



**Universidad
Rey Juan Carlos**

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

GRADO EN INGENIERÍA DEL SOFTWARE

Curso Académico 2021/2022

Trabajo Fin de Grado

**APRENDIZAJE Y PREDICCIÓN DE LA CALIDAD DEL
AIRE A PARTIR DE DATOS ABIERTOS**

Autor: Sergio Sánchez Vallés

Tutores: Holger Billhardt

1. Resumen

En este proyecto tenemos por objetivo la creación de un modelo de inteligencia artificial para el aprendizaje y la predicción de la calidad del aire en la ciudad de Madrid. Para la consecución del mismo, utilizaremos datos abiertos (datasets) de distintas estaciones de control de clima y contaminación.

Por ello, hemos realizado una investigación previa de los datos disponibles y dado que se trata de series temporales, ha sido necesaria una reestructuración al estar incorrectamente organizados. También se ha estudiado la correlación de los datasets de clima y contaminación para establecer su importancia.

Hemos empleado en el modelo distintas tecnologías, partiendo desde un lenguaje de programación Python, conocido por ser el máximo exponente en desarrollo de Machine Learning y Deep Learning en la actualidad. Destacamos el uso de las librerías para el análisis de datos, inteligencia artificial y la visualización y generación de gráficos (Pandas, Scikit-Learn, Keras y Matplotlib respectivamente). Como entorno de trabajo interactivo hemos usado Jupyter Notebook que permite ejecutar distintas celdas de código, visualizar gráficas y trabajar con datos de forma muy dinámica.

Se ha procurado seguir una metodología basada en proyectos de Data Science con una estructura que dividimos en: planteamiento del problema, recogida de datos, limpieza de datos, análisis exploratorio de datos (EDA), ingeniería de características y modelización. Se han desarrollado modelos de Deep Learning basados en las redes neuronales LSTM (Long short-term memory) para realizar los distintos entrenamientos, evaluaciones y predicciones.

El trabajo realizado en este proyecto se enmarca dentro del campo de la Inteligencia Artificial y el medio ambiente. Los resultados indican que la contaminación sigue una tendencia marcada que pueden ser objeto de predicciones, con lo cual se hace necesario el desarrollo de distintos Softwares para la predicción de la calidad del aire y la toma de medidas para mitigar dicha contaminación.

Palabras clave: Machine Learning, Deep Learning, Inteligencia Artificial, LSTM, predicción, contaminación, calidad del aire

2. Summary

In this project we aim to create an artificial intelligence model for learning and predicting air quality in the city of Madrid. In order to achieve it, we will use open data (datasets) from different climate and pollution control stations.

Therefore, we have carried out a previous investigation of the available data and since they consist of time series, we need to restructure them since they were incorrectly organized. We have also studied the correlation of the climate and pollution datasets to establish their significance.

We have used different technologies in the model, starting from the Python programming language, known for being the top language in Machine Learning and Deep Learning development nowadays. We highlight the use of libraries for data analysis, artificial intelligence and visualization and graphics generation (Pandas, Scikit-Learn, Keras and Matplotlib respectively). As interactive development environment we have used Jupyter Notebook that allows us to execute different code cells, visualize graphs and work with data in a very dynamic way.

We have tried to follow a methodology based on Data Science projects with a structure that we divide into: problem statement, data collection, data cleaning, exploratory data analysis (EDA), feature engineering and modeling. Deep Learning models based on LSTM (Long short-term memory) neural networks have been developed to perform the different training, evaluations and predictions.

The work carried out in this project is framed within the field of Artificial Intelligence and the climate environment. The results indicate that pollution follows a marked trend that can be subject to predictions, which makes necessary the development of different softwares for the prediction of air quality and taking measures to mitigate such pollution.

Keywords: Machine Learning, Deep Learning, Artificial Intelligence, LSTM, prediction, pollution, air quality.

3. Agradecimientos

Quiero dar las gracias a mi familia, sobretodo a mi madre y a mis abuelos, por estar siempre apoyándome en este duro camino, me han acompañado en toda mi carrera de ingeniería software y están muy orgullosos de la persona en la que me he convertido.

Contacto con el autor

Si desea ponerse en contacto con el autor de este documento puede escribir al email
serindio@gmail.com

Tabla de Contenidos

4. Tabla de Contenidos

1. Resumen.....	3
2. Summary.....	4
3. Agradecimientos.....	5
4. Tabla de Contenidos.....	7
5. Introducción.....	9
5.1. Precedentes.....	9
5.1.1. Europa.....	9
5.1.2. Madrid.....	11
5.2. Por qué la contaminación del aire nos afecta.....	12
5.3. ¿Se puede predecir la calidad del aire?.....	13
6. Objetivos del proyecto.....	17
7. Descripción informática.....	21
7.1. Recogida de datos.....	21
7.2. Limpieza de datos.....	25
7.3. Análisis exploratorio de datos (EDA).....	28
7.3.1. Correlación de los datos.....	29
7.3.2. Valores nulos.....	38
7.3.3. Índices de calidad del aire.....	39
7.4. Ingeniería de características (o de datos).....	44
7.5. Modelización, entrenamientos, evaluaciones y predicciones.....	47
8. Experimentos y validación.....	55
9. Conclusiones y trabajos futuros.....	59
10. Bibliografía.....	63
11. Apéndices.....	67
11.1. Instrucciones de instalación.....	67
11.2. Estructura del código.....	67
11.3. Manual de uso.....	68

5. Introducción

5.1. Precedentes

El aire de las ciudades, su polución, nos mata. La OMS estima que la inhalación de contaminantes mata a más de 7 millones de personas cada año [WHO], especialmente en países con bajos ingresos per cápita. El 90% de la población vive en áreas donde no se cumplen las pautas de calidad del aire que marca esta organización. Europa tiene algunas de las regulaciones de emisiones más estrictas del planeta, pero el aire contaminado sigue provocando más de 400.000 muertes prematuras al año en este continente. Aun en lugares donde no se sobrepasan las recomendaciones de la Organización Mundial de la Salud, se asocian muertes a la contaminación atmosférica, la evidencia científica respalda que no existe un umbral de exposición seguro para la contaminación.

5.1.1. Europa

Un estudio del instituto de investigación de Barcelona [ISGlobal], fue el primero en evaluar la carga de mortalidad atribuible a la contaminación del aire en más de 1,000 ciudades de Europa. El algoritmo se utilizó para calcular la puntuación de cada ciudad en función de la tasa de mortalidad de cada contaminante del aire, la tasa de mortalidad evitable y los años de vida perdidos.

Ciudades de España, Bélgica, Francia e Italia tienen muertes relacionadas con la inhalación de contaminantes, medidos por la concentración de dióxido de nitrógeno y partículas PM_{2,5} causadas por la oxidación del dióxido de azufre y nitrógeno y su interacción con el amoniaco. Brescia y Bérgamo son las ciudades más afectadas por la contaminación por PM_{2,5}, y Madrid y Amberes encabezan la lista de muertes por inhalación de dióxido de nitrógeno. En este estudio, creado a partir de datos de 2015, la mortalidad también es más alta en ciudades más pobladas como París, Barcelona, Milán y Bruselas, algunas de las ciudades vecinas se ven afectadas por la gran cantidad de residentes, esto es debido a que es más probable que viajen a una comunidad más grande cercana usando su automóvil privado, como explica el estudio.

También confirmaron que si todas las ciudades analizadas cumplieran con los niveles de PM_{2,5} y NO₂ recomendados por la OMS, cada una de estas medidas podrían evitar

Precedentes

entre 51.000 y 900 muertes prematuras cada año respectivamente. Además, si solo se redujera la contaminación al mismo nivel que las ciudades con las más bajas concentraciones se habría evitado aún más muertes, en concreto, 124.729 por la primera causa y 79.435 por la segunda. Reducir las emisiones puede ahorrar decenas de miles de muertes en toda Europa. “Encontramos que los resultados variaban ampliamente entre las diferentes ciudades analizadas. Los peores datos de mortalidad provienen principalmente del NO₂, un gas tóxico asociado con el tráfico rodado”, afirman los investigadores de ISGlobal.

Los expertos señalan que las ciudades con las tasas de mortalidad más altas se encuentran en la región italiana de la Llanura Padana, en el sur de Polonia y en el este de la República Checa, donde estas partículas en suspensión se emiten en exceso, producidas por múltiples fuentes además de los automóviles, como la industria, la calefacción doméstica, la quema de carbón y madera. Por ejemplo, en la ciudad italiana de Brescia, que ocupa el primer lugar en PM2,5, la tasa de muerte prematura provocada por estos finos contaminantes puede llegar a ser de hasta el 15%. En Madrid, esta cifra alcanza el 7%, según los investigadores, que podrían haberse evitado reduciendo las emisiones.

El estudio en cuestión hace algo más que analizar las ciudades más contaminadas, también compiló una lista de ciudades con las tasas de mortalidad por contaminación más bajas. Reikiavik, Tromso y Umea encabezan las muertes por PM2.5, mientras que Tromso, Umea (de nuevo) y Oulu encabezan las muertes por NO₂.

También se advierte que el máximo de concentraciones permitidas de NO₂ y PM2.5 deben revisarse para proteger mejor la salud pública. Este estudio es el primero de una serie de estudios que investigan los efectos en la salud de varios factores ambientales comunes a la vida urbana, como la contaminación del aire, contaminación acústica, la falta de acceso a espacios verdes y las olas de calor. Estas clasificaciones señaladas por la organización se basan en tasas de mortalidad estimadas por cada ciudad, y las estimaciones son algoritmos que tienen en cuenta la tasa de mortalidad, la tasa de mortalidad evitable y la esperanza de vida disminuida en cada ciudad.

Introducción

5.1.2. Madrid

En el estudio mencionado anteriormente [ISGlobal], Madrid figura en la lista de ciudades europeas con mayores tasas de mortalidad asociadas a la contaminación por NO₂. También concluye que el área metropolitana de Madrid es el área metropolitana del continente con mayor mortalidad asociada a la contaminación por dióxido de nitrógeno. La posición en este ranking de las diferentes ciudades de Europa depende de factores como el tamaño de la población, las muertes relacionadas con la contaminación y la esperanza de vida en cada ubicación. Desde el instituto de investigación aseguran que eran conocedores de los altos niveles de contaminación en Madrid, pero hasta el momento no se habían realizado estudios de mortalidad y métodos capaces de comparar tantas ciudades.

En el caso del dióxido de nitrógeno (NO₂) como en Madrid, el principal causante de la contaminación es el tráfico rodado, especialmente los vehículos diésel, que refleja altos niveles de dióxido de nitrógeno en el área metropolitana de Madrid. Sin embargo, las muertes atribuibles a la contaminación no es el único problema, múltiples problemas de salud son causados por la inhalación de estos.



Ilustración 1: Alerta de previsión de contaminación en una autopista de Madrid.

Los efectos de la contaminación provocan aumentos del riesgo de infecciones respiratorias, accidentes cerebro-vasculares, enfermedades cardíacas o cáncer de pulmón (este último afecta a los más vulnerables), estos datos están siendo muy reveladores, el propio portal [MadridSalud] de salud del Ayuntamiento de Madrid afirma que los altos niveles de dióxido de nitrógeno no solo pueden irritar los pulmones y reducir su función, sino tam-

Precedentes

bién reducir la resistencia a las infecciones respiratorias. Y de hecho, la irritación provocada por este contaminante está asociada a un aumento de la cantidad de mucosidad en las vías respiratorias superiores, lo que puede aumentar y exacerbar los síntomas de infecciones respiratorias en pacientes con enfermedades respiratorias crónicas, asma y alergias.

Si Madrid cumpliera las guías de contaminación más actuales (2021) de la Organización Mundial de la Salud [WHO], hasta 1.966 muertes se evitarían anualmente, en el caso de que cumpliera aunque sea con las recomendaciones de la guía de 2005, podrían evitarse 206 muertes anuales, y si consiguiéramos mantener los datos más bajos del año respecto a la propia ciudad, podríamos estar hablando de hasta 2.380 muertes menos por año calcula el estudio [ISGlobal].

5.2. Por qué la contaminación del aire nos afecta

Más de 10.000 litros de aire pasan cada día por nuestros pulmones, inhalando a la vez pequeñas partículas de menos de 2,5 μm de diámetro (PM2.5), que causan enfermedades cardiovasculares, respiratorias y cáncer. Una mezcla compleja de partículas, gotas y gases químicos de emisiones industriales, quema de combustibles, tráfico rodado y muchas otras fuentes. Inhalar aire contaminado tiene un efecto negativo en casi todos los órganos de nuestro cuerpo, incluido el cerebro. Cada vez que respiramos aire contaminado, llega a nuestros pulmones una mezcla de partículas tóxicas, de las cuales algunas pueden salir a cualquier parte e incluso atravesar la barrera placentaria.

Algunas muertes pueden también ser resultado de la suma de varios factores adicionales, por ejemplo fumar y la contaminación del aire exterior pueden causar cáncer de pulmón. Las muertes por algunos cánceres de pulmón se pueden evitar mejorando la calidad del aire interior y reduciendo el consumo de tabaco. Una evaluación de 2013 realizada por la Agencia Internacional de Investigación sobre el Cáncer de la OMS [IARC] encontró que la contaminación del aire exterior es cancerígena para los humanos y que las partículas de aire contaminado están fuertemente asociadas con un aumento de cáncer, especialmente el de pulmón. También se ha observado un vínculo entre la contaminación del aire exterior y el aumento del cáncer de vejiga y del tracto urinario.

Introducción

No todas las causas de la contaminación del aire exterior están fuera del control de la población, aunque no elegimos por donde respiramos, siempre se pueden mejorar hábitos y tomar medidas a nivel doméstico, aún así, mejorar la calidad del aire también requiere la acción de las ciudades, los alcaldes y, por supuesto, las organizaciones nacionales e internacionales en áreas como el transporte, la energía, la construcción y la agricultura. Combatir las causas del cambio climático y la contaminación del aire mejorará significativamente nuestras vidas, es una batalla que traerá grandes beneficios para nuestra salud, el tiempo corre en nuestra contra y ahora es el momento de actuar antes de que sea demasiado tarde para las actuales y futuras generaciones.

5.3. ¿Se puede predecir la calidad del aire?

En esta sección y como objetivo de este trabajo, vamos a estudiar las distintas posibilidades que existen para poder hacer predicciones de la calidad del aire, en este caso es fundamental crear modelos predictivos que ayuden a estimar los niveles de calidad del aire para un lugar, momento y condición determinados. De todas las ramas de la inteligencia artificial, vamos a ver cómo el aprendizaje profundo puede superar estas dificultades.

Vamos a explorar algunas posibilidades para estos modelos, el principal problema en estos casos es cómo combinar las principales fuentes de datos (contaminación, clima, tráfico, etc). Por otro lado, las estaciones meteorológicas terrestres recopilan datos constantemente, pero solo desde un número limitado de ubicaciones, al igual que con las estaciones del clima.

¿Se puede predecir la calidad del aire?

También se pueden usar sensores satelitales que proporcionan una medición de alta resolución espacial de la calidad del aire troposférico para capturar la variabilidad espacial de la contaminación del aire. En la imagen, podemos ver un ejemplo visual de la concentración cambiante de dióxido de nitrógeno en Europa antes y tras la pandemia del Covid-19 [ESA].

