hasta que existe la posibilidad de transmitir la enfermedad a un organismo nuevo.

- Virulencia: es el grado de patogenicidad de un microbio o microorganismo. La patogenicidad es la capacidad que tiene un microorganismo para generar una enfermedad después de infectar a un ser vivo.

- Número R_o reproductivo: estima la velocidad con la que una enfermedad se propaga entre cierta población. Cuando $R_o < 1$, la enfermedad muere después de un extenso periodo de tiempo. Sin embargo, cuando $R_o > 1$, la enfermedad sigue propagándose.

1.2. Motivación

España es un país en el que existe diversidad climatológica, de densidad poblacional y en el cual cada comunidad autónoma tiene un sistema sanitario propio, que puede haber estado más o menos preparado ante las necesidades de la pandemia. Todos estos factores pueden haber influido en las distintas incidencia que ha tenido el virus en cada una de las provincias, y por ende, en cada una de las comunidades autónomas de este país.

En las últimas pandemias los modelos matemáticos y estadísticos se han utilizado para predecir las pérdidas humanas durante periodos específicos de la pandemia. Esto ha ocasionado que también se hayan desarrollado distintos modelos para estudiar la evolución de la pandemia actual. También existen modelos de predicción para determinar como de grave puede acabar siendo la enfermedad par una persona según tenga una patologías determinadas, aunque esto no entra dentro del objeto de estudio de este trabajo [8].

Existe una gran cantidad de datos que cuantifican el alcance de la pandemia en nuestro país y en todo el mundo. Esta disponibilidad de datos sobre aspectos puramente relacionados con la pandemia, como son el número de infectados o fallecidos, ha motivado el uso de Machine Learning para poder procesarlos y relacionarlos con otros datos que no son indicadores epidemiológicos en sí, como la densidad poblacional o la temperatura atmosférica.

El Machine Learning es una disciplina de la inteligencia artificial que permite a las máquinas aprender automáticamente y puede llegar a predecir comportamientos futuros a partir de unos datos iniciales, haciendo sus propios cálculos sobre datos iniciales. Estos sistemas pueden mejorar de forma autónoma, sin necesidad de la intervención de un programador. El Machine Learning busca encontrar patrones y comportamientos en los datos, para ver como van a reproducirse éstos en el futuro. Estos algoritmos son más efectivos cuánto mayor sea el número de datos.

1.3. Objetivos

Además de las pérdidas humanas, la pandemia ha sido muy perjudicial para la economía, tanto nacional como internacional. A nivel mundial, la producción se ha visto afectada por el cierre de industrias y la interrupción en cadenas de suministro y distribución. Esto a nivel bursátil se traduce en un aumento del riesgo, lo que puede ocasionar una crisis financiera [9].
Como se ilustra en la Figura 1-1, sólo en España, el producto interior bruto (PIB) en el segundo trimestre de 2020 sufrió una bajada del 18.5%, que aunque se ha solventado en trimestres siguientes casi en su totalidad, ha destruido miles de empleos.

Figura 1-1. Variación trimestral del PIB en España 2007-2020 en porcentajes [10].

España es uno de los países más resentidos por esta crisis, ya que su economía depende en gran medida del turismo, el comercio y la industria automovilística. La situación de cuarentena generalizada durante los primeros meses de pandemia, las restricciones de movilidad entre países, comunidades autónomas y municipios, y las medidas impuestas a los comercios y establecimientos hosteleros durante la segunda y tercera ola han afectado de forma directa a miles de empresas en nuestro país, llevándolas a una situación de cuantiosas pérdidas económicas.

También se han visto afectadas por la estructura empresarial del país, ya que la mayoría de empresas son medianas o pequeñas, y esto hace que resistan peor a las grandes crisis por carecer de grandes recursos financieros. Solo el 1.01% de las empresas españolas tienen más de 50 trabajadores.

El alto porcentaje de contratos temporales que existe en España tampoco es ventajoso, ya que estos empleos son más sensibles a las caídas del PIB [10]. Otras de las desventajas a las que nos enfrentamos son la baja capacidad de teletrabajo, ya que solo el 30% de los trabajos en España se pueden realizar de forma online [11].

En Marzo de 2020, el mes en el que se decretó el primer estado de alarma, el paro en España registró el mayor aumento de la historia con 302365 desempleados nuevos. Aunque durante el transcurso de la pandemia, en algunos meses posteriores la cifra de variación de parados ha sido negativa, creándose en esos meses concretos más empleo del que se destruía, como en el mes de Julio y Septiembre, los datos en conjunto no solventan las grandes caídas de empleo en Marzo y Abril [12]. Estas variaciones se ilustran en la Figura 1-2.

Figura 1-2.Variación mensual del paro registrado en España [12].

En el gráfico que se muestra a continuación (Figura 1-3), están representadas las variaciones en el número de parados por sectores en el año 2020. Podemos ver que en todos los sectores el paro ha aumentado. El total de nuevos parados ascendió a un total de 634.284 personas, y la gran mayoría de ellas, en concreto el 65% , pertenecen al sector servicios, que como ya hemos dicho es de los más afectados por la pandemia, y de los más importantes en nuestro país [13].

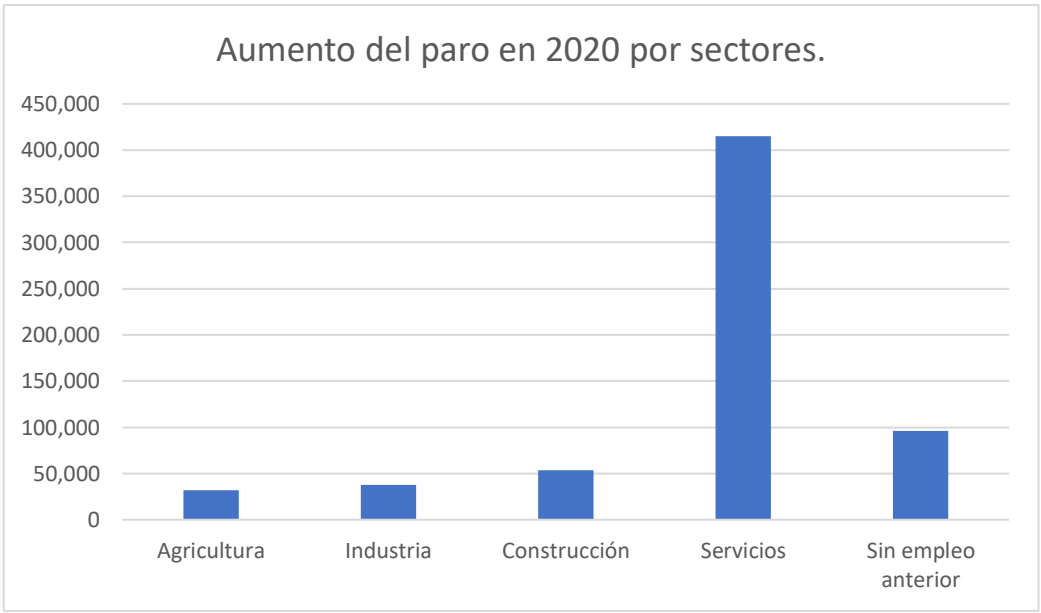


Figura 1-3. Aumento del paro por sectores. Año 2020 [13].

Este trabajo fin de grado busca servir de apoyo a empresas o instituciones que podrán optimizar la planificación de su producción en base a las predicciones obtenidas.

Este objetivo se va a implementar realizando un modelo matemático que relacione los datos de positivos y fallecidos por Covid-19 en cada una de las provincias españolas con la temperatura diaria de dichas comunidades, su población y su capacidad hospitalaria.

Con ello se pretende establecer un sistema predictivo, que ayude a conocer el estado actual y futuro de la pandemia, e informe de la importancia de las distintas variables en su crecimiento, para así ayudar a la toma de decisiones y medidas de contingencia contra la Covid-19.

1.4. Estructura del proyecto

La presente memoria se ha organizado en un total de cuatro capítulos. Cada uno de ellos se divide en varios subapartados. Los capítulos son los siguientes:

1. Introducción: donde se encuentran la motivación y objetivos del proyecto.
2. Machine Learning: sistema supervisado de regresión.
3. Descripción de los datos. Implementación en Python.
4. Resultados y posibles futuras líneas de investigación.
5. Conclusiones del proyecto.

MACHINE LEARNING: SISTEMA SUPERVISADO DE REGRESIÓN

En este capítulo se van a detallar la teoría y conceptos referentes al Machine Learning y a los modelos matemáticos elegidos. Estos modelos son la regresión lineal múltiple y regresión polinómica, con distintas variantes. También será explicado el cálculo de errores experimentales.

2.1. Machine Learning

El Machine Learning, que puede traducirse como aprendizaje automático, es un subcampo de la informática, que se ha definido como 'campo de estudio que da a los ordenadores la capacidad de aprender sin ser programados explícitamente'. Mediante el estudio de la construcción de algoritmos se pueden detectar patrones en grandes cantidades de datos que difícilmente serían captadas por los humanos. Con estos algoritmos, puede hacer predicciones e inferencias sobre el comportamiento de conjuntos de datos ya dados.

El proceso de aprendizaje que utiliza Machine Learning podría describirse como un lazo. Los pasos son los siguientes:

- Observación: se identifican patrones en los datos
- Planificación: encuentra todas las posibles soluciones
- Optimización: encuentra las soluciones óptimas de la lista de posibles soluciones.
- Actuación: se ejecuta la solución óptima anteriormente encontrada.
- Aprendizaje y adaptación: si el resultado es similar al esperado, finaliza el proceso. En caso contrario, se adapta de nuevo, volviendo al principio de los pasos.

Hemos definido el aprendizaje como un lazo por la estructura cíclica que presenta. Este lazo, representado en la Figura 2-1, se suele realizar mediante el uso de agentes inteligentes, es decir, un componente que pueda percibir su alrededor mediante algún tipo de sensor.

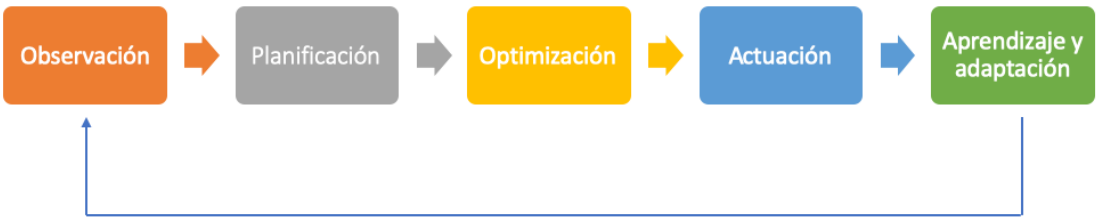


Figura 2-1. Representación del proceso de aprendizaje de Machine Learning.

Las diferentes categorías de Machine Learning se pueden separar en 3 grandes grupos, los cuales se diferencian por el tipo y cantidad de datos, así como el uso que se hace de ellos.

- Aprendizaje supervisado: Estos sistemas utilizan datos cuya salida es conocida, es decir, al algoritmo de aprendizaje automático se le proporciona un conjunto de datos de entrada cuya variable de salida asociada es conocida. El objetivo del algoritmo es aprender patrones en los datos y construir un conjunto general de reglas.
  Un ejemplo común de aprendizaje supervisado es el detector de spam en los servidores de correo electrónico, que etiquetan los diferentes correos como 'deseados' o 'no deseados'.

  Dentro del aprendizaje supervisado existen dos tipos principales:

  - Regresión: su objetivo es establecer un método para la relación entre un cierto número de valores y una variable objetivo o de salida continua. El ejemplo de regresión más común es la regresión lineal, en el cual se define una recta para proporcionar la tendencia de un conjunto de datos.

  - Algoritmos de clasificación: Se utilizan cuando la salida se encuentra en un conjunto finito de resultados. El algoritmo debe aprender los patrones en la entrada de cada una de las clases de datos de entrada, y predecir la salida considerando dicha entrada.

Existen 3 etapas dentro de los algoritmos de aprendizaje supervisado:

1. Entrenamiento: El algoritmo debe aprender los patrones de los datos de entrada y representarlos como una ecuación estadística, conocida habitualmente como modelo.
2. Prueba: En esta etapa se evalúa el rendimiento de el modelo representado en la etapa anterior, aplicándolo a un conjunto de datos distinto al usado en la primera etapa.

3. Predicción: Se aplica el modelo a un nuevo conjunto de datos, que no se ha utilizado anteriormente.

- Aprendizaje no supervisado: El aprendizaje no supervisado se utiliza cuando se desconoce la clase de salida deseada para los datos. Este tipo de algoritmo carece de un conocimiento previo. El objetivo es estudiar los patrones del conjunto de comportamiento de datos de entrada para identificar patrones y comportamientos similares que se puedan agrupar. Este tipo de algoritmos no requieren la intervención previa de expertos.

  Algunos ejemplos de aprendizaje no supervisado son los siguientes:

  - Clustering: El objetivo del Clustering es agrupar los datos de entrada en grupos lógicos de elementos relacionados, de manera que tengan características similares.

  - Reducción de la dimensionalidad: Su objetivo es simplificar un gran conjunto de datos de entradas asignándolos a un espacio de menor dimensión, para disminuir el número de variables.

  - Detección de anomalías: Se basa en la identificación de elementos que no se ajustan a un patrón o comportamiento esperado en comparación con otros elementos en un conjunto de datos. Las técnicas más utilizadas para detectar estas anomalías son el cálculo de la desviación estándar y la agrupación.

- Aprendizaje por refuerzo: su objetivo es que el algoritmo aprenda por su propia experiencia. Se determinan qué acciones se deben escoger para maximizar una 'recompensa' final. Para la toma de decisiones no sólo se debe considerar la recompensa inmediata, sino la posterior y todas las siguientes. Se diferencia del aprendizaje supervisado y no supervisado en que el modelo no se entrena con un conjunto inicial de datos, si no que aprende a base de prueba y error. Esto hace que después de una serie de decisiones acertadas el proceso resulte reforzado. En la actualidad se utiliza en las tecnologías de reconocimiento facial

  Algunos de los ejemplos de aprendizaje por refuerzo son:

  - La cadena de Markov: modelo estadístico que establece una fuerte dependencia ente un evento y otro suceso anterior. 'Si el estado actual X_n y los estados previos $X_{1, \dots,} X_{n-1}$ son conocidos, la probabilidad del estado futuro X_{n+1} no depende de los estados anteriores $X_1,\dots,X_{n-1}$, solamente depende del estado actual X_n' [14].

- El Q-learning: su objetivo es aprender una serie de normas para elegir qué decisión tomar bajo una serie de circunstancias concretas, de forma que esta decisión maximice la recompensa final.

- El método Monte-Carlo: método de simulación donde se obtienen soluciones a problemas matemáticos o físicos con pruebas aleatorias repetidas [15] [16] [17].

Existe otra forma de clasificar los algoritmos Machine Learning, en paramétricos y no paramétricos. Los modelos paramétricos son aquellos modelos que asumen que los datos de entrada se distribuyen con una probabilidad que puede ser descrita siguiendo un conjunto de parámetros. Al contrario, los modelos no paramétricos son aquellos cuya complejidad crece según crece el número de datos de entrada, y no se pueden describir por un conjunto de parámetros.

1. Métodos paramétricos: a continuación vamos a mencionar algunos algoritmos de Machine Learning que siguen métodos paramétricos
   - Regresión logística
   - Perceptrón
   - Análisis discriminante lineal

2. Métodos no paramétricos
   - Árboles de decisión
   - Naive Bayes
   - Support Vector Machine
   - Redes neuronales

Esta división se muestra en la figura a continuación (Figura 2-2).

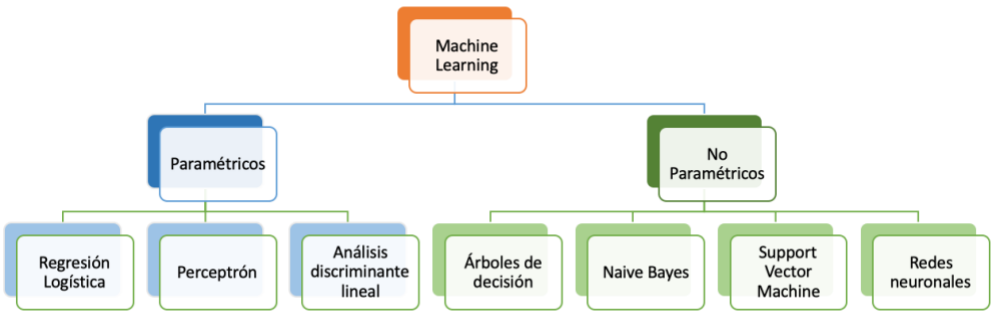


Figura 2-2. Clasificación de algoritmos de Machine Learning.

Los beneficios de los modelos paramétricos son su simplicidad, velocidad, y el escaso número de datos necesarios para obtener buenos resultados en comparación con otros métodos. Su principal inconveniente es que funcionan mejor con problemas simples

Los algoritmos no paramétricos son recomendables cuando disponemos de una gran cantidad de datos pero no mucha información sobre ellos. Sus ventajas son su flexibilidad y efectividad a la hora de procesar los datos. Sin embargo, sus limitaciones se basan en la lentitud y la alta probabilidad de overfitting (sobreajuste) ya que existe la posibilidad de perder calidad al predecir las salidas. A continuación se procede a explicar el significado de overfitting y el underfitting [18].

2.1.1. Overfitting y underfitting

La principal causa de errores en la predicción de datos en Machine Learning se da por el overfitting o underfitting de los datos. Al entrenar el modelo se intentar ajustar, en inglés, 'fit', los datos de entrada y salida. Si se da un fallo al ajustar un nuevo dato de entrada, será probablemente por alguna de estas dos razones.

El overfitting, que se podría traducir como 'sobreajuste', se da cuando el modelo es incapaz de reconocer un nuevo dato de entrada. Esto puede ocurrir porque nuestros datos de entrada tengan una característica común, pero no intrínseca de este tipo de dato, y nosotros intentemos introducir un nuevo dato que carezca de esta característica concreta.
Si caemos en el overfitting, nuestra máquina sólo se ajustará a casos particulares y no reconocerá nuevos datos de entrada. Si al introducir datos en el modelo introducimos muestras algo atípicas, las que serían normalmente consideradas como 'ruido', el modelo aprenderá este dato como si fuera un dato propio del modelo y encontrará dificultades para diferenciar entre los datos buenos y los datos que deberían ser considerados ruido. El overfitting es más probable en modelos paramétricos y no lineales. Es importante fijar un número de datos de muestra para evitar este problema.
Una técnica muy común para evitar el overfitting es *k-fold cross validation*, o *validación cruzada*, que permite entrenar el modelo k veces con k particiones de los datos de entrenamiento, para estimar el futuro desarrollo del modelo en los datos a introducir.

El problema de underfitting, o 'subajuste', ocurre cuando no le proporcionamos al modelo suficientes datos de entrenamiento, por lo que no es posible encontrar resultados correctos, ya que el modelo no reconoce patrones en los datos. La solución se encuentra facilitándole al modelo un mayor número de datos de entrada [19] .

2.1.2. Limitaciones

Para el correcto desarrollo del trabajo actual nos encontramos una serie de limitaciones que deben ser tenidas en cuenta a la hora de interpretar los resultados.

1. Limitación en los datos: durante en la primera ola de Covid-19 la disponibilidad de pruebas diagnóstico de Covid-19 en España y en la mayoría de países del resto del mundo era limitada. Esto puede ocasionar que el número de notificados como positivos sea menor al número real de infectados. Para nuestro periodo de estudio los datos de hospitalizaciones a causa del Covid-19 son limitados o inexistentes dependiendo de la provincia, lo cual imposibilita su uso.

2. Precisión en la predicción: Los resultados de la investigación pueden tener sesgos que podrían afectar a la interpretación de resultados y a la toma de medidas de contingencia. Por lo tanto es importante cuantificar el error que pueda producirse [8].

2.2. Series temporales

Una serie temporal o cronológica es una sucesión de observaciones de una variable medidos en determinados momentos y ordenados cronológicamente. Normalmente, estas observaciones se hacen es instantes de tiempo equiespaciados. Es importante saber que en las series temporales las observaciones no son independientes y que los datos deben analizarse teniendo en cuenta el orden temporal de las observaciones. Por todas estas características, los distintos datos recogidos en este trabajo son considerados series temporales. A continuación se explican algunas características más de las mismas.

Las series temporales pueden ser:

- Discretas o continuas: Dependiendo de cuándo se hayan tomado las observaciones.

- Determinísticas o no determinísticas: En el caso de que se pueda predecir sus valores con exactitud o no.

- Estocásticas: Si en una serie temporal solo se pueden determinar los valores futuros de forma parcial y no de forma exacta, se considera que los valores futuros están condicionados por los valores pasados, y que esta serie es estocástica.

El estudio de series temporales se hace con el primer objeto de observar los datos para comprobar si existen factores como una cierta tendencia, estacionalidad o valores atípicos en momentos determinados, motivados por una componente aleatoria. Esto se puede describir como se muestra en la Ecuación (2-1):

$$X_t = T_t + E_t + I_t \tag{2-1}$$

Donde T_t es la tendencia de los datos, E_t es la componente estacional y I_t la parte aleatoria. Para detectar estas componentes es muy útil representar gráficamente la serie. El segundo objetivo del estudio de las series temporales es la predicción de valores futuros.

Para decidir qué modelo probabilístico es el que más se acerca a nuestra serie temporal es interesante aislar la componente aleatoria, I_t. Para ello existen dos métodos:

- Enfoque descriptivo: Primero se calculan T_t y E_t de forma aproximada, y después se obtiene I_t despejando de la Ecuación (2-2).

$$I_t = X_t - T_t - E_t \tag{2-2}$$

- Enfoque de Box-Jenkins: Se identifica y selecciona el modelo, asegurándose de que las variables sean estacionarias, y se elimina la tendencia y parte estacional con el uso de transformaciones y filtros, para finalmente aplicar modelos paramétricos a la parte que queda [20].

2.3. Regresión lineal múltiple

La regresión lineal simple es un modelo matemático que pretende ajustar una variable dependiente 'Y' a unos valores independientes 'X_i', como se muestra en la Ecuación (2-3), donde β_i son los coeficientes y ε el posible error cometido. En la Figura 2-3 se ilustra un ejemplo de este tipo de regresión, en el que la línea roja, llamada línea de tendencia se ajusta a los valores de Y. El sistema asume que las relaciones entre variables son lineales.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \cdots + \beta_n X_n + \varepsilon \tag{2-3}$$

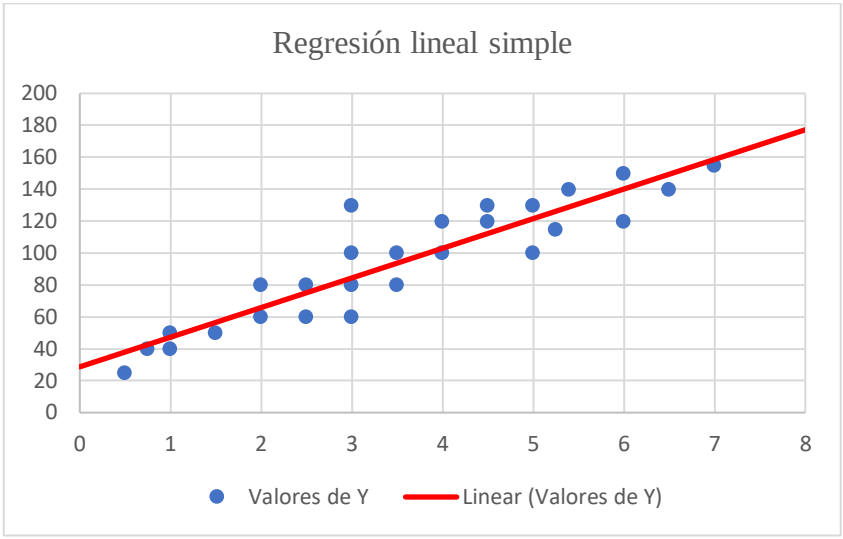


Figura 2-3. Ejemplo gráfico de una regresión lineal simple.

Como podemos ver, este modelo se utiliza cuando existe una sola variable. La regresión lineal múltiple es una extensión de esta regresión lineal simple. Sin embargo, ésta se utiliza para relacionar varias variables, como es nuestro caso. El modelo de esta regresión de forma matricial es el que se muestra a continuación en la Ecuación 2-4, para el caso de p observaciones y n variables.

$$\begin{matrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{matrix} = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1n} \\ 1 & x_{21} & \ldots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{p1} & \cdots & x_{pn} \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_p \end{pmatrix} \tag{2-4}$$

De esta expresión se deduce:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + \cdots + \beta_n x_{in} + \varepsilon_i \quad para\ i=1,2,3\ldots n \tag{2-5}$$

De la aplicación de este modelo se obtiene un error aleatorio ε, y n coeficientes β_i, cada uno referido a una de las variables, que nos indican cuánto influye dicha variable en el resultado de ajuste final.

2.4. Regresión polinómica

La regresión polinómica pertenece a la categoría de aprendizaje supervisado. Es una variante del análisis de regresión lineal múltiple en el que la relación entre la variable independiente 'x' y la variable dependiente 'y' se modela como un polinomio de grado 'n'.

2.4.1. Teoría y conceptos

Una regresión simple puede extenderse con variables polinómicas en los coeficientes. En el modelo linear estándar se tendría un caso así para datos bidimensionales:

$$\hat{y}(w,x) = w_o + w_1x_1 + w_2x_2 \tag{2-6}$$

Si queremos ajustar los datos a un paraboloide en vez de a un plano, tenemos que ajustar las funciones en polinomios de segundo orden

$$\hat{y}(w,x) = w_o + w_1x_1 + w_2x_2 + w_3x_1x_2 + w_4x_1^2 + w_5x_2^2 \tag{2-7}$$

Este sigue siendo un modelo lineal

$$z = [x_1, x_2, x_1 * x_2, x_1^2, x_2^2] \tag{2-8}$$

Con este nuevo etiquetamiento de los datos el problema se puede escribir como:

$$\hat{y}(w,x) = w_o + w_1z_1 + w_2z_2 + w_3z_3 + w_4z_4 + w_5z_5 \tag{2-9}$$

Este modelo tiene gran flexibilidad para adaptarse a los datos. El modelo de regresión polinomial puede considerarse como un caso concreto de la regresión lineal múltiple en que se busca determinar el mejor polinomio que represente un conjunto de datos. Este modelo corresponde a la Ecuación 2-10.

$$y_i = \beta_0 + \beta_1x_i + \beta_2x_i^2 + \beta_3x_i^3 + \cdots + \beta_nx_i^m + \varepsilon_i \text{ para i=1,2,3…n} \tag{2-10}$$

Éste también se puede representar como el modelo a continuación, representado de forma matricial en la Ecuación 2-11.

$$\begin{matrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{matrix} = \begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^m \\ 1 & x_2 & x_2^2 & \ldots & x_2^m \\ 1 & x_3 & x_3^2 & \ldots & x_3^m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^m \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \\ \beta_n \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{pmatrix} \tag{2-11}$$

El cual con notación matricial pura se representaría como:

$$Y = X\beta + \varepsilon \tag{2-12}$$

2.4.2. Particularización del modelo al caso de estudio

En la aplicación de este modelo a los datos del Covid-19 de los que disponemos, la variable dependiente ‘y’ corresponde al número de casos diarios por provincia, ‘num_casos’. Los coeficientes que multiplican a las variables independientes ‘x’ son los datos de temperatura máxima y mínima, ‘Tmax_10’ y ‘min_10’, tomados con diez días de antelación respecto al día que se ha calculado la variable dependiente ‘y’; el dato

mensual de pasajeros que han llegado a dicha provincia en el mes que se están realizando los cálculos, y la densidad de población de la provincia.

