Trabajo Fin de Máster

Caracterización del perfil de turismo a raíz de redes sociales

Guillermo Vilar González

Máster en Técnicas Estadísticas

Curso 2021-2022

Propuesta de Trabajo Fin de Máster

Título en galego: Caracterización do perfil de turismo a raíz de redes sociais
Título en español: Caracterización del perfil de turismo a raíz de redes sociales
English title: Characterization of the turism profile based on social media
Modalidad: Modalidad B
Autor/a: Guillermo Vilar González, Universidad de Santiago de Compostela
Director/a: Tomás Cotos Yáñez, Universidad de Vigo
Tutor/a: Ana Larrañaga Janeiro, Possible Incorporated S.L.
Breve resumen del trabajo: En este trabajo se busca crear un perfil de turismo de la gente que visita la ciudad de Vigo a partir de datos obtenidos de la red social de Instagram. Resultará de particular interés estudiar el posible efecto de la pandemia de 2020 por COVID-19 en estos perfiles de los visitantes.
Recomendaciones:
Otras observaciones:

Don Tomás Cotos Yáñez, Profesor Contratado Doctor de la Universidad de Vigo, y doña Ana Larrañaga Janeiro, personal colaborador de Possible Incorporated S.L., informan que el Trabajo Fin de Máster titulado

Caracterización del perfil de turismo a raíz de redes sociales

fue realizado bajo su dirección por don Guillermo Vilar González para el Máster en Técnicas Estadísticas. Estimando que el trabajo está terminado, dan su conformidad para su presentación y defensa ante un tribunal.

En Santiago de Compostela, a 13 de junio de 2022.

COTOS YAÑEZ, TOMAS RAIMUNDO (AUTENTICACIÓN)
Firmado digitalmente por COTOS YAÑEZ, TOMAS RAIMUNDO (AUTENTICACIÓN)
Fecha: 2022.06.13 10:13:08 +02'00'

El director:
Don Tomás Cotos Yáñez

LARRAÑAGA JANEIRO ANA - 39462206W
Firmado digitalmente por LARRAÑAGA JANEIRO ANA - 39462206W
Fecha: 2022.06.13 12:27:35 +02'00'

La tutora:
Doña Ana Larrañaga Janeiro

El autor:
Don Guillermo Vilar González

Declaración responsable. Para dar cumplimiento a la Ley 3/2022, de 24 de febrero, de convivencia universitaria, referente al plagio en el Trabajo Fin de Máster (Artículo 11, Disposición 2978 del BOE núm. 48 de 2022), **el/la autor/a declara** que el Trabajo Fin de Máster presentado es un documento original en el que se han tenido en cuenta las siguientes consideraciones relativas al uso de material de apoyo desarrollado por otros/as autores/as:

- Todas las fuentes usadas para la elaboración de este trabajo han sido citadas convenientemente (libros, artículos, apuntes de profesorado, páginas web, programas,...)
- Cualquier contenido copiado o traducido textualmente se ha puesto entre comillas, citando su procedencia.
- Se ha hecho constar explícitamente cuando un capítulo, sección, demostración,...sea una adaptación casi literal de alguna fuente existente.

Y, acepta que, si se demostrara lo contrario, se le apliquen las medidas disciplinarias que correspondan.

Agradecimientos

En primer lugar, me gustaría agradecer a todo el mundo que me ha ayudado en la elaboración de este trabajo. En particular a mi tutora Ana Larrañaga Janeiro, sin quien no habría sabido ni por donde empezar; y a Javier Martínez Torres de Possible Inc., por sus múltiples consejos en la elaboración de este documento.

A todo el personal de la Facultad de Matemáticas de la Universidad de Santiago de Compostela, y del Máster en Técnicas Estadísticas, a quienes les debo el conocimiento que espero haber plasmado correctamente en este trabajo.

Finalmente, a mi familia y amigos, por apoyarme siempre en todo, y que han logrado hacerme más feliz cada día.

Índice general

Resumen

Resumen en español

El objetivo de este trabajo es la creación de un perfil de turismo para los visitantes de Vigo a partir de datos obtenidos de redes sociales que permita detectar posibles efectos de la pandemia de 2020. Se empleará un conjunto de datos formado por publicaciones de la red social de Instagram con ubicación en Vigo y realizadas en los años 2018 y 2021. Se comienza repasando toda la información que podremos extraer de la imagen y el título de las publicaciones. En particular, se introducirán herramientas para obtener datos acerca del género, la edad, el nivel de educación y los sentimientos del autor, y se evaluarán sobre conjuntos de datos públicos etiquetados.

A continuación, a partir de una muestra de nuestro conjunto de datos etiquetada manualmente, se entrenará un modelo de clasificación supervisada que tenga como objetivo distinguir entre las publicaciones realizadas por cuentas personales, y aquellas realizadas por cuentas de negocios como restaurantes, clínicas de nutrición u otros comercios.

Finalmente, se realizará un análisis de las publicaciones que hayan sido clasificadas como provenientes de cuentas personales. Este análisis se centra en particular en estudiar si la pandemia de 2020 supuso un cambio significativo en alguno de los atributos mencionados anteriormente para los visitantes de Vigo; es decir, el género, la edad, el nivel de educación o los sentimientos. Estos cambios detectados se detallan en las secciones finales del trabajo.

English abstract

The goal of this study is the creation of a tourism profile for the visitors of Vigo using data obtained from social media that allows for the testing of possible effects of the pandemic of 2020. Such study will be performed through a dataset consisting on Instagram posts with a location set in Vigo, and posted between the years of 2018 and 2021. The first step is going over all of the information that can be extracted from the posts' image and title. Some tools will be specifically introduced to extract information about the age, gender, education level and sentiments of the author. The performance of those tools will be evaluated over publicly available labeled datasets.

Furthermore, a supervised classification model will be trained from a manually labeled sample from our data. This model will differentiate between posts from personal accounts, and those made by business accounts like restaurants or nutrition clinics.

Lastly, an analysis will be performed over those posts classified by the model as being made by personal accounts. It will be of particular interest the testing of whether the global pandemic of 2020 caused a significant change over any of the aforementioned attributes for the turists of Vigo; that is, the gender, the age, the education level or the polarity. The conclusions drawn from the study are gathered in the final sections of the study.

Capítulo 1

Introducción

Las redes sociales generan cada día un volumen de tráfico de información masivo en comparación con el que se puede obtener a través de plataformas más tradicionales. Por ejemplo, desde su aparición en 2012, la aplicación de Instagram cuenta ya con más de mil millones de usuarios activos, que se estima que comparten cada día más de 96 millones de publicaciones[1]. Esta gran cantidad de información disponible lleva a múltiples sectores como pueden ser el marketing o el turismo a emplear las redes sociales como fuente de la que obtener sus datos (Park et al. 2016, Tenkanen et al. 2017).

Este trabajo tiene como objetivo el realizar un estudio a partir de datos recogidos de la red social de Instagram, como una extrapolación de Kalra et al. (2018), centrado en la gente que visita la ciudad de Vigo. En particular, nos interesará estudiar la manera en la que la pandemia por COVID-19 de 2020 ha podido afectar al perfil de los visitantes de la ciudad. Para ello, recogimos un total de 10140 publicaciones con la geolocalización activada marcando la ubicación en Vigo. La mitad de estas publicaciones fueron realizadas en los meses de mayo y junio de 2018, y la otra mitad en los mismos meses de 2021. La información que recogemos está limitada al nombre de la cuenta que realizó cada publicación, la fecha y hora exactas a las que fue publicada, el título que la acompaña, y la imagen (en el caso de que no se trate de un vídeo). Dedicaremos este capítulo introductorio a presentar una visión general del análisis que podremos realizar sobre estos datos.