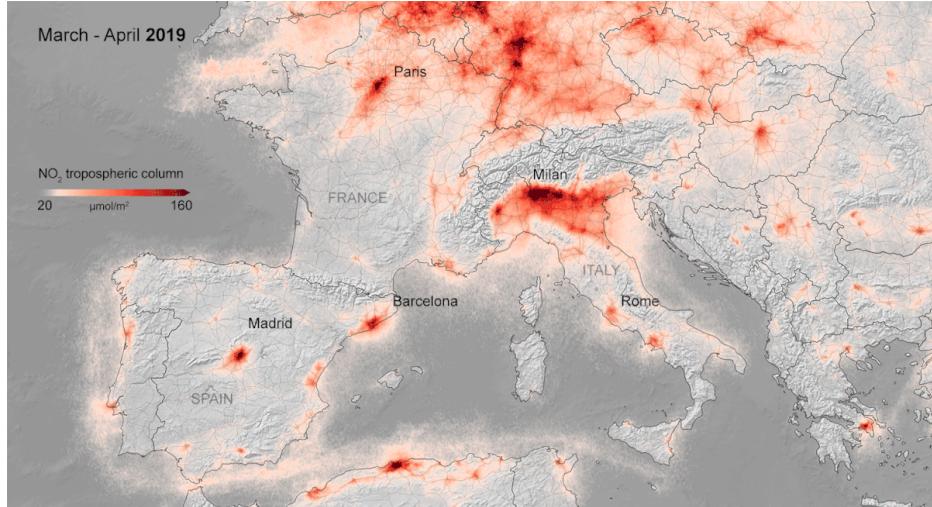


Ilustración 2: Concentraciones de dióxido de nitrógeno vistas por el satélite Copernicus Sentinel-5P de marzo a abril de 2019

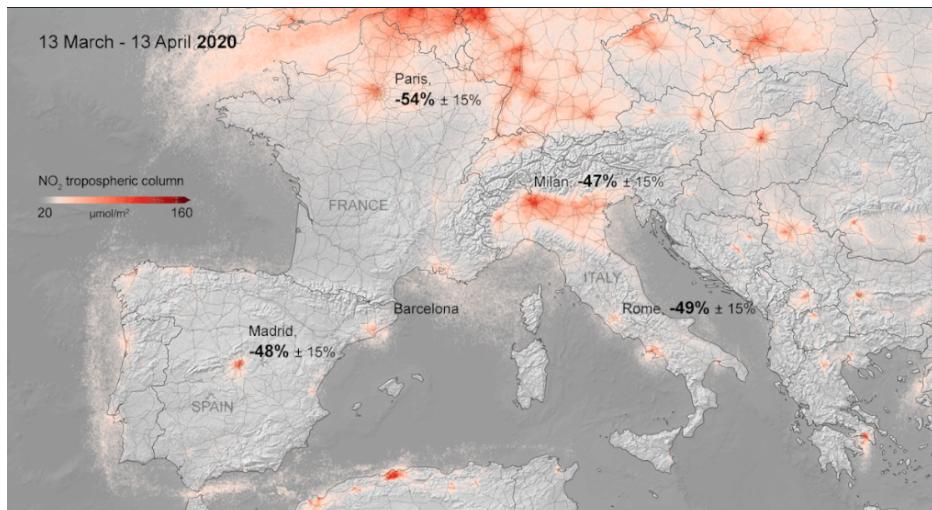


Ilustración 3: Concentraciones de dióxido de nitrógeno vistas por el satélite Copernicus Sentinel-5P de marzo a abril de 2020

Los sensores de tierra proporcionan datos muy precisos, pero tienen que estar muy bien ubicados, poniendo como ejemplo AirNow, una asociación que recopila datos de varias asociaciones del gobierno de EE.UU. y datos de sus embajadas y consulados en todo el

Introducción

mundo para proporcionar datos de contaminación del aire de alta calidad a propietarios de viviendas, investigadores, empresas y el público. El ayuntamiento de Madrid también proporciona datos de varias estaciones a lo largo de la ciudad, recopilando datos horarios, diarios desde hace más de 20 años.

De todo el campo de la inteligencia artificial, el deep learning o aprendizaje profundo es un subcampo dentro del aprendizaje automático que utiliza redes neuronales para recopilar representaciones de datos cada vez más significativas a través del aprendizaje basado en capas. Por esta razón, el aprendizaje profundo es especialmente adecuado para detectar patrones a partir de grandes cantidades de datos, incluidos los datos no estructurados, también realiza ingeniería de variables dentro de cada neurona para crear datos nuevos o desechar los menos relevantes.

Hay bastantes investigaciones sobre este tipo de predicciones, pero vamos a tomar como ejemplo y analizar el realizado por unos investigadores de Estados Unidos, su paper científico [DL-N02] se basa en modelos de redes neuronales profundas (Deep Neural Networks), en adelante DNN, que compara las observaciones terrestres directas con las observaciones satelitales, produce estimaciones más confiables de dióxido de nitrógeno y captura mejor la distribución espacial de las concentraciones de NO, agregar información como datos meteorológicos, altitud y la ubicación de estaciones terrestres, carreteras principales y centrales eléctricas ha aumentado la precisión del pronóstico. Esta alta resolución del espacio-tiempo es útil para estudiar la evolución de los contaminantes del aire, ya que también se puede aplicar a otros gases de efecto invernadero y a otras escalas geográficas.

También comparan dos de sus modelos, en uno de ellos utilizan interpolaciones de los datos de las estaciones para intentar averiguar los datos en otros lugares donde no estén las estaciones, en el otro directamente utilizan una matriz de DNN para hacer una cuadrícula de los datos en el mapa.

En conclusión podemos ver que este tipo de problemas ya se está modelando desde hace años con redes de aprendizaje profundo, sobretodo cuando tenemos una cantidad muy grande de datos.

6. Objetivos del proyecto

Los objetivos del presente proyecto es investigar el impacto de la contaminación de la ciudad de Madrid, realizando un estudio completo de Data Science con datos abiertos de varias fuentes, con el fin de desarrollar una inteligencia artificial capaz de dar estimaciones de contaminación a días futuros basándonos en la experiencia pasada, así como implementar todo el ciclo y estructura que este tipo de problemas conlleva.

Para lograr este objetivo general, se ha seguido la siguiente metodología, que veremos más en profundidad en la descripción informática:

- Planteamiento del problema. Con todo el conocimiento de los algoritmos de aprendizaje automático, es fácil olvidar que el propósito del aprendizaje automático es resolver problemas con datos. Esto puede incluir análisis predictivos que intentan predecir el futuro o análisis exploratorios que intentan responder preguntas sobre cómo y por qué sucedió algo. Usamos la ciencia de datos no porque queramos implementar redes neuronales complejas y utilizar tarjetas gráficas (GPU), sino simplemente porque queremos responder algunas preguntas que ayudarán a nuestra organización, nuestro entorno o nuestro planeta. Por lo tanto, es muy importante comenzar con lo que desea responder con los datos.
- Recogida de datos. Para poder resolver el problema en cuestión necesitamos recopilar los datos necesarios de las diferentes fuentes disponibles, en nuestro caso usaremos datos de dominio público suministrado directamente por gobiernos y ayuntamientos, Ayuntamiento de Madrid y la AEMET (Agencia Estatal de Meteorología). Los datos suelen dividirse en dos tipos: Estructurados y no estructurados. El ejemplo más sencillo de datos estructurados sería un archivo .xls o .csv en el que cada columna representa un atributo de los datos. Los datos no estructurados podrían estar representados por un conjunto de archivos de texto, fotos o vídeos. A menudo, el propio problema dicta cómo organizar la recopilación y el almacenamiento de los datos, en otros casos tenemos que buscar estas colecciones de datos en fuentes de terceros.
- Limpieza de datos. Como se mencionó anteriormente, para tomar decisiones importantes se necesita tener los datos limpios y lo más claros posibles. Debido a la gran

Objetivos del proyecto

cantidad de datos que fluyen entre múltiples fuentes, el uso de una herramienta de limpieza de datos que garantice la precisión de la información nos ayudará a tomar las decisiones correctas. La limpieza de datos de alta calidad nos ayuda a proporcionar información completa y confiable al proyecto, para que se puedan identificar las necesidades cambiantes del problema. Es importante eliminar las entradas de datos incorrectos o de alguna manera arreglarlos. De esta forma, se ayudará al rendimiento de los datos con los que se está trabajando, para esto se necesita interpretar los datos y comprender el valor que aportan y cómo se pueden utilizar para mejorar los resultados.

- Análisis exploratorio de datos (EDA). La finalidad del Análisis Exploratorio de Datos, en adelante EDA, es examinar los datos previamente a la aplicación de cualquier técnica estadística. Analizar e investigar conjuntos de datos y resumir sus características principales, a menudo utilizando métodos de visualización de datos. Ayuda a determinar la mejor manera de manipular las fuentes de datos para obtener las respuestas que necesita y facilita que los planificadores de preguntas identifiquen patrones, identifiquen evento anómalos, prueben o confirmen hipótesis. El EDA se usa principalmente para obtener una mejor comprensión de las variables en un conjunto de datos y las relaciones entre ellos. También puede ayudar a determinar si los métodos estadísticos que implementa para analizar los datos son apropiados.
- Ingeniería de características (o de datos). Los conjuntos de datos resultantes tendrán diferentes tamaños. Para responder algunas preguntas interesantes, es posible que no se necesiten todas las variables. Es importante informarse lo máximo posible y realizar las pertinentes investigaciones para poder tener un conocimiento más experto sobre la materia de los datos, las relaciones entre ellos. Dependiendo de su estudio, se puede ahorrar mucho tiempo decidiendo qué es lo que hay que conservar y cuáles se pueden eliminar del modelo. Otra estrategia podría ser usar las funciones propias de algoritmos de aprendizaje automático para ver la importancia relativa de las variables y así explicar la variación observada en el modelo. La tercera estrategia, muy científica, es utilizar el aprendizaje automático aplicado. Esto

Objetivos del proyecto

puede ser contrario a la intuición, ya que es posible que se pregunte cómo podemos usar machine learning cuando queremos usarlo para nuestro modelo al mismo tiempo. El análisis de componentes principales se puede utilizar para extraer características importantes de un conjunto de datos.

- Modelización, entrenamientos, evaluaciones y predicciones. Si hemos analizado correctamente el problema que tenemos que resolver, seguramente se puede entrever a qué categoría de problema de inteligencia artificial nos estamos enfrentando, las categorías en las que se puede dividir este tipo de problemas son: aprendizaje supervisado, aprendizaje semi-supervisado, aprendizaje no supervisado y aprendizaje por refuerzo. Cada una de estas tiene sus propios métodos de entrenamiento, pero al final con todas tenemos que iterar sobre los modelos con los distintos parámetros e hiperparámetros (parámetros de configuración para un entrenamiento en concreto), para evaluar estos modelos tenemos conjuntos de prueba o test con los que estimaremos unos resultados y finalmente realizar predicciones sobre otros datos.
 - Aprendizaje supervisado: Para poder entrenar este tipo de modelos se necesitan datos de entrenamiento previamente etiquetados con los que la inteligencia artificial aprenda de datos históricos y sepa aplicarlos para obtener la salida correcta o deseada.
 - Aprendizaje no supervisado: Este tipo de modelos carece de datos etiquetados en si, no hay ese conocimiento a priori, no necesita de un agente externo que enseñe, sino que aprende organizando los propios datos de forma autónoma, los tipos de problemas más comunes que se pueden resolver son: agrupamiento de datos, prototipado, extracción de variables, relacionar variables, análisis de los componentes principales, familiaridad de unos datos con otros.
 - Aprendizaje semi-supervisado: Se nutre de los datos previamente etiquetados como en los modelos de aprendizaje supervisado, pero también de datos sin etiquetado como en el no supervisado, normalmente se tiene una gran cantidad de datos sin etiquetar y un pequeño conjunto previamente etiquetado, esto hace

Objetivos del proyecto

que se pueda mejorar el rendimiento del aprendizaje ayudándonos de más datos aunque no los tengamos clasificados.

- Aprendizaje por refuerzo: Este tipo de modelos se basan en un agente dentro de un entorno con ciertas reglas, este agente tiene la capacidad de tomar distintas acciones con las que pueda modificar el estado del entorno, para que aprenda, se le proporciona al agente con recompensas tras evaluar su acción, que puede ser negativa o positiva.

7. Descripción informática

En este capítulo vamos a detallar el desarrollo del proyecto en profundidad, explicando por secciones todo el proceso.

7.1. Recogida de datos

Desde la página web de datos abiertos del Ayuntamiento de Madrid hemos recopilado los datasets de contaminación diarios, para complementar esta información, se han recogido también los datos de clima diarios sacados de la página de la Agencia Estatal de Meteorología [AEMET].

Los datos de calidad del aire nos proporcionan información detallada de la cantidad de contaminantes presentes desde el año 2001, estos son actualizados todos los días a través del portal web, pero también se pueden descargar utilizando la API REST a través del protocolo HTTP y en una gran variedad de formatos, hemos seleccionado el formato de archivos “.CSV” (Comma Separated Values), son archivos de texto delimitados que utilizan las comas para separar los distintos valores o columnas, estableciendo la primera fila como el nombre de las columnas, y de ahí en adelante el resto de valores con el mismo formato de comas, como vemos a continuación:

*PROVINCIA;MUNICIPIO;ESTACION;MAGNITUD;PUNTO_MUESTREO;ANO;
MES;D01;V01;D02;V02;D03;V03;D04;V04;D05;V05;D06;V06;D07;V07;D08;V08;D09;
V09;D10;V10;D11;V11;D12;V12;D13;V13;D14;V14;D15;V15;D16;V16;D17;V17;D18;V
18;D19;V19;D20;V20;D21;V21;D22;V22;D23;V23;D24;V24;D25;V25;D26;V26;D27;V2
7;D28;V28;D29;V29;D30;V30;D31;V31*

```
PROVINCIA;MUNICIPIO;ESTACION;MAGNITUD;PUNTO_MUESTREO;ANO;MES;D01;V01;D02;V02;D03;V03;D04;  
28;079;4;1;28079004_1_38;2001;01;00017;V;00015;V;00015;V;00016;V;00020;V;00025;V;  
28;079;4;1;28079004_1_38;2001;02;00040;V;00042;V;00038;V;00029;V;00018;V;00023;V;00016;V;  
28;079;4;1;28079004_1_38;2001;03;00019;V;00018;V;00013;V;00011;V;00015;V;00023;V;00015;V;  
28;079;4;1;28079004_1_38;2001;04;00014;V;00017;V;00013;V;00010;V;00009;V;00017;V;00011;V;  
28;079;4;1;28079004_1_38;2001;05;00010;V;00011;V;00016;V;00018;V;00015;V;00013;V;00013;V;  
28;079;4;1;28079004_1_38;2001;06;00013;V;00010;V;00012;V;00011;V;00012;V;00011;V;00012;V;  
28;079;4;1;28079004_1_38;2001;07;00014;V;00013;V;00012;V;00009;V;00007;V;00007;V;00009;V;  
28;079;4;1;28079004_1_38;2001;08;00007;V;00006;V;00007;V;00006;V;00006;V;00010;V;00009;V;  
28;079;4;1;28079004_1_38;2001;09;00006;V;00007;V;00007;V;00007;V;00007;V;00007;V;00007;V;
```

Ilustración 4: Muestra de los datos en bruto de calidad del aire.

Recogida de datos

Para poder tener cierta agilidad descargando, actualizando y borrando estos datos, se ha establecido un programa en Python que contiene dos funciones, una para conectarse a la web del Ayuntamiento de Madrid, que se encarga de leer un archivo de configuración “JSON” que hemos creado con antelación, especificando las rutas a los archivos de cada año para la contaminación del aire, descargando todos los datos.

La otra función se encarga de los datos del clima de la AEMET, este lee un archivo de configuración donde se encuentra la API KEY (necesaria para acceder), con esto, consigue conectarse. Según como funciona esta, primero accede a los datos de todas las estaciones que existen en España, filtraremos todas a las que se correspondan con la provincia de Madrid y lo guardaremos en una variable temporalmente, posteriormente se realizará otra petición a la API, iterando todas estas estaciones que habíamos guardado, pidiendo los datos diarios para todas ellas en las fechas que requerimos para este proyecto 2001-2021.

Todo esto se procesa como una “Pipeline” de datos en orden, tenemos un método que se encarga de limpiar nuestra carpeta de datos en bruto (“data/raw”) para que no se mezcle con los nuevos (se actualizan los datos diariamente y se realizaron varias pruebas con otras fechas antes de establecer las finales), después se descargan los datos de calidad del aire y justo después los de clima explicando en el párrafo anterior. Veremos cómo en pantalla nos aparece un “log” de todos los datos que estamos descargando en tiempo real para ver cómo va el progreso.

Junto con los archivos de calidad del aire, se nos proporciona también en la página web un texto en formato “PDF” explicando cada columna del conjunto de datos. Estos datos en bruto, distan mucho de ser ideales para poder trabajar con ellos de manera óptima, por lo que en adelante en otras secciones, vamos a mostrar cómo transformarlos para poder trabajar con ellos y entenderlos mejor.

Como se ve en la tabla, tenemos distintas estaciones a las que se les han atribuido los siguientes códigos:

Descripción informática

Las estaciones señaladas con un asterisco (), cambiaron su código a partir de la fecha que se indica para la adaptación a la codificación nacional de intercambio de datos de calidad del aire.*

28079001	Pº. Recoletos	Baja.- 04/05/2009 (14:00 h.)
28079002	Gta. de Carlos V	Baja.- 04/12/2006 (11:00 h.)
28079003	Pza. del Carmen	* Código desde enero 2011
28079035(*)		
28079004	Pza. de España	
28079005	Barrio del Pilar	* Código desde enero 2011
28079039(*)		
28079006	Pza. Dr. Marañón	Baja.- 27/11/2009 (08:00 h.)
28079007	Pza. M. de Salamanca	Baja.- 30/12/2009 (14:00 h.)
28079008	Escuelas Aguirre	
28079009	Pza. Luca de Tena	Baja.- 07/12/2009 (08:00 h.)
28079010	Cuatro Caminos	* Código desde enero 2011
28079038(*)		
28079011	Av. Ramón y Cajal	
28079012	Pza. Manuel Becerra	Baja.- 30/12/2009 (14:00 h.)
28079013	Vallecas	* Código desde enero 2011
28079040(*)		
28079014	Pza. Fdez. Ladreda	Baja.- 02/12/2009 (09:00 h.)
28079015	Pza. Castilla	Baja.- 17/10/2008 (11:00 h.)
28079016	Arturo Soria	
28079017	Villaverde Alto	
28079018	C/ Farolillo	
28079019	Huerta Castañeda	Baja.- 30/12/2009 (13:00 h.)
28079020	Moratalaz	* Código desde enero 2011
28079036(*)		
28079021	Pza. Cristo Rey	Baja.- 04/12/2009 (14:00 h.)
28079022	Pº. Pontones	Baja.- 20/11/2009 (10:00 h.)
28079023	Final C/ Alcalá	Baja.- 30/12/2009 (14:00 h.)
28079024	Casa de Campo	
28079025	Santa Eugenia	Baja.- 16/11/2009 (10:00 h.)
28079026	Urb. Embajada (Barajas)	Baja.- 11/01/2010 (09:00 h.)
28079027	Barajas	
28079047	Méndez Álvaro	Alta.- 21/12/2009 (00:00 h.)
28079048	Pº. Castellana	Alta.- 01/06/2010 (00:00 h.)
28079049	Retiro	Alta.- 01/01/2010 (00:00 h.)
28079050	Pza. Castilla	Alta.- 08/02/2010 (00:00 h.)
28079054	Ensanche Vallecas	Alta.- 11/12/2009 (00:00 h.)
28079055	Urb. Embajada (Barajas)	Alta.- 20/01/2010 (15:00 h.)
28079056	Plaza Elíptica	Alta.- 18/01/2010 (12:00 h.)
28079057	Sanchinarro	Alta.- 24/11/2009 (00:00 h.)
28079058	El Pardo	Alta.- 30/11/2009 (13:00 h.)
28079059	Parque Juan Carlos I	Alta.- 14/12/2009 (00:00 h.)
28079086	Tres Olivos	Alta.- 14/01/2010 (13:00 h.) *
28079060(*)		Código desde enero 2011

Ilustración 5: Códigos para las distintas estaciones de calidad del aire de Madrid.