2.5. Cálculo de errores

La precisión de los algoritmos Machine-Learning puede ser descrita por su relación entre el error de sesgo (también conocido como 'bias') y el error de varianza. Estos conceptos son muy importantes ya que el Machine Learning busca la máxima precisión posible. Como es poco probable modelar un modelo de forma totalmente precisa, es importante aprender a interpretar los errores cometidos.

El error en la predicción se divide en tres partes:

- Error de bias: se refiere a la diferencia entre la predicción esperada y el resultado obtenido. Los algoritmos paramétricos suelen obtener un alto bias, que los hace más fáciles de entender pero también menos flexibles. Suelen ser menos efectivos en problemas complejos. En los modelos con bajo bias se suelen dar menos suposiciones sobre la forma de la función objetivo, mientras que cuando existe un alto bias suelen darse mayores suposiciones sobre la forma de la función objetivo.

- Error de varianza: estima cuánto puede cambiar el resultado de la función objetivo si se usan distintos datos de entrenamiento. Si nuestro algoritmo es bueno, no debería cambiar demasiado de un conjunto de datos de entrenamiento a otro. Cuando mayor sea el valor de error de varianza, mayor dependerá el resultado del algoritmo del conjunto de datos que estemos utilizando. Los algoritmos que tienen grandes varianzas suelen estar influenciados por datos muy específicos.

- Error irreducible: al que anteriormente hemos llamado 'ruido', el cual no se puede eliminar. Este ruido se da por distintos factores, muchas veces desconocidos.

En cualquier algoritmo de Machine Learning se busca tanto un bias bajo como una varianza baja, para conseguir así buenas predicciones. El bias frente a la varianza se refiere a la precisión frente a la consistencia de los modelos entrenados por su algoritmo.

Normalmente al aumentar el bias, la varianza disminuirá, y al aumentar la varianza, el bias disminuirá. Sin embargo, lo más importante es disminuir el error total, que es la suma de ambos. Esto se consigue encontrando el equilibrio óptimo entre bias y varianza.

2.5.1. Métricas de evaluación

Para evaluar el rendimiento del modelo, se aplican métricas de evaluación. Estas métricas se refieren al error calculado desde la recta generada hasta los puntos reales. Los más comunes son los que se detallan a continuación:

- Error absoluto medio: El error absoluto medio, del ingles Mean Absolute Error o 'MAE' es la media de la diferencia en valor absoluto de los datos reales y los datos estimados.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y - \hat{y}| \tag{2-13}$$

- Error cuadrático medio: El error cuadrático medio , del inglés Mean Squared Error 'MSE' es la media de la diferencia de los datos reales y los datos estimados, al cuadrado. Se diferencia del método anterior en que penaliza más las diferencias mayores.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\mathrm{y} - \hat{y})^2 \tag{2-14}$$

- Raíz del error cuadrático medio: Es la raíz del error cuadrático medio, del inglés Root Squared Error o 'RMSE'. El objetivo de introducir la raíz es que la escala del error sea igual a la escala de los datos introducidos.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\mathrm{y} - \hat{y})^2} \tag{2-15}$$

- Coeficiente de determinación: El coeficiente de determinación R^2 o R cuadrado representa la varianza del modelo. Se obtiene dividiendo la Regresión de suma de cuadrados ('SSR') entre la Suma de cuadrados total ('SST'), los cuales se obtienen con las siguientes operaciones matemáticas [21].

$$R^2 = \frac{SSR}{SST} \tag{2-16}$$

$$SSR = \sum_{i=1}^{n}(y' - \bar{y'})^2 \tag{2-17}$$

$$SST = \sum_{i=1}^{n} (y - \bar{y})^2 \tag{2-18}$$

DESCRIPCIÓN Y TRATAMIENTO DE DATOS. IMPLEMENTACIÓN EN PYTHON.

En este capitulo se procede a hacer una descripción detallada de los datos y del tratamiento que se les ha hecho. También se explica el procedimiento que se ha seguido para implementar el modelo creado en Python, y se justifica el uso de este lenguaje.

3.1. Descripción de los datos

Los distintos datos utilizados en este trabajo provienen de diferentes fuentes de acceso público. En primer lugar, las pruebas de diagnostico de Covid-19 utilizadas y de las cuales tenemos datos son las siguientes:

- La PCR, 'Reacción en cadena de la Polimerasa' permite detectar un fragmento del material genético de un patógeno, en este caso una molécula de ARN (ácido ribonucleico). Esta prueba permite conocer si la persona está infectada o no por coronavirus. Aunque puede dar lugar a un falso negativo si la prueba no se ha hecho correctamente, la fiabilidad de esta prueba es muy alta. Esta prueba es compleja y necesita personal entrenado y preparado para su realización. Aunque el resultado de esta prueba se pueda conocer en 3-4 horas, la realidad es que por la saturación del sistema público de salud normalmente los posibles infectados tardan de 24 a 48 horas en saber los resultados.

- Los test de antígenos se realizan con una muestra nasal o de saliva y detectan la proteína del virus. Son más rápidos que la realización de una PCR, pero a su vez menos fiables.

- Los test de anticuerpos, a diferencia de la PCR, no identifican el ARN del virus sino posibles anticuerpos que el cuerpo de la persona enferma haya producido. Se realizan con una muestra de sangre, o con proteínas del exudado nasofaríngeo. Son muy rápidos, y se pueden hacer en el domicilio del sospechoso, supervisados por un profesional sanitario [22].

Aunque los test de antígenos pueden presentar ciertas ventajas frente a la realización de pruebas PCR, ya que son más económicos, y rápidos en su diagnóstico, la realidad es que no siempre son más aconsejables que las pruebas PCR, ya que sólo son efectivos durante los 5 primeros días de la enfermedad, siempre y cuando el paciente muestre síntomas. Esto hace que no sean muy útiles para la detección de asintomáticos ni para la ejecución de cribados masivos en zonas muy afectadas por la Covid-19 [23].

El gobierno de España proporciona los siguientes datos relativos a la pandemia, de forma diaria, y para cada una de las provincias y Comunidades Autónomas. Estos se encuentran en la página web facilitada por el gobierno de España en [24].

- ccaa_iso o provincia_iso: El código ISO de la CCAA o de la provincia. En el caso de la Figura 3-10, que está al final de este apartado, el código 'A' corresponde a la provincia de Alicante.

- fecha: desde el inicio de la pandemia hasta el 10 de mayo, la fecha de inicio de síntomas o, en su defecto, la fecha de diagnóstico menos 6 días. A partir del 11 de mayo, la fecha de inicio de síntomas, o en su defecto, la fecha de diagnóstico menos 3 días, o la fecha de diagnóstico para los casos asintomáticos.

- num_casos: el número de casos totales, confirmados o probables.

- num_casos_prueba_pcr: el número de casos con prueba de laboratorio PCR o técnicas moleculares.

- num_casos_prueba_test_ac: el número de casos con prueba de laboratorio de test rápido de anticuerpos.

- num_casos_prueba_otras: el número de casos con otras pruebas de laboratorio, mayoritariamente por detección de antígeno o técnica Elisa.

- num_casos_prueba_desconocida: el número de casos sin información sobre la prueba de laboratorio.

Respecto a los distintos datos sobre los casos que se proporcionan, en este trabajo se va a usar el dato num_casos por ser el más global. Para hacer una predicción más precisa, se ha escogido el dato de positivos por provincia, y no de alguna unidad territorial más extensa [25].

- num_casos_ant: con el objetivo de mejorar la eficacia del modelo, se ha usado como dato en Machine Learning el número de casos positivos (num_casos), por provincia, del día anterior al calculado.

- num_casos_7: por ser el dato 'num_casos_ant' algo limitante, se ha añadido también en algunos escenarios del trabajo el dato de positivos registrados 7 días antes del día que se calcula.

En la figura que se muestra a continuación, (Figura 3-1),se representa el total de casos positivos de Covid-19 en España desde el 1 de Febrero de 2020 hasta el 30 de Marzo de 2020.

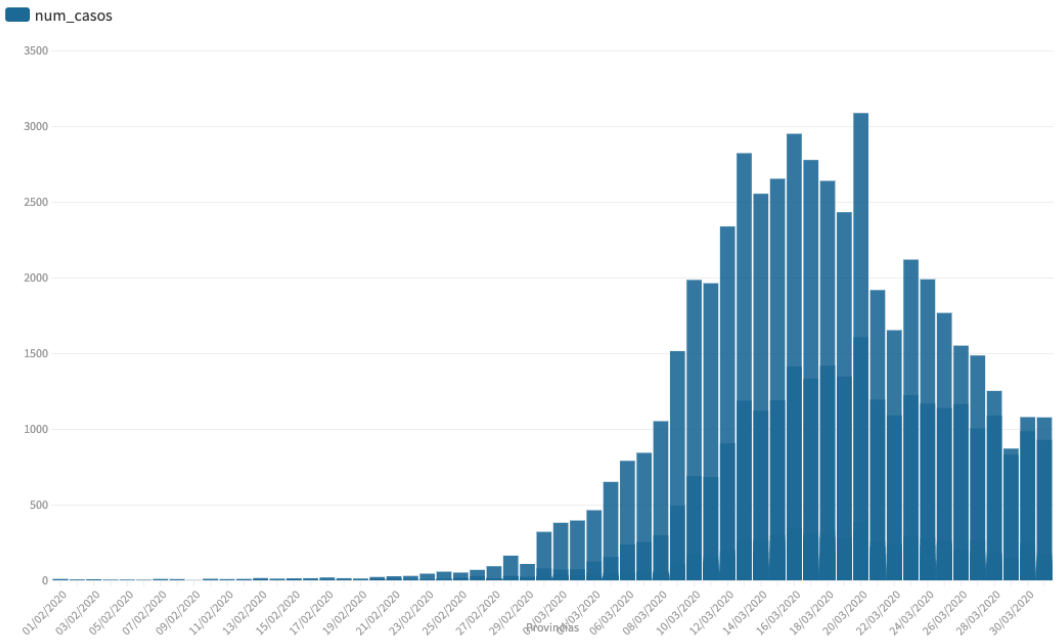


Figura 3-1.Datos totales de infectados por Covid-19 en España de 01/02/2020 hasta 30/03/2020 [17].

En segundo lugar, los datos diarios de temperatura en cada provincia se encuentran en la Agencia Estatal de Meteorología de España (AEMET). En su pagina web podemos encontrar un apartado de datos abiertos, llamado OpenData, ilustrado en la Figura 3-2, donde después de elegir una estación determinada, se pueden descargar los datos diarios para un intervalo de tiempo de temperatura máxima, mínima y media, presión, altitud y otras magnitudes. Los datos que utilizaremos en este trabajo son los de temperatura máxima y mínima diarias en alguna de las estaciones de cada una de las provincias españolas.

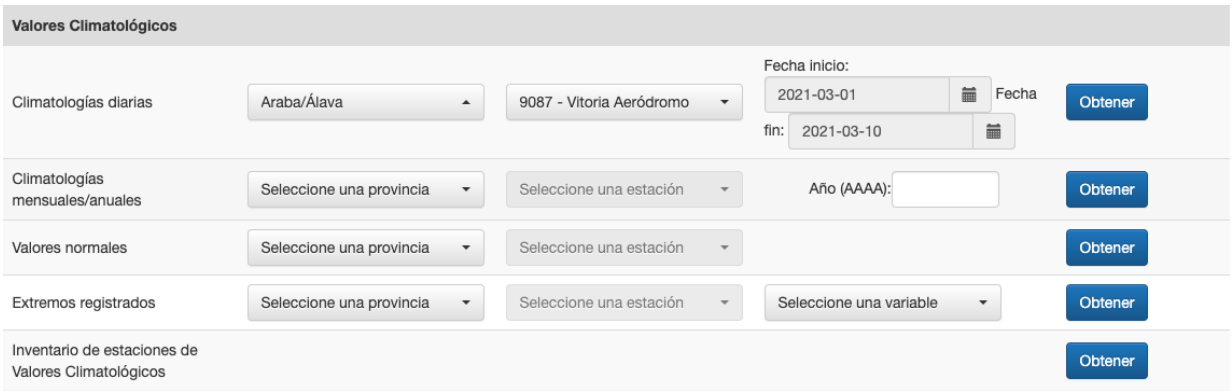


Figura 3-2. Parte de la interfaz del sistema OpenData de AEMET.

Por el periodo de incubación de la enfermedad, se han relacionado los datos de prueba de diagnóstico positiva en una provincia con los de la temperatura máxima y mínima en la provincia retrasados 10 días. Por esta razón llamaremos a cada uno de los datos de temperatura ‘Tmax_10’ y ‘Tmin_10’, refiriéndonos a la temperatura máxima y mínima registrada con 10 días de antelación, respectivamente. En la Figura 3-3 se muestra un gráfico con la temperatura máxima y mínima registrada desde el 22/01/2020 hasta el 20/03/2020 en la provincia de Alicante.

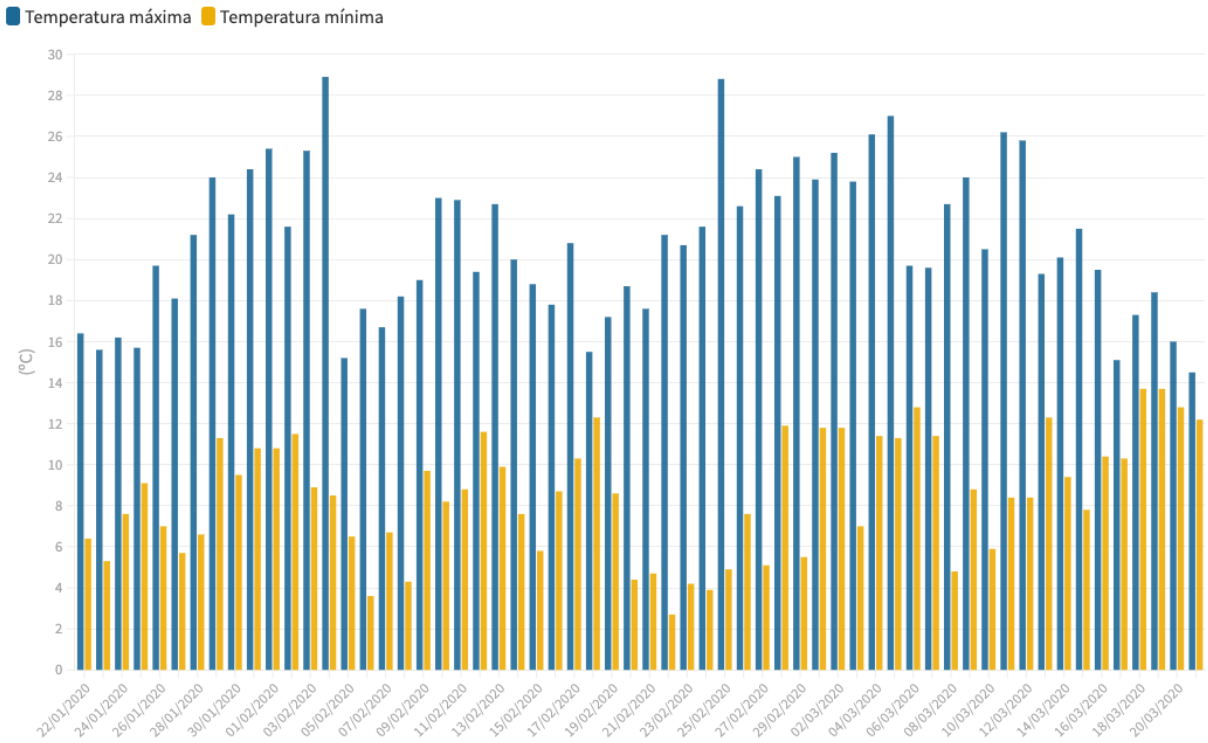


Figura 3-3. Temperatura máxima y mínima registradas desde el 22/01/2020 hasta el 20/03/2020 en la provincia de Alicante (Grados Celsius ºC) [18].

En tercer lugar tenemos los datos de los aeropuertos españoles. En la pagina web de AENA, la empresa pública dedicada a gestionar los aeropuertos de interés general españoles, se encuentran, para cada mes, los datos relativos al total de pasajeros, de mercancías y de operaciones, en formato Excel, como se ilustra en la Figura 3-4, en cada uno de los distintos aeropuertos que gestiona. Para este trabajo se van a utilizar los datos de pasajeros totales, teniendo en cuenta a que provincia pertenece cada uno de ellos, ya que como es natural hay provincias que no poseen ningún aeropuerto, y otras que tienen varios. AENA no proporciona los datos diarios si no los mensuales, por lo que éstos son los que se van a utilizar. A este dato lo denominamos como ‘Aena’.

Dirección de Operaciones, Seguridad y Servicios

Departamento de Estadísticas

AEROPUERTOS	PASAJEROS	
	Total	% Inc 2020 /s 2019
ADOLFO SUÁREZ MADRID-BARAJAS	4,663,407	6.3%
BARCELONA-EL PRAT J.T.	3,396,470	3.7%
GRAN CANARIA	1,166,742	-1.2%
MALAGA-COSTA DEL SOL	1,042,347	2.9%
TENERIFE-SUR	949,286	-6.8%
PALMA DE MALLORCA	820,661	-2.2%
ALICANTE-ELCHE	743,713	-5.3%
VALENCIA	561,235	3.2%
SEVILLA	546,798	4.7%
LANZAROTE-CESAR MANRIQUE	528,733	-7.1%
TENERIFE-NORTE	443,657	10.8%
FUERTEVENTURA	419,214	-5.4%
BILBAO	350,837	5.0%

Figura 3-4. Parte de los datos mensuales de tráfico de pasajeros en los aeropuertos españoles en Enero de 2020.

Como vemos en las siguientes imágenes (Figura 3-5 y Figura 3-6), la cantidad de pasajeros varía mucho por provincia, y esta cantidad disminuye cuantiosamente desde Febrero, el primer mes representado, hasta Marzo, que es el otro mes representado, por las consecuencias del estado de alarma.

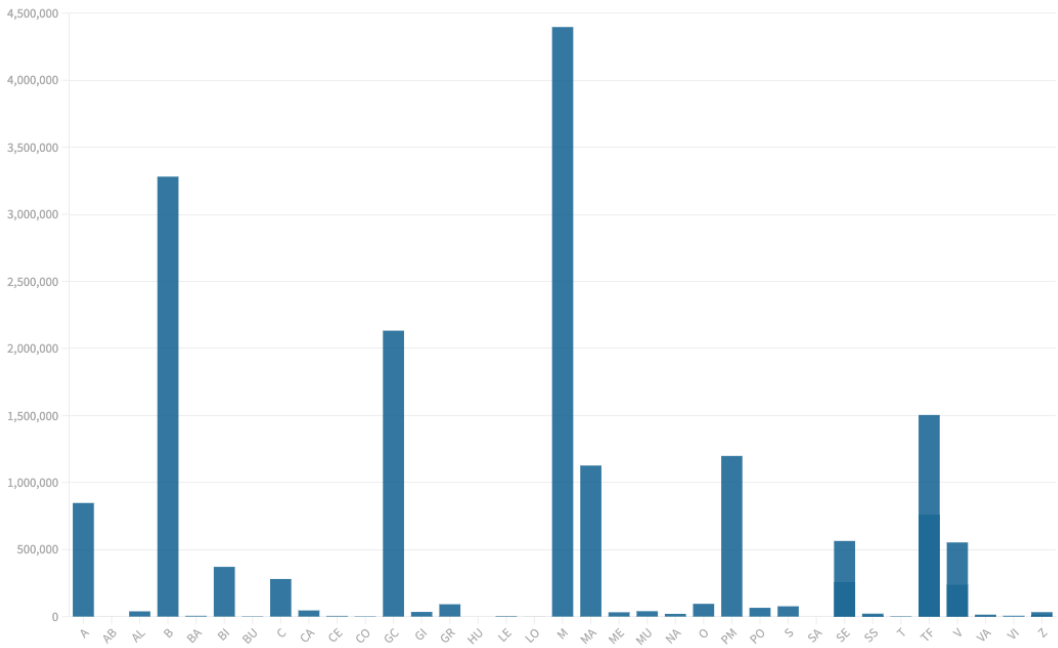


Figura 3-5. Pasajeros mensuales de AENA por provincia en España en el mes de Febrero de 2020.

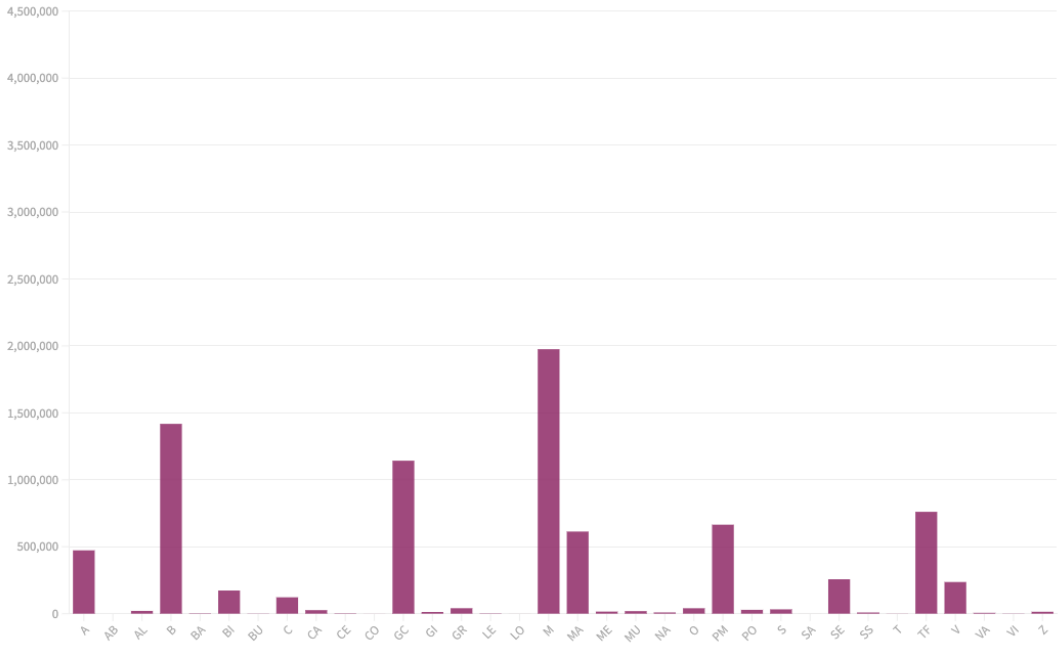


Figura 3-6. Pasajeros mensuales de AENA por provincia en España en el mes de Marzo de 2020 [26].

Otros datos reflejados en el archivo sobre el que se van a hacer los cálculos de este proyecto son:

- Mes: Mes en el que se notifican los resultados de las pruebas de Covid-19. Es simplemente el mes del apartado ‘Fecha’.

- Población: la población de cada una de las provincias españolas se puede encontrar en diferentes fuentes, la escogida para realizar este trabajo ha sido el Instituto Nacional de Estadística [27]. Este dato se ilustra en la Figura 3-7. Como se puede ver en el gráfico, el número de población es muy variante, y los mayores valores se dan en las provincias de Madrid y Barcelona.

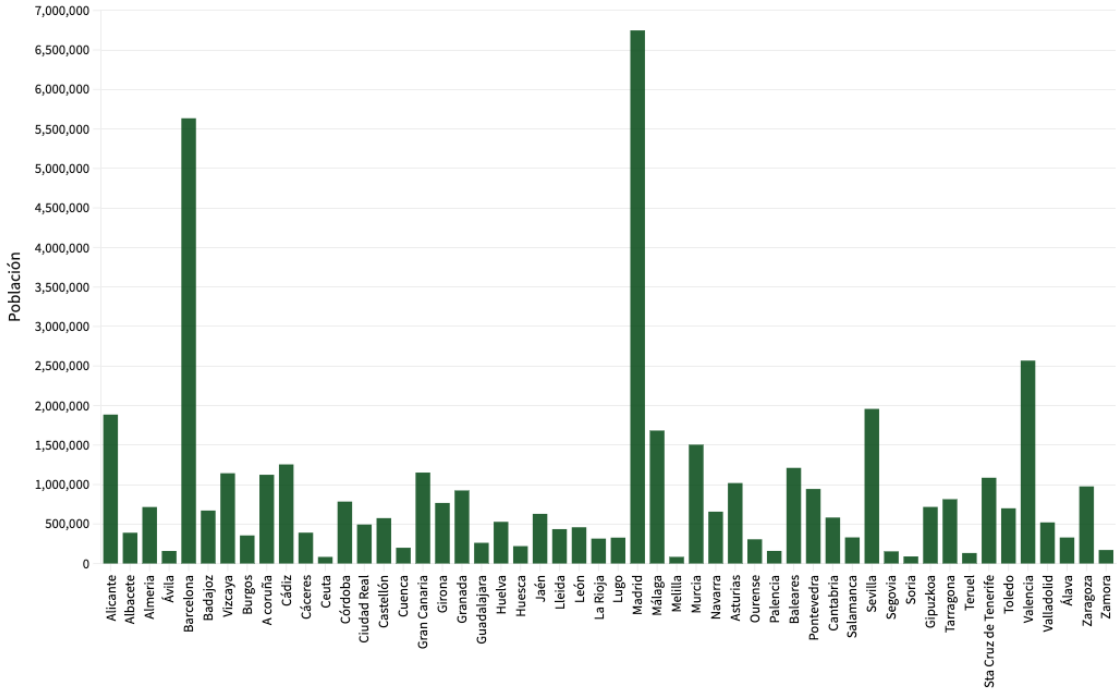


Figura 3-7. Población de las provincias y ciudades autónomas españolas.

- Extensión: Este dato también se da en distintas fuentes, la escogida ha sido [28]. La unidad usada son kilómetros cuadrados. En la Figura 3-8 se ilustran las distintas extensiones.