1.1. Detección de spam

Antes de realizar cualquier análisis, es importante tener en cuenta que no todas las publicaciones van a tener valor para el estudio. Nos interesan en concreto las cuentas personales, y por tanto será necesario encontrar una manera de filtrar las cuentas automatizadas, las de spam o las que pertenecen a organizaciones o marcas, entre otros. La manera de detectar estas cuentas es a través de distintos atributos de la misma o de una publicación en concreto que se quiera clasificar. Por ejemplo, para clasificar una cuenta pueden utilizarse la cantidad total de publicaciones de la cuenta, el número de seguidores, o la cantidad de números presentes en el nombre de usuario. Por otro lado, puede clasificarse una publicación en particular empleando atributos como el número de usuarios mencionados en la publicación, la cantidad de hashtags empleados, el número de veces que la publicación ha sido compartida, o la presencia o no de palabras pertenecientes a una lista de palabras de spam.

La metodología particular depende en gran medida del autor. En Stringhini et al. (2010) se miden 6 atributos de la cuenta para detectar si se trata de una cuenta de spam a través de un clasificador de bosque aleatorio. En Benevenuto et al. (2010) se hace un análisis sobre Twitter tomando 39 atributos relacionados con el contenido de las publicaciones, y otros 23 acerca del comportamiento del usuario, y se emplean máquinas de soporte vectorial para realizar la clasificación. Obtienen una precisión de 87,2 % para detectar publicaciones de spam, y un 84,5 % para detectar cuentas de spam empleando

[1] https://www.wordstream.com/blog/ws/2017/04/20/instagram-statistics

únicamente las 23 medidas del comportamiento de la cuenta y no de su contenido. En Ersahin et al. (2017) los autores aplican un algoritmo de Naive Bayes sobre 12 características generales de la cuenta, obteniendo una precisión del 90,9 % para detectar cuentas de spam tras haber realizado discretización en las variables (o del 86 % sin realizarla).

La detección de spam es vital en el sector turístico. Uno de sus usos puede ser distinguir entre las reseñas de un hotel veraces escritas por clientes, y aquellas que podrían ser clasificadas como spam realizadas de manera deshonesta. En Li et al. (2013) se emplea aprendizaje supervisado para estudiar si cada palabra de una reseña es más probable que forme parte de un comentario real o de uno de spam. La clasificación se realiza en base a los pesos de cada una de las palabras que forman el texto. En Yoo y Gretzel (2009) se busca detectar comentarios realizados por el propio establecimiento con el objetivo de autopromocionarse. En este caso la clasificación se hace a través de 7 contrastes que comparan las reviews de spam y las reales, como por ejemplo cuáles son más largas, o cuáles hacen más referencia al nombre del hotel.

1.2. Análisis de las imágenes de perfil: detección facial

Las publicaciones de redes sociales están acompañadas de la foto de usuario de su autor. En caso de que esta foto sea de la propia persona, puede proporcionar información que será utilizada junto con la contenida en la propia publicación para crear un perfil más preciso de la persona que la escribe. En particular, se utilizan técnicas de detección facial para predecir el género y la edad de la persona que se encuentra en la foto.

Los esfuerzos por crear técnicas de estudio facial en imágenes comenzaron a principios de la década de 1960 (Nilsson 2009, Insaf et al. 2020) cuando W. Bledsoe desarrolló para la CIA un proyecto de reconocimiento facial que tenía como objetivo, recibida una fotografía de la cara de una persona, clasificarla según a quien perteneciera de entre los integrantes de una base de datos. La clasificación se realizaba a partir de 20 distancias entre rasgos faciales, como por ejemplo la anchura de los ojos o la de los labios. La principal pega era que un operador debía introducir manualmente las coordenadas de los principales rasgos de la persona representada en la fotografía, como el centro de la pupila o las esquinas de los ojos. El autor estimaba que un operador podía procesar unas 40 fotografías por hora.

En Kanade (1977), el autor presentó uno de los primeros métodos de detección de rasgos faciales realizados por un ordenador. El método se basaba en, a partir de una fotografía de una cara, obtener las regiones en las cuales ocurría un mayor cambio en el brillo a través de un operador Laplaciano. Después, estas regiones se identificaban a través de un algoritmo con distintos rasgos faciales como la nariz o la boca. Una vez detectadas las posiciones de estos rasgos, vuelve a ser posible realizar reconocimiento facial basado en las distancias entre los mismos.

En Kirby y Sirovich (1990), se aplicó análisis de componentes principales sobre el problema de reconocimiento facial para el diseño de un sistema de eigenfaces, que permite la obtención de un conjunto de características principales a partir de una serie de imágenes de caras. Esta idea fue utilizada en Turk y Pentland (1991) para formular un nuevo método de reconocimiento facial, creando en primer lugar un espacio de caras a través de una serie de eigenfaces, para después proyectar imágenes en este espacio y detectar si se corresponden con caras (si están lo suficientemente cerca del espacio), y en caso afirmativo clasificarlas en base a los pesos para determinar si se corresponden con una persona de la base de datos o no.

En 1993 comenzó el desarrollo del programa FacE-REcognition Technology (FERET) (Rauss et al. 1997), un proyecto financiado por el gobierno que como parte de su contenido comparó dos algoritmos de reconocimiento facial sobre un conjunto de datos formado por imágenes de caras tomadas desde diferentes ángulos. El primero de ellos continuaba el análisis de componentes principales iniciado en Kirby y Sirovich (1990), mientras que el segundo proyectaba las imágenes en un conjunto de wavelets Gabor centradas en el mismo pixel pero con distintas escalas y orientaciones. En este caso, la clasificación de a quién pertenece una cara se realiza en base a los coeficientes de las proyecciones. Los resultados mostraron que este nuevo algoritmo parecía ser como mínimo tan bueno como el de las

eigenfaces. Otra de las metas de este proyecto era la propia creación del conjunto de datos utilizado en la evaluación de los algoritmos, con el objetivo de mejorar las comparaciones entre diferentes modelos, ya que hasta ese momento no existía un estándar en las bases de datos que cada investigador empleaba.

En los años posteriores se realizaron diversos esfuerzos para comparar los múltiples métodos de reconocimiento facial que iban surgiendo, como el Face Recognition Vendor Test en 2002 (Phillips et al. 2003) o el Face Recognition Grand Challenge en 2006 (Phillips et al. 2006).

La creación de métodos de aprendizaje profundo y redes neuronales artificiales que surgieron posteriormente trajeron consigo mejoras en términos de capacidad de aprendizaje y robustez (Guo y Zhang 2019, Hu et al. 2015). Estos métodos están generalmente basados en redes neuronales convolucionales, que aplican una operación de convolución en alguna de sus capas.

A día de hoy existen múltiples aplicaciones de detección y reconocimiento facial de uso comercial, como por ejemplo IBM Bluemix Visual Recognition, AWS Rekognition o Microsoft Azure Face API; que además de detectar rostros en imágenes, también pueden devolver una predicción acerca de su género, edad o etnia. Otros servicios como iPhone o Facebook tienen integrados sus propios sistemas de reconocimiento facial para uso interno. Se ha estudiado con anterioridad la eficacia de estas múltiples aplicaciones comparándolas entre si (Jung et al. 2018), y en este trabajo utilizaremos en particular la herramienta de Face++[2], que presenta resultados altamente aceptables con un plan gratuito.