Recogida de datos

Las estaciones dadas de alta a lo largo de los años son cambiantes. En el año 2009, podemos observar que la mayoría de estaciones se han dado de baja, esto puede suponer problemas a la hora de explorar los datos y posteriormente realizar predicciones, veremos cómo afrontar esto más adelante.

En cuanto a los datos del clima, hemos recurrido a los de la [AEMET], tienen un conjunto de todas las estaciones que monitorizan la meteorología en España, utilizando también su API REST por HTTP, y pidiendo acceso a los datos mediante previo registro en su portal web, conseguimos acceder al dataset también en formato “.CSV”. A diferencia de los de calidad del aire, tenemos acceso a un índice de todas las estaciones, teniendo que establecer un filtrado de estas para obtener las de la provincia de Madrid, una vez realizado este filtrado, obtenemos las estaciones deseadas con sus respectivos códigos y posiciones geográficas:

indicativo	nombre	latitud	longitud	altitud
2462	PUERTO DE NAVACERRADA	404735N	040038W	1894
3100B	ARANJUEZ	400402N	033246W	540
3110C	BUITRAGO DEL LOZOYA	410025N	033649W	1030
3111D	SOMOSIERRA	410808N	033449W	1450
3129	MADRID AEROPUERTO	402800N	033320W	609
3175	TORREJÓN DE ARDOZ	402919N	032637W	607
3191E	COLMENAR VIEJO	404146N	034554W	1004
3194U	MADRID, CIUDAD UNIVERSITARIA	402706N	034327W	664
3195	MADRID, RETIRO	402443N	034041W	667
3196	MADRID, CUATRO VIENTOS	402232N	034710W	690
3200	GETAFE	401758N	034320W	620
3266A	PUERTO ALTO DEL LEÓN	404223N	040831W	1532
3338	ROBLEDO DE CHAVELA	402540N	041500W	763

Ilustración 6: Códigos para las distintas estaciones del clima de Madrid.

Descripción informática

Una vez accedido a estos datos de climatología, hemos decidido descargar el mismo período temporal que los de calidad del aire, estos: desde el 1 de enero de 2001, hasta el 31 de diciembre de 2021, en total 21 años de información. Creemos que serán suficientes para poder realizar unas estimaciones bien formadas.

Estos datos se nos presentan con una mejor estructura que los anteriores, nos permite trabajar con ellos con bastante facilidad, destacando su claridad de formato:

fecha,indicativo,nombre,provincia,altitud,tmed,prec,tmin,horatmin,tmax,horatmax,velmedia,sol,presMax,horaPresMax,presMin,horaPresMin,dir,racha,horaracha

2001-01-01,2462,PUERTO DE NAVACERRADA,MADRID,1894,"0,3","10,1",-1,0",23:30,"1,6",15:00,"7,5","0,0","806,3",00,"800,1",17,,,

7.2. Limpieza de datos

Refiriéndonos al apartado anterior, muchos de estos datos que se han recopilado, distan de ser ideales para ser utilizados tanto en sistemas de inteligencia artificial como en la elaboración de estadísticas para entenderlos mejor, es por esto que tenemos que realizar una limpieza y cambios en los formatos. Para todo lo anterior vamos a utilizar la librería de [Pandas], esta herramienta de [código abierto] para Python nos permite la manipulación y análisis de datos, está basada en un tipo de dato llamado Dataframe, en el cual pueden ser cargados y modificados los datos, por lo que resulta ideal para este proyecto.

El dataset más complejo de manejar sin duda fue el de calidad del aire del Ayuntamiento de Madrid, debido a que en cada fila se representan los datos de cada mes y en las columnas se describen los días, junto a una variable de “validación” de los datos para cada día, que según nos indica la propia fuente de los datos, si presentan una “V”, quiere decir que son datos correctamente verificados y validados, con lo que también tenemos que filtrar los datos “no validos”.

Sumado a esto, cada fila solamente representa una magnitud de un contaminante, lo que dificulta la lectura de los datos, esto es otra de las transformaciones necesarias que ten-

Limpieza de datos

dremos que realizar. Según la tabla a continuación, vamos a sustituir los códigos de los contaminantes por sus propios nombres:

Magnitud		Abreviatura o fórmula	Unidad medida	Técnica de medida	
01	Dióxido de Azufre	SO ₂	µg/m ³	38	Fluorescencia ultravioleta
06	Monóxido de Carbono	CO	mg/m ³	48	Absorción infrarroja
07	Monóxido de Nitrógeno	NO	µg/m ³	08	Quimioluminiscencia
08	Dióxido de Nitrógeno	NO ₂	µg/m ³	08	Id.
09	Partículas < 2.5 µm	PM2.5	µg/m ³	47	Microbalanza
10	Partículas < 10 µm	PM10	µg/m ³	47	Id.
12	Oxidos de Nitrógeno	NOx	µg/m ³	08	Quimioluminiscencia
14	Ozono	O ₃	µg/m ³	06	Absorción ultravioleta Cromatografía de gases
20	Tolueno	TOL	µg/m ³	59	
30	Benceno	BEN	µg/m ³	59	Id.
35	Etilbenceno	EBE	µg/m ³	59	Id.
37	Metaxileno	MXY	µg/m ³	59	Id.
38	Paraxileno	PXY	µg/m ³	59	Id.
39	Ortoxileno	OXY	µg/m ³	59	Id.
42	Hidrocarburos totales (hexano)	TCH	mg/m ³	02	Ionización de llama
43	Metano	CH4	mg/m ³	02	Id.
44	Hidrocarburos no metánicos (hexano)	NMHC	mg/m ³	02	Id.

Ilustración 7: Códigos para los distintos contaminantes de las estaciones del clima de Madrid.

Nos encontramos con la necesidad de transformar estos datos, así, establecemos nuevas columnas y pivotamos otras. De esta forma crearemos un nuevo formato.

Para poder tratar mejor los datos, necesitamos tener las fechas claras, por lo que vamos a asignar la fecha con formato “año-mes-día”, esto será el nuevo índice para todas nuestras series temporales, esto nos permitirá en el futuro poder hacer búsquedas sobre los datos, sacar estadísticas y tener todo organizado. Como vemos en la tabla a continuación, otro de nuestros índices serán las estaciones, puesto que no todas las estaciones miden los mismos contaminantes y tenemos distinta cantidad de estaciones dependiendo de la fecha,

Descripción informática

se han creado valores vacíos en los que no haya datos de cierto contaminante para cada fecha y estación.

date	station	SO ₂	CO	NO	NO ₂	PM _{2,5}	PM ₁₀	NO _x	O ₃	TOL	BEN	EBE	TCH	CH ₄	NMH C
2012-05-27	2807 9024	2	0,2	1	10	6	13	12	64	0,5	0,3	0,5	1,35	1,1	0,25
2012-05-27	2807 9027			4	24			30	65				1,23	1,09	0,14
2012-05-27	2807 9035	2	0,2	7	32			43	55						
2012-05-27	2807 9036	3	0,2	4	24		19	31							
2012-05-27	2807 9038	1		10	26	6	13	41		1,5	0,2	0,5			

Los valores vacíos, también llamados nulos, son uno de los problemas más típicos en proyectos de data science, tenemos que tener mucho cuidado qué datos nulos introducimos y qué datos se puede considerar nulos también, para este proyecto los datos vacíos se han podido filtrar previamente por la propia fuente de los datos cuando se nos indicaba si un día era “V” válido o no, pero veremos más adelante si necesitamos crear más transformaciones.

En cuanto al dataset del clima de la AEMET, no hemos requerido de una limpieza de estos datos, pues ya se encontraban en un formato muy adecuado para este proyecto, lo único que hemos realizado han sido algunos cambios en el nombre de las columnas para que fueran más explicativas, hemos estudiado el significado de cada una de ellas también para poder llevar a cabo esta tarea.

Limpieza de datos

Para llevar a cabo todas las transformaciones que hemos descrito, se ha creado un script en Python, similar al que comentamos en el apartado de recogida de datos. Este script se encarga de iterar todos los datos que hemos descargado previamente con el formato “tipo-dato_año”, como por ejemplo: “air-quality_2013.csv”, “weather_2013.csv” de la carpeta “data/raw”, previamente categorizados en el script de descarga de los datasets. Las transformaciones de formato mencionadas en esta sección entran dentro de la categoría de limpieza de los datos, pues sin estas, sería inviable realizar un estudio de los datos.

A continuación el script de procesamiento de estos datos, va realizando las transformaciones necesarias a cada archivo dependiendo de si se trata de datos de la contaminación o de la climatología, en el método “process_data()”, según se van transformando los archivos, se vuelven a volcar, esta vez en la carpeta “data/interim”, con el mismo nombre de archivo pero con el formato deseado.

Pero además de estas transformaciones, el punto clave de este programa, es que en el siguiente método vamos a concatenar todos los archivos procesados anteriormente, manteniendo el orden temporal y también ordenado por el código de las estaciones, de forma que al final del procesado nos quedan únicamente dos archivos en “data/processed”, que serían “air-quality.csv” y “weather.csv”, conteniendo la totalidad de los datos de todos los años.

7.3. Análisis exploratorio de datos (EDA)

Seguramente este es uno de los puntos más importantes en todo proyecto de inteligencia artificial, pues es imposible tomar decisiones sobre un modelo de machine learning sin antes entender las necesidades y peculiaridades de nuestro dataset. Se trata del momento de sacar estadísticas, tener una toma de contacto con los datos más allá de ver filas y columnas, podemos generar gráficos de barras, matrices de correlación, gráficos de dispersión y mucho más. Podemos estudiar los datos del clima y de la calidad del aire por separado, juntos, combinando algunos únicamente o todos a la vez.

7.3.1. Correlación de los datos

Al recopilar todos estos datos, estamos suponiendo que tanto la calidad del aire como la climatología tienen algo que ver entre ellas, es inevitable hacernos varias preguntas ¿Cuánto influye la climatología en la contaminación del aire? -y viceversa- ¿Cuánto influyen los contaminantes sobre el clima? El sentido común o la experiencia nos podría preparar para responder a algunas de estas cuestiones, como por ejemplo, es comúnmente conocido que cuando llueve, se arrastra toda la contaminación del cielo al suelo, por lo que después tenemos menos contaminantes en el aire pero, ¿Es esto cierto? Vamos a intentar responder a esta y más cuestiones en este apartado.

La correlación no implica causalidad, hay que investigar más. Cuando analicemos los siguientes datos, veremos que existe una relación entre A y B, pero en muchos casos no implica que A suceda por B o viceversa. El estudio de la correlación entre dos variables es uno de los temas estudiados en estadística, se calcula con el coeficiente de correlación, en nuestro caso vamos a utilizar el coeficiente de correlación de [Pearson], una medida de correlación lineal entre dos datos.

Cuanto más cerca de 1, mayor es la correlación positiva entre las variables. Cuanto más cerca de -1, mayores son las correlaciones negativas entre las variables. Cuanto más cerca de cero, menor es la correlación entre las variables. En ninguna parte esta teoría permite afirmar tan a la ligera que el hecho de que exista una relación muy fuerte entre A y B significa que la variable A es la causante de la variable B. Muchas veces cuando decimos que la expresión correlación no implica causalidad, queremos decir que el hecho de que exista una relación entre dos variables no significa que una cause la otra, pero tampoco que si encontramos una asociación entre dos variables, podamos descartar automáticamente que uno sea la causa del otro. El problema de creer que una fuerte correlación implica alguna causalidad entre las variables, es que esto puede llevarnos a error, porque no es muy difícil encontrar una correlación entre dos variables que no tengan nada que ver entre ellas, esto puede estar dado por los sesgos de la propia persona que analice los datos.

Al tener una gran cantidad de variables distintas o columnas, y también una gran cantidad de datos, decidimos empezar realizando unas matrices de correlación de los datos

Análisis exploratorio de datos (EDA)

como vamos a ver a continuación, lo hemos mostrado en las siguientes figuras con códigos de color también. Para calcular las correlaciones hemos utilizado la función de correlación de [Pandas] y para la creación de estas visualizaciones hemos recurrido a la librería de alto nivel [Seaborn], la cual está basada a su vez en la famosa herramienta de [Matplotlib] y nos ofrece una sencilla API para facilitarnos el desarrollo.

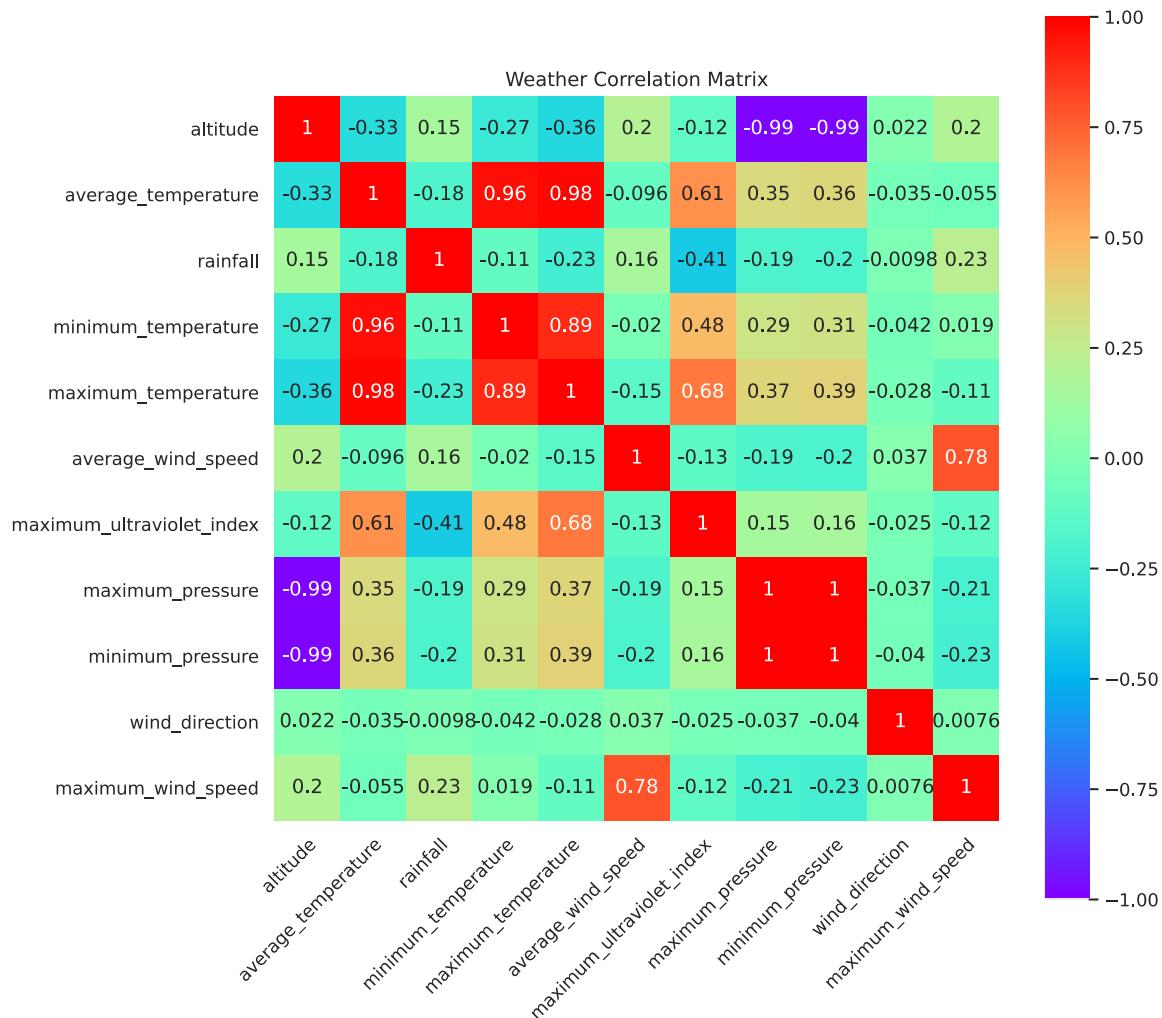


Ilustración 8: Matriz de correlación del clima.