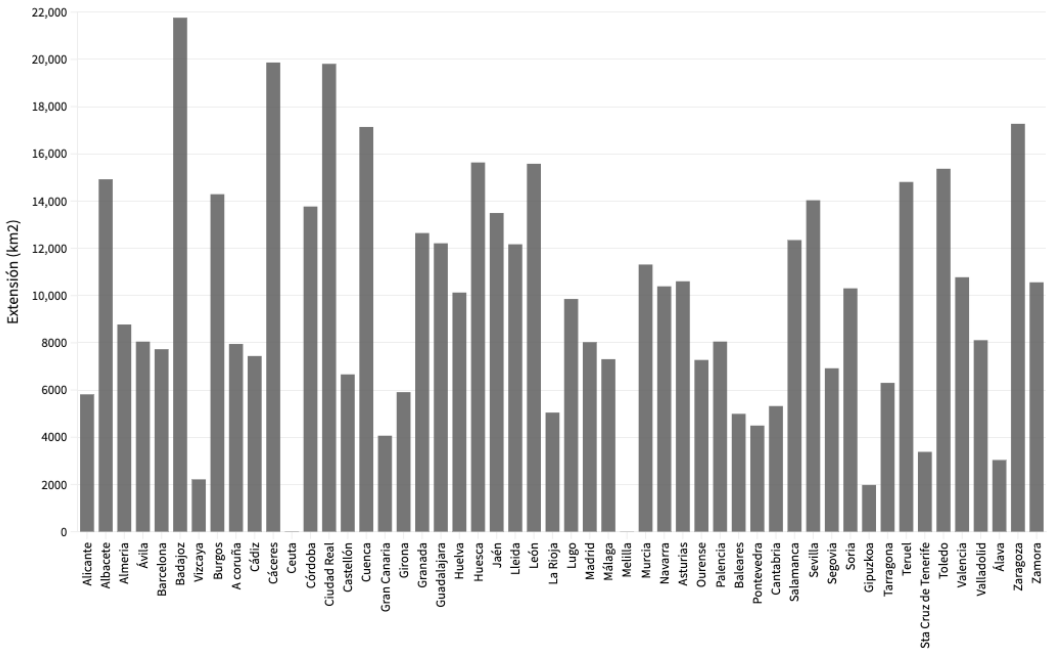


Figura 3-8. Extensión de las provincias y ciudades autónomas de España en kilómetros cuadrados.

- Densidad: España es uno de los países más poblados de la Unión Europea, aunque existe disparidad de densidades, debido a las diferencias entre zonas rurales y urbanas. Cabe destacar la especial

densidad de Ceuta y Melilla, con unas densidades de 4201 y 6869 personas por kilometro cuadrado respectivamente, que sobresalen respecto a las demás provincias. Esto se debe a su carácter de ciudades autónomas y será tenido en cuenta a la hora de interpretar los datos [29]. En el gráfico a continuación (Figura 3-9) se ilustra la densidad de cada una de estas provincias en personas/km^2.

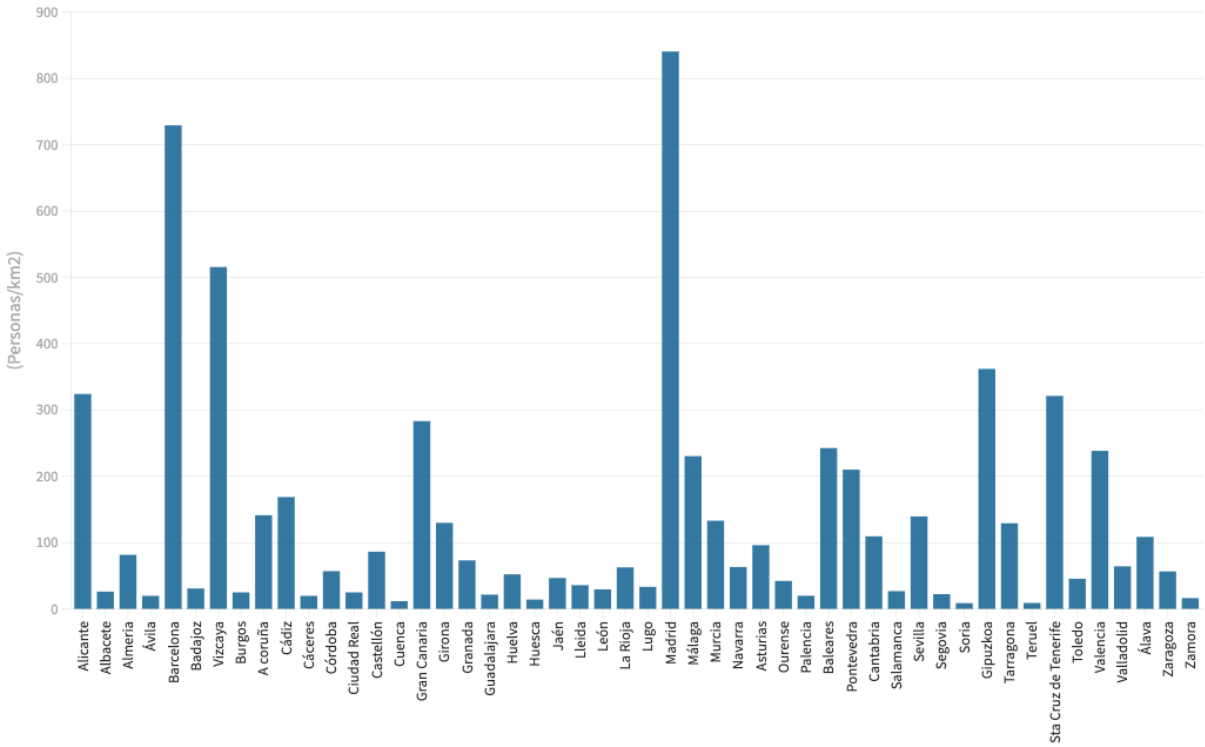
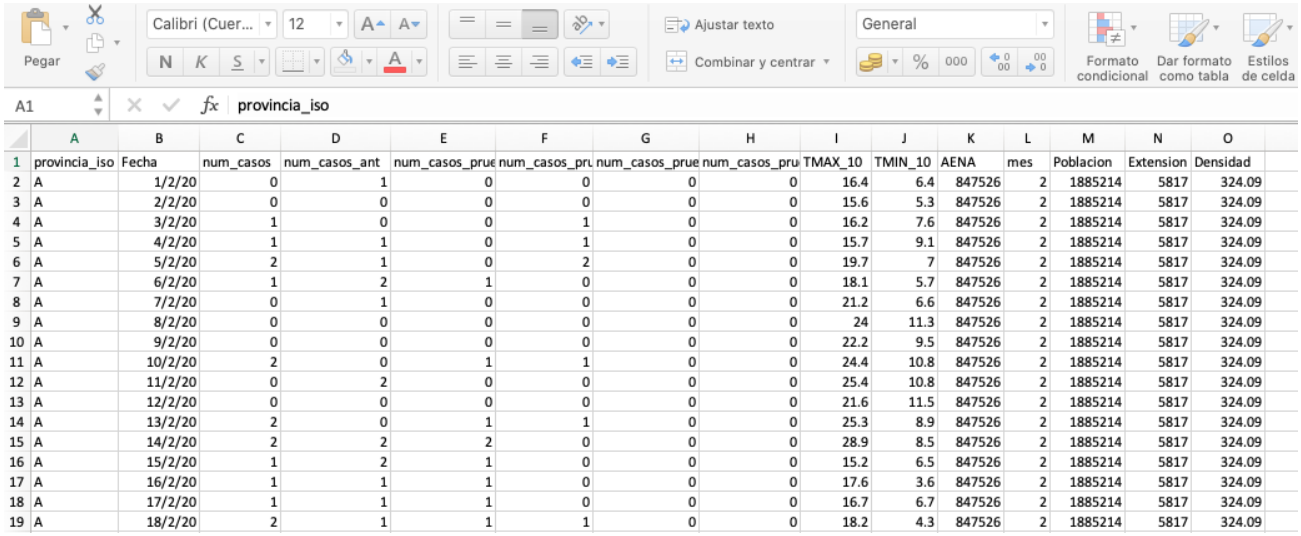


Figura 3-9. Densidad de las distintas comunidades y ciudades autónomas de España en 2021 (personas/km^2).

La densidad de cada una de las provincias se calcula simplemente dividiendo cada una de los términos 'Población' entre 'Extensión' (Ecuación 3-1). La unidad resultante es personas/km^2.

$$Densidad = \frac{Población}{Extensión} \tag{3-1}$$

El conjunto de los datos anteriores se refleja en un archivo csv de 3120 filas, referentes a los 60 días de estudio en cada una de las 52 provincias y ciudades autónomas de España. Parte de él se muestra en la Figura 3-10. Cómo se puede ver en la imagen, algunos de los datos son constantes para cada una de las provincias, como por ejemplo la población, extensión y densidad. Otros, como las temperaturas máximas y mínimas cambian cada día.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	provincia_iso	Fecha	num_casos	num_casos_ant	num_casos_prue	num_casos_pru	num_casos_prue	num_casos_pru	TMAX_10	TMIN_10	AENA	mes	Poblacion	Extension	Densidad
2	A	1/2/20	0	1	0	0	0	0	16.4	6.4	847526	2	1885214	5817	324.09
3	A	2/2/20	0	0	0	0	0	0	15.6	5.3	847526	2	1885214	5817	324.09
4	A	3/2/20	1	0	0	1	0	0	16.2	7.6	847526	2	1885214	5817	324.09
5	A	4/2/20	1	1	0	1	0	0	15.7	9.1	847526	2	1885214	5817	324.09
6	A	5/2/20	2	1	0	2	0	0	19.7	7	847526	2	1885214	5817	324.09
7	A	6/2/20	1	2	1	0	0	0	18.1	5.7	847526	2	1885214	5817	324.09
8	A	7/2/20	0	1	0	0	0	0	21.2	6.6	847526	2	1885214	5817	324.09
9	A	8/2/20	0	0	0	0	0	0	24	11.3	847526	2	1885214	5817	324.09
10	A	9/2/20	0	0	0	0	0	0	22.2	9.5	847526	2	1885214	5817	324.09
11	A	10/2/20	2	0	1	1	0	0	24.4	10.8	847526	2	1885214	5817	324.09
12	A	11/2/20	0	2	0	0	0	0	25.4	10.8	847526	2	1885214	5817	324.09
13	A	12/2/20	0	0	0	0	0	0	21.6	11.5	847526	2	1885214	5817	324.09
14	A	13/2/20	2	0	1	1	0	0	25.3	8.9	847526	2	1885214	5817	324.09
15	A	14/2/20	2	2	2	0	0	0	28.9	8.5	847526	2	1885214	5817	324.09
16	A	15/2/20	1	2	1	0	0	0	15.2	6.5	847526	2	1885214	5817	324.09
17	A	16/2/20	1	1	1	0	0	0	17.6	3.6	847526	2	1885214	5817	324.09
18	A	17/2/20	1	1	1	0	0	0	16.7	6.7	847526	2	1885214	5817	324.09
19	A	18/2/20	2	1	1	1	0	0	18.2	4.3	847526	2	1885214	5817	324.09

Figura 3-10. Archivo csv con los datos utilizados.

3.1.1. Análisis descriptivo de los datos

Una vez agrupados los datos requeridos para el proyecto, se ha hecho un primer estudio de ellos con algunas funciones de Python.

Disponemos de 3120 filas de datos, referentes a cada uno de los 60 días estudiados en cada una de las 52 provincias españolas y ciudades autónomas. En la Tabla 3-1 se recogen algunas características de los datos utilizados, como la media, la desviación estándar y el valor mínimo y máximo registrado. Estas se han obtenido con el comando *describe* de la librería *pandas* en Python.

Tabla 3-1. Descripción de los datos utilizados.

	num_casos	num_casos_ant	num_casos_7
Media	54.71	53.27	41.61
Desviación estándar	210.06	208.74	200.467
Mínimo	0	0	0
Máximo	3086	3086	3086
	Tmax_10	Tmin_10	AENA
Media	16.76	7.15	237896.26
Desviación estándar	4.72	4.57	645099.32
Mínimo	-4.1	-5.7	0
Máximo	31	23.8	4396896
	AENA	Población	Extensión
Media	237896.26	910191.94	9730.56
Desviación estándar	645099.32	1184276.42	5052.88
Mínimo	0	84032	12.3
Máximo	4396896	6747425	21766

Podemos observar que la media de casos notificados por provincia es de 54 casos diarios, y que el máximo en nuestro periodo de estudio fue de 3086 casos, una cifra muy superior a la media, en la provincia de Madrid, el 20 de marzo de 2020. La máxima temperatura registrada fue de 31ºC, en la provincia de Sevilla, y la mínima de -5,7ºC, en la provincia de Burgos. Los datos de AENA, es decir, el número de pasajeros en avión que vuelan a la provincia, presentan una gran desviación estándar, lo cual es lógico por la gran disparidad que existe en este aspecto entre provincias, ya que muchas ni siquiera disponen de aeropuerto y otras albergan aeropuertos muy concurridos. Respecto a la extensión también encontramos diversidad en los datos, de los cuales el mínimo se da en Melilla, que a su vez es la ciudad o provincia más densa, por su condición de ciudad autónoma, como ya se ha mencionado anteriormente en este trabajo. La mínima densidad se da en la provincia de Soria, probablemente por su carácter rural.

La desviación estándar (σ) es un indicador de la dispersión media de una variable (Ecuación 3-2). Es la raíz cuadrada de la varianza, y presenta las mismas unidades que las variables de medida. Por ser las unidades de medida de los datos muy distintas entre sí, se ha calculado el Coeficiente de variación de Pearson (r),

cuyo cálculo se muestra en la Ecuación 3-3. Este resulta de la división de la desviación típica entre la media en valor absoluto. Este coeficiente nos informa de la dispersión relativa del conjunto de datos estudiado.

$$\sigma = \sqrt{\frac{\sum_1^N (x_i - \bar{x})^2}{N}} \tag{3-2}$$

$$r = \frac{\sigma}{|\bar{x}|} \tag{3-3}$$

El resultado de calcular el Coeficiente de variación de Pearson a cada una de las variables es el que se muestra en la siguiente tabla.

Tabla 3-2. Coeficiente de variación de Pearson.

Variable	r
num_casos	3.84
num_casos_ant	3.92
num_casos_7	1.46
Densidad	3.20
Población	1.30
Extensión	0.52
Tmax_10	0.28
Tmin_10	0.64
AENA	2.71

Como podemos observar en la tabla anterior (Tabla 3-2), siguiendo este criterio los datos que presentan mayor dispersión es el número de casos positivos en Covid-19 en el día anterior, 'num_casos_ant', que como es lógico es también muy similar a la variable num_casos. También es significativa la gran dispersión hallada en la variable 'Densidad' y en el número de pasajeros de los aeropuertos, 'AENA'. El dato que menor dispersión presenta es la temperatura máxima registrada.

En segundo lugar, se ha calculado la matriz de covarianza, representada en la Tabla 3-3. La covarianza (Ecuación 3-4) indica cuánto varían dos variables aleatorias de forma conjunta respecto a sus medias. Es un dato útil para determinar si existe dependencia entre sus variables. Cuando el valor de la covarianza es positivo, existe una relación directa entre las variables, es decir, cuando la variable dependiente 'X' aumenta, la variable independiente 'Y' también lo hace. Al contrario, cuando el valor de la covarianza es negativo, esto significa que existe una relación indirecta entre las variables, y por lo tanto si la variable dependiente aumenta, la variable independiente disminuye. Finalmente si el valor de la covarianza resultase igual al cero, esto significaría que no existe relación entre 'X' e 'Y'. Esta matriz también resulta simétrica.

$$Cov(X,Y) = \frac{\sum_1^n (x_i - \bar{x})(y_i - \bar{y})}{n} \quad (3\text{-}4)$$

Tabla 3-3. Matriz de covarianza.

	num_casos	num_casos_7	num_casos_ant	Densidad	Población	Extensión	TMAX_10	TMIN_10	AENA
num_casos	46300								
num_casos_7	35900	40200							
num_casos_ant	43900	35800	43600						
Densidad	6130	5180	5780	1.18E+06					
Población	1.38E+08	1.11E+08	1.33E+08	2.45E+06	1.40E+12				
Extensión	-37900	-29600	-3.38E+04	-2.39E+06	-6.1E+08	2.55E+07			
TMAX_10	16.70	18.60	30.20	524	5.00E+05	-1030	18.20		
TMIN_10	28.90	37.60	41.70	1510	1.31E+06	-7300	7.87	20.10	
AENA	3.67E+07	2.93E+07	3.51E+07	4.12E+07	6.28E+11	-6.94E+08	3.05E+05	8.57E+05	4.16E+11

En este caso las covarianzas negativas también son todas relacionadas con la variable 'Extensión'. Vemos otras similitudes con la matriz de correlación, como que la relación positiva más fuerte se da entre 'AENA' y 'Población', y que 'num_casos' y 'Población' tienen también una relación relativamente fuerte. Sin embargo, la relación negativa más fuerte no coincide en este caso, ya que en la matriz de covarianza se encuentra en la relación entre 'Extensión' y 'AENA'.

Por último, aunque partimos de datos aparentemente independientes, se ha calculado la Matriz de correlación entre ellos, que se muestra en la Tabla 3-4. La matriz de correlación es una tabla simétrica, en la que se exponen los coeficientes de correlación de Pearson, 'r', entre cada variable. Este coeficiente toma siempre valores entre 0 y 1, indicando la relación lineal entre dos variables aparentemente independientes. siendo la diagonal siempre igual a 1. El coeficiente se calcula tal y como se indica en la Ecuación 3-5.

$$\rho_{X,Y} = \frac{\sigma_{X,Y}}{\sqrt{\sigma_X \sigma_Y}} \quad (3\text{-}5)$$

Tabla 3-4. Matriz de correlación.

	num_casos	num_casos_7	num_casos_ant	Densidad	Población	Extensión	TMAX_10	TMIN_10	AENA
num_casos	1.000								
num_casos_7	0.831	1.000							
num_casos_ant	0.976	0.855	1.000						
Densidad	0.026	0.024	0.026	1.000					
Población	0.542	0.466	0.539	0.002	1.000				
Extensión	-0.035	-0.029	-0.032	-0.437	-0.102	1.000			
TMAX_10	0.018	0.022	0.034	0.113	0.099	-0.048	1.000		
TMIN_10	0.030	0.042	0.045	0.310	0.246	-0.322	0.411	1.000	
AENA	0.264	0.227	0.261	0.059	0.822	-0.213	0.111	0.296	1.000

De esta matriz podemos observar que existe una relación positiva fuerte entre la variable 'Aena' y la Población de cada provincia, como es lógico. También existe cierta relación directa entre el número de casos y la población. El resto de correlaciones con valor positivo son más débiles, con valores entre el 0 y el 0.5, como por ejemplo las existentes entre la variable 'Tmax_10' y el resto de variables, exceptuando 'Tmin_10'. Respecto a la variable número de casos positivos, 'num_casos', que sería el dato más importante, podemos ver que la temperatura mínima, 'Tmin_10', está relacionada de una forma más fuerte con el número de casos que la temperatura máxima, 'Tmax_10'. Esto es significativo, ya que reafirma la teoría de que es más probable contagiarse en un clima frío.

Por otro lado, no existe ninguna relación inversa fuerte, de hecho son todas mayores a -0.5. Resulta llamativo que todas las correlaciones con valor negativo sean referidas a la variable 'Extensión'. En el caso de la correlación entre 'Extensión' y 'Densidad' y también en 'Extensión' y 'Población' es lógico, ya que por la formula de la densidad, podemos ver que éstas <son en ambos casos variables inversamente proporcionales. Otras relaciones inversas no son tan fáciles de explicar a simple vista, como la relación entre 'Aena' y 'Extensión'.

3.1.2. Procesamiento de datos

Al implementar Machine Learning con datos reales es común carecer de algunos datos concretos. En este trabajo se ha encontrado este problema con la temperatura de un día concreto, el 23 de febrero de 2020, en la provincia de La Rioja (Lo). Para este día desconocemos tanto la temperatura máxima registrada como la mínima registrada. En ciencia de datos se conoce a este fenómeno como huecos o 'missing values'.

Existen varias formas de tratar esta ausencia de datos, como eliminar las filas que carecen de datos o usar algoritmos para sustituir estos datos ausentes. La opción utilizada en este caso ha sido sustituir el valor que falta con la media de este tipo de dato, es decir la media de la Tmax_10 en la provincia de La Rioja para el periodo estudiado, y la media de Tmin_10 en la provincia de La Rioja para este mismo periodo. Esto es posible porque la variable es una variable continua, es decir, que puede tomar cualquier valor dentro de un

intervalo. La ventaja de este método comparado con, por ejemplo, borrar la fila referente a ese día en esta provincia, es que evitamos perder datos, y los inconvenientes son que puede provocar 'data leakage' o 'fuga de datos' y que no se tiene en cuenta la covarianza de los datos.
Los resultado son un Tmax_10 de 15.8 ºC y una Tmin_10 de 4ºC.

3.2. Implementación en Python

Python es un lenguaje de programación creado en los años 90. Es un lenguaje de programación de alto nivel, orientado a objetos y de tipo dinámico. En la filosofía de este lenguaje destaca la búsqueda de legibilidad de código y la practicidad, prefiriendo siempre lo simple frente a lo complejo, y lo explicito frente a lo implícito. Es un lenguaje de programación multiparadigma, esto implica que permite utilizar distintos estilos de programación. Estos estilos son los siguientes:

i. Programación orientada a objetos
ii. Programación imperativa
iii. Programación funcional
iv. Otros paradigmas que se soportan mediante extensiones

Posee una licencia de código abierta, denominada Python Software Foundation License. Para implementar Python en este trabajo se ha utilizado Spyder en Anaconda. Anaconda es una distribución libre y abierta de los lenguajes Python y R, comúnmente usado en Machine Learning. Dentro de éste se encuentra Spyder (Scientific Python Development Environment), el cual, como su nombre indica es un entorno para el desarrollo de códigos en lenguaje Python.

Para la ejecución de este proyecto se han usado algunas librerías, pero probablemente la más importante de ellas sea Scikit-Learn. Esta librería nos ofrece herramientas para implementar algoritmos de Machine Learning en Python, tanto supervisados como no supervisados. Se sustenta en la biblioteca NumPy, cuya principal habilidad es la capacidad para crear y trabajar con vectores y matrices de grandes dimensiones.

3.2.1. Separación en test y train

Aunque se ha nombrado en la introducción a Machine Learning del Capitulo 2, en este apartado se va a explicar más detenidamente la separación que se ha hecho de los datos existentes para conseguir una buena predicción al aplicar el algoritmo. Los datos se han dividido en dos subconjuntos:

- Conjunto de entrenamiento: subconjunto usado para entrenar el modelo. La calidad de nuestro modelo va a depender fuertemente de la calidad de los datos proporcionados. Estos datos son los que a continuación se denominan datos tipo 'train'.

- Conjunto de prueba: subconjunto para probar el modelo ya entrenado. Este conjunto, aunque es menor en número que el conjunto de entrenamiento, debe ser representativo del conjunto de datos globales. Estos datos son los que a continuación se denominan como datos tipo ‘test’.

En este caso hemos usado el 70% de los datos como conjunto de entrenamiento, y el 30% restante forma parte del conjunto de prueba. Esta división se ha hecho de forma aleatoria usando la función train_test_split() de scikit-learn.

RESULTADOS

En este apartado se procede a comentar los distintos resultados obtenidos al implementar los modelos, para después comparar los mismos. Es importante saber que estos cálculos se han hecho con una entrada 'x' que corresponde a una matriz cuyas filas son los distintos días de estudio en cada una de las provincias estudiadas y cuyas columnas corresponden a las variables 'num_casos_ant', o 'num_casos_7', 'Densidad', 'Población', 'Extensión', 'Tmax_10', 'Tmin_10' y 'AENA' respectivamente. La salida real con la que vamos a comparar los resultados es un vector 'y' con el número de casos positivos, 'num_casos' real que se produjo el día que se quiere calcular el número de contagiados. Cada uno de los apartados está dividido en tres escenarios distintos.

I. Escenario 0: Se utilizan variables no normalizadas.
II. Escenario 1: Se utilizan variables normalizadas.
III. Escenario 2: Se hacen predicciones semanales con variables normalizadas.

Cabe destacar que los resultados esperados de estos cálculos se basan en hallar una relación directa con algunas variables, como el número de casos registrados en una fecha anterior, la densidad de la provincia, y la variable 'Aena', y una relación inversa con las variables relacionadas con las temperaturas máximas y mínimas registradas en la provincia.

4.1. Regresión Lineal Múltiple

4.1.1. Escenario 0: Regresión Lineal Múltiple con variables no normalizadas

En primer lugar se va a aplicar la regresión lineal múltiple. Al implementarla hemos obtenido la predicción de la variable de salida 'y', un vector de coeficientes 'w' y un intercept 'b'. Cada uno de los componentes del vector 'w' corresponden a una de las columnas de la matriz 'x' que hemos utilizado en los cálculos. En la Tabla 4-1 se ilustran los valores de los coeficientes de 'w' y se indica a qué parámetro corresponden.

Tabla 4-1. Coeficientes de 'w' en Regresión Lineal múltiple con variables no normalizadas.

	num_casos_ant	Densidad	Población	Extensión	Tmax_10	Tmin_10	Aena
w	0.9800	0.0012	0.0000	-0.0001	-0.3650	-0.2710	0.0000

Como podemos observar, hay una gran diferencia entre el coeficiente que corresponde a 'num_casos_ant' y los demás. Esto sucede por dos razones, primero, porque el conjunto de valores 'número de casos positivos', al ser similar a una serie temporal, suele tener cierta continuidad y no dar saltos muy grandes de un día para otro. Por esto, es muy útil servirse del número de casos del día anterior para ver cuántos serán

los del día siguiente. La segunda razón es que nuestros datos tienen órdenes de magnitud muy distintas, por tanto es normal que para un dato como es el número de casos del día anterior, cuya orden de magnitud es pequeña, se tenga que maximizar su coeficiente para que sea relevante a la hora de determinar la salida de los datos. Dejando este factor atrás, entre los restantes, los datos con más relación entre el número de diagnosticados y el número de infectados son 'Tmax_10' y 'Tmin_10'. Estos datos tienen una relación inversa con la variable de salida, lo cuál es lógico, ya que cuánto mayor sea el valor de la temperatura, ya sea esta la máxima o mínima registrada, menor será el número de casos positivos de Covid-19. Las variables restantes tienen un coeficiente tan pequeño que no es relevante en los resultados. En la Figura 4-1 se han representado los coeficientes de 'w' en el modelo de Regresión Lineal para una mejor comprensión de los datos.