Los métodos de detección y reconocimiento facial también han sido utilizados con anterioridad en ámbitos relacionados con el turismo. Por ejemplo, en Ryu y Lee (2016) se emplean sistemas de reconocimiento facial en una aplicación de teléfono móvil para aumentar la comodidad de los turistas mediante un servicio adaptado a su trasfondo o motivo de visita. En González-Rodríguez et al. (2020) se emplean algoritmos de reconocimiento más especializados para detectar las emociones a partir de una imagen de rostro, y poder así evaluar la satisfacción de un grupo de turistas mediante no solo sus respuestas a una encuesta, sino también a través de sus expresiones faciales a lo largo de la experiencia.

1.3. Análisis del texto: fórmulas de lecturabilidad

En el texto de la publicación está contenida la mayor parte del mensaje. Analizar el tema concreto puede resultar una tarea demasiado compleja para ser realizada por una máquina, pero es posible intentar extraer conclusiones más generales tanto del mensaje como de su autor a través de distintos métodos.

Uno de esos métodos es a través de las fórmulas de lecturabilidad. Estas fórmulas tienen como objetivo establecer la dificultad de la lectura de un texto. Aplicar ese tipo de fórmulas puede resultar útil ya que permite obtener una posible estimación de la educación de la persona que se encuentra detrás de la publicación.

Uno de los pioneros en este campo fue Rudolf Flesch, que publicó en Flesch (1948) la conocida como prueba de legibilidad de Flesch. El autor presenta la siguiente fórmula:

$$206{,}835 - 1{,}015 \left(\frac{\text{nº de palabras}}{\text{nº de oraciones}} \right) - 84{,}6 \left(\frac{\text{nº de sílabas}}{\text{nº de palabras}} \right) \tag{1.1}$$

Se basa en el número total de oraciones, palabras y sílabas en el texto. Devuelve un valor más elevado cuanto más fácil sea la lectura del texto, ya que en ese caso estará formado por oraciones simples y palabras cortas. La fórmula está pensada para ser aplicada sobre textos escritos en inglés, pero ha sido adaptada a otros idiomas. En particular, destacamos la publicada en Fernández (1959), para textos en español, que viene dada por:

$$206{,}835 - 1{,}02 \left(\frac{\text{nº de palabras}}{\text{nº de oraciones}} \right) - 60 \left(\frac{\text{nº de sílabas}}{\text{nº de palabras}} \right) \tag{1.2}$$

[2] https://www.faceplusplus.com/

Cuadro 1.1: Interpretación del índice de lecturabilidad de Flesch

Puntuación	Grado escolar	Nivel
100.0 - 90.0	5º de primaria	Lectura muy fácil
90.0 - 80.0	6º de primaria	Lectura fácil, nivel conversacional
80.0 - 70.0	1º de la ESO	Lectura algo fácil
70.0 - 60.0	2º y 3º de la ESO	Lectura usual para un adulto
60.0 - 50.0	4º de la ESO y bachillerato	Lectura algo difícil
50.0 - 30.0	Estudiante universitario	Lectura difícil
30.0 - 10.0	Graduado de universidad	Lectura muy difícil
10.0 - 0.0	Profesional	Lectura extremadamente difícil

A pesar de que los valores que devuelven estas fórmulas no están necesariamente restringidos entre 0 y 100, se suelen interpretar sus resultados por medio de la tabla 1.1.

Otras medidas de lecturabilidad buscan darle más peso a las palabras más complejas. Un ejemplo es el Gunning Fog Index, presentado en Gunning (1952), aplicable a textos en inglés. El autor define las palabras complejas como aquellas de 3 o más sílabas, sin contar nombres propios o palabras compuestas, e ignorando los sufijos más comunes del inglés como *-ed* o *-ing*. La fórmula, pensada para ser aplicada en pasajes de unas 100 palabras, es:

$$0{,}4\left[\left(\frac{\text{nº de palabras}}{\text{nº de oraciones}}\right)+100\left(\frac{\text{nº de palabras complejas}}{\text{nº de palabras}}\right)\right] \tag{1.3}$$

Destacamos también la fórmula presentada en Spaulding (1956), pensada para el español, y que considera en este caso a las palabras complejas como aquellas que no figuran en una lista de más de 1500 palabras proporcionada por el autor. La fórmula viene dada por:

$$22+1{,}609\left(\frac{\text{nº de palabras}}{\text{nº de oraciones}}\right)+331{,}8\left(\frac{\text{nº de palabras complejas}}{\text{nº de palabras}}\right) \tag{1.4}$$

Para estas dos últimas fórmulas, se considera que un texto es más complicado de leer cuanto mayor sea la puntuación obtenida.

1.4. Análisis del texto: estudio de sentimientos

Además de obtener información del autor a partir del texto de la publicación, también obtendremos información del mensaje en sí. Una posibilidad es agrupar las publicaciones por temáticas en base a bancos de palabras, pero ante el riesgo de añadir un sesgo en base a los grupos que decidamos crear, o que no tomemos un rango lo suficientemente amplio y acabemos con un número demasiado elevado de textos sin clasificar, nos decantamos por realizar análisis de sentimientos.

El concepto de análisis de sentimientos engloba a una serie de procedimientos y herramientas que permiten extraer información subjetiva de un texto (Mäntylä et al. 2018). Los primeros estudios de la realización de este tipo de análisis a través de ordenadores fueron publicados en la década de 1990, como trabajo de la Asociación de Lingüística Computacional fundada en 1962. En Wiebe et al.

(1999) se presentó un método para clasificar enunciados mediante el uso de un ordenador según sean considerados objetivos o subjetivos.

En los años posteriores, el análisis de sentimientos comenzó a centrarse en estudiar la polaridad de un texto supuesta ya su subjetividad. En este contexto fueron publicados Pang et al. (2002) y Turney (2003), que presentaban métodos para clasificar reseñas de internet en base a si recomendaban o no el objeto o servicio en cuestión. En particular, el estudio en Pang et al. (2002) se realiza sobre críticas de cine, alcanzando una precisión de más del 82 %. Por otro lado, en Turney (2003) se aplica aprendizaje no supervisado sobre opiniones de automóviles, bancos, películas y destinos turísticos, obteniendo una precisión final superior al 74 %.

La línea de análisis de sentimientos empleada en este trabajo sigue con este estudio de la polaridad. En particular, consideraremos el uso del modelo VADER (Hutto y Gilbert 2014). Se trata de un modelo diseñado para trabajar con datos obtenidos de redes sociales, que otorga ciertas puntuaciones al vocabulario presente en un texto, y que además tiene en cuenta el uso de signos de puntuación o mayúsculas para enfatizar las emociones. Devuelve una puntuación para la intensidad emocional del texto global entre -1 y 1, correspondiéndose el 1 con un texto extremadamente positivo y el -1 con uno extremadamente negativo. Los autores obtuvieron con este método una puntuación $F1$ de 0,96, contra un 0,84 de un clasificador humano.

El análisis de sentimientos ha sido ampliamente utilizado en el campo del análisis de turismo, en gran parte debido a que las críticas de internet de ciertos servicios como hoteles son conjuntos de datos buenos y fácilmente accesibles sobre los cuales trabajar. En Kirilenko et al. (2018) se recogen varios modelos de análisis de sentimientos empleados por distintos autores en análisis de turismo. En general existen dos enfoques a la hora de construir modelos: el primero de ellos es el basado en el léxico utilizado, como el que sigue VADER, mientras que el segundo es el que utiliza modelos de aprendizaje automático, como máquinas de soporte vectorial o Naive Bayes. La conclusión a la que llegan los autores es que es recomendable el uso de un modelo basado en aprendizaje automático en caso de que se realice una calibración entrenándolo en una muestra representativa de los datos. Si embargo, si esto no es factible debido a restricciones de tiempo o coste, es preferible un modelo basado en léxico, aunque no haya sido calibrado para el conjunto de datos en particular.