En esta matriz de la climatología de la figura 8, se observa la correlación de las propias variables del clima entre sí, casi todas tienen algo que ver entre ellas. Por ejemplo vemos que la temperatura media (average temperature) tiene una correlación del 0.96 y 0.98 con la temperatura mínima y máxima del día respectivamente, esto no es ninguna sor-

Descripción informática

presa y tampoco es un dato que nos aporte nada, porque en la propia temperatura media se incluyen datos de estas variables, así que vamos a observar las que no son de este tipo.

Vamos a destacar los valores más significativos, la correlación entre las **temperaturas máximas y el máximo índice ultravioleta** registrado en el mismo día. Esto tiene relación puesto que el índice ultravioleta [UV] no es otra cosa que un indicador de la intensidad de radiación ultravioleta proveniente del Sol en la superficie terrestre, así que tiene que ver con las temperaturas.

También encontramos correlación negativa entre el **máximo índice ultravioleta** y las **lluvias**, esto nos podría indicar que a mayor cantidad de lluvias, menor índice ultravioleta y viceversa, si investigamos más sobre este suceso, nos damos cuenta de que en realidad esta correlación tiene más que ver con las propias nubes que provocan la lluvia que con la lluvia en sí, aunque la intensidad de los rayos UV es mayor cuando no hay nubes, puede ser alta incluso cuando está nublado.

Otro dato significativo es el de la **altitud**, existe correlación negativa de esta con todas las variables que tienen que ver con la **temperatura**, esto es debido a que a mayor altitud, más bajas son las temperaturas, no es lo mismo la temperatura mínima en la sierra de Madrid que en un entorno urbano más masificado y con menos árboles como podría ser en la Puerta del Sol. Además, tiene una fuerte correlación negativa también con las **presiones mínimas y presiones máximas**, ya que las fluctuaciones de la presión atmosférica presentes en diferentes puntos del planeta dependen de la altitud y la temperatura. Cuanto mayor sea la altitud, menor será la presión, mientras que cuanto menor sea la altitud y más cerca del nivel del mar, mayor será la [presión].

A continuación vamos a estudiar la correlación que existe entre las variables de calidad del aire con ellas mismas, de esta manera estudiaremos las posibles causas de que haya un aumento de ciertos contaminantes cuando otros aumentan y/o disminuyen.

Análisis exploratorio de datos (EDA)

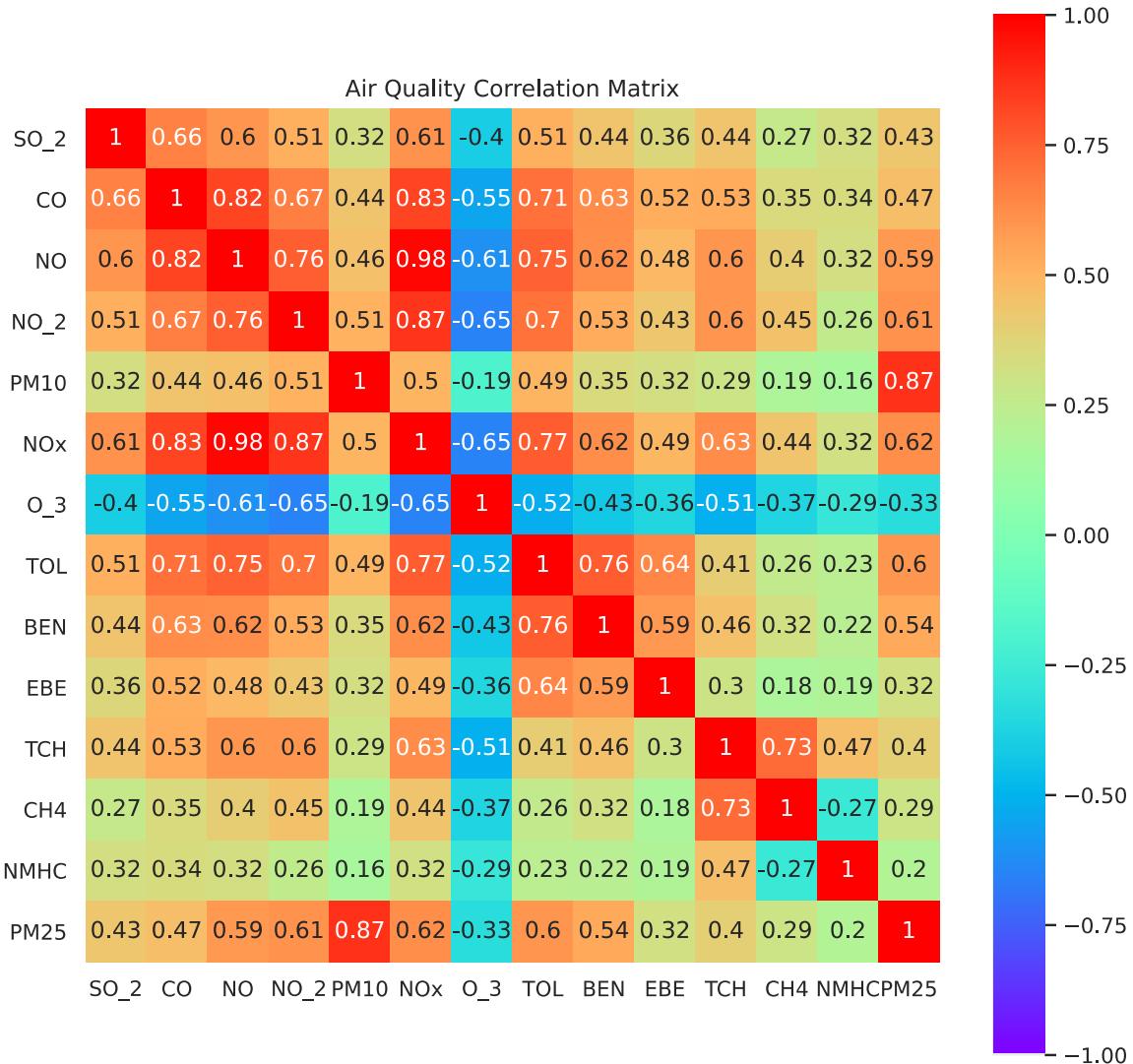


Ilustración 9: Matriz de correlación de la calidad del aire.

Podemos apreciar a simple vista un patrón de colores con el O₃, mantiene correlaciones negativas con el resto de contaminantes, así, esto será sujeto de estudio en los siguientes párrafos. También encontramos correlaciones positivas de otros contaminantes en comparación con el resto, como en el caso del NO_x, este contaminante aúna todos los óxidos de nitrógeno en una misma medida.

Todas estas relaciones nos serán de utilidad para que el futuro modelo de inteligencia artificial aprenda y cree nuevos caminos en sus redes neuronales.

Descripción informática

A diferencia del [ozono] estratosférico, que se forma de forma natural en la atmósfera superior y nos protege de los daños rayos ultravioleta del sol, el ozono a nivel del suelo (o troposférico) se crea a través de las interacciones de las emisiones artificiales (y naturales) de compuestos orgánicos volátiles y óxidos de nitrógeno en presencia del calor y la luz solar.

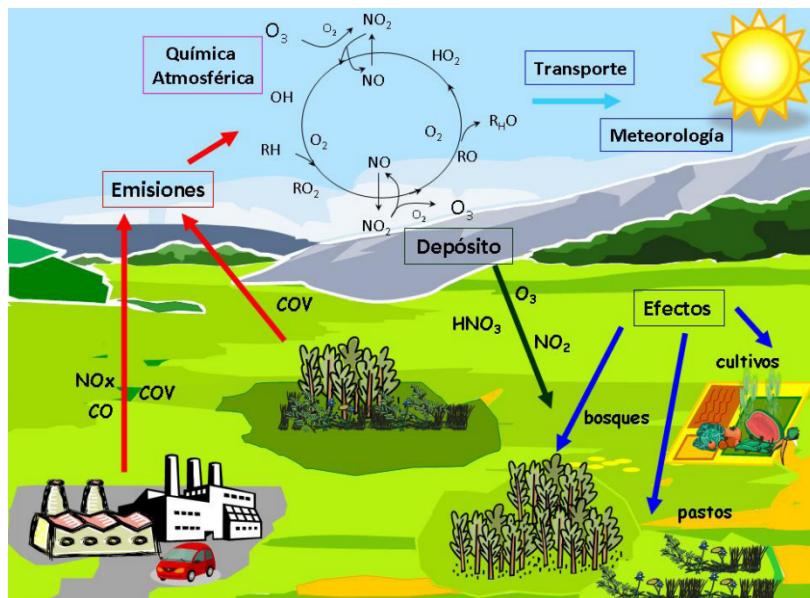


Ilustración 10: Ciclo simplificado del ozono y procesos relacionados.

También se puede apreciar una fuerte correlación entre la materia particulada de menos de 2.5 micras y las de menos de 10 micras, las partículas “finas” se definen como partículas de 2,5 micras o menos de diámetro (PM2. 5). Por lo tanto, las PM2. 5 comprenden una parte de las PM10, por ello la correlación de 0.87.

Para poder analizar las relaciones de todos los contaminantes se necesitaría de un experto en química. Para nuestro caso de uso estas relaciones explicadas nos son suficientes para tener una idea de cómo se comportan y generan estos contaminantes.

Una vez vistas las relaciones entre los contaminantes y las relaciones de la climatología, nos preguntamos si entre ellas tendrán relación. Como hemos visto antes en este proyecto, tienen bastante relación, es por ello que vamos a desglosar las correlaciones y vamos a proceder a estudiarlas a fondo.

Análisis exploratorio de datos (EDA)

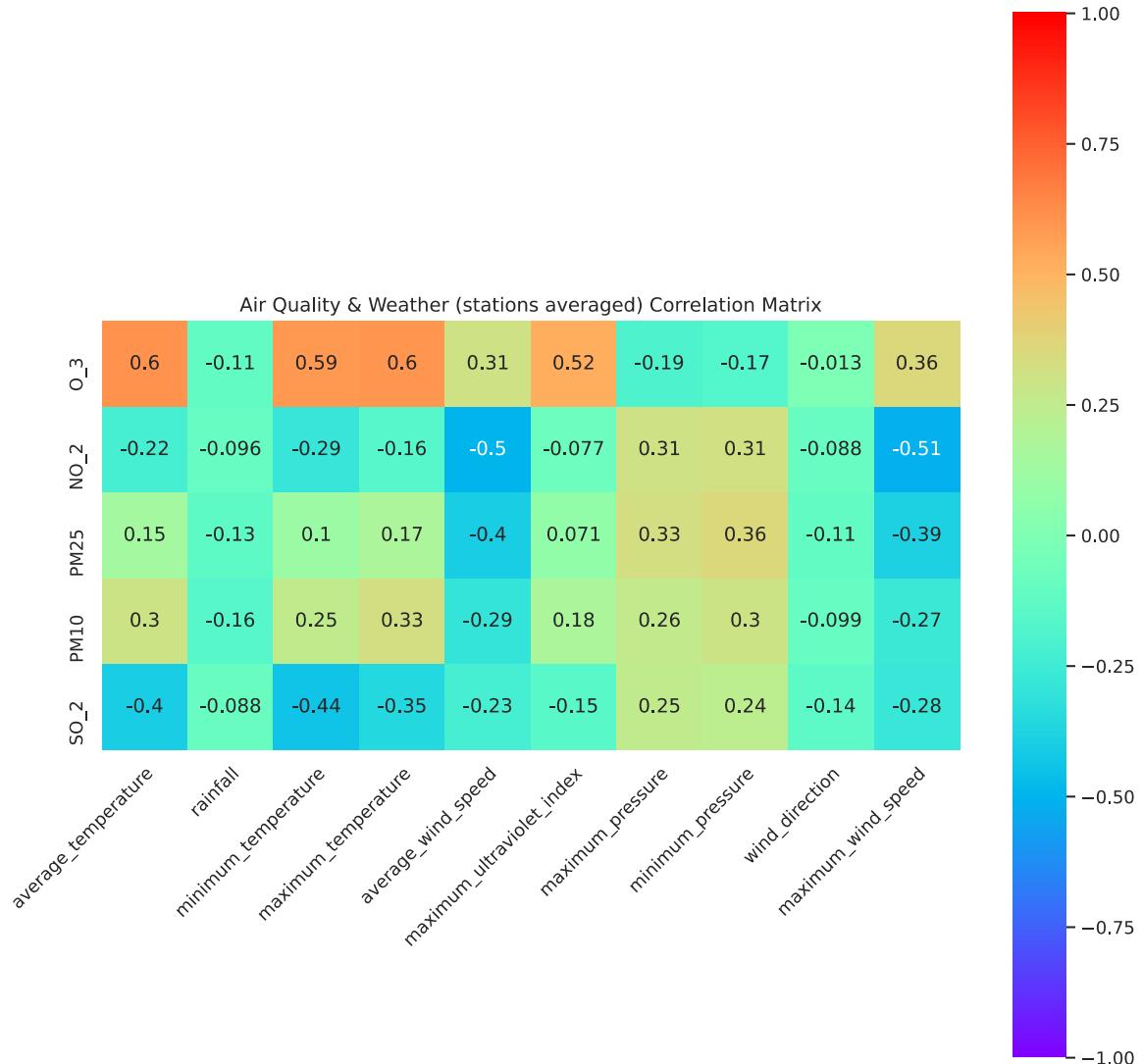


Ilustración 11: Matriz de correlación de la calidad del aire con la climatología.

Debido a la gran cantidad de datos disponibles de la calidad del aire, hemos escogido los 5 contaminantes que resultan más dañinos para la salud y que son los más significativos según nos indica la guía de la Organización Mundial de la Salud [WHO].

Como se puede ver en la matriz de correlación, hay varios contaminantes que tienen una correlación de al menos un 50% con distintas medidas del clima, como puede ser las temperaturas y el viento.

Descripción informática

Una vez vistas las correlaciones de los datos vamos a intentar darles significado mediante unos gráficos de dispersión. Para estas visualizaciones hemos utilizado la librería de [Plotly Express] para Python, se trata de un módulo de alto nivel, construido sobre [Matplotlib], que nos permite crear ricas visualizaciones interactivas, incluye funciones para los gráficos más utilizados en este sector.

Se puede apreciar una fuerte correlación entre las temperaturas e índice ultravioleta con el O₃, como hemos referenciado anteriormente, las altas temperaturas y los rayos del sol hacen que se propicien reacciones químicas del [ozono] con el resto de contaminantes.

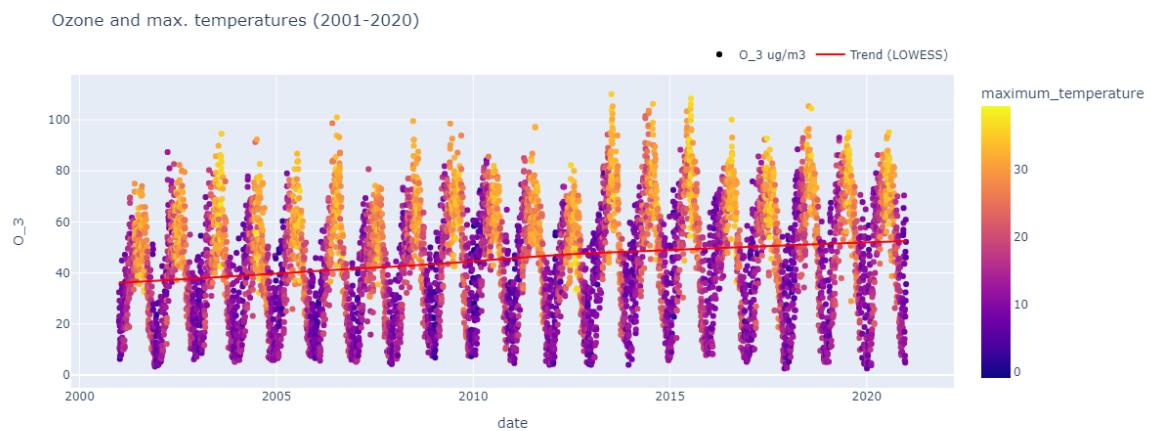


Ilustración 12: Ozono y temperaturas máximas (2001-2020)

Como vemos en la ilustración 12, el Ozono es mayor cuanto más temperatura hace, las curvas se presentan en los meses más calurosos del año, en verano. Existe una tendencia alcista de contaminación del ozono desde el 2001 hasta el 2020.