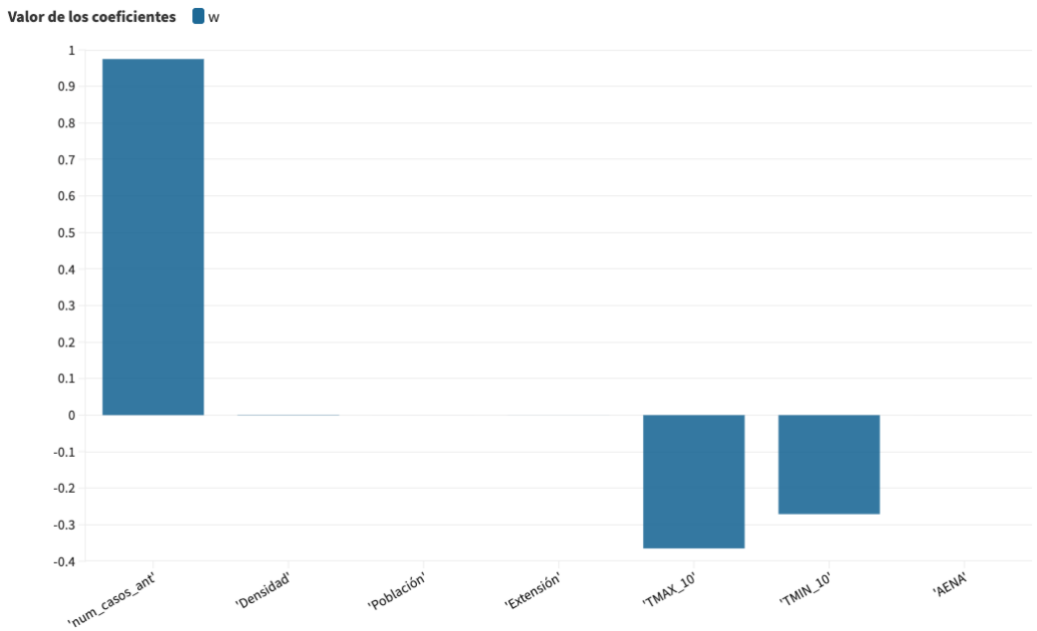


Figura 4-1. Representación del valor de los coeficientes de 'w' en Regresión Lineal múltiple con variables no normalizadas.

También se ha obtenido un intercept 'b' de valor b= 3.15, pero esto no tiene un sentido matemático que podamos aplicar a la interpretación de los datos. A continuación se representan en las Figuras 4-2 y 4-3 los gráficos en los que se comparan el número total de positivos calculado y el número real de enfermos, tanto para los datos denominados 'train', como para los datos 'test'. En la Figura 4-2 los datos en color verde corresponden a los valores reales, mientras que los datos en color rojo corresponden a los valores calculados con los datos de entrenamiento tipo 'train'. Los valores del eje 'y' representan el número de positivos totales, y los valores del eje 'x' son los días desde el 1 de Febrero de 2020 hasta el 31 de Marzo de 2020. En todas las figuras que muestran esta representación en los distintos apartados que siguen el trabajo, el código de colores y las leyendas siguen la misma norma.

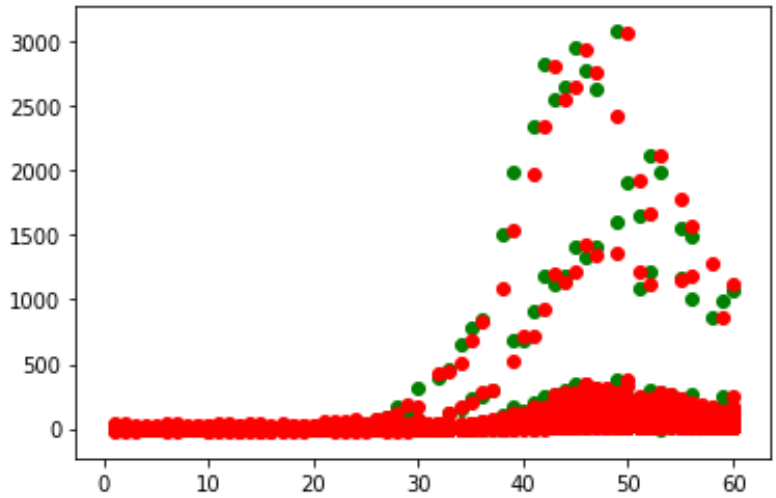


Figura 4-2 Representación de la comparación de las predicciones para los datos de 'train' (color rojo) calculados con Regresión lineal múltiple y los datos reales de positivos en España (color verde) en el periodo estudiado.

En la Figura 4-3, los datos de color verde corresponden nuevamente a los datos reales y los datos en color azul corresponden a los valores calculados con los datos tipo ‘test’.

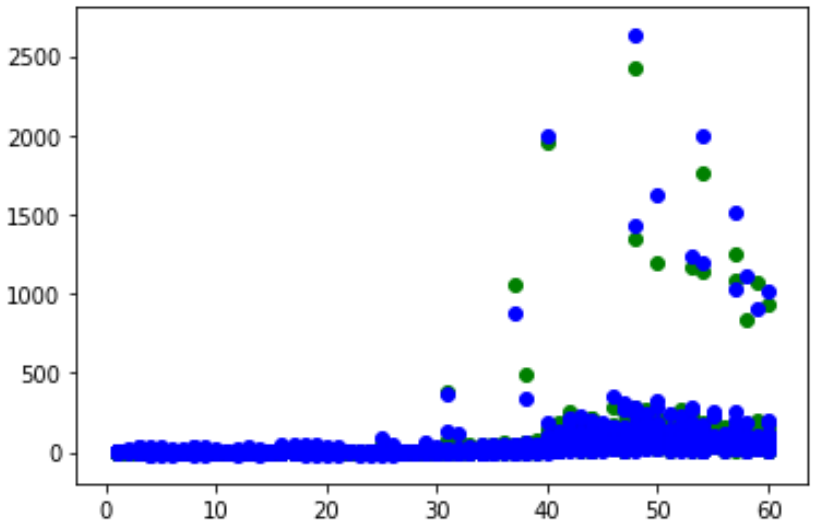


Figura 4-3. Representación de la comparación de las predicciones para los datos de 'test' (color azul) calculados con Regresión lineal múltiple y los datos reales de positivos en España (color verde) en el periodo estudiado.

Los errores cometidos tanto para los datos de entrenamiento ‘train’ como para los de evaluación ‘test’ se representan a continuación, en la Tabla 4-2.

Tabla 4-2. Errores en el modelo de Regresión Lineal.

	MSE	RMSE	MAE	R^2
Train	1956.56	44.23	13.58	1.00
Test	877.97	29.63	12.75	1.00

En conjunto los errores son algo elevados, aunque el coeficiente de correlación es perfecto, por eso en los siguientes apartados se van a normalizar las variables de entrada para mejorar los resultados.

Observando estos resultados, podemos concluir que 'num_casos_ant' es una variable fundamental para el buen funcionamiento de este modelo de regresión lineal, aunque otras variables también son importantes, como la temperatura máxima y mínima.

4.1.2. Escenario 1: Regresión lineal múltiple con transformación de variables

Como se ha comentado en el apartado anterior, es complicado cuantificar la importancia de cada una de las variables en el vector de coeficientes 'w' porque nuestras variables tienen órdenes de magnitud muy distintos. Para solucionar este problema se ha planteado normalizar cada una de estas variables con el comando *MinMaxScaler()*, también de la librería *sklearn.* Los resultados se muestran a continuación. En primer lugar el vector 'w_norm', que corresponde a los coeficientes que resultan de aplicar regresión lineal múltiple a los datos normalizados se muestra en la Tabla 4-3.

Tabla 4-3. Coeficientes de 'w' en Regresión lineal normalizada múltiple y variables normalizadas.

	num_casos_ant	Densidad	Población	Extensión	Tmax_10	Tmin_10	Aena
w_norm	0.9560	0.0008	0.0000	-0.0001	-0.3680	-0.2450	0.0000

Una vez más, la relación con el número de casos anteriores es mucho más fuerte que las demás. En la Figura 4-4 se ilustra la comparación entre los coeficientes resultantes de aplicar regresión lineal múltiple sin normalizar ('w') y normalizada ('w_norm'). Observando la imagen, podemos ver que la relación de cada una de las variables con la variable de salida es muy similar a la hallada anteriormente. Como en el apartado anterior, las únicas variables con coeficientes no despreciables son 'num_casos_ant' y la temperatura máxima y mínima.

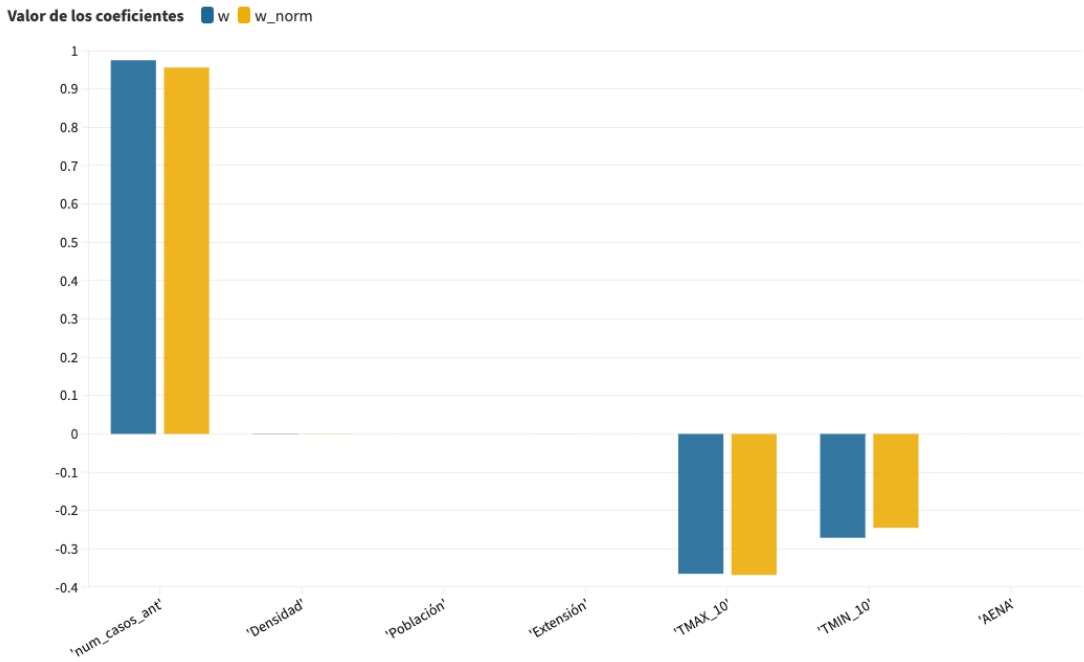


Figura 4-4. Comparación de los componentes de 'w' y 'w_norm'.

A continuación, en las Figuras 4-5 y 4-6, se representa la comparación entre datos reales y calculados en Regresión Lineal normalizada. Los datos reales se representan en color verde, y los puntos de datos de color rojo corresponden a los calculados con las variables de entrada tipo ‘train’. En la Figura 4-6 los datos reales se representan en color verde y los datos calculados con las variables ‘test’ se representan en azul.

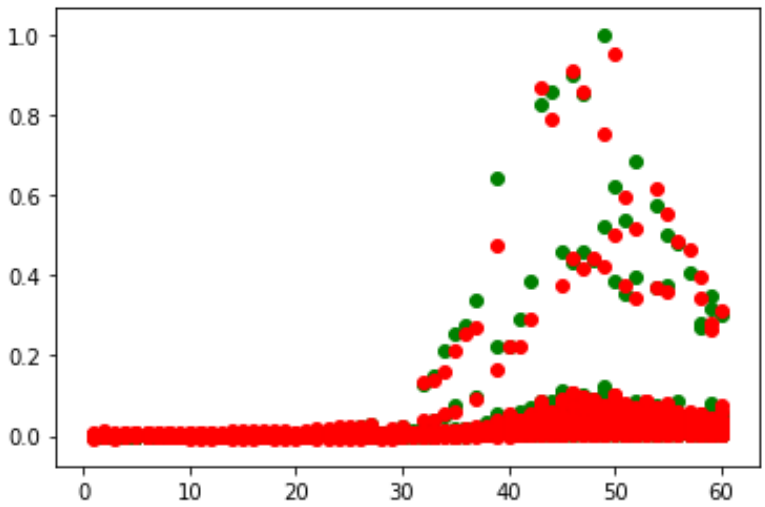


Figura 4-5. Representación de la comparación de las predicciones para los datos de 'train' (color rojo) calculados con Regresión lineal múltiple normalizada y los datos reales de positivos en España (color verde) en el periodo estudiado.

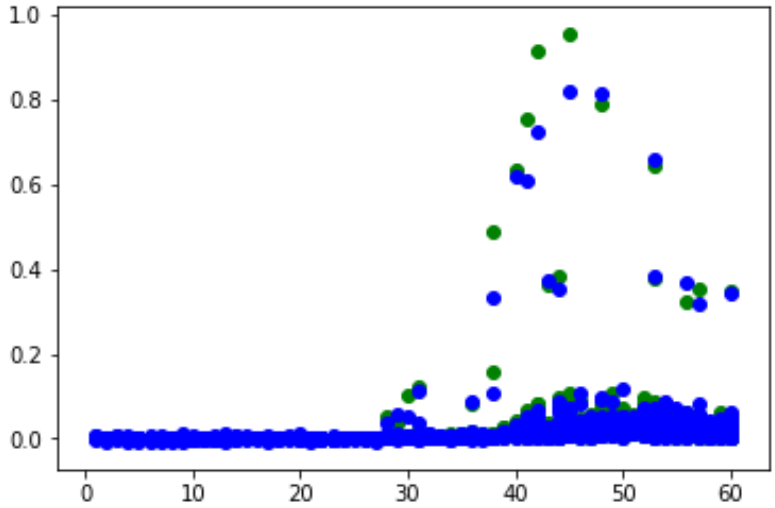


Figura 4-6. Representación de la comparación de las predicciones para los datos de 'train' (color rojo) calculados con Regresión lineal múltiple normalizada y los datos reales de positivos en España (color verde) en el periodo estudiado.

El error cometido tras normalizar los datos se muestra en la Tabla 4-5. Podemos ver que tras este cambio, el error ha disminuido cuantiosamente en todos los casos, y el coeficiente de determinación R^2 sigue teniendo un valor igual a la unidad, es decir

Tabla 4-4. Error cometido en la Regresión Lineal múltiple normalizada.

	MSE	RMSE	MAE	R^2
Train	0.0296	0.1720	0.0584	1.0000
Test	0.0539	0.2320	0.0663	1.0000

4.1.3. Escenario 2: Regresión lineal múltiple para predicciones semanales con transformación de variables

Ya que el uso de la variable ‘num_casos_ant’ es algo limitante, ya que solo permite calcular el número de positivos previsto para el día siguiente, se ha optado por añadir una nueva variable, llamada ‘num_casos_7’ referente al número de positivos hallado con una semana de anterioridad, para así predecir los de la semana próxima. Por tanto, en este escenario la variable x_1 pasa a ser ‘num_casos_7’ y la variable ‘num_casos_ant’ no se utiliza. En este escenario, las variables siguen estando normalizadas. Los resultados son los siguientes:

En primer lugar, se ha decidido nombrar al vector de coeficientes como ‘w_norm_7’, indicando que las variables son normalizadas y que el dato de positivos de entrada pertenece al notificado la semana anterior.

En la Tabla 4-5 se ilustra el valor de los coeficientes de este nuevo vector, y en la Figura 4-7 se comparan los coeficientes de este vector con los calculados anteriormente.

Tabla 4-5. Coeficientes de 'w_norm_7'.

	num_casos_7	Densidad	Población	Extensión	Tmax_10	Tmin_10	Aena
w_norm_7	0.7910	0.0130	0.1530	-0.0058	0.0010	-0.0228	-0.1000

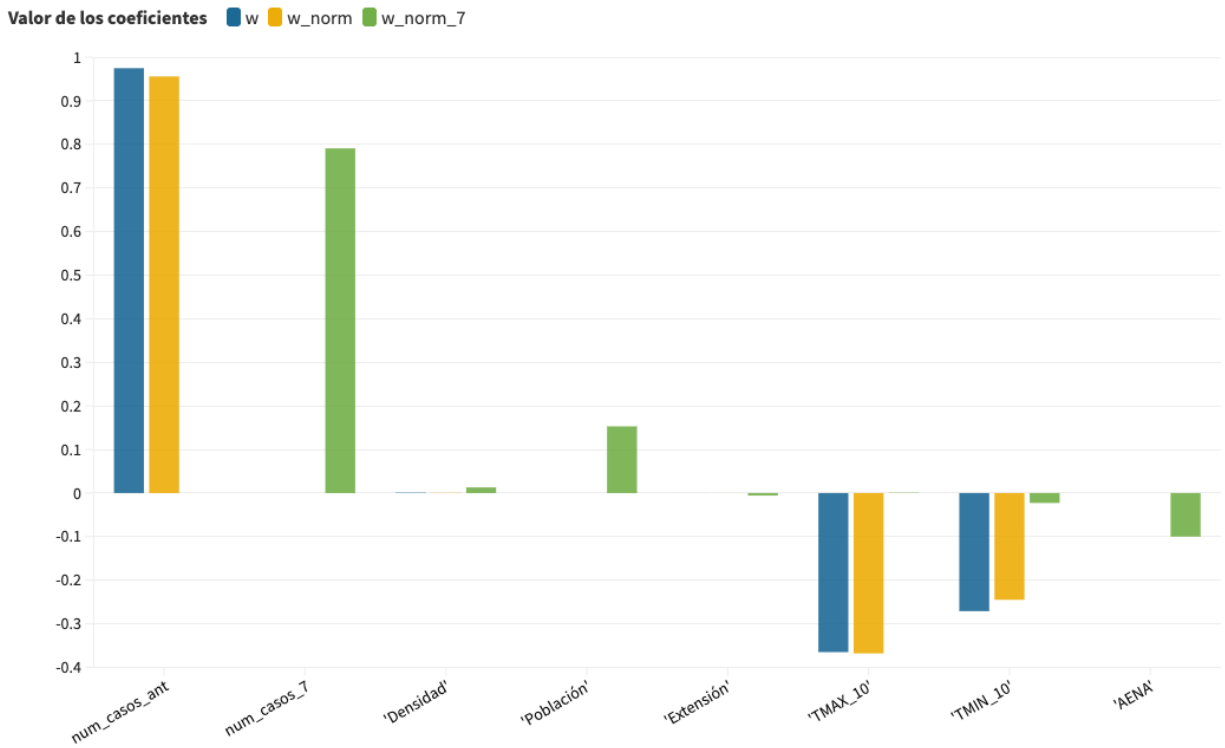


Figura 4-7. Comparación de los coeficientes calculados en los distintos escenarios para Regresión Lineal Múltiple.

De este nuevo cálculo sacamos varias conclusiones distintas. En primer lugar, vemos que en este caso el número de casos notificados anteriormente pierde importancia respecto a cuando usábamos el número de casos del día anterior. A pesar de esto, sigue siendo el dato con mayo importancia. Por otro lado, resulta llamativo que en este caso las variables relacionadas con la temperatura pierdan importancia y pasen a ser irrelevantes, mientras que otras variables, que hasta ahora no habían tenido mucho interés, como ‘Aena’, ‘Población’, y en menor medida ‘Densidad’ cobran mayor importancia a la hora de predecir el número de contagios. El incluir la predicción semanal arroja datos interesantes, como que tanto la densidad como la población de la provincia tienen un valor de correlación positivo, lo que va acorde con la hipótesis inicial de que es más probable contagiarse en lugares con mayor densidad de población. Al contrario, nuestra variable ‘Aena’, tiene una relación negativa, lo cual va en contra de nuestra hipótesis inicial de que los contagios aumentan si llega un mayor número de pasajeros a la provincia estudiada.

En las Figuras 4-8 y 4-9 se representa la comparación entre datos reales y datos calculados. En el Eje X se representa el eje temporal de los días de estudio, es decir, desde el 1 de Febrero de 2020 hasta el 31 de Marzo de 2001. En el Eje Y se representa el número de positivos reales y predichos, ambos normalizados, es decir, el valor de 'y' igual a 1 corresponde con el máximo de positivos registrado, 3086, y el resto de valores están normalizados en esa escala.

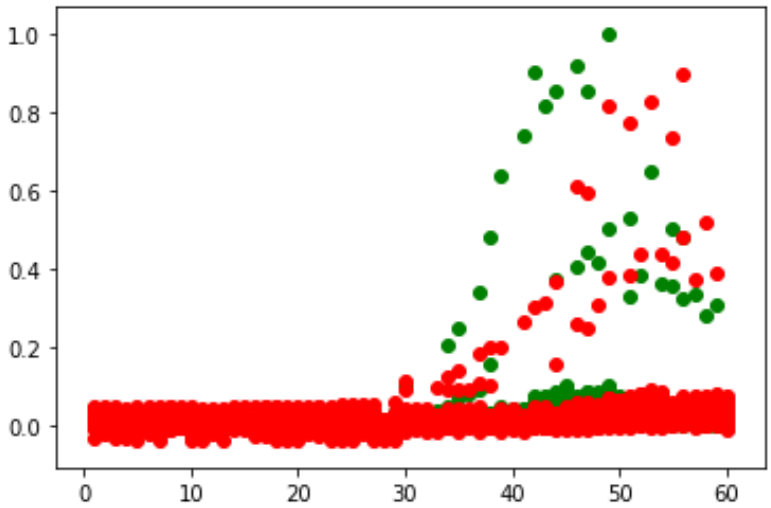


Figura 4-8. Representación de la comparación de las predicciones para los datos de 'train' (color rojo) calculados con Regresión lineal múltiple normalizada para predicciones semanales y los datos reales de positivos en España (color verde) en el periodo estudiado.

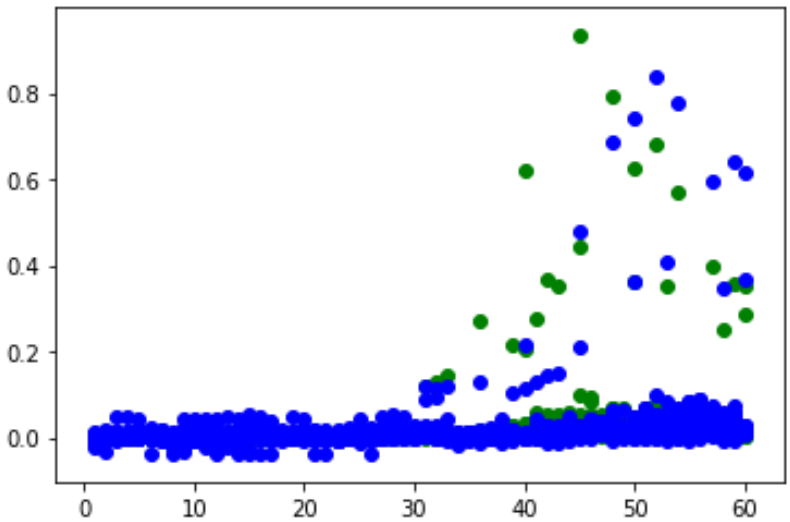


Figura 4-9. Representación de la comparación de las predicciones para los datos de 'test' (color azul) calculados con Regresión lineal múltiple normalizada para predicciones semanales y los datos reales de positivos en España (color verde) en el periodo estudiado.

A pesar de que en las figuras anteriores parece que la predicción se ajusta a los datos reales peor que en otros casos, los errores calculados son menores a los de apartados anteriores. Esto puede ser debido a haber

normalizado la variable de salida, ya que se pierde precisión en la gráfica. Estos errores se representan en la Tabla 4-6.

Tabla 4-6. Errores calculados en Regresión Lineal Múltiple con variables normalizadas para predicciones semanales.

	MSE	RMSE	MAE	R^2
Train	0.0012	0.0344	0.0126	1.00
Test	0.0012	0.0340	0.0140	1.00

Al igual que en el apartado anterior, el error en todos los casos es muy pequeño, esto sucede por haber normalizado las variables. Podemos concluir que tras modificar la variable 'num_casos_ant' por la variable 'num_casos_7', obtenemos una predicción para un intervalo mayor de tiempo sin perder precisión en los datos.

4.2. Regresión polinómica de segundo grado

4.2.1. Escenario 0: Regresión polinómica de segundo grado con variables no normalizadas

A la hora de implementar el modelo de regresión polinómica el resultado varía en función del grado del polinomio escogido. Por ello, este el capitulo 4 contiene dos subapartados más, uno para los cálculos de regresión polinómica con grado 2, y otro para los cálculos de regresión polinómica con grado 3.

Para el caso de un polinomio de grado 2, al transformar la entrada en polinómica y ajustarla mediante *LinearRegression,* obtenemos un vector de coeficientes 'w' de 36 elementos, y un intercept 'b'. En este caso, relacionar con qué elemento corresponde cada coeficiente de 'w' es un poco más complicado que en la regresión lineal, por el hecho de haber transformado los datos de entrada en polinómicos. En la Tabla 4-7, se ilustra el coeficiente calculado para cada una de las variables, variables al cuadrado, y variables multiplicadas entre sí. Para facilitar la lectura, en la Tabla 4-8 se indica la correspondencia de variables.

Tabla 4-7. Vector 'w' en Regresión Polinómica de grado 2.

1	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_1^2
-1.104E-05	1.24E+00	-6.99E-03	5.83E-03	-1.30E-04	9.51E-01	-4.79E-01	4.65E-05	-1.31E-04
x_1x_2	x_1x_3	x_1x_4	x_1x_5	x_1x_6	x_1x_7	x_2^2	x_2x_3	x_2x_4
3.95E-05	2.52E-08	6.18E-06	-1.80E-02	-1.07E-02	7.93E-08	3.11E-07	3.66E-08	-5.87E-03
x_2x_5	x_2x_6	x_2x_7	x_3^2	x_3x_4	x_3x_5	x_3x_6	x_3x_7	x_4^2
1.30E-04	2.86E-05	-4.37E-08	-2.07E-12	-3.37E-10	2.57E-06	-3.74E-07	-1.96E-12	3.46E-09
x_4x_5	x_4x_6	x_4x_7	x_5^2	x_5x_6	x_5x_7	x_6^2	x_6x_7	x_7^2
-9.87E-07	4.62E-05	-1.26E-09	-5.04E-02	3.08E-02	-3.15E-06	-1.46E-02	1.25E-06	7.55E-12

Tabla 4-8. Correspondencia de variables.

x_1	x_2	x_3	x_4	x_5	x_6	x_7
num_casos_ant	Densidad	Población	Extensión	Tmax_10	Tmin_10	AENA

En la Figura 4-10 se representan gráficamente los coeficientes de esta regresión. Aunque en este caso los coeficientes son más difíciles de interpretar, podemos ver que el mayor valor de forma positiva se da en x_1, que corresponde al número de casos del día anterior. El siguiente corresponde a 'Tmax_10', lo cual no concuerda con el resultado esperado, ya que al ser una temperatura esperábamos un resultado de relación inversa, con signo negativo, para así justificar que una temperatura más elevada supondrá un menor número de contagios. El valor de 'x_6' sí concuerda con lo esperado, ya que se refiere al coeficiente de temperatura mínima y en este caso sí existe signo negativo. Este es el coeficiente negativo con mayor valor absoluto. Los restantes coeficientes tienen escaso valor absoluto. De ellos el mayor corresponde a 'x_1x_5', la multiplicación de 'num_casos_ant' por 'Tmax_10', y el siguiente a 'Tmax_10' al cuadrado, con valor negativo, lo cual si sigue nuestra hipótesis inicial, ya que así existe una relación inversa con la temperatura máxima. El valor del intercept es de 0.76.