Capítulo 2

Modelos de clasificación

Se entiende por modelo de aprendizaje automático, o *machine learning model*, como un modelo que tiene la capacidad de mejorar su eficiencia a partir de la experiencia (Mitchell 2006). Existen dos principales categorías en las cuales se pueden englobar a estos modelos: los de aprendizaje supervisado y los de aprendizaje no supervisado. Los modelos de aprendizaje no supervisado realizan un análisis exploratorio de las observaciones en base a unas covariables medidas sobre las mismas, llamadas usualmente variables predictoras. En el caso de los modelos de aprendizaje supervisado existe además una variable respuesta, y se tratan de métodos predictivos.

Según la naturaleza de esta la variable respuesta, se distingue entre modelos de aprendizaje supervisado de regresión (variable respuesta numérica) o de clasificación (variable respuesta categórica). La mayoría de los modelos que veamos pueden ser adaptados para ambos casos, pero en este capítulo nos centraremos en su uso como modelos de clasificación, que emplearemos más adelante en el trabajo.

2.1. Árboles de decisión y bosques aleatorios

La idea tras los árboles de decisiones consiste en dividir el espacio de variables predictoras de manera recurrente en subregiones más simples. Partiendo del espacio completo, en cada paso se realiza una división del espacio en función de una de las variables (y cada uno de estos subespacios será a su vez dividido en la siguiente etapa). La selección de la variable sobre la cual se va a realizar la partición en cada etapa, y el valor de esta variable sobre el cual hacer el corte, son seleccionados a partir de un método de optimización que veremos en esta sección.

Los subespacios que conforman la partición del espacio original una vez terminado el algoritmo reciben el nombre de nodos terminales. A la hora de clasificar una nueva observación, se calcula a qué nodo terminal pertenece a partir de sus variables predictoras. La predicción se hará en función de la moda de la variable respuesta para las observaciones de entrenamiento que caen en ese nodo terminal.

Una primera versión de este tipo de algoritmos fue presentada en Belson (1959), con ciertas limitaciones. En el artículo, el autor presenta un modelo con estructura de árbol, pero que solo puede ser entrenado para variables categóricas (o al menos no presenta una manera de realizar la clasificación para variables predictoras continuas en un tiempo razonable). El autor propone que en cada nodo se compruebe, para cada variable predictora, cómo resultaría la división de las observaciones en función de los valores de dicha variable. Para cada predictor, define su poder predictivo como la diferencia entre la proporción de individuos de una categoría que hay en un subespacio generado al dividir por ese predictor, y los que hay en el conjunto de entrenamiento completo. Entonces, se particiona el espacio en función de la variable con el mayor poder predictivo (que es la que haya separado a los individuos más diferente de lo que lo hubiera hecho una separación aleatoria).

El problema de este algoritmo es que no proporcionaba una solución satisfactoria en caso de que alguna variable sea continua. Posteriormente, en las décadas de 1970 y 1980, surgieron múltiples

algoritmos para entrenar un árbol de decisión, como el ID3 (Quinlan 1986) o el CART (Breiman et al. 1984). En este trabajo nos centraremos en el modelo CART.

El algoritmo CART se basa en realizar divisiones binarias del espacio de predictores en función de la impureza de Gini. En cada nodo del árbol, la impureza de Gini mide la probabilidad de que un individuo sea clasificado de manera incorrecta, suponiendo que sus etiquetas fueron generadas aleatoriamente según la distribución de etiquetas de individuos del nodo, y que la clasificación se hace proporcionalmente en función a los individuos en cada nodo.

Por ejemplo, supongamos que tenemos dos categorías para la respuesta, y que en un nodo en particular hay 7 individuos de la primera categoría y 3 de la segunda. La probabilidad de generar un individuo de la clase 1 en este nodo es de 0,7, y de 0,3 para la clase 2. Si un individuo pertenece a la clase 1, la probabilidad de que fuera incorrectamente clasificado (como perteneciente a la clase 2) sería igual a la proporción de individuos pertenecientes a la clase 2, es decir, 0,3. De igual manera, para la clase 2 sería de 0,7. Por tanto, la impureza de Gini en este nodo sería de $0{,}3 \cdot 0{,}7 + 0{,}7 \cdot 0{,}3 = 0{,}42$.

De manera general, supongamos que tenemos I categorías diferentes para la respuesta, y que en un nodo en particular la proporción de individuos de la clase i es p_i, $i = 1, \ldots, I$. La probabilidad de clasificar mal a un individuo de la clase i será de $\sum_{k \neq i} p_k = 1 - p_i$. Por tanto, la impureza de Gini del nodo, I_G, vendría dada por la expresión

$$I_G = \sum_{i=1}^{I} p_i(1 - p_i) = 1 - \sum_{i=1}^{I} p_i^2. \tag{2.1}$$

El modelo de árboles de decisión CART implementa la impureza de Gini de la siguiente manera: sea $\vec{X} = (X_1, \ldots, X_p)$ el vector de las p variables predictoras. En cada nodo del árbol, y comenzando con el nodo que contiene al espacio de predictores completo, se seleccionan el predictor X_i y el punto de corte k_i con $i \in \{1, \ldots, p\}$ de manera que, si dividimos el espacio en los dos subespacios generados a partir de este corte, $R_1 = \{\vec{X} = (X_1, \ldots, X_p) | X_i \leq k_i\}$ y $R_2 = \{\vec{X} = (X_1, \ldots, X_p) | X_i > k_i\}$, la media de las impurezas de Gini de los dos nodos resultantes ponderadas por el número de observaciones de entrenamiento en cada subespacio sea la menor posible. De este modo se selecciona el corte binario que minimiza la proporción de individuos mal clasificados.

Como criterio de parada se pueden emplear la profundidad máxima (cadena de mayor longitud posible de subespacios consecutivos) o el tamaño mínimo de un nodo (es decir, de las observaciones de entrenamiento que caen en ese nodo). Es importante que no se debería continuar dividiendo un nodo si el mínimo de las impurezas de Gini ponderadas de sus posibles nodos hijos es mayor que la impureza de Gini del propio nodo, ya que entonces se empeoraría la clasificación.

Una vez se ha obtenido el árbol, el siguiente paso del algoritmo es podarlo. Podar un árbol consiste en eliminar nodos intermedios con el fin de reducir su complejidad y eliminar parte del sesgo de las observaciones de entrenamiento. Existen varios métodos para hacerlo, y en esta sección explicaremos la poda de coste-complejidad. Esta se basa en minimizar la suma de residuos al cuadrado (RSS), pero introduciendo una penalización por complejidad en base al número de nodos terminales del árbol. Se busca entonces el árbol obtenido colapsando nodos del árbol original, y que minimice la puntuación

$$RSS - aT, \tag{2.2}$$

donde los residuos son las proporciones de observaciones mal clasificadas en cada nodo terminal por tratarse de un modelo de clasificación, T es el número de nodos terminales del nuevo árbol, y a es un hiperparámetro que tiene el objetivo de controlar la complejidad del árbol. Cuanto mayor sea este hiperparámetro, más penaliza el modelo el hecho de que el nuevo árbol tenga una cantidad elevada de nodos terminales.