Análisis exploratorio de datos (EDA)

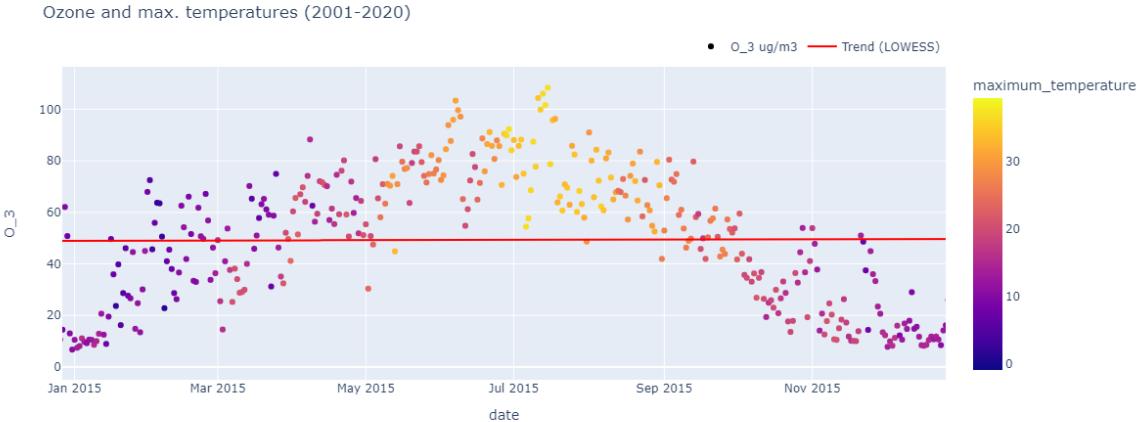


Ilustración 13: Ozono y temperaturas máximas (2015).

Desde mayo empieza a incrementar el ozono significativamente hasta el mes de septiembre, esto coincide con el final de la primavera e inicio de la temporada de verano como se puede apreciar en la ilustración 13.

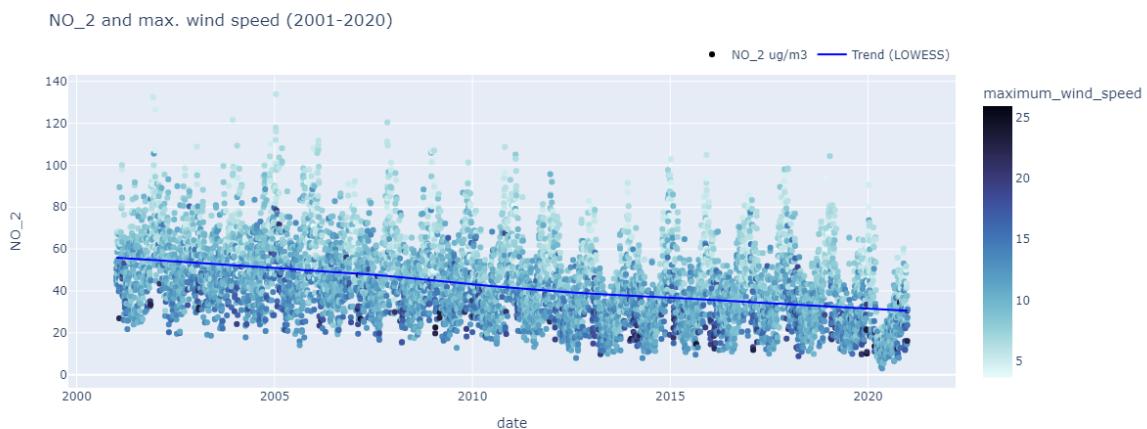


Ilustración 14: Dióxido de nitrógeno y velocidades máximas de viento (2001-2020).

Encontramos una fuerte relación entre las velocidades máximas del viento con las concentraciones del dióxido de nitrógeno (NO_2), este contaminante es generado por la acción del ser humano, la alta densidad del tráfico, el viento y las diferencias de temperatura entre distintas capas del aire. Este gas irritante es un contaminante oxidante perteneciente a los [óxidos nitrosos], sirve como precursor de otras reacciones químicas que provoca que

Descripción informática

aumente el efecto de otros contaminantes. El viento toma parte en las capas del aire donde se mezcla el dióxido de nitrógeno, es por eso que palia parcialmente sus efectos.

Sigue una tendencia descendente con el paso de los años, lo que significan buenas noticias para los ciudadanos madrileños, esta tendencia está generalizada a todo el mundo y se suele concentrar más en focos localizados, como episodios de alta contaminación por alta densidad de tráfico y aglomeraciones.

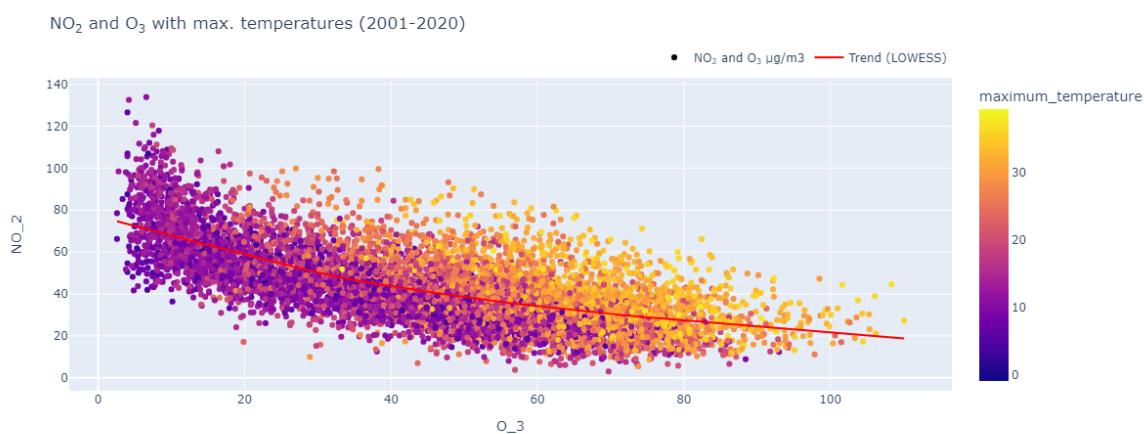


Ilustración 15: Dióxido de nitrógeno, ozono y temperaturas máximas (2001-2020).

En la ilustración 15 queda reflejado el proceso químico que hemos descrito en anteriores párrafos de la intervención del [ozono] troposférico con los [óxidos nitrosos], la mezcla de estos contaminantes sumado a las altas temperaturas son la combinación ideal para desarrollar más contaminantes.

Los días de máximas temperaturas se llega a unos niveles muy altos de ozono, también ocurre a la inversa pero en menor medida con el dióxido de nitrógeno, los días más fríos son los que presentan máximas de este contaminante y menos del ozono (O₃).

7.3.2. Valores nulos

Como vimos en el apartado de Recogida de datos, no todas las estaciones existen desde el primer año de nuestras medidas (2001), ni todas las estaciones miden los mismos contaminantes, incluso hay estaciones que empezaron midiendo unos contaminantes y han terminado tomando medidas de otros distintos. Es por esto que necesitamos alguna forma de visualizar los datos que tenemos y que faltan para cada una de las fechas de nuestro proyecto. Utilizaremos el módulo [missingno], este nos permite obtener un rápido resumen visual de la integridad de nuestro conjunto de datos.

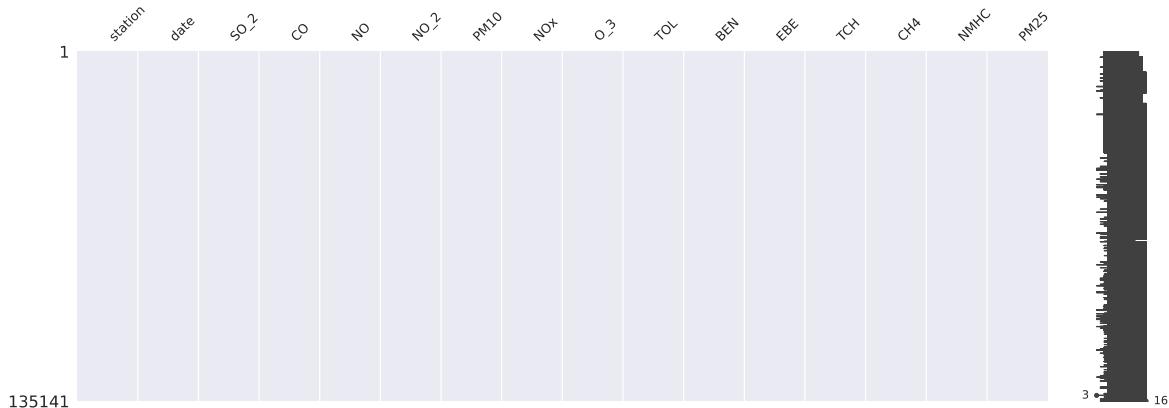


Ilustración 16: Valores nulos para la calidad del aire (2001-2020).

La figura 16 es una representación gráfica de los valores nulos para la serie de datos de calidad del aire, para todas las estaciones y para cada fecha existen algunos huecos para ciertos contaminantes, los que están marcados en oscuro quiere decir que existe el dato para ese preciso instante, los huecos indican la falta de datos.

Los contaminantes de los que tenemos más datos son los óxidos de nitrógeno, tanto NO, NO₂ y NO_x. Seguidos de el dióxido de azufre (SO₂), monóxido de carbono (CO), partículas finas de menos de 10μm (PM₁₀) y Ozono (O₃). El resto de contaminantes tienen bastantes huecos porque son medidos por una menor cantidad de estaciones, esto puede suponer dificultades a la hora de realizar estadísticas o predicciones con estos datos. Con estos datos también podemos ver qué contaminantes son más interesantes para los expertos en calidad del aire y cuáles podríamos priorizar para nuestro proyecto.

Descripción informática

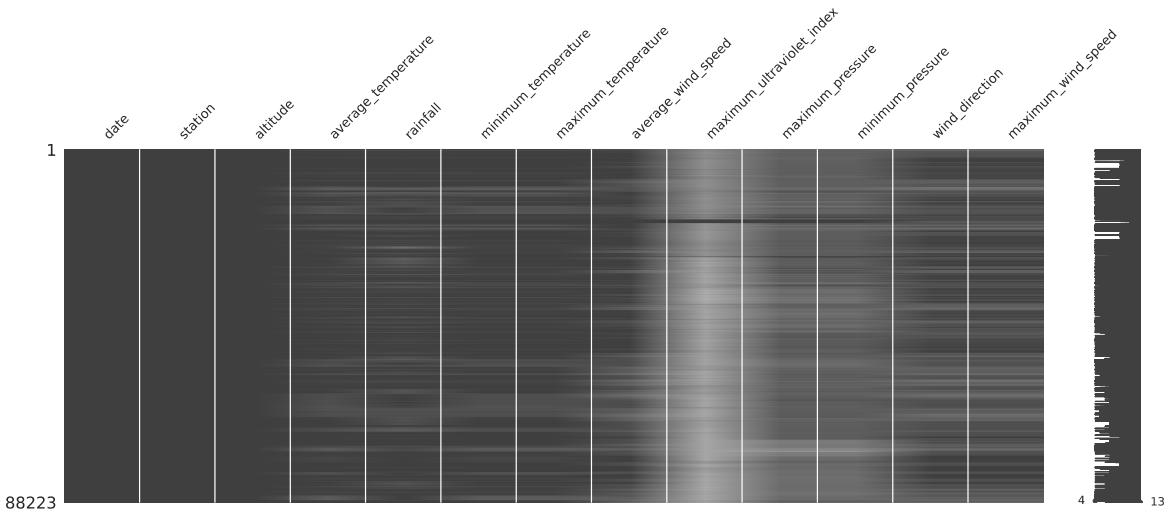


Ilustración 17: Valores nulos para la climatología (2001-2020).

A diferencia de los valores que faltan en la calidad del aire, para el caso de los datos del clima nos encontramos con que tenemos casi la totalidad del dataset, solo faltan datos para algunos días de pocas estaciones y no suelen coincidir, por lo que no tenemos que hacer prácticamente nada de imputaciones de valores nulos.

La Agencia Estatal de Meteorología [AEMET] lleva recopilando datos de la climatología de toda España desde el año 1911, y digitalizado los datos de estudio del clima de la historia más reciente del país. Las estaciones de calidad del aire son mucho más recientes y han sufrido de muchos más cambios en los últimos años, por ello creemos que la consistencia de los datos meteorológicos es más adecuada para este tipo de problemas.

7.3.3. Índices de calidad del aire

Son múltiples los estudios de contaminación del aire que se han llevado a cabo en este siglo, casi todos los países desarrollados cuentan con equipamientos y sistemas especializados en la realización de estas medidas. Pero a la hora de compartir estos datos de forma internacional o nacional, se hace difícil saber con exactitud qué cantidad de cada contaminante es más o menos nociva, comparar las distintas ciudades de un país o incluso comparar los países de la Unión Europea. La interpretabilidad de estos datos también se

Análisis exploratorio de datos (EDA)

hace muy compleja para los ciudadanos de a pie, que también son los afectados por las medidas que tomen sus gobernantes para paliar los efectos de la contaminación.

Es por todo esto que se hace necesaria una medida estandarizada para llevar un control de la contaminación. Las medidas de contaminación, desgraciadamente, no son universales, por temas políticos o de diferencias culturales no se ha llegado a un acuerdo con todos los países del planeta para establecer una única medida unificada de la contaminación, por ello, se han incrementado los esfuerzos en ciertas organizaciones de países como la Unión Europea, mundiales como la Organización Mundial de la Salud, incluso nacionales y múltiples estudios científicos de contaminación internacionales para poder establecer limitaciones y reglas para mejorar la calidad del aire de todos los ciudadanos.

Debido a la diversidad de índices de calidad del aire y la disparidad de fórmulas que se han creado por el mundo, para este proyecto hemos valorado varios índices de calidad del aire con los que ampliar el estudio de mi problema, en los que destaco los más comúnmente utilizados, el Air Quality Index (AQI) de la Agencia de Protección Ambiental de los Estados Unidos de América [EPA], el European Air Quality Index de la Agencia Ambiental Europea [EEA] o el Indice Nacional de Calidad del Aire [ICA] del Ministerio para la Transición Ecológica y el Reto Demográfico español.

Descripción informática

Pollutant	Index level (based on pollutant concentrations in µg/m ³)					
	Good	Fair	Moderate	Poor	Very poor	Extremely poor
Particles less than 2.5 µm (PM _{2.5})	0-10	10-20	20-25	25-50	50-75	75-800
Particles less than 10 µm (PM ₁₀)	0-20	20-40	40-50	50-100	100-150	150-1200
Nitrogen dioxide (NO ₂)	0-40	40-90	90-120	120-230	230-340	340-1000
Ozone (O ₃)	0-50	50-100	100-130	130-240	240-380	380-800
Sulphur dioxide (SO ₂)	0-100	100-200	200-350	350-500	500-750	750-1250

Ilustración 18: Índice de Calidad del Aire Europeo.

Cada uno de los índices tiene rangos distintos de calidad, esto hace difícil tener una consistencia con los datos por lo que finalmente vamos a utilizar el índice de calidad del aire de la Agencia Ambiental Europea, en un intento de tener unos rangos de contaminación iguales a otros países de nuestro mismo continente.

Este índice se divide en 6 categorías de peor a mejor calidad del aire como vemos en la figura 18, con que haya uno de los contaminantes en ese rango, se coge el de peor categoría como referencia, es decir, si tenemos por ejemplo un día en el que todos los contaminantes están en la categoría de “Good” pero hay 1 solo que está en la categoría de “Moderate”, automáticamente se marca ese punto con “Moderate” para todo ese rango de tiempo, lo que permite ensalzar los malos datos.