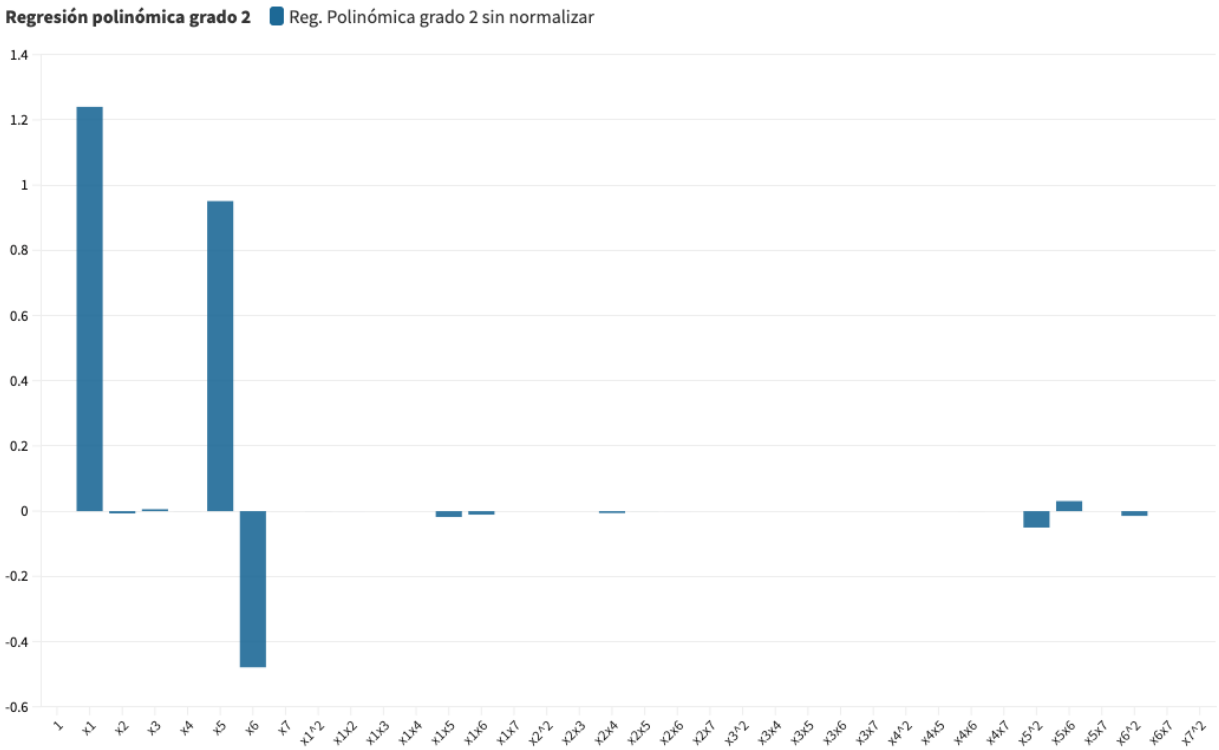


Figura 4-10. Valor de coeficientes en Regresión Polinómica de segundo grado sin normalizar.

Para ilustrar los resultados, en las Figuras 4-11 y 4-12 se puede ver en verde el número de positivos real y el número de positivos calculado. El color verde representa el número de infectados real, en la Figura 4-11, el color rojo corresponde al conjunto de datos pertenecientes a 'train'. En la Figura 4-12, de nuevo el color verde representa el número de infectados real y el color azul los correspondientes al tipo 'test'.

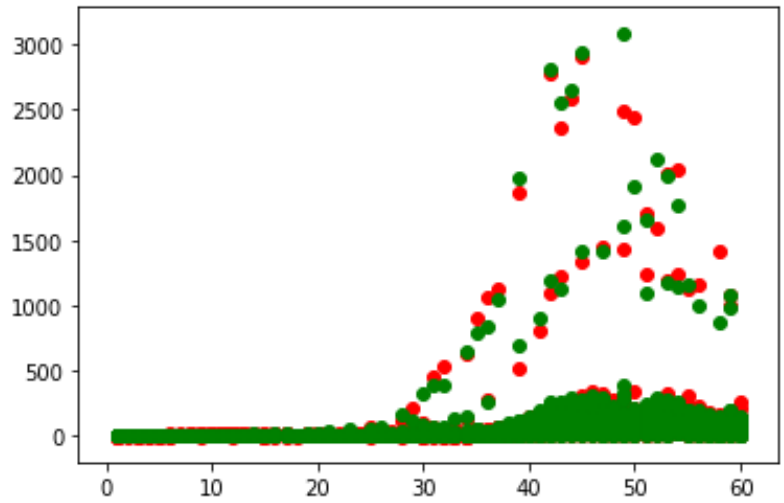


Figura 4-11. Datos correspondientes al resultado de las predicciones para los datos de 'train' (color rojo) con Regresión Polinómica de grado 2 y datos reales de positivos en España (color verde). en el periodo estudiado

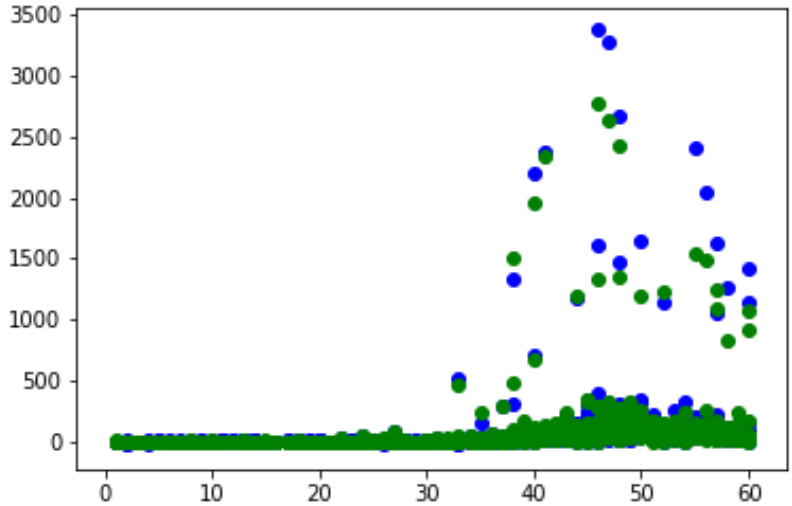


Figura 4-12. Datos correspondientes al resultado de las predicciones para los datos de 'test' con Regresión Polinómica de grado 2 (color azul) y datos reales de positivos en España (color verde) en el periodo estudiado.

Los errores calculados en este modelo se muestran en la Tabla 4-9. Podemos ver que el error cuadrático medio y el error absoluto son algo altos, aunque el coeficiente de determinación se acerca mucho al valor de la unidad, por lo que es un buen ajuste.

Tabla 4-9. Errores calculados para la Regresión Polinómica de grado 2.

	MSE	RMSE	MAE	R^2
Train	1250.00	35.40	11.70	0.9740
Test	1490.00	38.50	0.9580	0.9670

4.2.2. Escenario 1: Regresión polinómica de grado 2 con transformación de variables

Para conseguir unos mejores resultados hemos normalizado de nuevo las variables en valores entre 0 y 1. Los valores de los coeficientes del vector 'w_norm' se muestran en la Tabla 4-10.

Tabla 4-10. Valor de los coeficientes 'w_norm' en la Regresión Polinómica de grado 2 con variables normalizadas.

1	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_1^2
-0.138E-11	1.15	4.13E-02	-4.27E+00	1.12E-01	4.69E-03	-7.24E-04	4.66E-02	-5.28E-01
x_1x_2	x_1x_3	x_1x_4	x_1x_5	x_1x_6	x_1x_7	x_2^2	x_2x_3	x_2x_4
3.52E-01	-1.58E-01	1.53E-01	-1.11E-01	-5.27E-01	1.35E+00	-9.53E-04	4.55E-01	9.39E+01
x_2x_5	x_2x_6	x_2x_7	x_3^2	x_3x_4	x_3x_5	x_3x_6	x_3x_7	x_4^2
1.44E-02	1.28E-02	-3.43E-01	-3.47E-02	-7.53E-03	1.22E-01	4.45E-02	-3.50E-02	8.64E-04
x_4x_5	x_4x_6	x_4x_7	x_5^2	x_5x_6	x_5x_7	x_6^2	x_6x_7	x_7^2
5.67E-03	1.12E-02	-4.63E-02	-1.66E-02	7.08E-03	-1.20E-01	-1.15E-02	3.64E-02	5.61E-02

Comparando esta tabla con la Tabla 4-6, que contiene los coeficientes sin normalizar las variables, vemos que la variable x_1, que corresponde al número de casos en el día anterior, sigue siendo importante pero en este caso no es la que tiene el mayor coeficiente. En la Figura 4-13 se representan dichos coeficientes.

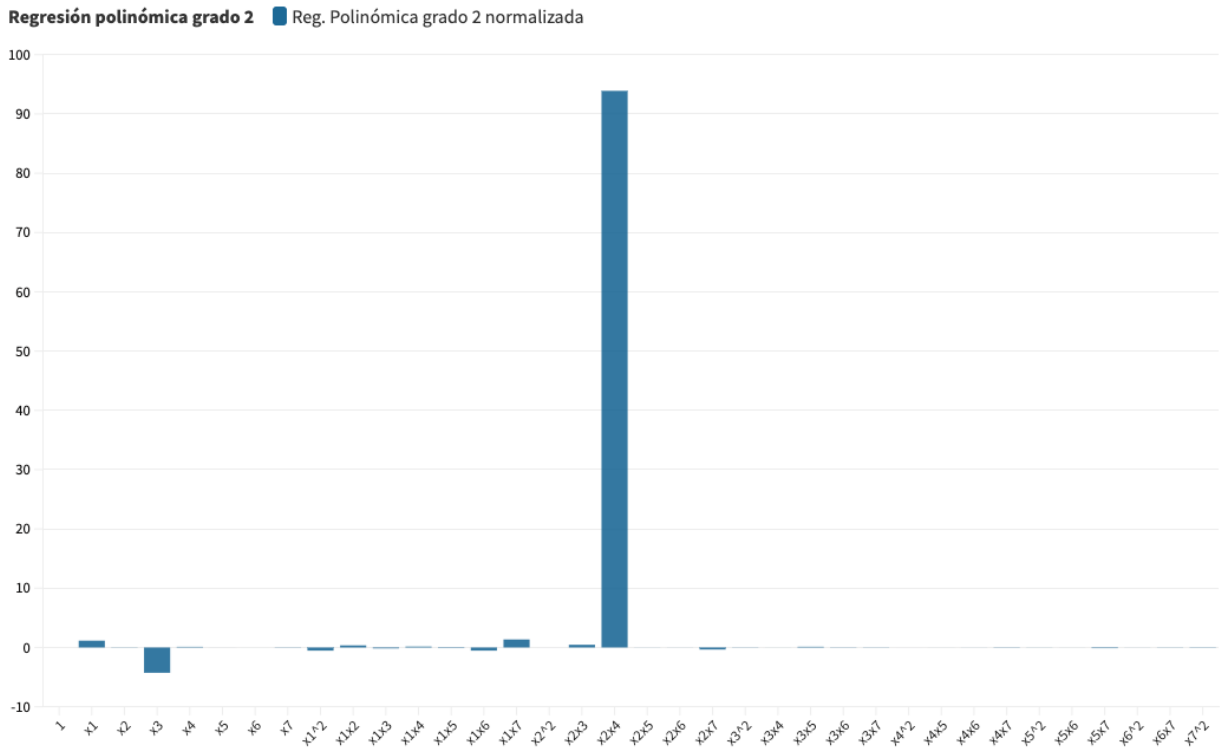


Figura 4-13. Valor de coeficientes en Regresión Polinómica de segundo grado normalizada.

Podemos observar que la variable 'x_2x_4' tiene asociado el coeficiente más alto, con mucha diferencia con los restantes. Como podemos ver en la Ecuación 4-1, este coeficiente corresponde al término 'Población', Es importante resaltar que por tanto este modelo no es tan dependiente del número de positivos registrado en una fecha anterior como los estudiados anteriormente, lo cual es positivo para demostrar que el número de contagiados depende de algunas variables como la densidad, y no solo del estudio de datos de contagios anteriores.

$$x_2x_4 = Densidad * Extensión = \ Población \tag{4-1}$$

Se puede observar un cambio significativo en las variable x_3 y x_3^2, referentes a 'Población', que pasan a tener valores muy negativos en el valor de los coeficientes, cuando antes de normalizar, x_3 tenía un valor positivo. La variable x_5 y la variable x_5^2, referentes a la Temperatura máxima pasan de ser determinantes en la versión normalizada a perder importancia en esta nueva versión del modelo. Al igual que en la regresión lineal, el error ha disminuido cuantiosamente tras normalizar los datos, en este caso esto sucede en todos los tipos de errores. Estos se ilustran en la Tabla 4-11.

Tabla 4-11. Errores cometido en Regresión Polinómica de grado 2 con valores normalizados.

	MSE	RMSE	MAE	R^2
Train	0.0001	0.0119	0.0041	0.9720
Test	0.0001	0.0112	0.0040	0.9650

En las figuras mostradas a continuación, Figura 4-14 y Figura 4-15, se muestra la comparativa entre datos reales y estimados, tanto para la parte de 'train', como la de 'test'. El código de colores es similar al usado anteriormente.

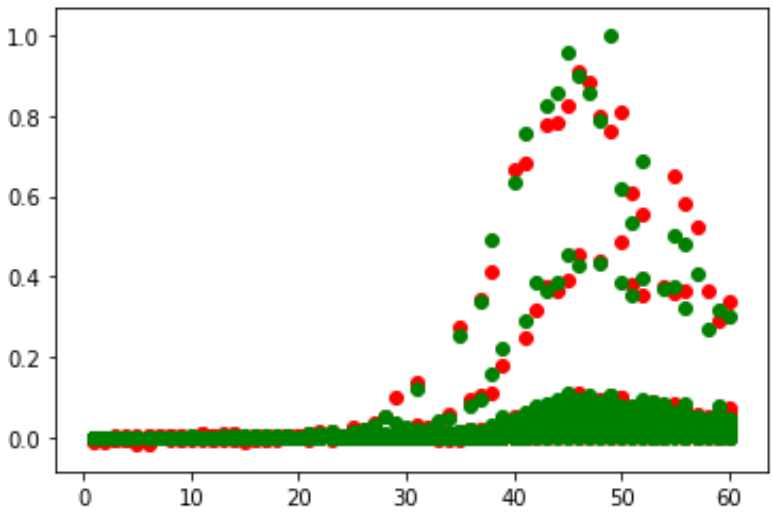


Figura 4-14. Datos correspondientes al resultado de las predicciones para los datos de 'train' (color rojo) con Regresión Polinómica de grado 2 con datos normalizados y datos reales de positivos en España (color verde) en el periodo estudiado.

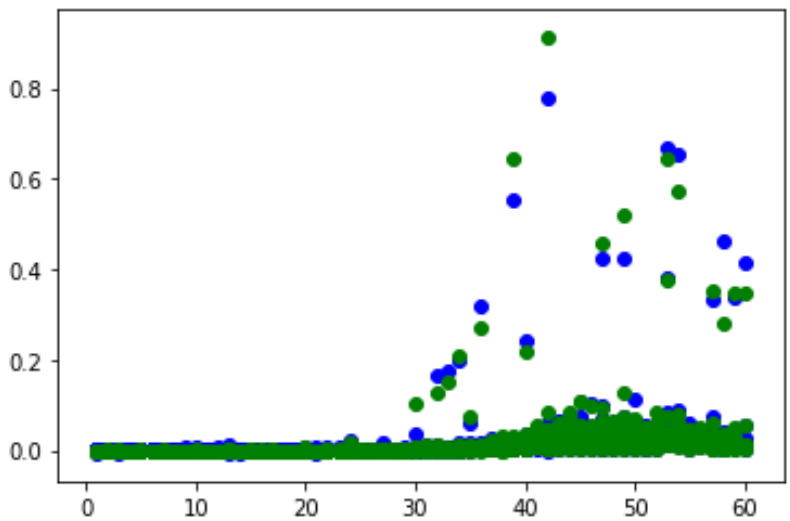


Figura 4-15. Datos correspondientes al resultado de las predicciones para los datos de 'test' (color azul) con Regresión Polinómica de grado 2 con datos normalizados y datos reales de positivos en España (color verde) en el periodo estudiado.

4.2.3. Escenario 2: Regresión polinómica de grado 2 para predicciones semanales con transformación de variables

Al igual que en el apartado 4.1, se ha aplicado al modelo de regresión polinómica normalizada el uso de la variable 'num_casos_7', para así poder calcular la predicción semanal de contagios. Los coeficientes calculados en este apartado se muestran en la Tabla 4-12. Al igual que en el Escenario 2 de la regresión lineal múltiple, se ha llamado a este vector 'w_norm_7', por estar usando variables normalizadas y por ser el dato de positivos de entrada el referente al ocurrido con 7 días de anterioridad. En la Figura 4-16 se representan estos coeficientes.

Tabla 4-12. Coeficientes de 'w_norm_7'.

1	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_1^2
-0.519E-11	1.22E+00	4.93E-01	-4.27E+00	1.10E-00	-2.94E-02	1.16E-04	1.19E-01	-2.66E+00
x_1x_2	x_1x_3	x_1x_4	x_1x_5	x_1x_6	x_1x_7	x_2^2	x_2x_3	x_2x_4
-7.27E-01	-2.85E-01	-2.37E-01	2.36E01	-3.39E+00	5.38E+00	4.25E-02	2.06E+00	8.70E+02
x_2x_5	x_2x_6	x_2x_7	x_3^2	x_3x_4	x_3x_5	x_3x_6	x_3x_7	x_4^2
-2.94E-02	-2.71E-02	-3.18E+00	1.13E-01	1.79E-01	-3.24E-01	1.75E-01	3.52E-03	-6.50E-03
x_4x_5	x_4x_6	x_4x_7	x_5^2	x_5x_6	x_5x_7	x_6^2	x_6x_7	x_7^2
1.73E-02	-4.07E-03	-4.18E-01	7.63E-03	5.90E-02	2.26E-01	-3.01E-02	-1.41E-01	3.89E-02

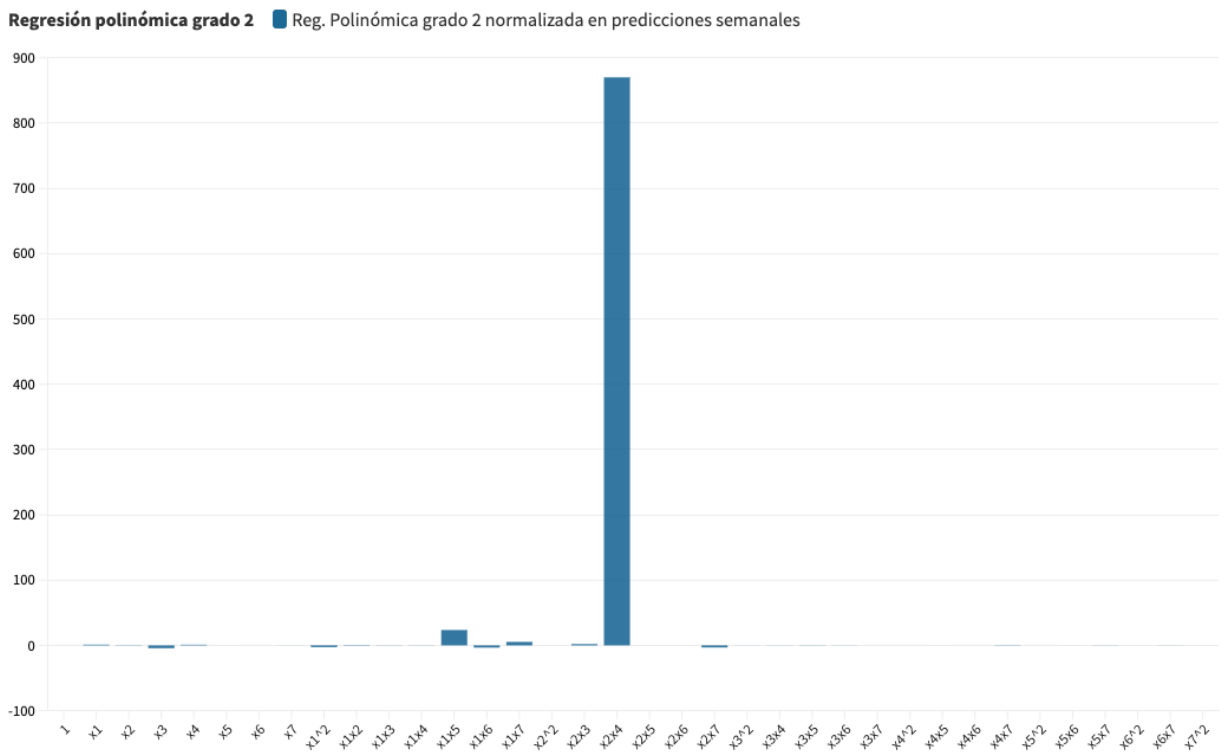


Figura 4-16. Valor de coeficientes en Regresión Polinómica de segundo grado normalizada para predicciones semanales.

Podemos observar que la variable con mayor coeficiente positivo, con gran distancia a los restantes es 'x_2x_4', como ya ocurría en el apartado anterior, y esta variable se detalla en la Ecuación 4-1. Por tanto, este

modelo también es muy dependiente del término 'Población'. Hay otras variables que tienen relación positiva, como 'x_1x_5'. Como se representa en la Ecuación 4-2. Esta corresponde al número de casos registrados hace 7 días multiplicados por la temperatura máxima registrada, por lo tanto alguna de estas dos variables, presumiblemente el número de casos registrados, tenga algo de relevancia en la salida de datos.

$$x_1x_5 = num\ casos\ 7 * Tmax\_10 \tag{4-2}$$

Los coeficientes restantes tienen valores escasos o irrelevantes, por lo que podemos concluir que la salida de datos es dependiente únicamente de las variables ya nombradas.

En las Figuras 4-17 y 4-18 se muestra la comparación entre el número de infectados real y el número de infectados calculado con la modificación de variables de este apartado.

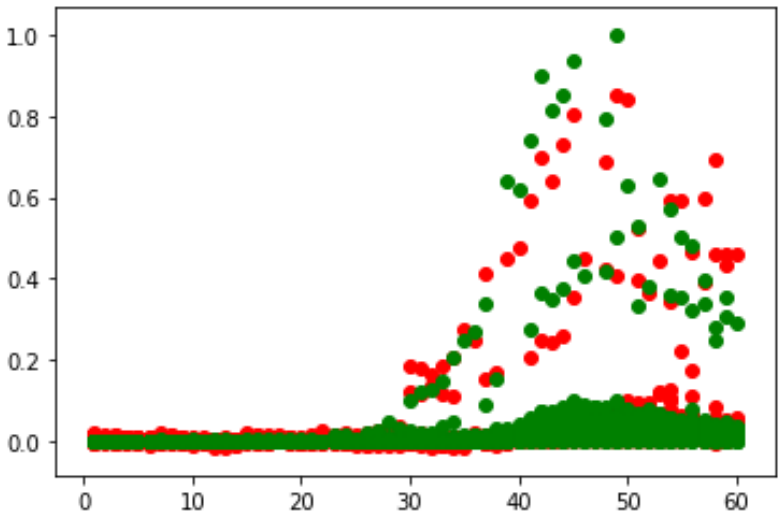


Figura 4-17. Datos correspondientes al resultado de las predicciones semanales para los datos de 'train' (color rojo) con Regresión Polinómica de grado 2 con datos normalizados y datos reales de positivos en España (color verde) en el periodo estudiado.

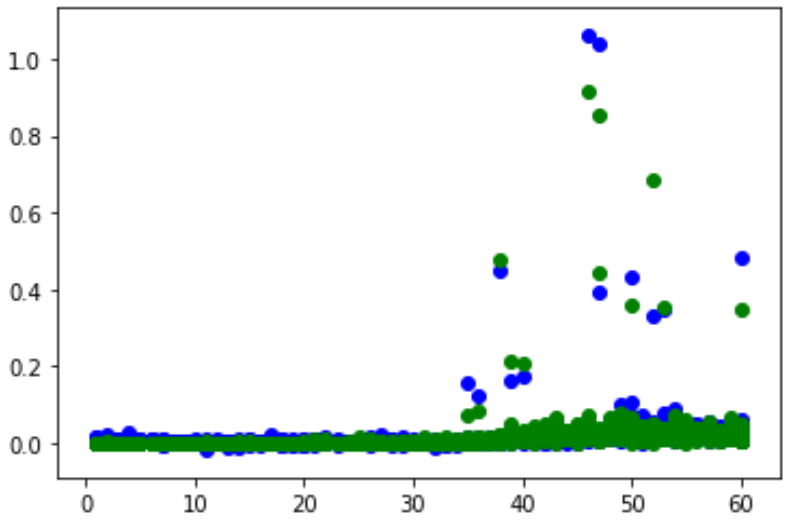


Figura 4-18. Datos correspondientes al resultado de las predicciones semanales para los datos de 'test'' (color azul) con Regresión Polinómica de grado 2 con datos normalizados y datos reales de positivos en España (color verde) en el periodo estudiado.

A continuación se muestra en la Tabla 4-13 los errores calculados para este apartado. Al igual que en regresión lineal múltiple el error cuadrático medio y el error absoluto son muy pequeños. Esto se debe al hecho de haber normalizado las variables, y al hecho de usar el número de positivos ocurridos en una fecha cercana, en este caso, con 7 días de anterioridad. Por otro lado, el coeficiente de correlación R^2 es inferior al calculado anteriormente, lo cual no es beneficioso para la precisión del modelo.

Tabla 4-13. Errores calculados en las predicciones semanales normalizadas en Regresión Polinómica de segundo grado.

	MSE	RMSE	MAE	R^2
Train	0.0005	0.0222	0.0094	0.9010
Test	0.0003	0.0182	0.0079	0.8980

4.3. Regresión polinómica de grado 3

En este apartado se va a aplicar Regresión polinómica de tercer grado a las mismas variables usadas anteriormente. Al igual que en el apartado de regresión lineal múltiple, y el de regresión polinómica de segundo grado, existen tres escenarios distintos detallados a continuación.

4.3.1. Escenario 0: Regresión polinómica de tercer grado con variables no normalizadas

En este primer escenario se utilizan las variables sin normalizar. Para el caso de un polinomio de tercer grado el procedimiento es muy similar al explicado anteriormente en el apartado 'Regresión Polinómica de

segundo grado', aunque el vector de coeficientes 'w' pasa a ser de en esta ocasión 120 elementos, por lo que vamos a nombrar solo los coeficientes más significativos en las Tabla 4-14, y la representación de algunas de las variables con relación positiva se encuentran en la primera columna, mientras que aquellas con valor negativo están en la segunda. La variable de entrada referida al número de casos en un día previo al calculado es en este apartado 'num_casos_ant', que se refiere al día inmediatamente anterior al del cálculo. En la Figura 4-14 se representan algunos de estos coeficientes.

Tabla 4-14. Algunas variables con relación significativa con la variable de salida en la Regresión Polinómica de tercer grado.