Las principales ventajas de los árboles de decisión son su simpleza, fácil interpretabilidad y rapidez del entrenamiento. Sin embargo, esto viene a costa de tener una varianza muy elevada. El tipo de aprendizaje que realizan los árboles de decisión es greedy, lo cual quiere decir que se centra en minimizar el error en cada etapa de manera independiente, sin preocuparse por lo que pase en etapas posteriores.

Con el objetivo de mejorar modelos de rápido entrenamiento y varianza elevada, como los árboles, se idearon las técnicas de agregación bootstrap, o bagging. Este concepto, basado en las técnicas bootstrap, fue presentado en Breiman (1996). Se basa en generar, a partir del conjunto de entrenamiento original, un número elevado de nuevos conjuntos de entrenamiento, obtenidos a partir de muestreo uniforme con reemplazo. Para cada una de estas muestras bootstrap se entrena un modelo, y como predicción se considera la moda de las predicciones de cada uno de los modelos.

Los bosques aleatorios son un tipo de modelos presentados en Ho (1995) que parten de la idea de aplicar agregación bootstrap con árboles de decisiones, pero van un paso más allá con el objetivo de reducir aún más la varianza, y de disminuir la correlación entre los árboles que lo forman. Durante el entrenamiento de cada uno de los árboles, normalmente se considerarían a todas las variables predictoras como candidatas sobre las cuales realizar cada corte en cada uno de los nodos. Sin embargo, al entrenar un bosque aleatorio, en cada uno de los nodos de cada árbol se seleccionará al azar un subconjunto de predictores candidatos para dividir el espacio, y los demás no serán tenidos en cuenta en ese paso.

Si el conjunto de entrenamiento tiene en total p variables predictoras diferentes, para los problemas de clasificación se suelen considerar en cada nodo $\sqrt{p}$ predictores seleccionados aleatoriamente, aunque esta cantidad puede ser considerada como un hiperparámetro que ajustar. El otro hiperparámetro propio de los bosques aleatorios es el número de árboles que lo componen. Es necesario tomar un número lo suficientemente grande de árboles como para que el bosque aleatorio pueda ser entrenado de manera apropiada, pero no tanto como para que se de sobreajuste. Por lo general, será necesaria una cantidad más elevada de árboles si el conjunto de entrenamiento tiene más predictores.

2.2. Máquinas de soporte vectorial

Las máquinas de soporte vectorial son modelos de clasificación de aprendizaje supervisado que parten de la idea base de dividir el espacio de variables predictoras mediante un hiperplano que separen a las observaciones en función de su respuesta. La primera versión de una máquina de soporte vectorial fue descrita en 1964 por V. N. Vapnik y A. Ya. Chervonenkis (Vapnik y Chervonenkis 1964, Chervonenkis 2013).

La versión más simple de estos modelos se da cuando hay 2 posibles categorías en la respuesta, y las observaciones son linealmente separables para estas categorías. Supongamos que tenemos n observaciones con p variables predictoras diferentes, entonces estas observaciones se corresponden con puntos de un espacio p-dimensional. Que sean linealmente separables respecto a la respuesta quiere decir que existe un hiperplano de dimensión $p-1$ tal que todas las observaciones de una de las categorías se sitúan a uno de los lados del hiperplano, y las de la otra categoría se encuentran en el otro lado.

Un hiperplano seguirá la fórmula $h(\vec{X}) = \vec{X}'\vec{\beta} + \beta_0 = 0$, donde $\vec{X} = (X_1, \ldots, X_p)$ es el vector p-dimensional de variables predictoras, $\vec{\beta} = (\beta_1, \ldots, \beta_p)$ es el vector de coeficientes y β_0 es el coeficiente independiente. Si se etiqueta la variable respuesta como $Y \in \{-1, 1\}$, y se restringe el vector de coeficientes $\vec{\beta}$ a un vector unitario (es decir, que su norma $\|\vec{\beta}\| = \sqrt{\sum_{j=1}^{p} \beta_j^2}$ sea 1), se cumple que el producto de la respuesta de una observación particular con el hiperplano representa la distancia perpendicular positiva entre dicha observación y el hiperplano. Si se calcula esta distancia para cada observación de entrenamiento, a la menor de ellas se le llama margen del hiperplano.

Lo ideal sería que este margen fuese lo más grande posible, con el objetivo de mejorar las clasificaciones de nuevas observaciones. Esto es debido a que cuanto más próximo se encuentre el hiperplano de alguna observación del conjunto de entrenamiento, menor es la variación en los valores de sus predictores que tiene que darse para que dicha observación pase a estar en el lado incorrecto del hiperplano. Por ello, al considerar nuevas observaciones con distribuciones similares a las que siguen los datos del conjunto de entrenamiento, será más probable que estas sean clasificadas correctamente por el modelo cuanto mayor sea la distancia entre el hiperplano y los datos de entrenamiento.

Por tanto, para las n observaciones de entrenamiento, sean $\vec{x}_1, \ldots, \vec{x}_n$ los vectores p-dimensionales de sus variables predictoras, y sean $y_1, \ldots, y_n$ sus respuestas (que recordamos que toman valores 1

o -1). Encontrar el hiperplano separador de máximo margen equivale a maximizar el margen del hiperplano (es decir, maximizar la distancia entre el hiperplano y la observación que esté más cercana al mismo), el cual ha sido denotado por M, a partir del siguiente problema de optimización:

$$\max M \tag{2.3}$$

$$\text{sujeto a } \|\vec{\beta}\| = 1 \tag{2.4}$$

$$y_i(\vec{x}_i'\vec{\beta} + \beta_0) \geq M, \;\; i = 1, \ldots, n \tag{2.5}$$

Generalmente se emplea por conveniencia la formulación del problema de minimización equivalente obtenida al sustituir el margen por la inversa de la norma de $\vec{\beta}$, es decir, al tomar $M = 1/\|\vec{\beta}\|$, que pasa a ser:

$$\min \|\vec{\beta}\| = \sqrt{\sum_{j=1}^{p} \beta_j^2} \tag{2.6}$$

$$\text{sujeto a } y_i(\vec{x}_i'\vec{\beta} + \beta_0) \geq 1, \;\; i = 1, \ldots, n \tag{2.7}$$

A la hora de clasificar una observación, es suficiente con calcular en qué lado del hiperplano se sitúa. El problema de este tipo de clasificadores es que en muchos casos los datos no serán linealmente separables. Por este motivo se introduce el concepto de margen débil, a través de la formulación

$$y_i(\vec{x}_i'\vec{\beta} + \beta_0) \geq M(1 - \xi_i), \;\; i = 1, \ldots, n, \tag{2.8}$$

donde ξ_i, $i = 1, \ldots, n$, son variables de holgura no negativas. Cuando estas variables toman valores positivos, permiten que la distancia del hiperplano a ciertas observaciones sea menor que el margen M definido antes. Se permite además que algunas observaciones estén clasificadas en el lado incorrecto del hiperplano, siempre y cuando sus variables de holgura asociadas tengan valores lo suficientemente elevados (ya que si la observación i-ésima está clasificada en el lado incorrecto, el valor de $y_i(\vec{x}_i'\vec{\beta} + \beta_0)$ es negativo, y ξ_i necesariamente habrá tomado un valor lo suficientemente elevado como para que $M(1 - \xi_i)$ sea menor que ese valor negativo y cumpla 2.8). Concretamente, si $\xi_i > 1$, la observación de entrenamiento i-ésima fue incorrectamente clasificada, mientras que si $1 > \xi_i > 0$, la observación fue clasificada correctamente, pero se encuentra a menor distancia del hiperplano que el margen M. Si $\xi_i = 0$, significa que la observación i-ésima cumple le ecuación 2.5 y por tanto fue correctamente clasificada y se encuentra a una distancia del hiperplano superior al margen M. Finalmente, el caso particular de $\xi_1 = 1$ estaría reservado para el caso de que la observación i-ésima cayese en el hiperplano.