Análisis exploratorio de datos (EDA)

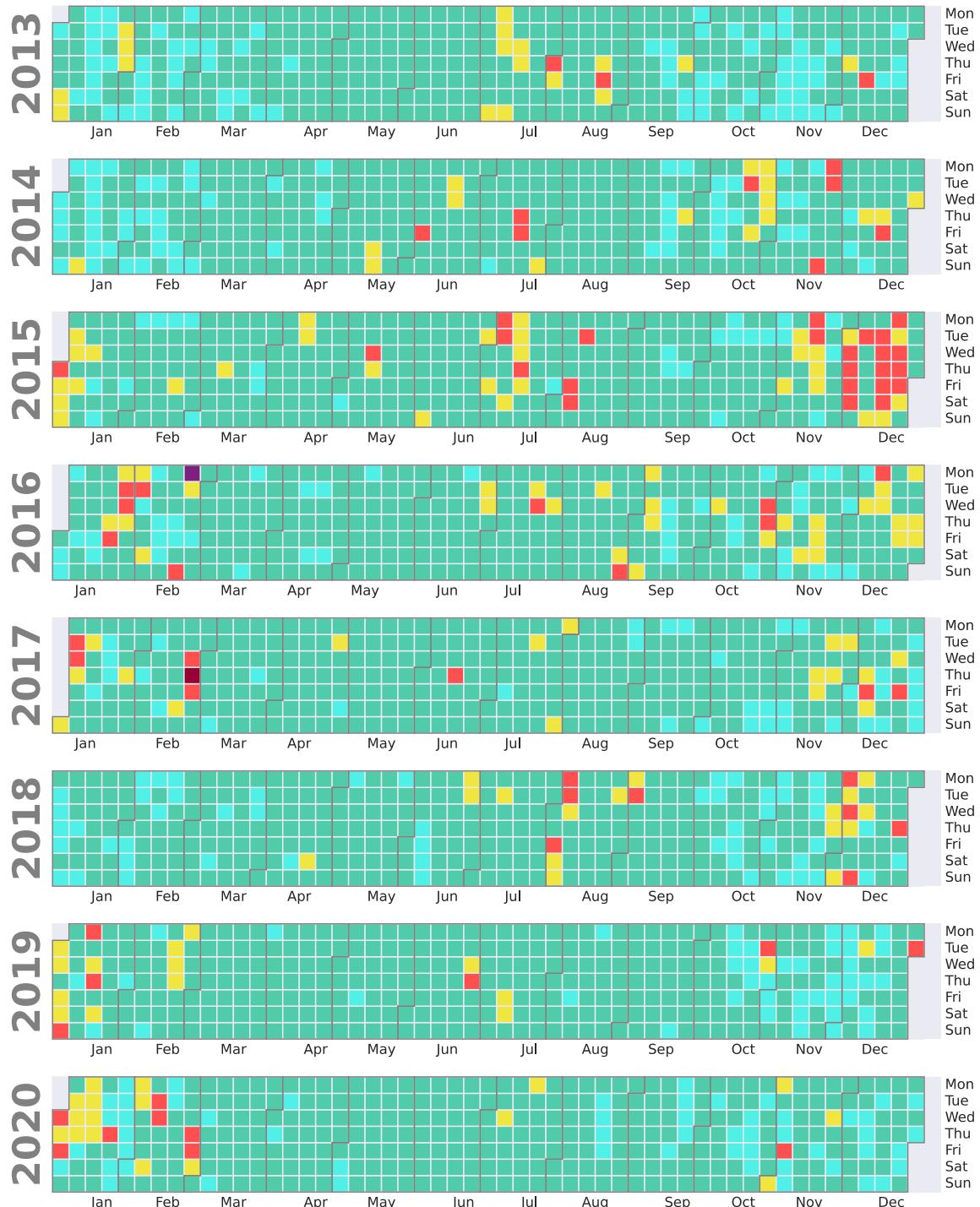


Ilustración 19: Índice de Calidad del Aire (2013-2020).

Una vez tenemos claro los rangos y las medidas para calcular este índice de calidad del aire, se procede a aplicar estas reglas en nuestros datos diarios y a representarlo de manera gráfica con el patrón de colores marcado para cada categoría, esto es posible

Descripción informática

utilizando [Matplotlib] y una librería especializada en la creación de calendarios a partir de Dataframes de [Pandas] llamada [calplot], como vemos en la figura 19. Para los datos de los últimos 7 años, observamos como la mayor parte del tiempo la calidad es “Fair”, o “justa” en español, es el término medio entre buena y moderada.

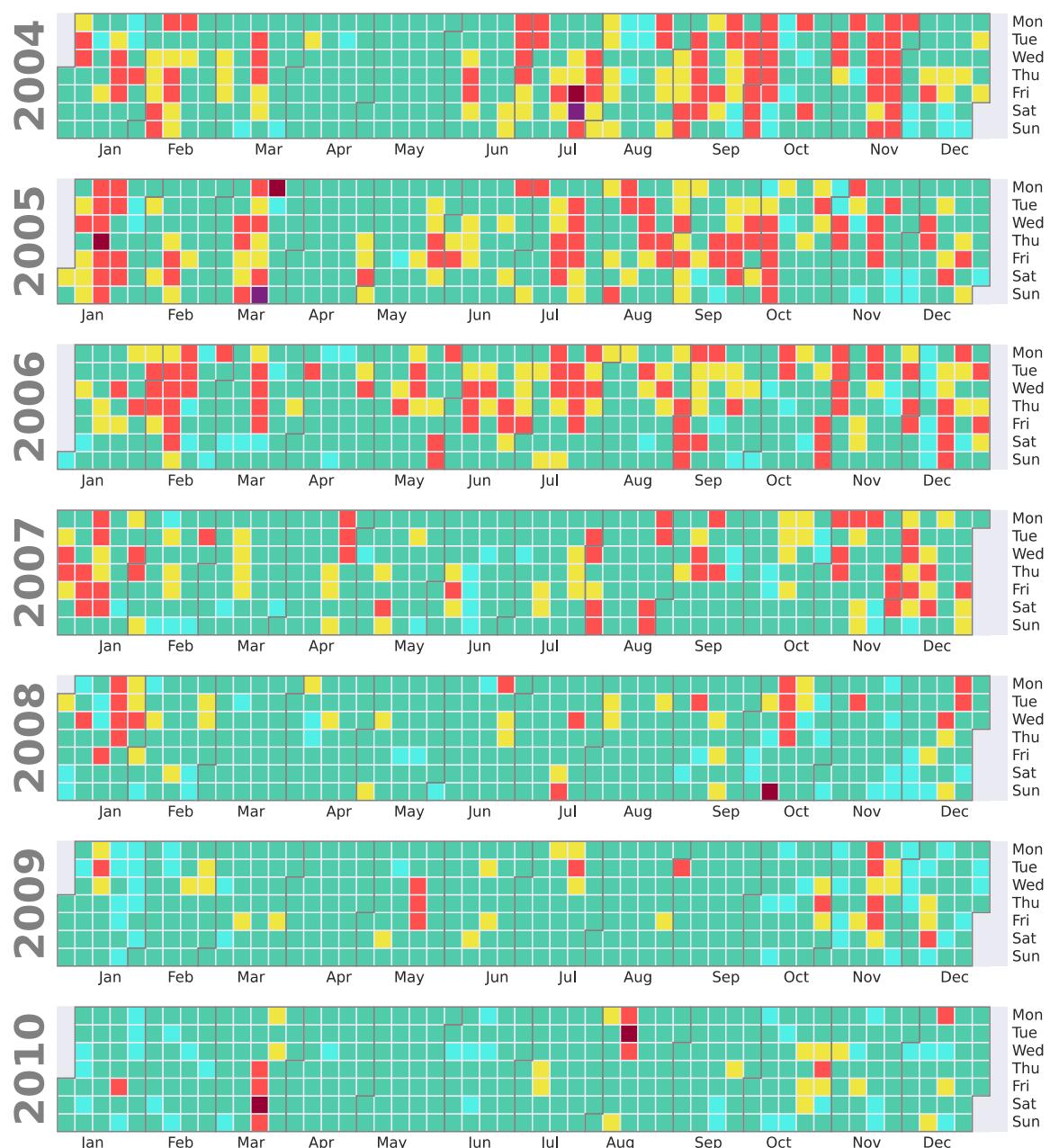


Ilustración 20: Índice de Calidad del Aire (2004-2010).

Esta representación en forma de calendario nos permite analizar en profundidad eventos concretos en el tiempo, como por ejemplo, en diciembre de 2015 vemos un aumento

Análisis exploratorio de datos (EDA)

to de contaminación que dura prácticamente todo el mes. Si investigamos este hecho, en los datos del clima, según la AEMET se registró un [anticiclón], lo que provocó la bajada de algunas capas atmosféricas, favoreciendo así la contaminación a nivel del suelo.

También se ha mencionado anteriormente los cambios de estaciones de calidad del aire y cambio en las medidas de los contaminantes, podemos verlo representado en la figura 20, si nos fijamos, a simple vista podemos contemplar una gran mejora de la calidad del aire, sobretodo en los años 2009 y 2010, se producen hasta 13 bajas de estaciones, lo que puede provocar este cambio repentino de los datos. Esto resulta muy problemático pues no sabemos a ciencia cierta si mejoró la contaminación por algún tipo de medida o fue única y exclusivamente debido a estos cambios de estaciones (y de contaminantes que miden) a unas ubicaciones con mejor calidad del aire.

7.4. Ingeniería de características (o de datos)

La ingeniería de características es el proceso de selección, manipulación y transformación de los datos brutos en características que puedan utilizarse en el aprendizaje supervisado. Para que el aprendizaje automático funcione bien en nuevas tareas, puede ser necesario diseñar y entrenar mejores características. La ingeniería de características, en términos sencillos, es el acto de convertir las observaciones en bruto en características deseadas utilizando enfoques estadísticos o de aprendizaje automático.

Este proceso es necesario cuando se trabaja con modelos de aprendizaje automático. Una buena ingeniería de características requiere de expertos con el contexto adecuado de la aplicación, las fuentes de datos y las formas en que los datos han sido procesados y gestionados.

Tras la adaptación de los datos a nuestro propio formato, más propicio para este tipo de problemas, nos queda intentar mejorar, añadir, eliminar o incluso modificar algunos datos directamente.

El primer problema que vamos a abordar es el de la diferencia de ubicaciones entre las estaciones de calidad del aire y las estaciones meteorológicas, lo ideal sería que tuviéra-

Descripción informática

mos la climatología de las mismas ubicaciones exactas que las estaciones de calidad del aire. La solución más directa a este problema sería directamente utilizar para cada estación de calidad del aire, los datos de climatología más cercanos por distancia, intentando así aproximar esos datos, ya que probablemente sea más fácil que sean similares. Pero esto al final es un proceso un poco inexacto, ya que contamos con distintas altitudes para los datos y distan de muchos kilómetros algunas.

Finalmente tras un largo estudio de alternativas, se ha decidido interpolar los datos climatológicos en cada estación de los contaminantes, estableciendo así una relación 1:1, teniendo cada estación por cada fecha, datos de polución y del clima de ese mismo punto, facilitando mucho los cálculos e intentando que nuestro futuro modelo tenga mejores datos.

El estudio de la interpolación de datos es un gran tema de estudio, sobretodo en el estudio de la geografía, y más en concreto en el análisis de datos geoespaciales como los que tenemos en nuestro proyecto. Hay múltiples maneras de interpolar los datos y algunos son más o menos adecuados para coordenadas geográficas.

Primeramente necesitamos obtener las coordenadas de todas las estaciones. Las estaciones de climatología se encuentran en el formato de grados, minutos y segundos, para facilitar nuestros cálculos lo convertimos a grados decimales, las esta-

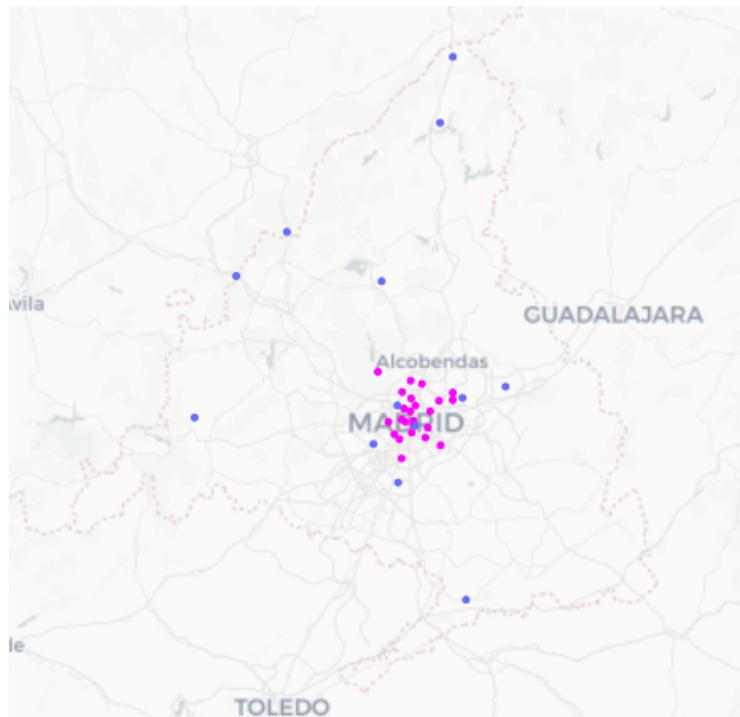


Ilustración 21: Mapa de las estaciones de calidad del aire (color rosa) y de climatología (color azul).

Ingeniería de características (o de datos)

ciones de calidad del aire se encuentran ya en este último formato así que en ese aspecto no tenemos más que hacer. Hemos plasmado en un mapa todas las estaciones para hacernos una mejor idea de cómo están distribuidas todas las estaciones, esto ha sido posible gracias a la librería [Plotly Express] mencionada anteriormente en el Análisis exploratorio de datos (EDA).

Para poder interpolar los datos, necesitamos calcular las distancias de cada una de las estaciones de climatología con las de calidad del aire y viceversa, para esta tarea vamos a construir una matriz de distancias, que contiene todos estos datos. Para hacer el cálculo de las distancias vamos a utilizar la distancia [Minkowski], se trata de una fórmula matemática que generaliza el cálculo de la distancia euclídea y de la distancia Manhattan, para nuestro caso nos es más interesante calcular la distancia Euclídea, finalmente obtenemos las distancias ordinarias entre dos puntos del espacio.

Utilizaremos únicamente la latitud y la longitud para los cálculos de las distancias euclidianas, ya que las diferencias de altitud entre las estaciones no son tan grandes como para que supongan una diferencia significativa en las distancias y también simplifica los cálculos, con esto ya tenemos todo lo necesario para iniciar la interpolación.

Los métodos de [interpolación] valorados han sido los siguientes: Kriging, distancia inversa ponderada, función de base radial (ej. gaussiana, spline) e interpolación polinómica. Para poder decidirnos por una u otra hemos tenido que recurrir a publicaciones científicas de expertos, con un enfoque en climatología por este tipo de datos en concreto. Muchas de estas funciones mencionadas anteriormente son bastante complejas de implementar en programación desde cero, no existe demasiada información y la poca que hay son para expertos en geografía o para informes del clima. Finalmente valorando todas estos factores hemos decidido utilizar la distancia inversa ponderada, esta función además nos permite modular cuánta importancia le damos a las estaciones que estén más cercanas comparadas con las que estén más alejadas y podemos tener en cuenta todas las estaciones para realizar estos cálculos y obtener una interpolación lo más acertada posible.

Una vez obtenidas todos los nuevos datos de las interpolaciones vamos a combinar ambos datasets sobre las fechas y sobre las estaciones de calidad del aire, obteniendo defi-

Descripción informática

nitivamente un formato unificado para todos los datos. Teniendo para cada estación unos únicos datos de contaminación y de clima, simplificando mucho el proceso para nuestros futuros modelos de inteligencia artificial. Todos estos cálculos son facilitados por la librería [SciPy] para Python, aúna en sus módulos los algoritmos más utilizados en distintos ámbitos científicos, optimización, integración, interpolación y muchas más, ahorrando al usuario tener que implementar dese cero todos estos.

Así quedarían las columnas del nuevo dataset:

station, date, SO_2, CO, NO, NO_2, PM25, PM10, NOx, O_3, TOL, BEN, EBE, TCH, CH4, NMHC, average_temperature, rainfall, minimum_temperature, maximum_temperature, wind_direction, average_wind_speed, maximum_wind_speed, maximum_ultraviolet_index, maximum_pressure, minimum_pressure

7.5. Modelización, entrenamientos, evaluaciones y predicciones

Para este problema de inteligencia artificial tenemos que crear algún modelo que pueda predecir los datos de nuestro problema algunos días en el futuro, para ello vamos a necesitar establecer algunas reglas.

Primero vamos a establecer que el modelo debe predecir los datos de calidad del aire para una **ventana retrospectiva de 14 días y una previsión de 1 día** utilizando los datos de entrenamiento. Esto quiere decir que con 2 semanas de datos nuestro modelo podría intentar predecir el día siguiente (día 15) con suficiente exactitud.

De todas las columnas que tenemos, debemos predecir las de contaminación, aún así tendríamos que predecir más de 10 tipos distintos de contaminantes, esto podría ser una limitación muy grande debido a la gran cantidad de datos que tenemos y a las limitaciones computacionales como veremos a continuación. Es por todo esto que hemos decidido predecir únicamente los elementos contaminantes que a nuestro criterio y según hemos visto en el análisis exploratorio de datos, tienen más impacto para el cálculo del índice de la calidad del aire y sobretodo para la ciudadanía, quedándonos así con los siguientes: **PM_{2.5}, PM₁₀, NO₂, SO₂ y O₃**.

Para las predicciones hemos decidido prescindir de los datos del año 2020 ya que la pandemia del COVID-19 ha provocado una disminución significativa de los contaminantes por las [restricciones] de movilidad con el estado de alarma, ya que el tráfico rodado es la principal causa.

Existen multitud de maneras de abordar este problema, a la hora de elegir un modelo nos hemos decantado por la rama del deep learning y las redes neuronales por varias razones, la primera de ellas porque para los problemas de series temporales multivariable, los mejores modelos predictivos se encuentran ahí, teniendo cada ciertos años un nuevo paradigma y mejora del estado del arte para estos problemas, segundo, para tener en cuenta tantas variables las redes neuronales son especialmente buenas en analizar patrones y aprenderlos. Por último tenemos que tener en cuenta que al no ser expertos en esta materia, nos es complicado tener un análisis totalmente formado de nuestros datos, la complejidad de la química de la atmósfera y la troposfera, junto con la climatología, abarca un gran espectro de conocimientos, es por ello que a la hora de realizar estadísticas e Ingeniería de características (o de datos) nos vemos muy limitados a pesar de las investigaciones que hemos realizado, aquí es donde las redes neuronales también juegan un papel clave, ya que son perfectas para ello, en los entresijos de estas redes profundas se crean relaciones y multiplicaciones de todos los datos entre sí, lo que nos puede ayudar a encontrar estas nuevas variables para llevar a cabo mejores predicciones.