Variable	Coeficiente en 'w'	Variable	Coeficiente en 'w'	Variable	Coeficiente en 'w'	Variable	Coeficiente en 'w'
x_1	6.97E-07	x_2^2	-4.55E-05	x_4x_5	4.99E-06	$x_6x_1x_2$	3.04E-05
x_2	-1.05E-06	x_2x_3	5.18E-08	x_4x_6	1.68E-06	$x_1x_4x_5$	2.41E-07
x_3	-5.74E-06	x_2x_4	-5.22E-06	x_4x_7	-3.99E-06	$x_1x_4x_6$	4.78E-07
x_4	-5.49E-07	x_2x_5	3.77E-07	x_6^2	4.24E-10	$x_1x_5^2$	2.71E-06
x_5	-5.33E-07	x_2x_6	5.68E-08	x_6x_7	1.79E-05	$x_1x_5x_6$	1.70E-06
x_7	-1.34E-06	x_2x_7	-8.08E-07	$x_1^2x_4$	-1.01E-07	$x_1x_6^2$	1.27E-06
x_1^2	1.12E-06	x_3x_4	1.33E-06	$x_1^2x_5$	1.89E-05	$x_2x_4^2$	-1.33E-06
x_1x_2	1.93E-06	x_3x_5	3.49E-07	$x_1^2x_6$	-5.43E-06	$x_2x_5^2$	2.79E-05
x_1x_3	5.47E-07	x_3x_6	-1.88E-07	$x_2^2x_1$	-1.97E-07	$x_2x_6x_5$	1.36E-05
x_1x_4	1.05E-04	x_3x_7	1.51E-07	$x_4x_1x_2$	5.65E-07	$x_2x_6^2$	6.90E-06
x_1x_7	-2.15E-06	x_4^2	-2.14E-07	$x_5x_1x_2$	4.50E-05		

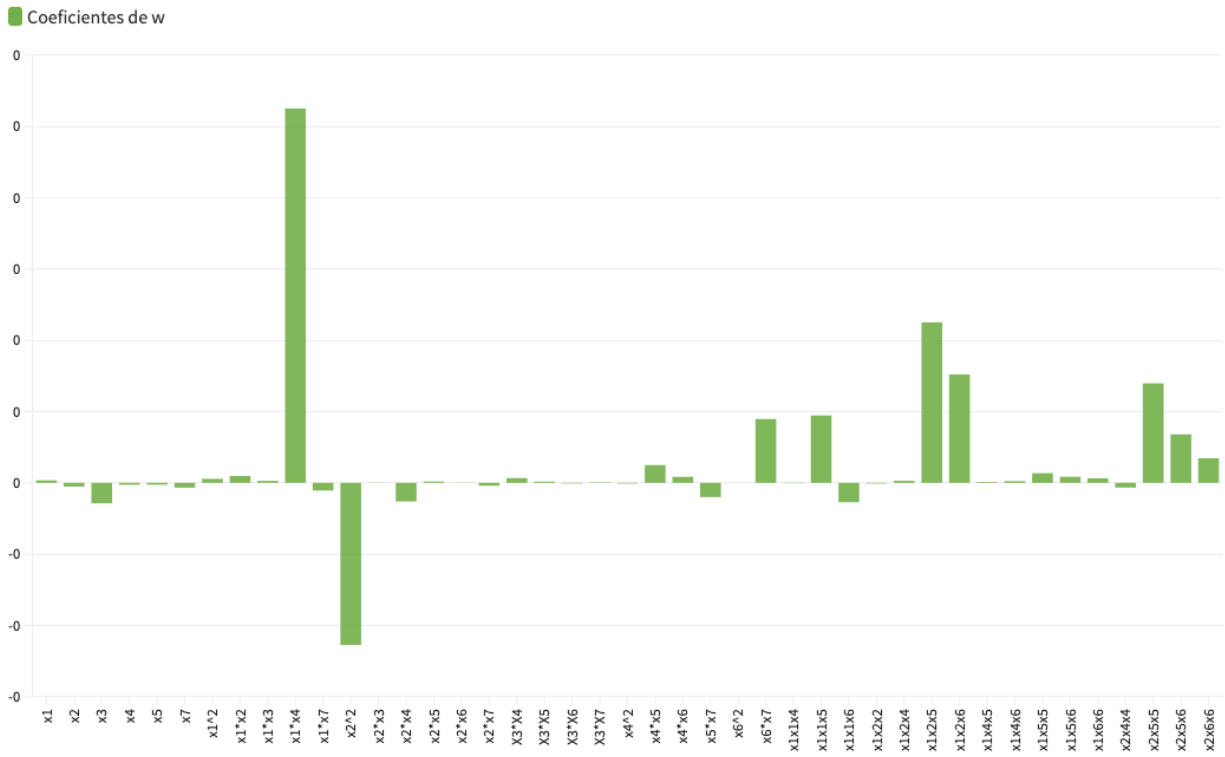


Figura 4-19. Algunos coeficientes de 'w' de Regresión polinómica de tercer grado sin normalizar.

Es difícil sacar conclusiones de estos datos ya que a diferencia de cálculos anteriores muchas de las variables tienen relación con la salida de los datos, aunque sea mínima, y muchas de ellas se ve representadas de forma tanto directa como inversa. La variable con mayor valor absoluto, y gran diferencia con el resto es 'x_1x_4'. Como se indica en la Ecuación 4-3, esta se relaciona con el número de casos del día anterior y la extensión.

$$x_1x_4 = num\ casos\ ant * Extensión \tag{4-3}$$

El siguiente elemento con mayor relación de forma directa es $x_5x_1x_2$. Este multiplica la densidad, el número de casos del día anterior, y la temperatura máxima registrada. El siguiente valor $x_6x_1x_2$ es idéntico, pero refiriéndose a la temperatura mínima registrada en vez de máxima. Por último $x_2x_5^2$, se relaciona de forma directa con la densidad y la temperatura máxima. Observando los valores negativos, podemos ver que el segundo valor con mayor valor absoluto es x_2^2, por lo que este modelo se relaciona aparentemente de forma indirecta con la densidad. En segundo lugar x_3, también tiene una relación inversa con el número de contagiados. Tras describir algunos de los coeficientes más elevados de este modelo, concluimos que algunas de las variables se ven relacionadas con la salida de datos de forma tanto directa como inversa, por lo que es difícil sacar conclusiones claras de como se comporta este modelo.

En la Figura 4-16 y 4-17 se ilustra la comparación entre datos reales de positivos y los datos calculados con nuestro modelo.

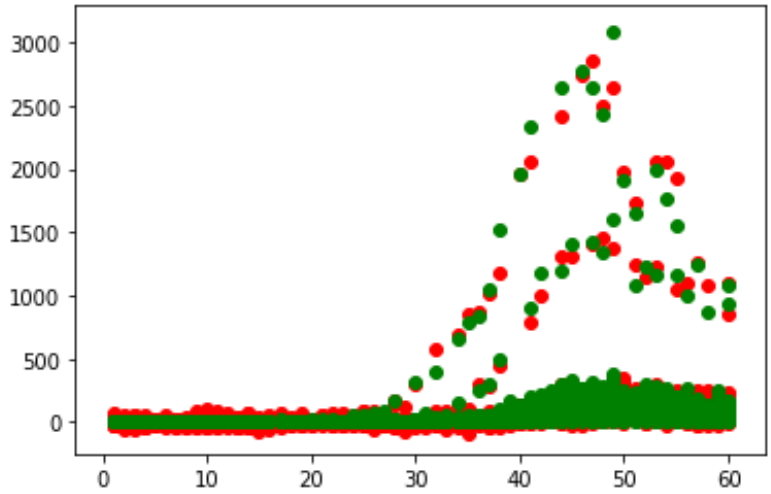


Figura 4-20. Datos correspondientes al resultado de las predicciones para los datos de 'train' con Regresión Polinómica de grado 3 (color rojo) y datos reales de positivos en España en el periodo estudiado (color verde).

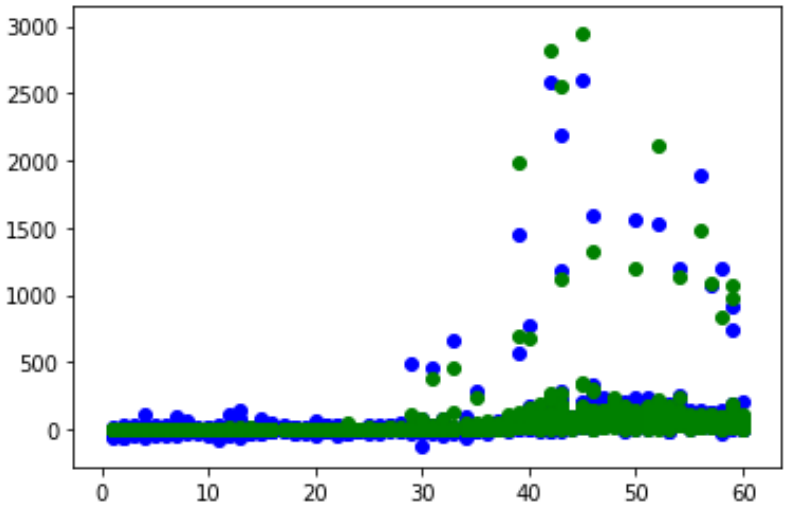


Figura 4-21. Datos correspondientes al resultado de las predicciones para los datos 'test' con Regresión Polinómica de grado 3 (color azul) y datos reales de positivos en España en el periodo estudiado (color verde).

Los errores calculados en esta regresión polinómica de tercer grado se representan en la Tabla 4-15.

Tabla 4-15. Errores cometidos en regresión polinómica de grado 3.

	MSE	RMSE	MAE	R^2
Train	846.2140	29.0890	11.5040	0.9773
Test	5615.9810	74.9390	16.5920	0.9065

Comparando este resultado con el que se da al aplicar regresión polinómica de segundo grado sin normalizar, vemos que aunque la predicción tipo 'train' ha mejorado bastante, el error de los datos tipo 'test' ha empeorado, por lo que en principio no parece beneficioso aplicar el grado 3 en este modelo. El coeficiente de correlación R^2 se mantiene relativamente constante respecto al calculado en Regresión polinómica de segundo grado.

4.3.2. Escenario 1: Regresión polinómica de grado 3 con transformación de variables

Como en casos anteriores, se ha procedido a normalizar las variables de 'x' para conseguir así mejores resultados. Por el gran número de elementos que tiene este vector de coeficientes, al que llamaremos 'w_norm', es más complicado evaluar qué variables son las más importantes en este caso. Se muestran algunos de los coeficientes más extremos en la Tabla 4-16. De nuevo la primera columna corresponde a los coeficientes de valor positivo, y los de la segunda a coeficientes negativos. En la Figura 4-22 se representan estos coeficientes.

Tabla 4-16. Algunos valores de los coeficientes de 'w_norm' en Regresión polinómica de grado 3 con variables normalizadas.

Variable	Coeficientes de 'w_norm'	Variable	Coeficientes de 'w_norm'
x_1x_3	-9.57E+03	$x_4x_1x_2$	2.14E+05
x_2x_3	1.30E+04	$x_4x_2^2$	-2.91E+05
x_2x_4	8.04E+02	$x_4x_3x_2$	8.36E+04
x_3^2	-3.73E+03	$x_2x_4^2$	-4.61E+03
x_3x_4	3.13E+02	$x_4x_5x_2$	-4.34E+03
x_3x_5	1.94E+02	$x_4x_6x_2$	-4.73E+03
x_3x_6	2.11E+02	$x_4x_7x_2$	3.28E+04
x_3x_7	-1.46E+03		

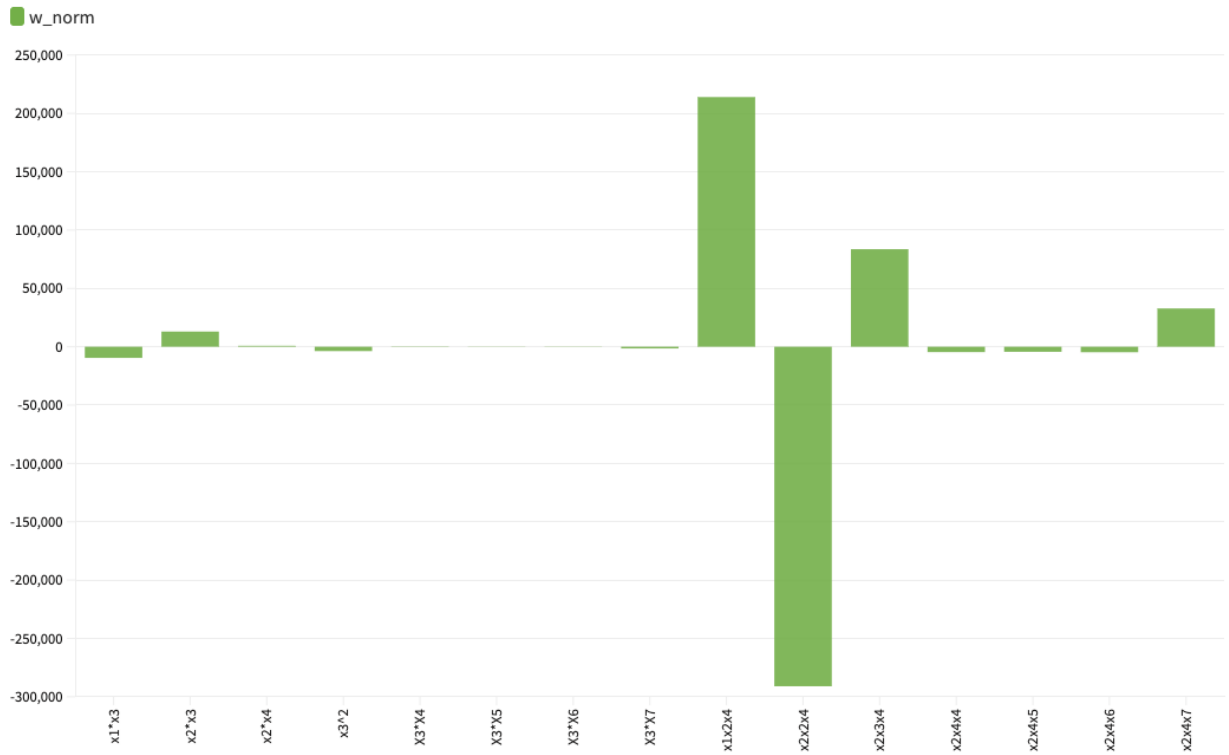


Figura 4-22. Coeficientes de 'w_norm' en regresión polinómica de tercer grado normalizada.

La variable con mayor valor absoluto corresponde a $x_2^2 x_4$ que se representa en la Ecuación 4-4. Esta tiene una relación negativa.

$$x_2^2 x_4 = Densidad^2 * Extensión = \frac{Población^2}{Extensión^2} * Extensión = \frac{Población^2}{Extensión} \quad (4\text{-}4)$$

Por tanto vemos que en este caso hay una relación inversa fuerte con la población. Esto va en contra de nuestra hipótesis inicial. La segunda variable con mayor valor absoluto es $x_1 x_2 x_4$, que como vemos en la Ecuación 4-5, está fuertemente relacionada con el número de casos del día anterior y la población.

$$x_1 x_2 x_4 = num\ casos\ ant * Densidad * Extensión = num\ casos\ ant * Población \quad (4\text{-}5)$$

De $x_2 x_3 x_4$ también sacamos en conclusión que la población es importante en este modelo de forma directa, como se puede ver en la Ecuación 4-6.

$$x_2 x_3 x_4 = Densidad * Población * Extensión = Población^2 \quad (4\text{-}6)$$

Por último también vemos que la variables $x_2 x_4 x_7$ también tienen representación, lo que significa una influencia directa de la variable 'Aena', y de nuevo, 'Población' .

$$x_2 x_4 x_7 = Densidad * Extensión * Aena = Población * Aena \quad (4\text{-}7)$$

Teniendo en cuenta ahora los restantes coeficientes de la tabla, vemos que la mayoría de variables están muy igualadas y aparecen tanto de forma negativa como positiva, por lo que es difícil sacar mayores conclusiones de la interpretación de coeficientes. Esto también sucede con la variable 'Población', como podemos ver en las ecuaciones anteriores. Podemos destacar que la variable 'x_2' referente a 'Densidad', aparece de forma más fuerte como variable directa que indirecta, por lo que seguramente exista una relación positiva con esta variable, lo cual confirmaría nuestra hipótesis de que el número de casos es directamente proporcional a la densidad de la población.

En las Figuras 4-23 y 4-24, se representan las predicciones calculadas en este subapartado.

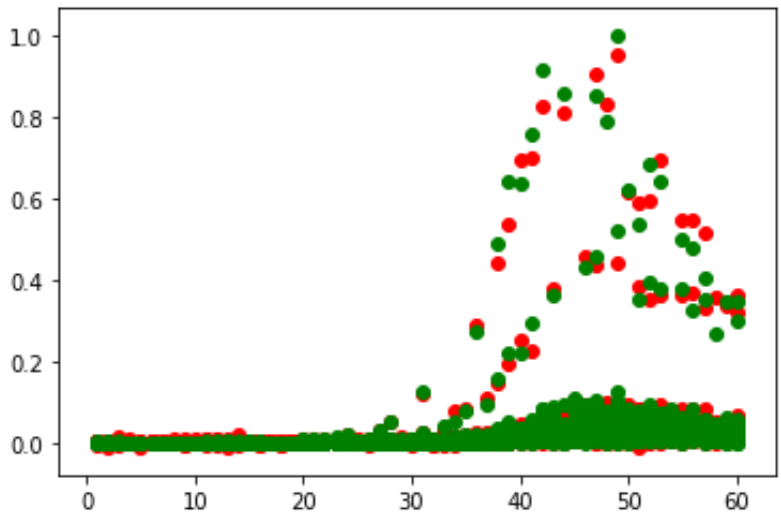


Figura 4-23. Representación correspondiente al resultado de las predicciones para los datos de 'train' con Regresión Polinómica de grado 3 (color rojo) y datos reales de positivos en España en el periodo estudiado (color verde), con las variables normalizadas.

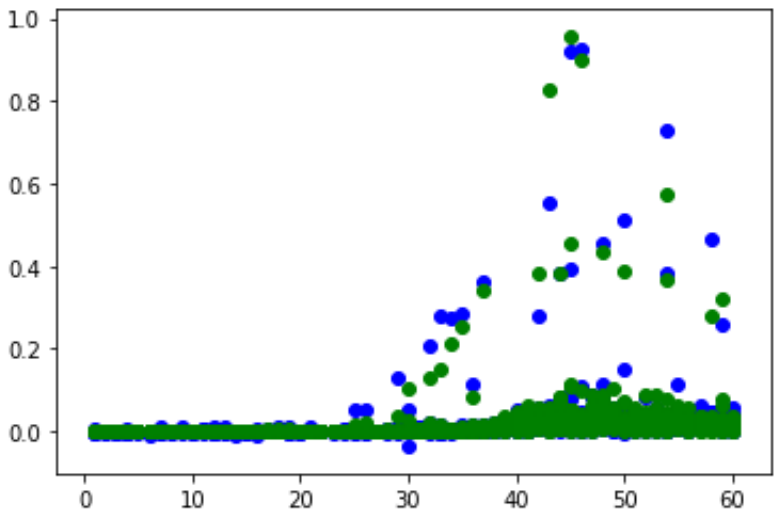


Figura 4-24. Representación correspondiente al resultado de las predicciones para los datos de 'test' con Regresión Polinómica de grado 3 (color rojo) y datos reales de positivos en España en el periodo estudiado (color verde), con las variables normalizadas.

Tabla 4-17. Errores en Regresión Polinómica de grado 3 normalizada y no normalizada.

	MSE	RMSE	MAE	R^2
Train	0.0001	0.0088	0.0040	0.9830
Test	0.0003	0.0169	0.0054	0.9370

Como podemos observar en la Tabla 4-17, todos los tipos de error disminuyen cuantiosamente respecto al apartado anterior tras aplicar la normalización de variables. La precisión calculada con el coeficiente de determinación también es superior a la anterior, por lo que concluimos que esta normalización de variables es muy beneficiosa.

4.3.3. Escenario 2: Regresión polinómica de grado 3 para predicciones semanales con transformación de variables

Al igual que en apartados anteriores, se han hecho predicciones semanales con las variables normalizadas para la Regresión Polinómica de grado 3. Los resultados se muestran a continuación:

En primer lugar, obtenemos un vector de coeficientes ‘w_norm_7’ de 120 elementos. Por ser este número muy elevado, en la Tabla 4-18, se ilustran algunos de los más valores más significativos para su estudio, que a su vez están representados en la Figura 4-25. En este caso la variable x_1 corresponde con ‘num_casos_7’, en vez de ‘num_casos_ant’, que no se utiliza en este escenario.

Tabla 4-18. Algunos coeficientes de la Regresión Polinómica normalizada de grado 3 para predicciones semanales.

Variables	Coeficientes de 'w_norm_7'
x_1x_3	-1.04E+04
x_2x_3	7.52E+04
x_2x_4	3.04E+04
x_7x_3	-2.39E+04
$x_1x_2x_4$	2.33E+05
$x_4x_2^2$	-1.68E+06
$x_2x_3x_4$	5.35E+04
$x_2x_4^2$	-4.45E+04
$x_2x_4x_5$	1.27E+04
$x_2x_4x_6$	-3.08E+04
$x_2x_4x_7$	5.35E+05

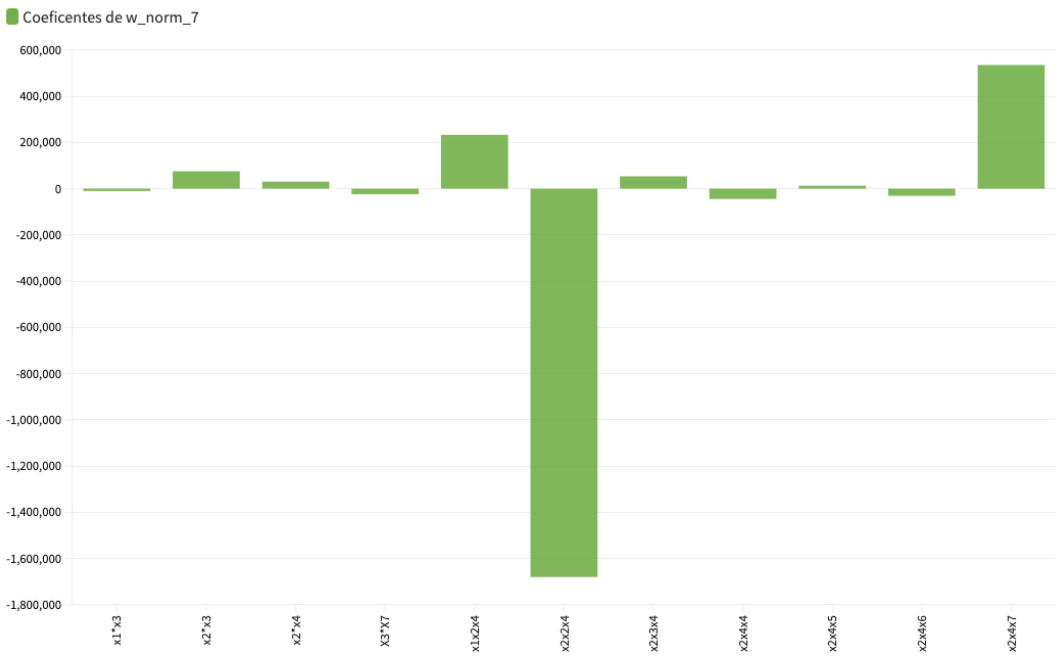


Figura 4-25. Coeficientes de 'w_norm_7' en regresión polinómica de tercer grado normalizada para predicciones normalizadas.

Podemos observar que la variable más determinante es $x_2^2x_4$, con relación negativa, que como vemos en la Ecuación 4-8, está relacionada de cierta forma con la densidad.

$$x_2^2x_4 = Densidad^2 * Extensión = \frac{Población^2}{Extensión^2} * Extensión = \frac{Población^2}{Extensión} \quad (4\text{-}8)$$

Esto se contrarresta con las variables $x_2x_4x_7$, $x_1x_2x_4$, y x_2x_3, que como podemos ver en las Ecuaciones 4-9, 4-10 y 4-11, también tienen coeficientes significativos y están relacionadas de forma directa con algunas variables como 'Aena', 'num_casos_7', y especialmente 'Población', que aparece en todos los términos.

$$x_2x_4x_7 = Densidad * Extensión * Aena = Población * Aena \quad (4\text{-}9)$$

$$x_1x_2x_4 = num\_casos\_7 * Densidad * Extensión = num\_casos\_7 * Población \quad (4\text{-}10)$$

$$x_2x_3 = Densidad * Población = \frac{Población^2}{Extensión} \quad (4\text{-}11)$$

En conclusión, podemos destacar que este modelo es muy dependiente de la variable 'Población'.

A continuación, en la Figuras 4-26 y 4-27, se representa la comparación entre los datos predichos y reales en la regresión polinómica de tercer grado para predicciones semanales.

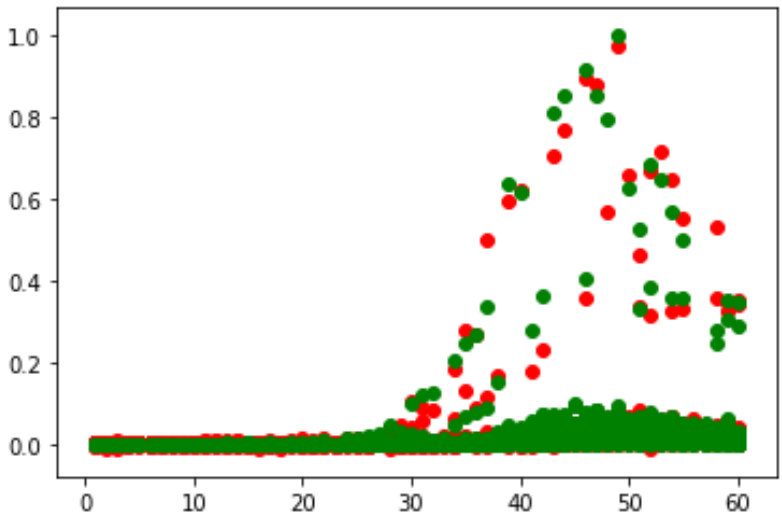


Figura 4-26. Representación correspondiente al resultado de las predicciones semanales para los datos de 'train' con Regresión Polinómica de grado 3 (color rojo) y datos reales de positivos en España en el periodo estudiado (color verde), con las variables normalizadas.