Es necesario limitar los valores que pueden tomar las variables de holgura para que el problema esté acotado. Esto se hace mediante la restricción

$$\sum_{i=1}^{n} \xi_i \leq K, \tag{2.9}$$

donde K será un hiperparámetro del modelo, que controla el grado de violaciones que puede ocurrir en total con respecto al margen y al hiperplano. Cuanto mayor sea el valor de K, más le estamos permitiendo al modelo que realice malas clasificaciones al encontrar el hiperplano. Teniendo todo esto en cuenta, el modelo se formula a través del problema de optimización:

$$\min \|\vec{\beta}\| = \sqrt{\sum_{j=1}^{p} \beta_j^2} \tag{2.10}$$

$$\text{sujeto a } y_i(\vec{x}_i'\vec{\beta} + \beta_0) \geq 1 - \xi_i, \;\; i = 1, \ldots, n \tag{2.11}$$

$$\sum_{i=1}^{n} \xi_i \leq K \tag{2.12}$$

$$\xi_i \geq 0, \;\; i = 1, \ldots, n \tag{2.13}$$

En la práctica, se suele usar la siguiente reformulación equivalente del problema:

$$\min \frac{1}{2}\|\vec{\beta}\|^2 + C\sum_{i=1}^{n} \xi_i \tag{2.14}$$

$$\text{sujeto a } y_i(\vec{x}_i'\vec{\beta} + \beta_0) \geq 1 - \xi_i, \;\; i = 1, \ldots, n \tag{2.15}$$

$$\xi_i \geq 0, \;\; i = 1, \ldots, n \tag{2.16}$$

En esta reformulación, las variables de holgura ya no están acotadas. En vez de eso, sus valores son controlados al añadirlas a la función objetivo. Aparece ahora el hiperparámetro C, que representa la penalización por realizar malas clasificaciones. Como el problema es de minimizar, se está permitiendo al modelo realizar un número más elevado de malas clasificaciones cuanto menor sea el valor de C, mientras que un C grande penalizará en gran medida estas malas clasificaciones.

La principal restricción de este nuevo modelo, que ya no requiere que las observaciones sean linealmente separables, es que sigue suponiendo que la frontera entre las regiones de las categorías de la respuesta sea lineal. Puede demostrarse que la solución al problema de optimización toma la forma

$$f(\vec{x}) = \beta_0 + \sum_{i=1}^{n} \alpha_i \langle \vec{x}, \vec{x}_i \rangle, \tag{2.17}$$

donde $\langle \vec{x}, \vec{x}_i \rangle$ es el producto interior entre el vector de predictores de una observación a clasificar y la observación i-ésima de entrenamiento respectivamente, y α_i, $i = 1, \ldots, n$ son unos coeficientes desconocidos que cumplen que $\alpha_i = 0$ si la observación i-ésima de entrenamiento fue correctamente clasificada y se encuentra a una distancia del hiperplano mayor que el margen M. Esto quiere decir que estos puntos, no afectan al cálculo del hiperplano. En consecuencia, el modelo es robusto para observaciones atípicas que se encuentren a gran distancia de las demás. Las observaciones que influyen en la expresión del hiperplano, es decir, las incorrectamente clasificadas y las correctamente clasificadas que están a una distancia del hiperplano menor o igual que el margen M, reciben el nombre de vectores soporte.

Otra pega de este modelo es que la hipótesis de frontera lineal supone una limitación en cuanto al número de problemas para los cuales tiene sentido aplicarlo. En Boser et al. (1996), los autores eliminan esta hipótesis sustituyendo el producto interior $\langle \vec{x}, \vec{x}_i \rangle$ de la expresión anterior por una función núcleo general $K(\vec{x}, \vec{x}_i)$, que cambiará en función del problema. Hasta ahora habríamos visto el caso particular que toma el núcleo lineal

$$K(\vec{x}, \vec{x}_i) = \langle \vec{x}, \vec{x}_i \rangle \tag{2.18}$$

obteniendo un hiperplano. Algunos núcleos no lineales comúnmente empleados son el polinomial

$$K(\vec{x}, \vec{x}_i) = (\gamma \langle \vec{x}, \vec{x}_i \rangle + c_0)^d, \tag{2.19}$$

el radial

$$K(\vec{x}, \vec{x}_i) = \exp\{-\gamma \|\vec{x} - \vec{x}_1\|^2\}, \tag{2.20}$$

o la tangente hiperbólica

$$K(\vec{x}, \vec{x}_i) = \tanh(1 + \gamma \langle \vec{x}, \vec{x}_i \rangle). \tag{2.21}$$

Estos núcleos permiten fronteras entre las categorías mucho más flexibles. En las expresiones anteriores, los valores γ, y de c_0 y d en el caso concreto del núcleo polinomial, son hiperparámetros que deberán ser hallados por algún método de selección de modelos, al igual que el tipo de núcleo que se va a emplear. En particular, el parámetro γ representa el radio de influencia de cada observación de manera individual, con valores grandes del parámetro siendo equivalentes a radios más pequeños.

A la hora de aplicar el modelo de máquinas de soporte vectorial a un problema de clasificación multiclase, es posible tomar el enfoque *one-versus-all* (OVA) o el *one-versus-one* (OVO). Supongamos que hay m clases en total, el enfoque OVA entrena, para cada grupo, un modelo que separe a los miembros de ese grupo de los miembros de todos los demás grupos, haciendo un total de m modelos. Por otro lado, el enfoque OVO aplica el algoritmo para cada par de clases, ignorando para cada par a las observaciones de todos los demás grupos, y entrenando para cada par un modelo de clasificación binaria, para un total de $\frac{m(m-1)}{2}$ modelos.

2.3. K-vecinos más próximos

El modelo de k vecinos más próximos es un tipo de algoritmo de clasificación y regresión supervisado. Se basa en la idea de realizar las predicciones en función de las respuestas de las observaciones más próximas, con respecto a los valores de los predictores. En esta sección nos centraremos en la versión del algoritmo para realizar clasificación, que es la que emplearemos en el trabajo. Este modelo fue presentado por primera vez en Fix y Hodges (1989), y expandido posteriormente en Altman (1992).

Supongamos que sobre el conjunto de los n datos de entrenamiento se han medido p variables predictoras diferentes. Entonces los datos se pueden situar en un espacio p-dimensional. Se fija entonces un hiperparámetro k, y se considera una distancia en ese espacio. Para realizar la clasificación de una observación, se buscan las k observaciones del conjunto de entrenamiento que estén más próximas a esta (con respecto a la distancia que estemos considerando), y se toma como predicción la moda de las respuestas de estas k observaciones más próximas.

Este modelo genera en el espacio de predictores una frontera de decisión en función de las categorías de la respuesta. Para valores bajos de k, esta frontera es más fragmentada, y pueden darse problemas de sobreajuste. En cambio, seleccionar un k demasiado elevado puede causar infraajuste en el modelo. Por eso es necesario obtener el valor de k mediante un método de selección de hiperparámetros.

Este algoritmo presenta problemas cuando las clases están altamente desbalanceadas, ya que la clase predominante puede acabar afectando demasiado a las predicciones de otros grupos. Para evitar esto, en vez de realizar la clasificación simplemente por la moda, se puede ponderar la clase de los k datos más cercanos en función de su distancia al punto a clasificar. Por tanto, se le daría más peso en la clasificación a los datos que se encuentren más cerca de la observación. Esto consigue reducir el sesgo sin aumentar en gran medida la varianza (Altman 1992).