En concreto vamos a estar utilizando un tipo de redes neuronales recurrentes, estas están especializadas en datos secuenciales, como los que tenemos nosotros en las series temporales, son capaces de aprender las dependencias temporales de nuestros datos y así determinar una predicción en nuestro caso, pero también sirven para clasificación, regresión, etc.

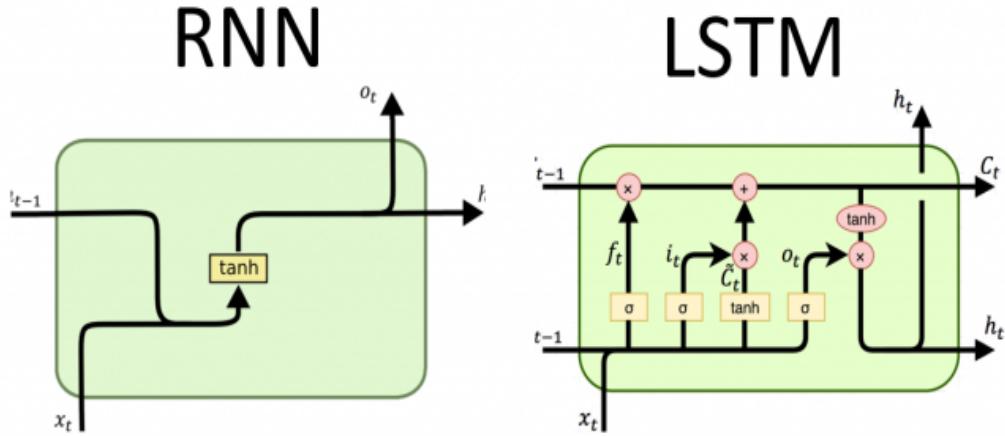


Ilustración 22: Una célula de una Red Neuronal Recurrente (RNN) y una Long Short-Term Memory (LSTM).

De las redes neuronales recurrentes, de ahora en adelante RNN, se encuentra un tipo de red llamada Long-Short Term Memory. Como su propio nombre indican son capaces de retener información tanto a corto como largo plazo, este era uno de los problemas principales de los modelos, no eran capaces de retener correctamente la información más antigua y se iba desvaneciendo el aprendizaje a largo plazo, aprendiendo de más la mayoría de los casos con la información más reciente.

La ventaja de la red de LSTM sobre otras redes recurrentes que datan de 1997, vino de un método mejorado de [propagación] hacia atrás del error. El problema que tenían era la explosión y desvanecimiento de los gradientes que se produce en el paso de propagación hacia atrás.

Las redes LSTM funcionan con distintas compuertas como vemos en la figura 22 en color naranja, vamos a explicarlas:

1. Compuerta de olvido (f): Decide qué información olvidar del estado de la célula utilizando una función σ que modula la información entre 0 y 1. Olvida todo lo que es 0, recuerda todo lo que es 1 y todo lo que está por el medio son posibles candidatos.

2. Compuerta de entrada o de memoria (i): Decide cuáles de los nuevos candidatos son relevantes para este paso de tiempo también con la ayuda de una función σ .
3. Compuerta de salida (o): Crea un nuevo conjunto de candidatos que se almacenan en el estado de la célula. La relevancia de estos nuevos candidatos será modulada por la multiplicación con los elementos de la capa de entrada. Determina qué partes del estado de la célula se emiten. El estado de la célula se normaliza mediante una función tanh y se multiplica elemento a elemento por la puerta de salida que decide qué nuevo candidato relevante debe salir por el estado oculto.

Una vez tenemos claro cómo funciona nuestro modelo necesitamos preparar los datos, al tratarse de un modelo de aprendizaje supervisado tenemos que separar el dataset en datos de entrenamiento y datos de validación para evitar que memorice los datos provocando un sobre-ajuste, comúnmente llamado [overfitting], sino que aprenda a generalizar. Esto se consigue aprendiendo de los datos de entrenamiento y validando los resultados con los datos de validación para ver si está aprendiendo a resolver nuestro problema correctamente. Para este proyecto hemos dividido los datos de la siguiente manera:

- Datos de entrenamiento abarcan el intervalo [2001, 2015], lo que supone aproximadamente el 74% de los datos.
- Datos de validación abarcan el intervalo [2016, 2019], lo que supone aproximadamente el 26% de los datos.

Normalmente se suele utilizar un 80% de entrenamiento y 20% de validación pero al tratarse de series temporales anuales, hemos aproximado estas medidas tomando los datos de los años enteros desde enero a diciembre incluidos.

Necesitamos realizar un preprocesamiento de los datos, hay que acomodar los datos al formato de nuestra red LSTM, para ello primeramente vamos a normalizarlos, esto lo que hará será conseguir que todas las características del modelo tengan la misma importancia, utilizando MinMaxScaler de la librería [scikit-learn], escalando cada columna a un rango [0, 1]. Con esto evitaremos que la red neuronal cree sesgos que solo empeorarían

Descripción informática

nuestro modelo. Hay que tener siempre en cuenta que esta normalización se hace utilizando únicamente los datos de entrenamiento y luego la aplicamos a los de entrenamiento y, ahora sí, a los datos de validación, de esta manera evitas filtrar información de los datos de entrenamiento a la validación.

Se nos presenta un problema, si tenemos distintas estaciones, ¿Cómo seremos capaces de realizar predicciones para cada una de ellas?, esto es un tema muy extenso, hay muchas maneras de abordar este tipo de problemas como hemos mencionado anteriormente, lo ideal para este proyecto, por su simplicidad y por nuestra limitación computacional (para realizar estos entrenamientos necesitaremos GPUs para agilizar los cálculos), sería desarrollar un modelo único que sirva para predecir después individualmente para cada estación los datos de contaminación. Otras soluciones se expondrán en las Conclusiones y trabajos futuros.

id	color
1	red
2	blue
3	green
4	blue

One Hot Encoding

id	color_red	color_blue	color_green
1	1	0	0
2	0	1	0
3	0	0	1
4	0	1	0

Ilustración 23: Ejemplo de One-hot encoding para distintas categorías de colores.

Después de mucha investigación de cómo realizar esto de la mejor manera posible, hemos llegado a la conclusión de que debemos diferenciar cada estación utilizando una técnica de codificación llamada [One-hot] (encoding) como se muestra en el ejemplo de la figura 23. Para todas estas codificaciones y preprocessado de los datos hemos utilizado la librería de [scikit-learn], que además de ser una de los módulos de facto para algoritmos de machine learning, contiene una gran API de procesamiento de datos.

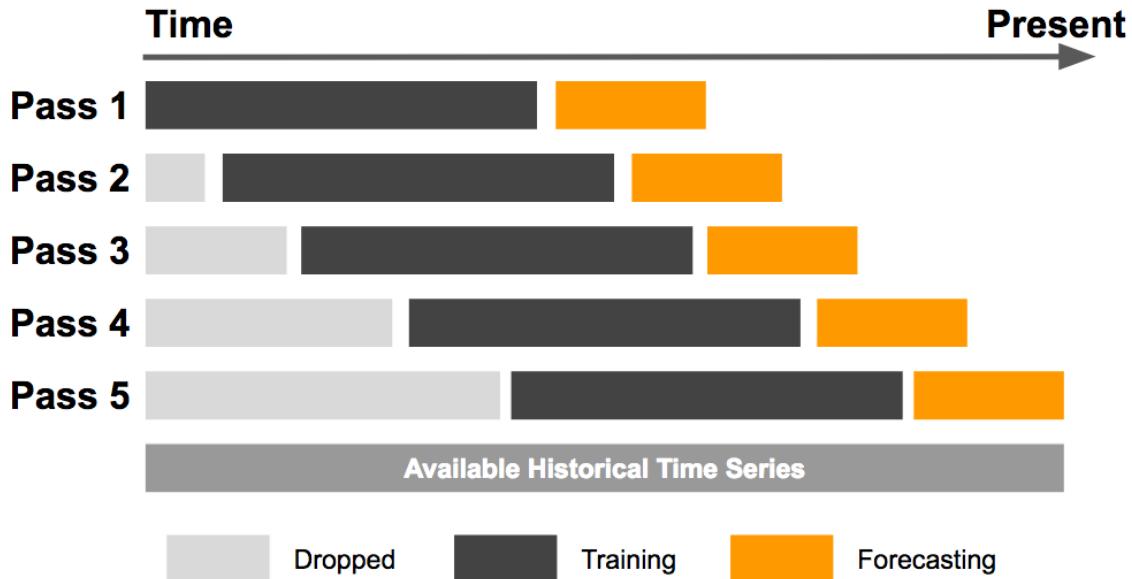


Ilustración 24: Esquema del método de ventana deslizante para series temporales.

También debemos que generar los datos para este problema de apredizaje supervisado, necesitamos crear lotes de datos de 14 días y de 1 día (el día a predecir, el día 15), para ello vamos a utilizar una técnica de ventana deslizante o [sliding window] como vemos en la figura 24. Para nuestro ejemplo cogeríamos el intervalo de días [1, 14] y se entrenaría para predecir el día número 15, después [2, 15] para predecir el 16 y así sucesivamente con todos los datos tanto de entrenamiento como de validación para todas las estaciones.

Estas técnicas han sido implementadas empleando la librería de [NumPy], especializada en vectores y cálculos matriciales que da soporte a una gran cantidad de datos, estamos ante la librería en la que se fundamentan las demás para realizar cálculos matemáticos, destaca por su velocidad y versatilidad, podemos verla siendo utilizada en [Pandas], [Matplotlib], [SciPy], [scikit-learn], etc.

Para introducir los lotes en la red, necesitamos usar específicamente un [formato LSTM] que se trata de un vector de tres dimensiones como vamos a ver a continuación:

1. Muestras. Una secuencia es una muestra. Un lote se compone de una o varias muestras.

Descripción informática

2. Pasos de tiempo. Un paso de tiempo es un punto de observación en la muestra. 14 días.
3. Características. Una característica es una observación en un paso de tiempo. Las columnas de nuestro dataset.

Los datos de entrada para nuestra red neuronal se compone de (muestras, pasos de tiempo, características) y además de un vector con los datos de salida de los contaminantes a predecir, que mencionamos al inicio del capítulo, este vector se trata del objetivo a predecir o la [variable target].

Por último, todo modelo de deep learning necesita de funciones que capacita al modelo para aprender. Necesitamos un método de optimización, como su propio nombre indica, se encarga de optimizar y actualizar los valores de los parámetros (pesos y sesgos) de nuestra red para reducir el error cometido con la [propagación] hacia atrás. Las funciones de activación tienen la tarea de decidir qué neuronas se activan en ciertos momentos y también determina cómo aprende el modelo, cuando es utilizada en la capa de salida (última) permite establecer el tipo de predicciones que esta red puede hacer,

Una función de pérdida, que establece una medida de lo bueno que es el modelo en realizar predicciones de nuestros resultados esperados. Tamaño de lote, comúnmente conocido como batch size, es el número de muestras que se van a entrenar antes de propagar el error hacia atrás, lo que podría aumentar el rendimiento si es ajustado correctamente. El learning rate o el porcentaje de aprendizaje, se utiliza en el entrenamiento de las redes neuronales y tiene un valor positivo, a menudo en el rango entre 0,0 y 1,0, sirve para determinar la cantidad de pesos que se actualizan durante el entrenamiento. Y para finalizar, el número de épocas que queremos entrenar nuestro modelo, puede variar dependiendo del problema o de la situación de la red neuronal.

8. Experimentos y validación

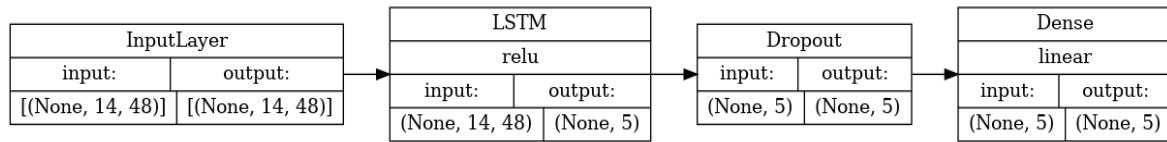


Ilustración 25: Arquitectura de nuestro modelo.

Como vemos en la figura 25, el modelo elegido se compone de las siguientes capas, totalmente conectadas cada una con la siguiente hasta el final. Todo el desarrollo de modelos ha sido íntegramente programado en Python utilizando el famoso ecosistema de deep learning, [TensorFlow], esta librería contiene un módulo de alto nivel llamado Keras. Utilizando la API de [Keras] se pueden crear modelos secuenciales a base de bloques (capas) con pocas líneas y nos permite iterar rápidamente en el desarrollo de los mismos.

1. Una primera capa de entrada que simplemente establece el formato y los datos que vamos a utilizar.
2. Capa LSTM con 50 neuronas, esta es la primera capa de entrada del modelo, se ha elegido el tamaño de 50 neuronas ya que nuestro problema tiene 25 columnas de datos de calidad del aire y del clima y después 23 columnas [One-hot] de las estaciones, 24 estaciones pero con esta técnica puedes representar una más (cuando todas están con valor 1), haciendo un total de 48 características. Estas neuronas adicionales sirven para facilitarle a la red más medios para el aprendizaje de esta gran cantidad de datos.
3. Capa de Dropout que sirve para evitar el [overfitting] inutilizando una parte de las neuronas aleatoriamente en cada entrenamiento.
4. Capa simple neuronal densa de 5 neuronas para predecir los 5 contaminantes (PM2.5, PM10, NO2, SO2 y O3). Esta sería la última capa y la salida que nos dé los resultados del modelo.

En total nos encontramos con 1,110 parámetros a entrenar para la arquitectura elegida. A continuación vamos a explicar los parámetros elegidos para nuestro modelo y el contexto de estos.

Experimentos y validación

Los distintos parámetros de los modelos se denominan hiperparámetros y no hay una única forma universal de elegirlos, por lo que tenemos que basarnos en investigación para determinar los mismos, y sobretodo en mucha prueba y error, a continuación vamos a presentar los que hemos elegido finalmente para nuestro modelo:

- Optimizador: **adam**. La función de optimización Adam es la versión extendida del descenso de gradiente estocástico, el nombre deriva de la estimación adaptativa de momentos, utiliza estimaciones del primer y segundo momento del gradiente para adaptar la tasa de aprendizaje de cada peso de la red neuronal. Utilizamos esta función porque generalmente suele dar mejores resultados que otros en la mayoría de los casos y es de las más rápidas computacionalmente hablando.
- Función de pérdida: **mae**. El Mean Absolute Error (MAE) o el error absoluto medio, mide la magnitud media de los errores en las predicciones, sin considerar su dirección. Es una forma de medir la precisión en variables continuas. Hemos elegido esta función de pérdida por su fácil interpretabilidad y simplicidad.
- Batch size: **32**. Este “tamaño de los lotes” se ha escogido por ser popularmente utilizado por los expertos en deep learning, nos encontramos muchas veces con este tamaño siendo usado de facto.
- Función de activación: **ReLU**. La Rectified Linear Unit o ReLU únicamente permite activar la neurona cuando el valor es mayor que 0 y si el valor es negativo, devuelve 0. Es una función de activación muy simple de calcular, puede devolver un valor 0 absoluto a diferencia de otras como tanh y además actúa como si se tratase de una función de activación lineal, lo que favorece mucho la optimización. Se trata de una de las más populares de los últimos años para entrenar redes neuronales muy profundas por todo lo mencionado anteriormente. A pesar de obtener muy buenos resultados con redes neuronales densas, suelen ser mucho peores si se usan en modelos LSTM o RNN, por lo que solamente lo hemos propuesto en la parte de nuestro modelo que tiene la red densa.

Experimentos y validación

- Dropout: **20%**. El dropout es una técnica de regularización que sirve principalmente para evitar el overfitting, desactiva neuronas de manera aleatoria durante el entrenamiento con el objetivo de que el modelo generalice mejor. Se ha elegido un 20% porque es la medida que se usa de facto en el campo del deep learning y porque no hemos sufrido de overfitting en nuestro caso.
- Número de neuronas LSTM: **50**. El número de neuronas debería aproximarse al número de los datos de entrada y de salida, es por esto que hemos elegido 50. Es una cuestión que tiene difícil respuesta clara pero con este valor de 50 nos aseguramos de no provocar cuellos de botella en el aprendizaje del modelo.
- Épocas de entrenamiento: **Early Stopping** con paciencia de 10 épocas. El Early Stopping es una forma de regularización, con la que se decide en qué época se debe dejar de entrenar el modelo. Hemos elegido una paciencia de 10 épocas, lo que quiere decir que vamos a dejar que entrene sin preocuparnos por el número de épocas, pero en el momento en el que se sucedan 10 épocas en las que el modelo para los datos de validación no mejore, para el entrenamiento y se queda con la mejor época. Es otra de las medidas que se toman para evitar el overfitting, que memorice los datos de entrenamiento y no generalice correctamente para los de validación, lo que provoca que la función de pérdida de resultados cada vez menores en los datos de entrenamiento y mayores en los de validación.