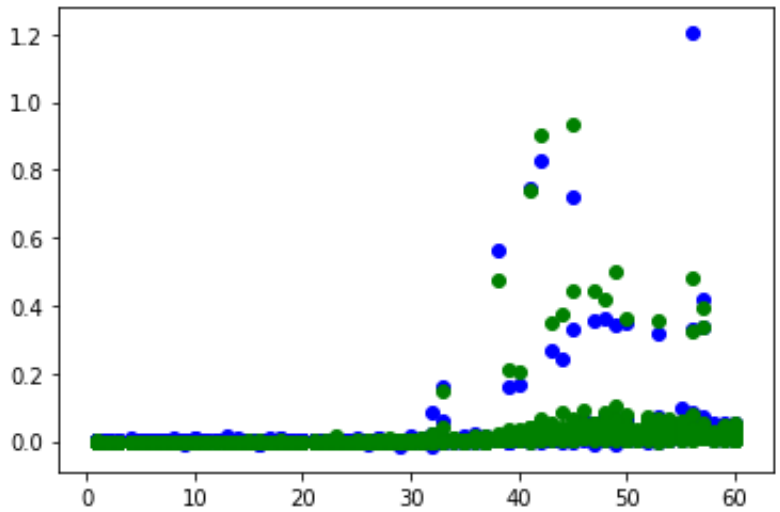


Figura 4-27. Representación correspondiente al resultado de las predicciones semanales para los datos de 'test' con Regresión Polinómica de grado 3 (color rojo) y datos reales de positivos en España en el periodo estudiado (color azul), con las variables normalizadas.

En la Tabla 4-19 podemos observar los distintos errores cometidos en las predicciones semanales. Comparando estos resultados con los del apartado anterior vemos que el todos los tipos de error han empeorado ligeramente, por lo que nuestras predicciones eran mejores al usar variables normalizadas y el dato del día anterior. Al contrario, el coeficiente de determinación R^2 ha mejorado, también levemente. En conclusión estos resultados son muy similares a los obtenidos aplicando variables normalizadas con el dato del día anterior, que por lo tanto como ya se ha explicado en el apartado anterior, son mejores que los obtenidos al usar variables no normalizadas.

Tabla 4-19. Errores cometidos en la regresión polinómica de grado 3 para predicciones semanales normalizadas.

	MSE	RMSE	MAE	R^2
Train	0.0002	0.0137	0.0064	0.96
Test	0.0008	0.0280	0.0076	0.84

4.4. Comparativa

Con todos los coeficientes calculados, podemos sacar algunas conclusiones sobre cuáles resultan más determinantes en conjunto. En primer lugar, observamos que los coeficientes resultan de un valor absoluto muy pequeño antes de normalizar los datos, tanto en regresión lineal como polinómica. De la regresión lineal, podemos destacar que depende fuertemente del número de casos registrados anteriormente, y se relaciona de forma inversa con la temperatura, lo que sigue nuestras hipótesis iniciales. La regresión

polinómica de segundo grado resulta muy dependiente de la población de forma directa, y en menor medida del número de casos anterior, en la versión sin normalizar. No obstante, estos también van en la línea de nuestra hipótesis. Por último, los resultados de la regresión de tercer grado, son más variables. Existe una relación inversa fuerte con la densidad y la población, pero también vemos estos elementos relacionados de forma directa en otros términos. También existe una relación directa más fuerte que en cálculos anteriores con las temperaturas, lo cual tampoco sigue nuestras hipótesis. De la variable 'Aena', podemos decir que en ninguno de los casos ha sido determinante, y que se ha hallado relación tanto directa como inversa con la salida de datos, pero como ya se ha mencionado, de forma poco relevante.

En la Tabla 4-20 se encuentran todos los distintos tipos de error calculados para cada uno de los escenarios, así como el coeficiente de determinación R^2 en cada uno de ellos.

Tabla 4-20. Conjunto de errores MSE, RMSE, MAE y coeficientes de determinación calculados.

			MSE	RMSE	MAE	R^2
Regresión lineal múltiple	No normalizado	Train	1960.00	44.200	13.600	1.0000
		Test	878.00	29.600	12.800	1.0000
	Normalizado	Train	0.0296	0.1720	0.0584	1.0000
		Test	0.0539	0.2320	0.0663	1.0000
	Normalizado semanal	Train	0.0012	0.0344	0.01	1.0000
		Test	0.0012	0.0340	0.01	1.0000
Regresión polinómica segundo grado	No normalizado	Train	1250.00	35.40	11.70	0.9740
		Test	1490.00	38.50	0.96	0.9670
	Normalizado	Train	0.0001	0.0119	0.0041	0.9720
		Test	0.0001	0.0112	0.0040	0.9650
	Normalizado semanal	Train	0.0005	0.0222	0.0094	0.9010
		Test	0.0003	0.0182	0.0079	0.8980
Regresión polinómica tercer grado	No normalizado	Train	846.00	29.10	11.50	0.9770
		Test	5620.00	74.90	16.60	0.9070
	Normalizado	Train	0.0001	0.0088	0.0040	0.9830
		Test	0.0003	0.0169	0.0054	0.9370
	Normalizado semanal	Train	0.0002	0.0137	0.0064	0.9570
		Test	0.0008	0.0280	0.0076	0.8380

Ya que los valores de los distintos errores tienen ordenes de magnitud muy distintos se han dividido los gráficos representativos en distintos grupos para facilitar su comprensión. En primer lugar, en las dos figuras que se exponen a continuación, la Figura 4-28 y la Figura 4-29 se comparan todos los errores pero solo en datos no normalizados. En primer lugar, en la Figura 4-28 se compara únicamente los errores MSE en este grupo, ya que estos son tan altos que es difícil compararlos gráficamente con los restantes. Podemos ver en esta gráfica, que el menor MSE en datos no normalizados se da en la Regresión polinómica de tercer grado con datos 'train'. Para los datos 'test', que representan más fielmente el rendimiento del modelo, el

mínimo error se da en la Regresión Lineal Múltiple. Por ello, se concluye, que según el MSE el mejor modelo para realizar predicciones diarias y utilizando datos no normalizados es la Regresión Lineal Múltiple. Por otro lado, para la regresión polinómica de segundo y tercer grado con datos no normalizados, ilustrado en la Figura 4-29, observamos que la regresión polinómica de tercer grado ofrece los menores errores para los datos tipo 'train', pero el mayor error para los datos tipo 'test', por lo tanto, es más conveniente usar la regresión lineal múltiple, ya que presenta los menores errores en el conjunto 'test', que es el más relevante para ver la precisión del modelo.

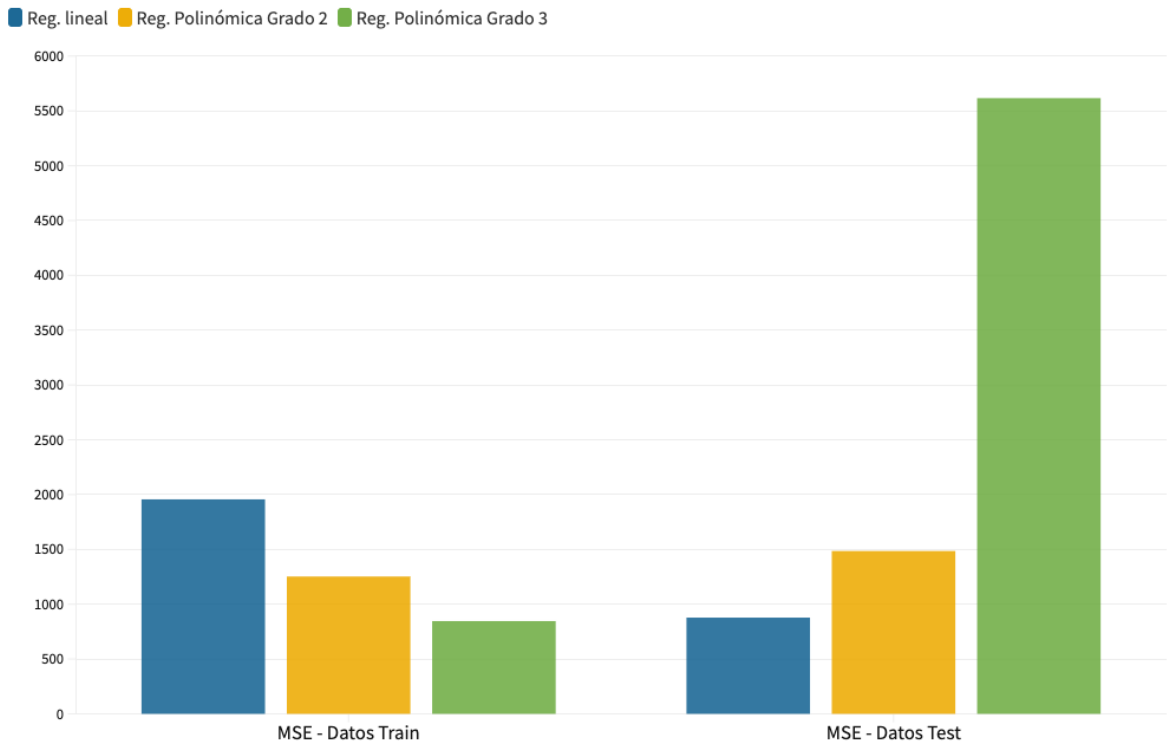


Figura 4-28. Comparación de los valores MSE para datos no normalizados.

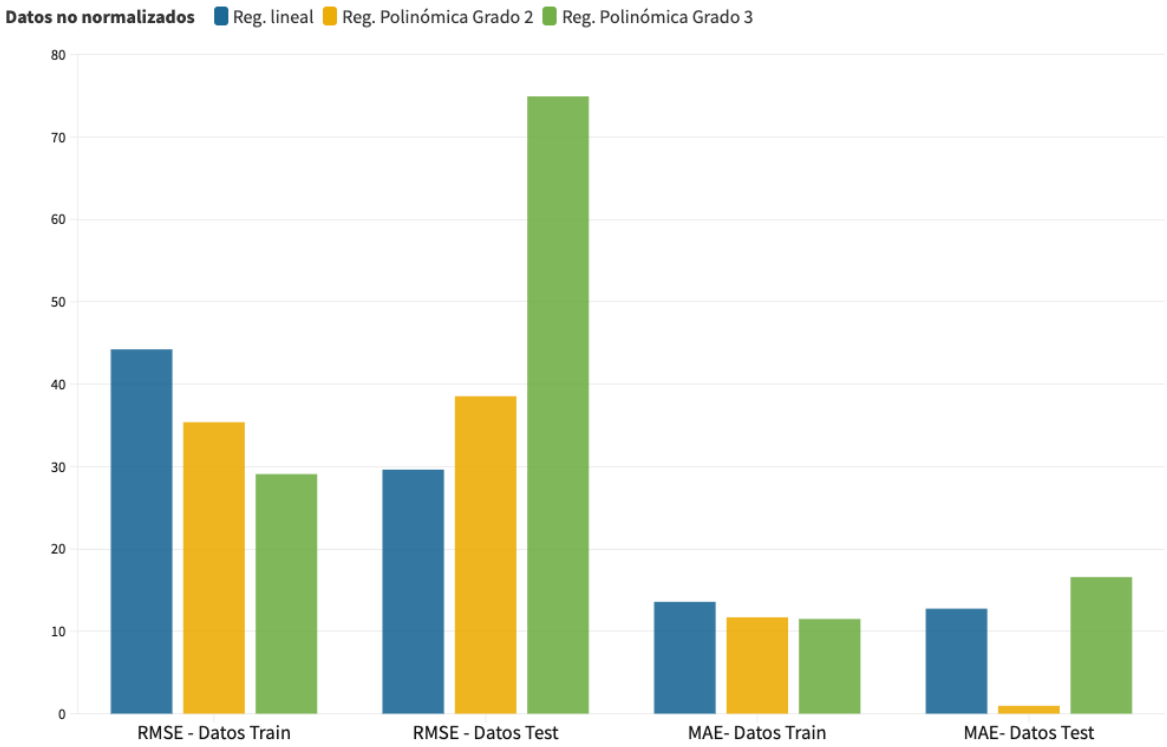


Figura 4-29. Comparación de los valores RMSE y MAE para datos no normalizados.

A continuación, se representan los errores para datos normalizados. En primer lugar, en la Figura 4-30 se representan los errores en datos con predicciones diarias, que son muy inferiores a los calculados en los casos no normalizados. Comparando los errores del conjunto de cálculos, podemos ver que los mayores errores, tanto RMSE como MAE, se dan en la regresión lineal múltiple. El menor error para los datos 'train' se da en la Regresión Polinómica de tercer grado normalizada, y el menor error de datos 'test' se da en la Regresión Polinómica de segundo grado normalizada, por lo que en este análisis este último sistema parece el modelo más recomendable para este trabajo. No obstante, todos estos errores son aceptables para confirmar la eficacia del modelo.

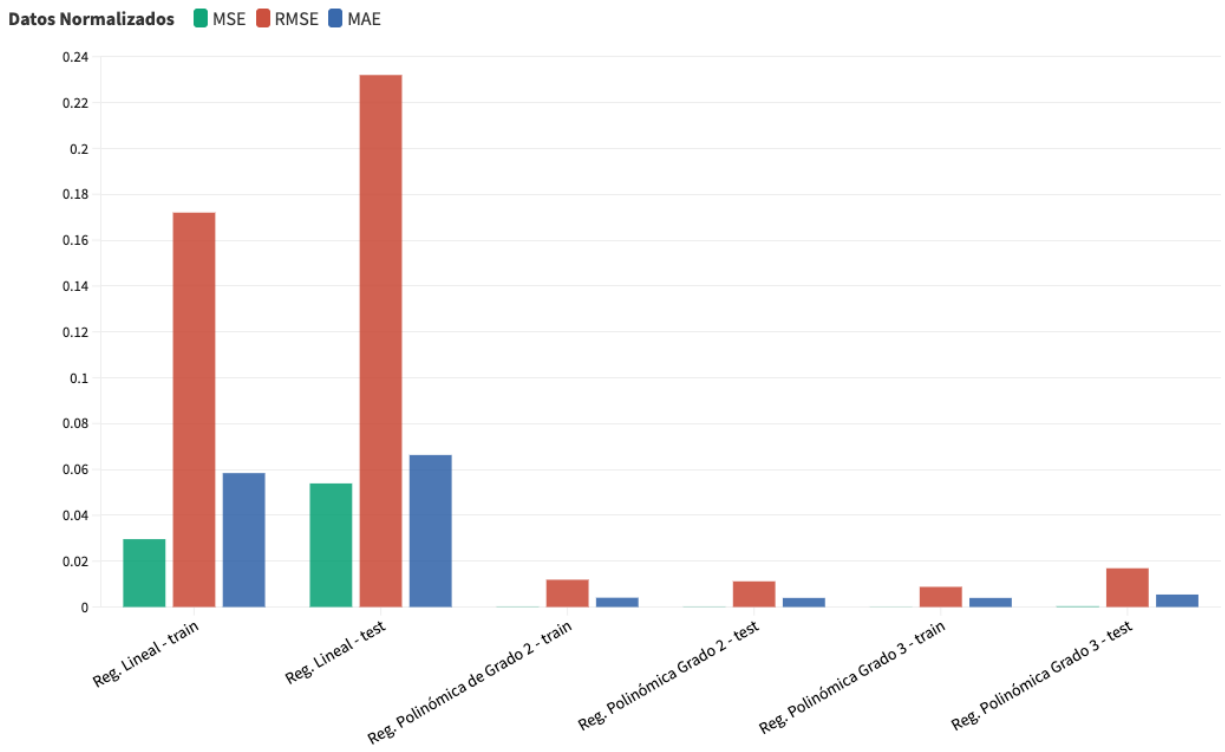


Figura 4-30. Comparación de errores MSE, RMSE y MAE en Regresión Lineal Múltiple normalizada y Regresión Polinómica normalizada para predicciones diarias.

Por otro lado, los errores normalizados en las predicciones semanales se representan en la Figura 4-31. Podemos ver que respecto a la figura anterior, ha disminuido el error en la regresión lineal múltiple, no obstante ha aumentado ligeramente en las regresiones polinómicas. En este caso, de nuevo el modelo que menor error presenta es la regresión polinómica de segundo grado. Aunque el error haya aumentando respecto al calculado en las predicciones diarias, sigue siendo muy aceptable. Esto es muy beneficioso para empresas u organizaciones que puedan ayudarse de este modelo para conocer el número de contagios en fechas cercanas, ya que las predicciones semanales pueden resultarles mucho más útiles que las diarias.

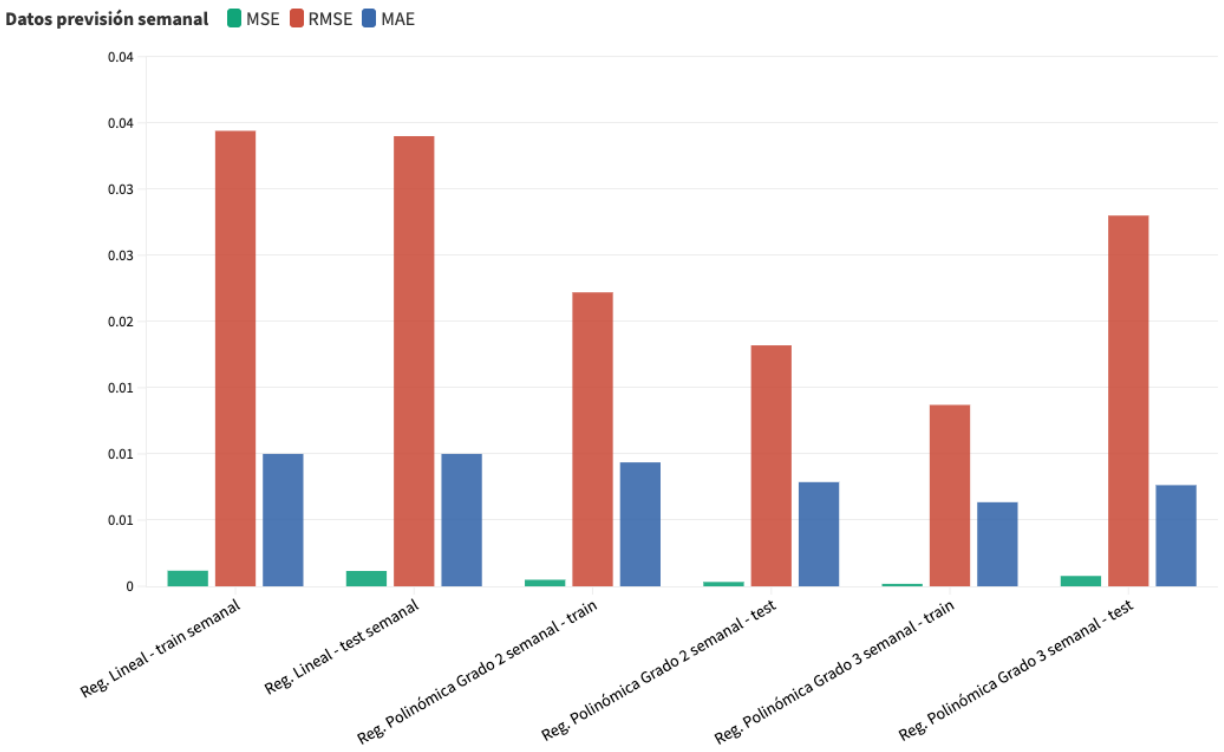


Figura 4-31. Comparación de errores MSE, RMSE y MAE en Regresión Lineal Múltiple normalizada y Regresión Polinómica normalizada para predicciones semanales.

Respecto al coeficiente de determinación, este se ha representado para todos los casos en la Figura 4-32. El coeficiente de determinación representa el ajuste del modelo estimado con los datos de entrenamiento a los datos de test, es decir, el porcentaje de la variable real de salida y del conjunto de test que se explica con las variables independientes. El modelo que mejor se ajusta, y por lo tanto mayor coeficiente R^2 tiene es el modelo regresión lineal múltiple, que tiene un valor de R^2 igual a la unidad, tanto en su versión normalizada como no normalizada. Por lo tanto podríamos decir que el modelo que mejor se ajusta es la regresión lineal múltiple, aunque por otro lado tiene los mayores errores cuadrático y medio. El modelo que peor se ajusta es la Regresión Polinómica de tercer grado en predicciones semanales.

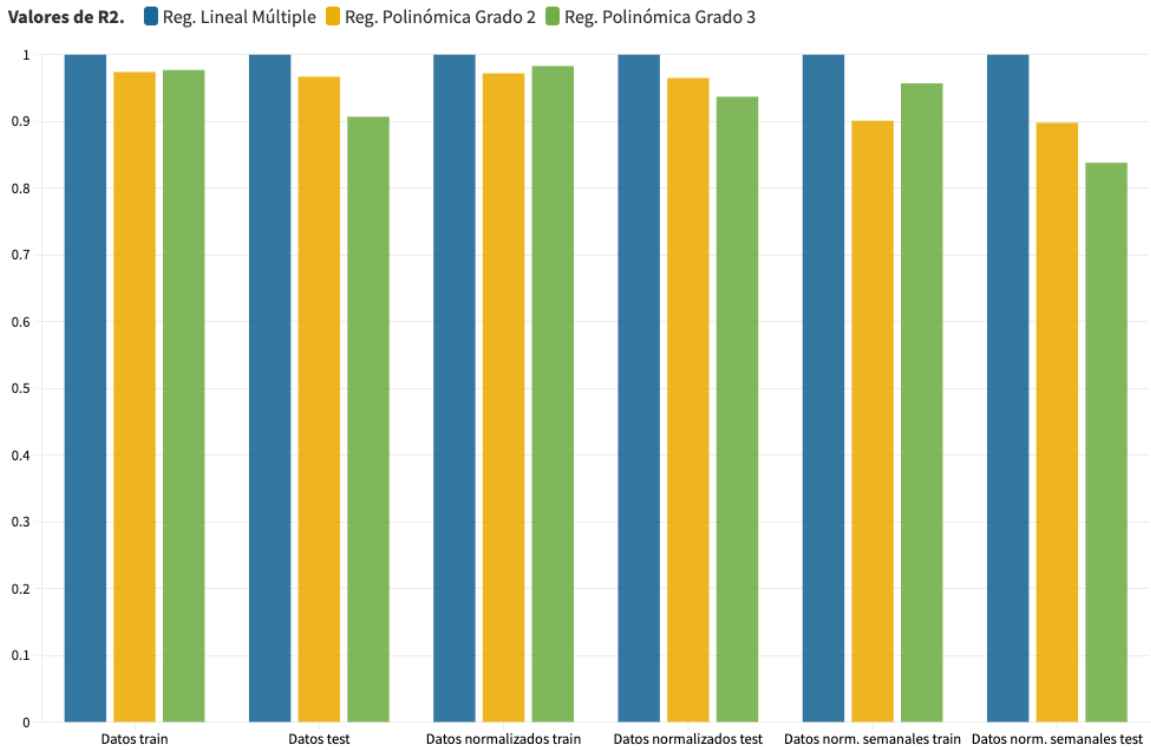


Figura 4-32. Gráfico comparativo de los distintos valores de R^2.

4.5. Análisis económico

Como ya se comentó en la introducción de este trabajo la crisis sanitaria del Covid-19 ha tenido un gran impacto en la economía tanto mundial como española. Un gran número de empresas han tenido que cerrar sus puertas al haber disminuido su nivel de negocio, a causa de la parada de actividad comercial por el confinamiento al principio de la pandemia, las restricciones a la movilidad, la interrupción de la cadena de suministro o el impacto de la inestabilidad en los índices bursátiles.

Uno de los objetivos de este trabajo es ayudar a algunas de las empresas que están viviendo está situación de inestabilidad con un sistema predictivo con el que puedan estimar la situación epidemiológica en su zona en un corto plazo. Algunos de los sectores que se han visto más afectados por esta situación han sido el sector turístico y el sector de la hostelería. En concreto solo en España se produjeron en 2020 unas pérdidas de 70000 millones de euros y se cerraron de forma definitiva 85000 bares y restaurantes [30]. Las empresas se han adaptado a esta nueva situación, un ejemplo de ello son los restaurantes, que han respondido a la disminución de flujo adaptando su modelo operativo y centrándose en el cliente, creando o reforzando su servicio de comida a domicilio, ya que a ciertas horas no pueden prestar servicio en sus instalaciones por las restricciones horarias que se han tomado en determinados momentos como plan de contingencia contra el Covid-19. Esta nueva situación, supone que la demanda de estos establecimientos y de mucho otros, aumenta y disminuye repentinamente, también los plazos entrega y la fiabilidad de estas, lo que genera presión en la cadena de suministro de las compañías [31]. Con los cálculos realizados en este trabajo,

añadiendo nuevas variables si fuera necesario, empresas como las dedicadas a la restauración o al sector hotelero pueden hacer una estimación de cuál va a ser el número de positivos en su provincia en las próximas semanas, y por lo tanto también estimar qué restricciones va a aprobar su gobierno autonómico respecto a los horarios de apertura y cierre o número de comensales por mesa. Con estos datos podrán calcular con mayor eficacia la cantidad de materia prima a comprar, la cantidad de stock óptima o las futuras necesidades de personal y obtener unos mejores resultados económicos.

CONCLUSIONES Y FUTURAS LÍNEAS DE INVESTIGACIÓN

En este trabajo se ha intentado encontrar una relación matemática entre el número de contagiados por Covid-19 en España durante un periodo concreto de la primera ola, y algunas variables, como la densidad de población, extensión, población en términos absolutos, temperatura máxima y mínima registrada en la provincia, y el número de pasajeros que han llegado a la provincia por vía aérea. El objeto de este trabajo es demostrar que existe una relación directa entre el número de contagiados, la densidad de población de cada provincia y el número de pasajeros llegados en avión a cada provincia o ciudad autónoma, y a su vez, una relación inversa con la temperatura, ya que existen estudios que demuestran que este tipo de virus se transmiten más fácilmente en ambientes más fríos. Para demostrar estas relaciones, se han usado las técnicas Machine Learning implementadas mediante el lenguaje Python. Las técnicas usadas han sido Regresión Lineal Múltiple y Regresión Polinómica de segundo y tercer grado. Dentro de estos cálculos, se han implementado distintos escenarios, normalizando variables y haciendo predicciones tanto para un día de posterioridad, como para siete días. Una vez ejecutado el modelo y descritos los resultados, llegamos a las siguientes conclusiones:

- Los distintos modelos matemáticos se ajustan bien a la predicción, por lo que existen relaciones reales entre las variables estudiadas y el número de contagios.
- Las temperaturas registradas tienen en la mayoría de casos una relación inversa con el número de contagios. Por lo tanto se verifica que es más probable que se dé un mayor número de contagios en una provincia con un clima más frío.
- El número de contagios está en la mayoría de cálculos determinado por la población o densidad de provincia, por lo que estas variables también son determinantes para calcular el número de infectados.
- Algunas de las variables estudiadas, como el número de pasajeros que llega por vía aérea a la provincia, o la extensión de cada una de las provincias, no son determinantes para el cálculo del número de futuros contagiados.
- Es mejor para disminuir el error normalizar las variables ya que estas tienen órdenes de magnitud muy diversos.
- Es muy beneficioso para la predicción insertar como dato de entrada el número de contagiados en una fecha cercana. En este trabajo se ha usado tanto el dato de contagios del día anterior, como el

número de contagios con una semana de anterioridad, y aunque se consiguen mejores resultados con los datos del día anterior, no existe una diferencia significativa, por lo que se puede usar el dato semanal para hacer predicciones más extensas en el tiempo.