2.4. Perceptrones multicapa

Los perceptrones multicapa son un tipo de modelos de clasificación de aprendizaje supervisado. La estructura de estos modelos está compuesta por capas; la primera de ellas es la capa de entrada que contiene las variables originales, mientras que la última es la capa de salida, donde están las predicciones. Entre ambas, hay una o más capas denominadas ocultas que es donde ocurren todas las operaciones que realiza el modelo.

Una versión altamente simplificada de este tipo de modelos es el perceptrón. El perceptrón es un modelo de clasificación binaria, que calcula la combinación lineal de las p variables predictoras $x_1, \ldots, x_p$, mediante la expresión

$$\sum_{i=1}^{p} w_i x_i + w_0, \tag{2.22}$$

donde w_i, $i = 1, \ldots, p$ son los pesos, y w_0 se conoce como sesgo (todos estos parámetros se obtienen durante el entrenamiento). Una vez calculada la suma anterior, realiza la clasificación en base a su signo. Sea $\vec{x} = (x_1, \ldots, x_p)$, se calcula

$$f(\vec{x}) = \begin{cases} 1, & \sum_{i=1}^{p} w_i x_i + w_0 \geq 0 \\ 0, & \sum_{i=1}^{p} w_i x_i + w_0 < 0 \end{cases} \tag{2.23}$$

Es decir, que el modelo asigna unos pesos a las variables en función de su importancia para la predicción, y si la suma de los valores de las variables ponderados por sus pesos supera al sesgo o *threshold*, se clasifica a la observación como 1, y en caso contrario como 0. Este modelo, presentado en Rosenblatt (1957) tiene el problema de que es demasiado simple, y está limitado por su estructura lineal. En particular, no es posible representar la estructura lógica XOR con un perceptrón. Los perceptrones multicapa existen con el fin de superar estas limitaciones.

El perceptrón simple que acabamos de ver tiene una única capa oculta, que simplemente supone una transformación lineal de los datos. El perceptrón multicapa en cambio puede tener más de una capa oculta, y estas capas ocultas estarán compuestas a su vez por nodos. Cada uno de estos nodos puede ser entendido como un perceptrón simple. Es decir, que cada una de las k capas ocultas está compuesta por N_i nodos o perceptrones simples, cada uno de ellos con sus propios pesos.

Una vez se calcula en cada nodo la combinación lineal de sus entradas, a esta se le aplica una transformación no lineal a través de una función de activación, que elimina la linealidad del modelo. Las más comúnmente utilizadas son la función sigmoide

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2.24}$$

y el rectificador (*rectified linear unit*, ReLU)

$$f(x) = \max\{0, x\}. \tag{2.25}$$

Las capas del modelo están conectadas en serie, lo cual quiere decir que las salidas de los nodos de una capa serán las entradas de los nodos de la siguiente. Además, hay una conexión entre todo par de nodos de dos capas consecutivas, por lo que cada nodo de una capa tiene tantas entradas como la cantidad de nodos de la capa anterior. Sin embargo, la información solo puede ir en un sentido, así que una capa no puede aprender de la siguiente. Por esto se dice que el modelo es prealimentado, o *feed-forward*.

Durante el entrenamiento, el modelo deberá buscar los coeficientes y sesgos en cada nodo. Esto se hace mediante un método iterativo: en cada paso, se calcula una función de coste para las predicciones (como el error cuadrático medio), se aplica un algoritmo de descenso de gradiente, y se actualizan los valores de los parámetros para la siguiente iteración, hasta cumplir un criterio de parada. Este tipo de entrenamiento se conoce como retropropagación, o *backpropagation*.

Al diseñar un perceptrón multicapa, es necesario fijar los hiperparámetros del modelo, que serían el número de capas ocultas, y la cantidad de nodos en cada capa oculta. Existen estudios acerca de la mejor manera de fijar estos hiperparámetros (Sheela y Deepa 2013). Sin embargo, en muchos casos se recomienda realizar una búsqueda por ensayo y error, ya que el comportamiento del modelo depende enormemente de la estructura de los datos.

Como ya dijimos, la principal ventaja del perceptrón multicapa es que permite estructuras muy complejas. Sin embargo, esto viene a costa de unos tiempos de entrenamiento muy elevados, por lo

que encontrar unos hiperparámetros del modelo que funcionen bien puede acabar siendo una tarea bastante laboriosa.

2.5. K medias

El algoritmo de k medias es el único modelo de aprendizaje no supervisado que veremos en esta sección. Se trata de un modelo de *clustering*, o creación de grupos, cuya idea consiste en dividir una muestra en k grupos de manera que se minimice la varianza de los predictores dentro de cada grupo formado. La primera versión de este algoritmo fue presentada en Lloyd (1957), aunque no fue publicada hasta 25 años más tarde, en Lloyd (1982). Su primer uso público tuvo lugar en MacQueen (1967).

El modelo realiza la creación de los grupos de manera iterativa, y a partir de sus centroides. En cada iteración, el modelo parte de k centroides obtenidos en el paso anterior (o inicializados como veremos a continuación para la primera iteración). Se calcula a qué grupo pertenece cada observación de entrenamiento en función del centroide que tenga a menor distancia y, una vez se tengan los grupos formados, se recalcula el centroide de cada uno, que será utilizado en la siguiente iteración. Al moverse los centroides de los grupos en cada paso del modelo, también se espera que los grupos que definen varíen, al cambiar para ciertas observaciones el centroide que tengan más próximo.

Es decir, que partimos de una muestra $\vec{x}_1, \ldots, \vec{x}_n$ de n vectores p-dimensionales de variables. Una vez fijado el parámetro k, el primer paso es inicializar los k centroides, para lo cual existen métodos de complejidad variable. Para complejidad lineal, podemos por ejemplo asignar a cada observación un grupo inicial aleatoriamente. Para complejidad cuadrática existe entre otros el modelo presentado en Kaufman y Rousseeuw (1990), que toma como primer centroide la media de todas las observaciones, y va obteniendo los $k-1$ restantes uno a uno como los puntos que más minimicen la suma de residuos al cuadrado.

En Celebi et al. (2013) se realiza una comparación de varios de estos métodos. Los autores consideran que, para aprovechar al máximo la complejidad lineal del algoritmo de k medias, lo mejor es emplear también un método de inicialización de complejidad lineal. Uno de los métodos que mejores resultados muestra en todos los tests que realizan es el presentado en Bradley y Fayyad (1998).

Este método particiona la muestra de entrenamiento en J grupos de manera aleatoria, y a cada uno de estos grupos se le aplica el algoritmo de k medias, inicializando en cada grupo los k centroides como k observaciones elegidas al azar (con la idea de que hay más probabilidad de que pertenezcan a zonas de alta densidad de datos, y por tanto sean representativas). Se obtienen así J conjuntos de centroides, con k centroides cada uno, los cuales se juntan en un solo conjunto más grande, al cual se denota por C. A este conjunto C se le aplica el algoritmo de k medias J veces, donde cada una de las veces se inicializa con uno de los J conjuntos diferentes de centroides obtenidos antes. Se busca cual de los J algoritmos produce la solución con una menor suma de errores al cuadrado sobre el conjunto C, y se usa esta solución como distribución inicial de centroides para el algoritmo. El método es no determinista, ya que no se garantiza que la configuración de centroides que devuelve sea la misma en cada ejecución.