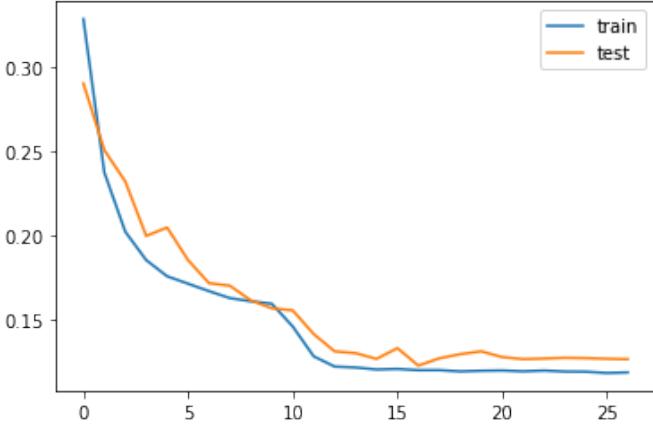


Ilustración 26: Visualización del entrenamiento del modelo.

Experimentos y validación

Como vemos en la figura 26, el modelo converge en torno a la época 12, encontrando una solución bastante óptima, a partir de ahí no se aprecian mejoras significativas. Tenemos que aclarar que el error observado en esta gráfica se corresponde con el de los datos normalizados mencionados en el apartado de Modelización, entrenamientos, evaluaciones y predicciones, por lo que tendremos que calcular el error real de los datos de validación una vez realizada la inversa de la normalización.

Una vez tenemos el modelo totalmente entrenado y revertida la normalización, vamos a ver qué resultados nos puede ofrecer y cómo de buena es esta solución para la predicción de la calidad del aire. Los errores medios absolutos para el modelo presentado son los siguientes (junto con los rangos de valores del Índice de Calidad del Aire Europeo de la figura 18 como referencia) :

- PM_{2.5}: **9.06** [0-800]
- PM₁₀: **8.48** [0-1200]
- NO₂: **13.74** [0-1000]
- SO₂: **5.98** [0-800]
- O₃: **21.17** [0-1250]
- Media total: **11.68**

Estos resultados son bastante buenos teniendo en cuenta el tamaño de los datos y de la red, aunque sí que observamos que hay claras diferencias de error para distintos contaminantes, siendo las mejores predicciones las del SO₂, PM₁₀ y PM_{2.5}, las peores predicciones se corresponden con el NO₂ y el O₃.

9. Conclusiones y trabajos futuros

Este TFG ha supuesto un gran reto, desde la investigación sobre los contaminantes y la climatología, hasta las distintas técnicas y modelos de inteligencia artificial. Mi objetivo era aprender toda la metodología de un proyecto de Data Science, adentrándome en cada tema con cierta profundidad pero sobretodo procurando no pasar ningún detalle por alto.

La calidad del aire y la contaminación son un tema crucial en la sociedad moderna, este tema me motivó a encontrar un enfoque en el que la ingeniería software estuviera claramente implicada, es por ello que presento los resultados y los objetivos alcanzados.

La aportación de este trabajo de investigación sirve como modelo para otros problemas y enfoques de la misma índole. El estudio realizado nos ha servido para entender el porqué de la contaminación, las relaciones entre estas y con la climatología, así como todo el desarrollo software que hay detrás de este tipo de proyectos. Se ha podido encontrar un modelo de inteligencia artificial adecuado para nuestro problema y transformar los datos adecuadamente, pero sobretodo, hemos conseguido comprender y explicar todo el proceso que conlleva el análisis y la toma de decisiones a la hora de diseñar redes neuronales.

Nos encontramos a lo largo de este proyecto con mucha problemática, principalmente proveniente de los datos, la mayor parte del trabajo ha sido dedicada a examinar y crear transformaciones de los datos, ya que contábamos con unos formatos de muy mala calidad y con bastantes datos nulos. Estos problemas con los datos serían fácilmente solucionados si de origen el Ayuntamiento de Madrid tuviera unos mínimos de calidad a la hora de recoger, limpiar, organizar y generar los datasets, de inicio estos datos son totalmente ilegibles por cualquier persona, también hay que tener en cuenta que no todo el mundo tiene los suficientes conocimientos en software como para poder convertir los datos y poder entenderlos como se ha hecho en el presente documento. La ofuscación de unas métricas tan esenciales que afectan a la salud de todos los ciudadanos no debería ser posible. En cuanto a los datos del clima se puede decir todo lo contrario, son un ejemplo a seguir de una buena bases de datos y con un sencillo acceso.

El otro gran problema que hemos tenido que abordar es la gran cantidad de datos que teníamos, junto a las limitaciones computacionales. Cuando se tienen varias estacio-

Conclusiones y trabajos futuros

nes, es difícil establecer un criterio para realizar las predicciones, ha sido uno de los momentos más clave de este proyecto, del que podemos decir que hemos conseguido resolverlo de manera perspicaz.

Nuestro modelo de redes LSTM pretende predecir la contaminación de Madrid de 1 día partiendo de los 14 días de datos anteriores, tanto de clima como de calidad del aire. El modelo consigue capturar las características del problema y obtiene conocimiento real de las relaciones de todas las variables. Por los resultados obtenidos podemos llegar a la conclusión de que estamos yendo por el buen camino, pero que todavía hay mucho margen de mejora, el modelo es capaz de predecir correctamente las tendencias, pero con una precisión un poco “vaga”. Da una buena idea de qué podemos esperar, pero para responder a necesidades sanitarias es necesaria una mejora sustancial.

Aún con todo, podemos afirmar que la predicción de la contaminación es una cuestión ideal para utilizar en modelos de inteligencia artificial. Se ha podido predecir la contaminación con una precisión aceptable para las limitaciones y el punto de inicio del que partíamos. Son unos de los caminos a seguir para la predicción de la contaminación (incluso de la meteorología), de hecho esto ya se hace a día de hoy en varias organizaciones y empresas como mencionamos al inicio.

Como hemos mencionado, las limitaciones computacionales que hemos tenido, pueden ser asumibles para las organizaciones gubernamentales, si se dispusiera de unos medios de GPUs, bases de datos, procesos software automatizados y de expertos en la materia, estoy seguro de que se podría conseguir un modelo con una precisión tan buena que podría ser utilizado para luchar contra el cambio climático.

Recopilar más datos haría que mejoraran inmediatamente los resultados, habría que establecer posibles relaciones de la contaminación con otro tipo de datos. Con más poder computacional habría incluido datos del tráfico de toda la ciudad o al menos de las carreteras más transitadas y cercanas a estas estaciones. Imágenes satelitales que ofrecen un amplio espectro de fotografía de las distintas capas de la atmósfera y también son capaces de detectar concentraciones de gases a nivel visual, así como satélites meteorológicos. Nu-

Conclusiones y trabajos futuros

trirse de distintos datasets podría resultar en nuevos caminos neuronales para entender mejor la contaminación.

Para buscar mejoras más significativas en las predicciones también debemos explorar más modelos de inteligencia artificial y de machine learning, probar múltiples parámetros e hiperparámetros utilizando técnicas como [GridSearch] para explorar un espacio de búsqueda más amplio. Modelos como XGBoost, Transformers e incluso técnicas de auto machine learning, las cuales exploran automáticamente distintos modelos. Se podrían hasta crear algoritmos de consenso entre todos estos para tratar de mejorar las predicciones.

Finalmente podemos decir que hemos establecido unas buenas bases para poder continuar esta línea de trabajo, siguiendo las pautas del proyecto e implementando las propuestas de futuro, estaríamos ante una herramienta de predicción de la calidad del aire que podría ser utilizada para tener un impacto significativo en la salud de todos.

10. Bibliografía

- WHO: World Health Organization. (2021). World Health Organization. (2021). WHO global air quality guidelines: particulate matter (PM_{2.5} and PM₁₀), ozone, nitrogen dioxide, sulfur dioxide and carbon monoxide. <https://apps.who.int/iris/handle/10665/345329>.
- ISGlobal: ISGlobal. (2021). Premature mortality due to air pollution in European cities: a health impact assessment. [https://doi.org/10.1016/S2542-5196\(20\)30272-2](https://doi.org/10.1016/S2542-5196(20)30272-2).
- MadridSalud: Madrid Salud. (2016). Dióxido de nitrógeno y Salud. <https://madridsalud.es/dioxido-de-nitrogeno-y-salud/>.
- IARC: International Agency for Research on Cancer. (2013). Air Pollution and Cancer. <https://publications.iarc.fr/Book-And-Report-Series/Iarc-Scientific-Publications/Air-Pollution-And-Cancer-2013>.
- ESA: Agencia Espacial Europea. (2020). La ESA en un mundo pos-Covid-19. https://www.esa.int/Space_in_Member_States/Spain/La_ESA_en_un_mundo_pos-Covid-19.
- DL-N02: Manzhu Yu, Qian Liu. (2021). Deep learning-based downscaling of tropospheric nitrogen dioxide using ground-level and satellite observations. <https://doi.org/10.1016/j.scitotenv.2021.145145>.
- AEMET: Agencia Estatal de Meteorología. (). AEMET OpenData - Sistema para la difusión y reutilización de la información de AEMET. <https://opendata.aemet.es/centrodedescargas/inicio>.
- Pandas: NumFOCUS. (). About pandas. <https://pandas.pydata.org/about/>.
- código abierto: Wikipedia. (). Código abierto. https://es.wikipedia.org/wiki/C%C3%B3digo_abierto.
- Pearson: Wikipedia. (). Pearson correlation coefficient. https://en.wikipedia.org/wiki/Pearson_correlation_coefficient.
- Seaborn: Michael Waskom. (). seaborn: statistical data visualization. <https://seaborn.pydata.org/>.
- Matplotlib: NumFOCUS. (). Matplotlib: Visualization with Python. <https://matplotlib.org/>.
- UV: AEMET. (). Radiación UV. <https://www.aemet.es/es/eltiempo/prediccion/radiacionuv/ayuda>.
- presión: UCM. (). Presión atmosférica. <http://meteolab.fis.ucm.es/meteorologia/presion-atmosferica--2>.

Bibliografía

ozono: Ministerio de Medio Ambiente y Medio Rural y Marino. (). El ozono troposférico y sus efectos en la vegetación.. https://www.miteco.gob.es/es/calidad-y-evaluacion-ambiental/temas/atmosfera-y-calidad-del-aire/Ozono_tcm30-188049.pdf.

Plotly Express: Plotly. (). Plotly Express in Python. <https://plotly.com/python/plotly-express/>.

óxidos nitrosos: Ministerio para la Transición Ecológica y el Reto Demográfico. (). Óxidos de Nitrógeno. <https://www.miteco.gob.es/es/calidad-y-evaluacion-ambiental/temas/atmosfera-y-calidad-del-aire/calidad-del-aire/salud/oxidos-nitrogeno.aspx>.

missingno: Aleksey Bilogur. (). missingno. <https://github.com/ResidentMario/missingno#readme>.

EPA: EPA. (). Outdoor Air Quality Data. <https://www.epa.gov/outdoor-air-quality-data/about-air-data-reports#aqi>.

EEA: European Environment Agency. (). European Air Quality Index. <https://airindex.eea.europa.eu/Map/AQI/Viewer/>.

ICA: Ministerio para la Transición Ecológica y el Reto Demográfico. (). Índice de Calidad del Aire. <https://www.miteco.gob.es/es/calidad-y-evaluacion-ambiental/temas/atmosfera-y-calidad-del-aire/ICA.aspx>.

calplot: tomkwok. (). Calendar heatmaps from Pandas time series data. <https://github.com/tomkwok/calplot>.

anticiclón: Telemadrid. (2015). El anticiclón favorecerá la contaminación durante por lo menos una semana más. <https://www.telemadrid.es/noticias/madrid/anticiclon-favorecera-contaminacion-semana-0-1743725633--20151201115024.html>.

Minkowski: Wikipedia. (). Minkowski distance. https://en.wikipedia.org/wiki/Minkowski_distance.

interpolación: Wu Hao, Xu Chang. (2013). Comparison of Spatial Interpolation Methods for Precipitation in Ningxia, China. <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.680.5221&rep=rep1&type=pdf>.

SciPy: SciPy. (). Fundamental algorithms for scientific computing in Python. <https://scipy.org/>.

restricciones: Dirección General de Gobernanza Pública. (2022). Crisis sanitaria COVID-19: Normativa e información útil.

Bibliografía

https://administracion.gob.es/pag_Home/atencionCiudadana/Crisis-sanitaria-COVID-19.html.

propagación: Tiago Miguel. (2021). How the LSTM improves the RNN. <https://towardsdatascience.com/how-the-lstm-improves-the-rnn-1ef156b75121>.

overfitting: Wikipedia. (). Overfitting. <https://en.wikipedia.org/wiki/Overfitting>.

scikit-learn: scikit-learn. (). Machine Learning in Python. <https://scikit-learn.org/stable/>.

One-hot: Wikipedia. (). One-hot. <https://en.wikipedia.org/wiki/One-hot>.

sliding window: Uber Engineering. (2018). Omphalos, Uber's Parallel and Language-Extensible Time Series Backtesting Tool. <https://eng.uber.com/omphalos/>.

NumPy: NumPy. (). The fundamental package for scientific computing with Python. <https://numpy.org/>.

formato LSTM: Jason Brownlee. (2019). How to Reshape Input Data for Long Short-Term Memory Networks in Keras. <https://machinelearningmastery.com/reshape-input-data-long-short-term-memory-networks-keras/>.

variable target: H2O.ai Wiki. (). Target Variable. <https://h2o.ai/wiki/target-variable/>.

TensorFlow: Google Brain Team. (). Plataforma de extremo a extremo de código abierto para el aprendizaje automático. <https://www.tensorflow.org/>.

Keras: TensorFlow. (). Deep learning for humans.. <https://keras.io/>.

GridSearch: Jason Brownlee. (). Hyperparameter Optimization With Random Search and Grid Search. <https://machinelearningmastery.com/hyperparameter-optimization-with-random-search-and-grid-search/>.

JupyterLab: Jupyter. (). JupyterLab: A Next-Generation Notebook Interface. <https://jupyter.org/>.

11. Apéndices

11.1. Instrucciones de instalación

1. Descargar el archivo “aereeze.zip” incluido en el anexo.
2. Descomprimir el archivo en la carpeta deseada.
3. Descargar e instalar Python 3.8.13
4. Ejecutar el comando “*pip install - r requirements.txt*” desde la raíz del proyecto para instalar las librerías necesarias.
5. Para visualizar los Python Notebooks es necesario tener algún tipo de extensión si se está abriendo el proyecto con un IDE o editor de código, también serviría utilizar una interfaz como [JupyterLab].

11.2. Estructura del código

```
└── LICENSE
└── README.md           ← The top-level README for developers using this project.
└── data
    ├── external        ← Data from third party sources.
    ├── interim          ← Intermediate data that has been transformed.
    ├── processed         ← The final, canonical data sets for modeling.
    └── raw               ← The original, immutable data dump.

    └── docs              ← The Project's documentation

    └── models            ← Trained and serialized models, model predictions, or model summaries

    └── notebooks          ← Jupyter notebooks.

    └── references         ← Data dictionaries, manuals, and all other explanatory materials.

    └── reports            ← Generated analysis as HTML, PDF, LaTeX, etc.
        └── figures          ← Generated graphics and figures to be used in reporting

    └── requirements.txt    ← The requirements file for reproducing the analysis environment, e.g.
                            generated with `pip freeze > requirements.txt`

    └── src                ← Source code for use in this project.
        └── data              ← Scripts to download or generate data.
```

11.3. Manual de uso

Para reproducir este proyecto en el mismo orden que se pensó, primero debemos ir a la carpeta raíz del proyecto, a continuación debemos dirigirnos a la página web de la AEMET ya que para hacer uso de su API y descargar los datos de meteorología, necesitamos una clave (API Key) personal, simplemente procedemos al enlace [“https://opendata.aemet.es/centrodedescargas/obtencionAPIKey”](https://opendata.aemet.es/centrodedescargas/obtencionAPIKey) e introducimos nuestra dirección de correo electrónico, nos llegará un correo electrónico, tras una confirmación en este último, recibiremos finalmente nuestra propia clave que deberemos de introducir en el archivo “src/data/aemet_api_key.json”, de la siguiente manera:

```
{  
    "api_key": "API_KEY"  
}
```

Ejecutaremos el script de descarga de datos con el comando “*python src/data/download_dataset.py*”. Para procesar los datos deberemos ejecutar el script “*src/data/process_dataset.py*”.

Finalmente nos queda explorar la carpeta de los Python Notebooks en la ruta “*notebooks*”, allí encontraremos las visualizaciones, mapas, etc. Para conseguir reproducir los datos totalmente debemos ejecutar por orden el notebook “*notebooks/process_weather_stations.ipynb*”, seguidamente de “*notebooks/interpolation.ipynb*”.

Una vez realizados los pasos anteriores tenemos todos los datos listos para ser procesados por nuestro modelo de inteligencia artificial, esto último lo conseguiremos ejecutando “*notebooks/predictions.ipynb*”.