- Aunque la regresión lineal obtiene un mejor coeficiente de determinación que la regresión polinómica, se dan errores menores absolutos y cuadráticos en la regresión polinómica. El mínimo error se da en la regresión polinómica de segundo grado con variables normalizadas y usando el dato de contagiados del día anterior como variable de entrada. Por lo tanto se concluye que el modelo que da resultados óptimos es la regresión polinómica de segundo grado.
- En Febrero de 2021, 207000 empresas habían cerrado en España desde el comienzo de la crisis del Covid-19 [32]. Para evitar que se siga produciendo esta situación, las empresas que consideren pueden ayudarse de este modelo, especialmente de las predicciones semanales, para estimar el número de contagios en su provincia en fechas cercanas, utilizando así estos cálculos para planificar su stock, compras y ventas de forma efectiva, lo que puede influir de forma directa en sus cuentas.

Una vez concluido el trabajo, se proponen algunas posibles futuras líneas de investigación. Existen muchos factores relacionados con la transmisión del Covid-19, y estos podrían ser cuantificados y introducidos como nuevas variables. Algunos de ellos son la intensidad de utilización del transporte público en cada ciudad o municipio, el tamaño medio de hogar, el número de pasajeros llegados al país desde países con altos niveles de incidencia o nuevas cepas del virus, y el porcentaje de personal sanitario infectado (en cálculos para la primera y segunda ola, previos a la introducción de vacunas). Por otra parte, se podría hacer un estudio que introdujese como nueva variable de salida el número de muertes. Aunque este dato es proporcional al número de contagiados, se podrían añadir nuevas variables de entrada, como el número de camas de hospital ocupadas por la enfermedad, número previo de camas disponibles en cada provincia, la edad de la población y el estado de salud de ésta (número de personas con patologías previas, ya sean cardiovasculares o respiratorias). Otro posible cambio en los cálculos es el uso de otros modelos de predicción, que quizás puedan dar mejores a la estimación del número de enfermos o muertes. Por ejemplo, el modelo SEIR, es comúnmente usado en el análisis de epidemias, aunque también podrían ser útiles las redes neuronales y otros modelos como el Random Forest. También concluimos en el análisis económico que en el caso de utilizar este modelo muchas empresas podrían mejorar sus resultados económicos, ajustándose mejor a las necesidades del mercado limitado por las restricciones o confinamientos.

REFERENCIAS

[1] MITECO, “Primeros indicios de correlación entre variables meteorológicas y propagación de la enfermedad COVID-19 y del virus SARS-CoV-2 en España.” pp. 1–20, 2020, Accessed: Nov. 19, 2020. [Online]. Available: moz-extension://5ef145fa-d2be-2941-9233-4f3e383d7f8b/enhanced-reader.html?openApp&pdf=https%253A%252F%252Fwww.mscbs.gob.es%252Fprofesionales%252FsaludPublica%252Fccayes%252FalertasActual%252FnCov%252Fdocumentos%252FCOVID19_Plan_de_respuesta_temprana_escen.

[2] Organización Mundial de la Salud (OMS), “Preguntas y respuestas sobre la enfermedad por coronavirus (COVID- 19),” *Organizacion mundial de la salud*, no. October 2018. pp. 1–3, 2020, Accessed: Nov. 19, 2020. [Online]. Available: https://www.who.int/es/emergencies/diseases/novel-coronavirus-2019/advice-for-public/q-a-coronaviruses.

[3] OMS, “Información basíca sobre la COVID-19,” *OMS*, 2020. https://www.who.int/es/news-room/q-a-detail/coronavirus-disease-covid-19 (accessed Nov. 19, 2020).

[4] J. Wang and G. Du, “COVID-19 may transmit through aerosol,” *Ir. J. Med. Sci.*, vol. 189, no. 4, pp. 1143–1144, Nov. 2020, doi: 10.1007/s11845-020-02218-2.

[5] “Las fechas de la desescalada: las fases 0, 1, 2 y 3 y el calendario completo en España | Las Provincias.” https://www.lasprovincias.es/sociedad/fechas-desescalada-fases-espana-20200429111640-nt.html (accessed Nov. 19, 2020).

[6] “La Moncloa. 21/06/2020. Estado de situación del COVID-19 [Prensa/Actualidad/Sanidad],” *2020*. https://www.lamoncloa.gob.es/serviciosdeprensa/notasprensa/sanidad14/Paginas/2020/210620datos-covid19.aspx (accessed Nov. 19, 2020).

[7] M. Coiras, J. Alcamí, J. A. Plaza-Ramos, and G. de A. C. de C. del I. (GACC-ISCIII), “INFORME DEL GRUPO DE ANALISIS CIENTÍFICO DE CORONAVIRUS DEL ISCIII (GACC-ISCIII) NECESIDAD DE ENCONTRAR FÁRMACOS FRENTE A LA ENFERMEDAD COVID-19 16 de abril de 2020,” p. 1, 2020, Accessed: Nov. 19, 2020. [Online]. Available: https://doi.org/10.1101/2020.03.22.002386.

[8] H. Swapnarekha, H. S. Behera, J. Nayak, and B. Naik, “Role of intelligent computing in COVID-19 prognosis: A state-of-the-art review,” *Chaos, Solitons and Fractals*, vol. 138, p. 109947, Sep. 2020, doi: 10.1016/j.chaos.2020.109947.

[9] Deloitte, “El impacto económico del COVID-19 | Deloitte España,” *Deloitte*, 2020, Accessed: Feb. 14, 2021. [Online]. Available: https://www2.deloitte.com/es/es/pages/about-deloitte/articles/impacto-economico-del-covid19.html.

[10] “El PIB se hunde un 18,5% en el segundo trimestre y la economía entra en recesión técnica.” https://www.abc.es/economia/abci-hunde-185-por-ciento-segundo-trimestre-y-221-por-ciento-

primeros-seis-meses-202007310905_noticia.html (accessed Feb. 12, 2021).

[11] ElEconomista.es, “Las cuatro razones por las que España es el mayor perdedor en la crisis del coronavirus,” 2020.

[12] “La Moncloa. 04/11/2020. El paro registrado sube en 49.558 personas en el mes de octubre, la segunda menor subida en este mes en los últimos 13 años [Prensa/Actualidad/Trabajo y Economía Social].” https://www.lamoncloa.gob.es/serviciosdeprensa/notasprensa/trabajo14/Paginas/2020/041120-paro.aspx (accessed Feb. 12, 2021).

[13] SEPE, “Resumen datos estadísticos | Servicio Público de Empleo Estatal,” 2020. https://www.sepe.es/HomeSepe/que-es-el-sepe/estadisticas/datos-avance/paro.html (accessed Feb. 12, 2021).

[14] P. Estoc, “Capítulo 10 Cadenas de Markov,” pp. 103–127, 2009.

[15] “Los métodos de Montecarlo,” pp. 1–6, 2011, Accessed: Feb. 14, 2021. [Online]. Available: http://www.sc.ehu.es/sbweb/fisica_/numerico/montecarlo/montecarlo.html.

[16] M. Swamynathan, *Mastering Machine Learning with Python in Six Steps*. 2017.

[17] “Qué son regresión y clasificación en Machine Learning,” 2020. https://agenciab12.com/noticia/que-son-regresion-clasificacion-machine-learning (accessed Feb. 14, 2021).

[18] “Repaso didáctico sobre machine learning | La Pastilla Roja.” https://lapastillaroja.net/2015/02/ml-algols/ (accessed Feb. 19, 2021).

[19] J. Ignacio, “Qué es overfitting y underfitting y cómo solucionarlo | Aprende Machine Learning,” *Aprende Machine Learning*, 2017.

[20] S. Fernandéz, “Series Temporales Introducción,” 2017. Accessed: Mar. 11, 2021. [Online]. Available: http://halweb.uc3m.es/esp/Personal/personas/jmmarin/esp/EDescrip/tema7.pdf.

[21] “Machine Learning Supervisado: Fundamentos de la Regresión Lineal | by Victor Roman | Ciencia y Datos | Medium,” 2019. https://medium.com/datos-y-ciencia/machine-learning-supervisado-fundamentos-de-la-regresión-lineal-bbcb07fe7fd (accessed Nov. 19, 2020).

[22] Instituto de Salud Carlos III. Ministerio de Ciencia e Innovación. Gobierno de España., “Pruebas de diagnóstico del coronavirus: ¿Qué es la PCR?, ¿qué son los test rápidos? ¿en qué se diferencian?,” 2020, Accessed: Nov. 19, 2020. [Online]. Available: https://www.isciii.es/InformacionCiudadanos/DivulgacionCulturaCientifica/DivulgacionISCIII/Paginas/Divulgacion/COVID19_PCR_test.aspx.

[23] “Covid: ¿cuándo se debe usar PCR y cuándo test de antígenos?” https://www.redaccionmedica.com/secciones/sanidad-hoy/coronavirus-cuando-usar-pcr-test-de-antigenos-cribado-8882 (accessed Feb. 14, 2021).

[24] Centro Nacional de Epidemiología, “COVID-19. Documentación y datos.,” *2021*. https://cnecovid.isciii.es/covid19/#documentación-y-datos (accessed Nov. 19, 2020).

[25] “Evolución de enfermedad por el coronavirus (COVID-19) - Conjunto de datos | datos.gob.es.” https://datos.gob.es/es/catalogo/e05070101-evolucion-de-enfermedad-por-el-coronavirus-covid-19

(accessed Feb. 18, 2021).

[26] "Aena.es." http://www.aena.es/csee/Satellite?SiteName=Estadisticas&anyo=2020&c=Page&cid=1144247795704&pagename=Estadisticas%2FEstadisticas&periodoInforme=Mensual (accessed Mar. 10, 2021).

[27] Instituto Nacional de Estadística (INE), "Población residente por fecha, sexo y edad(31304)," 2021. https://www.ine.es/jaxiT3/Tabla.htm?t=31304 (accessed Feb. 18, 2021).

[28] Instituto Nacional de Estadística, "Extensión superficial de las Comunidades Autónomas y Provincias, por zonas altimétricas," 1994. Accessed: Mar. 10, 2021. [Online]. Available: https://www.ine.es/inebaseweb/pdfDispacher.do?td=154090&L=0.

[29] Instituto Nacional de Estadística, "Población por comunidades y ciudades autónomas y tamaño de los municipios.(2915)," *Población por comunidades y ciudades autónomas y tamaño de los municipios.*, 2019. https://www.ine.es/jaxiT3/Datos.htm?t=2915#!tabs-tabla (accessed Feb. 18, 2021).

[30] "Unos 85.000 bares y restaurantes ya han cerrado definitivamente en España." https://www.cronicabalear.es/2021/la-hosteleria-cerro-2020-con-perdidas-de-70000-millones-y-el-cierre-definitivo-de-85000-establecimientos/ (accessed Apr. 29, 2021).

[31] G. Fornos, "COVID-19: Un desafío para las cadenas de suministro," *KPMG TENDENCIAS*, 2020. https://www.tendencias.kpmg.es/2020/03/covid-19-cadenas-de-suministro/ (accessed Apr. 24, 2021).

[32] "El Covid se lleva por delante a 207.000 empresas y 323.000 autónomos en apenas medio año | Economía | Cinco Días." https://cincodias.elpais.com/cincodias/2021/02/03/economia/1612367119_734627.html (accessed Apr. 13, 2021).

ANEXO: CÓDIGO

Los códigos utilizados en este trabajo se exponen a continuación:

Regresión Lineal Múltiple

Escenario 0: Regresión lineal múltiple con variables no normalizadas

```
# Importación de las librerías necesarias
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
import matplotlib.pyplot as plt
from sklearn.metrics import mean_absolute_error

# Lectura de datos
datos = pd.read_csv('datos_provincias.csv',header=0,sep=';',encoding='latin-
1')
x = datos[['num_casos_ant',
'Densidad','Poblacion','Extension','TMAX_10','TMIN_10', 'AENA']]
y= datos[['num_casos']]
f = datos[['n']]

# Se utiliza este comando para pasar de tipo de dato DataFrame a float
x= x.to_numpy()
y = y.to_numpy()
f = f.to_numpy()

# Se dividen los datos en datos de entrenamiento y datos de prueba
xtrain, xtest, ytrain, ytest,ftrain,ftest = train_test_split(x, y, f,
test_size=0.3)#separo el 30%
# Selección del modelo: regresión lineal
regresion_lineal = LinearRegression() # Creamos una instancia de
LinearRegression

# Se instruye a la regresión lineal para que aprenda de los datos
```

```
regresion_lineal.fit(xtrain, ytrain)
pred_train = regresion_lineal.predict(xtrain)
pred_test = regresion_lineal.predict(xtest)

# Se obtienen los parámetros que ha estimado la regresión lineal
b = regresion_lineal.intercept_
w = regresion_lineal.coef_ #Los coeficientes
#Se muestra por pantalla
print('w = ' + str(regresion_lineal.coef_) + ', b = ' +
str(regresion_lineal.intercept_))

# Representación de las gráficas de comparación
plt.scatter(ftrain,ytrain,c="green")
plt.scatter(ftrain,pred_train,c="red", label='train')
plt.show()

plt.scatter(ftest,ytest,c="green")
plt.scatter(ftest,pred_test,c="blue")
plt.show()

# Cálculo del Error Cuadrático Medio de los datos train (MSE = Mean Squared
Error)
mse = mean_squared_error(ytrain, pred_train)

# La raíz cuadrada del MSE es el RMSE
rmse = np.sqrt(mse)
print('Error Cuadrático Medio de datos de entrenamiento(MSE) = ' + str(mse))
print('Raíz del Error Cuadrático Medio datos train (RMSE) = ' + str(rmse))
mae=mean_absolute_error(ytrain, pred_train)
print('Error absoluto del train (MAE)= ' + str(mae))
print()

# Cálculo del Error Cuadrático Medio (MSE = Mean Squared Error)
mse = mean_squared_error(ytest, pred_test)#no se como compararlo con y pq son
dimensiones distintas
rmse = np.sqrt(mse)
print('Error Cuadrático Medio de datos test(MSE) = ' + str(mse))
print('Raíz del Error Cuadrático Medio (RMSE) = ' + str(rmse))
```

```
mae=mean_absolute_error(ytest, pred_test)
print('Error absoluto del test (MAE)= ' + str(mae))

print()

# Cálculo del coeficiente de determinación R2 para los datos train
r2 = regresion_lineal.score(xtrain, pred_train)
print('Coeficiente de Determinación R2 de train= ' + str(r2)) # es del ajuste

# Cálculo del coeficiente de determinación R2 para los datos test
r2 = regresion_lineal.score(xtest, pred_test)
print('Coeficiente de Determinación R2 de test = ' + str(r2)) # es del ajuste
```

Escenario 1: Regresión lineal múltiple con variables normalizadas

```
# Importación de las librerías necesarias
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
import matplotlib.pyplot as plt
from sklearn.metrics import mean_absolute_error

# Lectura de datos
datos = pd.read_csv('datos_provincias.csv',header=0,sep=';',encoding='latin-
1')
x = datos[['num_casos_ant',
'Densidad','Poblacion','Extension','TMAX_10','TMIN_10', 'AENA']]
y= datos[['num_casos']]
f = datos[['n']]

# Normalización de los datos
scaler = MinMaxScaler()
x = scaler.fit_transform(x)
x = pd.DataFrame(x, columns=['num_casos', 'Densidad','Poblacion',
'Extension', 'TMAX_10','TMIN_10', 'AENA'])
y = scaler.fit_transform(y)
```

```
# Se utiliza este comando para pasar de tipo de dato DataFrame a float
x= x.to_numpy()
y = y.to_numpy()
f = f.to_numpy()

# Se dividen los datos en datos de entrenamiento y datos de prueba
xtrain, xtest, ytrain, ytest,ftrain,ftest = train_test_split(x, y, f,
test_size=0.3)#separo el 30%
# Utilizo la regresión lineal
regresion_lineal = LinearRegression() # Creamos una instancia de
LinearRegression

# Se instruye a la regresión lineal para que aprenda de los datos
regresion_lineal.fit(xtrain, ytrain)
pred_train = regresion_lineal.predict(xtrain)
pred_test = regresion_lineal.predict(xtest)

# Se obtienen los parámetros que ha estimado la regresión lineal
b = regresion_lineal.intercept_ # Es el intercept
w = regresion_lineal.coef_ #Los coeficientes
#Se muestra por pantalla
print('w = ' + str(regresion_lineal.coef_) + ', b = ' +
str(regresion_lineal.intercept_))

# Representación de las gráficas de comparación
plt.scatter(ftrain,ytrain,c="green")
plt.scatter(ftrain,pred_train,c="red", label='train')
plt.show()

plt.scatter(ftest,ytest,c="green")
plt.scatter(ftest,pred_test,c="blue")
plt.show()

# Cálculo del Error Cuadrático Medio de los datos train (MSE = Mean Squared
Error)
mse = mean_squared_error(ytrain, pred_train)
```

```
# La raíz cuadrada del MSE es el RMSE
rmse = np.sqrt(mse)
print('Error Cuadrático Medio de datos de entrenamiento(MSE) = ' + str(mse))
print('Raíz del Error Cuadrático Medio datos train (RMSE) = ' + str(rmse))
mae=mean_absolute_error(ytrain, pred_train)
print('Error absoluto del train (MAE)= ' + str(mae))
print()

# Cálculo del Error Cuadrático Medio (MSE = Mean Squared Error)
mse = mean_squared_error(ytest, pred_test)#no se como compararlo con y pq son
dimensiones distintas
rmse = np.sqrt(mse)
print('Error Cuadrático Medio de datos test(MSE) = ' + str(mse))
print('Raíz del Error Cuadrático Medio (RMSE) = ' + str(rmse))
mae=mean_absolute_error(ytest, pred_test)
print('Error absoluto del test (MAE)= ' + str(mae))

print()

# Cálculo del coeficiente de determinación R2 para los datos train
r2 = regresion_lineal.score(xtrain, pred_train)
print('Coeficiente de Determinación R2 de train= ' + str(r2)) # es del ajuste

# Cálculo del coeficiente de determinación R2 para los datos test
r2 = regresion_lineal.score(xtest, pred_test)
print('Coeficiente de Determinación R2 de test = ' + str(r2)) # es del ajuste
```

Escenario 2: Regresión lineal múltiple con variables normalizadas y predicciones semanales

Se ha sustituido la variable ‘num_casos_ant’ del Escenario 1.1 por ‘num_casos_7’.

Regresión polinómica de grado 2

Escenario 0: Regresión polinómica de segundo grado con variables no normalizadas

```
# Librerias
from sklearn.linear_model import LinearRegression # Se utiliza para generar
características polinómicas
from sklearn.model_selection import train_test_split
```

```
from sklearn.preprocessing import PolynomialFeatures
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error

# Lectura de datos
datos = pd.read_csv('datos_provincias.csv',header=0,sep=';',encoding='latin-
1')
x = datos[['num_casos_ant',
'Densidad','Poblacion','Extension','TMAX_10','TMIN_10', 'AENA']]
y= datos[['num_casos']]
f= datos[['n']]

# Se utiliza este comando para pasar de tipo de dato DataFrame a float
x= x.to_numpy()
y = y.to_numpy()

# División de los datos en entrenamiento y test
xtrain, xtest, ytrain, ytest, ftrain,ftest = train_test_split(x, y, f,
test_size=0.3)#separo el 30%
pf = PolynomialFeatures(degree = 2)  # Elección del grado del polinomio
Xtrain = pf.fit_transform(xtrain.reshape(-1,7)) # Transformación de la
entrada en polinómica
Xtest = pf.fit_transform(xtest.reshape(-1,7))
regresion_lineal = LinearRegression() # Se crea una instancia de
LinearRegression

# Se instruye a la regresión lineal para que aprenda de los datos (ahora
polinómicos) (X,y)
regresion_lineal.fit(Xtrain, ytrain)
b = regresion_lineal.intercept_
w = regresion_lineal.coef_

# Impresión por pantalla de los parámetros que se han estimado
print('w = ' + str(regresion_lineal.coef_) + ', b = ' +
str(regresion_lineal.intercept_))
```

```
# Predicción de los valores ‘y’ para los datos usados en el entrenamiento. Impresión de gráfica comparativa.
pred_train = regresion_lineal.predict(Xtrain)
plt.scatter(ftrain,pred_train,c="red")
plt.scatter(ftrain,ytrain,c="green")
plt.show()

# Predicción de los valores ‘y’ para los datos usados en el test. Impresión de gráfica comparativa.
pred_test = regresion_lineal.predict(Xtest)
plt.scatter(ftest,pred_test,c="blue")
plt.scatter(ftest,ytest,c="green")
plt.show()

# Evaluación el modelo de entrenamiento
# Cálculo del Error Cuadrático Medio (MSE = Mean Squared Error) de los datos train
mse = mean_squared_error(ytrain, pred_train)
# La raíz cuadrada del MSE es el RMSE
rmse = np.sqrt(mse)
print('Error Cuadrático Medio del entrenamiento(MSE) = ' + str(mse))
print('Raíz del Error Cuadrático Medio del entrenamiento(RMSE) = ' + str(rmse))
mae=mean_absolute_error(ytrain, pred_train)
print('Error absoluto del train (MAE)= ' + str(mae))

# Calculamos el coeficiente de determinación R2
r2 = regresion_lineal.score(Xtrain, ytrain)
print('Coeficiente de Determinación R2 del entrenamiento = ' + str(r2))
print()

# Evaluación el modelo de entrenamiento
# Cálculo del Error Cuadrático Medio (MSE = Mean Squared Error) de los datos train
mse = mean_squared_error(ytest, pred_test)

# La raíz cuadrada del MSE es el RMSE
```

```
rmse = np.sqrt(mse)
print('Error Cuadrático Medio del test(MSE) = ' + str(mse))
print('Raíz del Error Cuadrático Medio del test(RMSE) = ' + str(rmse))
mae=mean_absolute_error(ytest, pred_test)
print('Error absoluto del test (MAE)= ' + str(mae))

# Cálculo del coeficiente de determinación R2
r2 = regresion_lineal.score(Xtest, ytest)
print('Coeficiente de Determinación R2 del test= ' + str(r2))
print()
```

Escenario 1: Regresión polinómica de segundo grado con variables nomalizadas

```
# Librerías
from sklearn.linear_model import LinearRegression # Para generar
características polinómicas
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import PolynomialFeatures
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error

# Lectura de datos
datos = pd.read_csv('datos_provincias.csv',header=0,sep=';',encoding='latin-
1')
x = datos[['num_casos_ant',
'Densidad','Poblacion','Extension','TMAX_10','TMIN_10', 'AENA']]
y= datos[['num_casos']]
f= datos[['n']]

# Normalización de los datos
scaler = MinMaxScaler()
x = scaler.fit_transform(x)
x = pd.DataFrame(x, columns=['num_casos', 'Densidad','Poblacion',
'Extension', 'TMAX_10','TMIN_10', 'AENA'])
y = scaler.fit_transform(y)
```

```
# Se utiliza este comando para pasar de tipo de dato DataFrame a float
x= x.to_numpy()
y = y.to_numpy()

# División de los datos en entrenamiento y test
xtrain, xtest, ytrain, ytest, ftrain,ftest = train_test_split(x, y, f,
test_size=0.3)#separo el 30%
pf = PolynomialFeatures(degree = 2)    # Elegimos el grado del polinomio
Xtrain = pf.fit_transform(xtrain.reshape(-1,7))  # Transformamos la entrada
en polinómica
Xtest = pf.fit_transform(xtest.reshape(-1,7))
regresion_lineal = LinearRegression() # Creamos una instancia de
LinearRegression

# Se instruye a la regresión lineal para que aprenda de los datos (ahora
polinómicos) (X,y)
regresion_lineal.fit(Xtrain, ytrain)
b = regresion_lineal.intercept_
w = regresion_lineal.coef_

# Impresión por pantalla de los parámetros que se han estimado
print('w = ' + str(regresion_lineal.coef_) + ', b = ' +
str(regresion_lineal.intercept_))

# Predicción de los valores ‘y’ para los datos usados en el entrenamiento.
Impresión de gráfica comparativa.
pred_train = regresion_lineal.predict(Xtrain)
plt.scatter(ftrain,pred_train,c="red")
plt.scatter(ftrain,ytrain,c="green")
plt.show()

# Predicción de los valores ‘y’ para los datos usados en el test. Impresión
de gráfica comparativa.
pred_test = regresion_lineal.predict(Xtest)
plt.scatter(ftest,pred_test,c="blue")
plt.scatter(ftest,ytest,c="green")
plt.show()
```

```
# Evaluación el modelo de entrenamiento
# Cálculo del Error Cuadrático Medio (MSE = Mean Squared Error) de los datos
train
mse = mean_squared_error(ytrain, pred_train)
rmse = np.sqrt(mse)
print('Error Cuadrático Medio del entrenamiento(MSE) = ' + str(mse))
print('Raíz del Error Cuadrático Medio del entrenamiento(RMSE) = ' +
str(rmse))
mae=mean_absolute_error(ytrain, pred_train)
print('Error absoluto del train (MAE)= ' + str(mae))

# Calculamos el coeficiente de determinación R2
r2 = regresion_lineal.score(Xtrain, ytrain)
print('Coeficiente de Determinación R2 del entrenamiento = ' + str(r2))
print()

# Evaluación el modelo de entrenamiento
# Cálculo del Error Cuadrático Medio (MSE = Mean Squared Error) de los datos
train
mse = mean_squared_error(ytest, pred_test)
rmse = np.sqrt(mse)
print('Error Cuadrático Medio del test(MSE) = ' + str(mse))
print('Raíz del Error Cuadrático Medio del test(RMSE) = ' + str(rmse))
mae=mean_absolute_error(ytest, pred_test)
print('Error absoluto del test (MAE)= ' + str(mae))

# Cálculo del coeficiente de determinación R2
r2 = regresion_lineal.score(Xtest, ytest)
print('Coeficiente de Determinación R2 del test= ' + str(r2))
print()
```

Escenario 2: Regresión polinómica de segundo grado con variables normalizas para predicciones semanales

Se ha sustituido la variable ‘num_casos_ant’ del Escenario 2.1 por ‘num_casos_7’.

Regresión polinómica de tercer grado

Escenario 0: Regresión polinómica de tercer grado con variables no normalizadas

Este código es idéntico al del apartado 2 del anexo, excepto por el hecho de que se ha sustituido

```
pf = PolynomialFeatures(degree = 2)
```
por
```
pf = PolynomialFeatures(degree = 3)
```

Escenario 1: Regresión polinómica de tercer grado con variables normalizadas

Este código es idéntico al del apartado 2.1. del anexo, excepto por el hecho de que se ha sustituido

```
pf = PolynomialFeatures(degree = 2)
```
por
```
pf = PolynomialFeatures(degree = 3)
```

Escenario 2: Regresión polinómica de tercer grado con variables normalizadas para predicciones semanales

Este código es idéntico al del apartado 2.2. del anexo, excepto por el hecho de que se ha sustituido

```
pf = PolynomialFeatures(degree = 2)
```
por
```
pf = PolynomialFeatures(degree = 3)
```