Ahora que hemos visto como inicializar los centroides, podemos pasar a describir cada iteración del modelo. El primer paso consiste en asignar un grupo a cada dato de la muestra de la siguiente manera: partiendo de los centroides $\vec{c}_1, \ldots, \vec{c}_k$ obtenidos en el paso anterior, el dato $\vec{x}_j$, $j = 1, \ldots, n$ será asignado al grupo i tal que cumple

$$d(\vec{x}_j, \vec{c}_i) = \min_{t \in \{1,\ldots,k\}} d(\vec{x}_j, \vec{c}_t), \tag{2.26}$$

siendo $d(\vec{x}_j, \vec{c}_i)$ la distancia que estemos considerando en el espacio p-dimensional de predictores medida entre la observación de entrenamiento j-ésima y el centroide i-ésimo. Es decir, que a cada observación se le asigna el grupo cuyo centroide se encuentre a menor distancia de la misma.

El siguiente paso es actualizar el valor de los centroides. Sean $\vec{x}^i_1, \ldots, \vec{x}^i_{n_i}$ las n_i observaciones que están actualmente clasificadas como pertenecientes al grupo i, $i = 1, \ldots, k$. Entonces se considerará

como nuevo centroide del grupo i-ésimo al punto

$$\vec{c}_i = \frac{1}{n_i} \sum_{l=1}^{n_i} \vec{x}_l^i. \tag{2.27}$$

Una vez hayamos obtenido los nuevos centroides habrá que volver al paso anterior de asignar un grupo a cada observación, y esta nueva asignación producirá a su vez un nuevo grupo de centroides. Como criterio de parada se puede fijar un número máximo de iteraciones, pero teniendo en cuenta que el algoritmo se detendría antes de tiempo si en algún paso no ocurren reasignaciones con respecto al grupo de cada observación (ya que en este caso al recalcular los centroides se obtendrían los mismos que en el paso anterior, y el resultado no cambiaría hasta la iteración final del algoritmo).

A la hora de entrenar este modelo, habrá que seleccionar de antemano el número de grupos a formar. Dependiendo de la estructura de los datos, es posible que sea necesario seleccionar un k mayor al número de categorías posibles para la respuesta que existen (o que sospechamos que existe). En tal caso, habría al menos un par de grupos creados por el algoritmo que se corresponden con una única categoría de la respuesta.

Se trata de un modelo sencillo, de complejidad lineal, ya que tan solo requiere calcular el centroide de cada grupo (que simplemente es la media de los predictores), y una distancia como por ejemplo la euclídea. Sin embargo, tiene como uno de sus principales problemas que los datos atípicos afectan en gran medida a los grupos formados, hasta el punto de que una observación atípica puede acabar siendo el único integrante de uno de los k grupos.

2.6. Validación cruzada

A lo largo de este capítulo hemos visto múltiples modelos de aprendizaje, cada uno de ellos con al menos un hiperparámetro que debe ser fijado antes del entrenamiento, y cuyo valor dependerá del problema particular con el que se trabaje. En esta sección veremos cómo fijar de manera óptima los valores de estos hiperparámetros.

Una de las maneras más sencillas de hacerlo es mediante el uso de un conjunto de validación. Generalmente cuando se construye un modelo de aprendizaje, se divide el conjunto de datos que se va a emplear en un conjunto de entrenamiento y otro conjunto de test. El conjunto de entrenamiento es el que se alimenta al modelo durante su creación para que aprenda a realizar predicciones sobre nuevos datos. Una vez entrenado el modelo, se calculan las predicciones para los datos del conjunto de test, y se comparan estas predicciones con sus valores reales de la variable respuesta. Como los datos del conjunto de test no han formado parte de la creación del modelo, se puede emplear estas comparaciones para evaluar el rendimiento del modelo, tomando por ejemplo la media de los errores al cuadrado para un modelo de regresión, o la proporción de clasificaciones acertadas para un modelo de clasificación.

En caso de que se quiera realizar una selección de modelos, como al seleccionar el valor de un hiperparámetro, se puede dividir la muestra en 3 conjuntos distintos: el conjunto de entrenamiento, el de validación y el de test. En primer lugar se entrena a partir de la muestra de entrenamiento a todos los modelos entre los cuales se quiera elegir. A continuación, se mide empleando el conjunto de validación el rendimiento de cada uno de los modelos, de igual manera que se hizo antes sobre el conjunto de test. Se selecciona el modelo con el mejor rendimiento sobre el conjunto de validación, y se evalúa su rendimiento final con el conjunto de test. La razón por la que hay que distinguir entre el conjunto de validación y el de test es para evitar un posible sobreajuste en los datos de validación.

El empleo de los datos de validación requiere que la muestra sea lo suficientemente grande como para poder ser dividida en 3 conjuntos diferentes. En la práctica, se suele realizar la selección de modelos mediante métodos de validación cruzada, que tan solo requiere dividir la muestra en un conjunto de entrenamiento y otro de test.

La versión más básica de validación cruzada es la denominada como validación cruzada dejando uno fuera, o *leave-one-out cross-validation* (Stone 1974). Esta consiste en, para cada dato de la muestra

de entrenamiento, entrenar el modelo empleando todo el conjunto de entrenamiento con excepción de ese dato particular, que será utilizado para evaluar el modelo como si se tratase de un conjunto de validación formado por una sola observación. Se obtiene así una medida de error para las predicciones para cada dato del conjunto de test, y al promediar estas medidas se habrá obtenido una medida global para evaluar las predicciones del modelo haciendo uso únicamente del conjunto de test. Se realiza este procedimiento para cada modelo considerado y se selecciona el que haya mostrado una mejor capacidad predictiva. Finalmente se calcula el rendimiento final del modelo seleccionado a partir del conjunto de test.

La validación cruzada dejando uno fuera requiere entrenar cada uno de los modelos para cada observación de entrenamiento, lo cual puede resultar en tiempos de computación extremadamente altos. Para solucionar esto se puede emplear la validación cruzada en k iteraciones, o *k-fold cross-validation*, que divide el conjunto de entrenamiento en k grupos, y realiza un procedimiento similar al de la validación cruzada dejando uno fuera, pero empleando estos k grupos en lugar de cada una de las observaciones de manera individual. Es decir, que para cada uno de los k grupos, se entrena el modelo sobre el conjunto formado por los otros $k-1$ grupos, y se evalúa su rendimiento sobre el grupo no utilizado. Se realiza el promedio del rendimiento sobre todos los grupos, y después de hacer esto para cada modelo considerado, se selecciona el modelo con el mejor rendimiento promedio.

2.7. Evaluación de las predicciones de un modelo

Una vez entrenado un modelo de aprendizaje, es necesario evaluar su capacidad predictiva sobre un conjunto de test antes de trabajar con él. Para ello se pueden emplear una serie de medidas acerca de lo buenas que son las predicciones, y que serán diferentes para los modelos de regresión y de clasificación.

Comenzaremos con las de los modelos de regresión. Supondremos que el conjunto de test está formado por n observaciones, $X_1, \ldots, X_n$, y sean $Y_1, \ldots, Y_n$ sus respuestas, e $\hat{Y}_1, \ldots, \hat{Y}_n$ las predicciones de estas respuestas a través del modelo de regresión. El error de la i-ésima predicción vendrá dado por $Y_i - \hat{Y}_i$. Nos interesa encontrar el modo de resumir estos n errores en un único valor más fácilmente interpretable. La primera medida que usaremos será el error medio, que es simplemente la media de todos los errores de las predicciones:

$$ME = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i) \tag{2.28}$$

El error medio nos aporta información acerca de posibles sesgos positivos o negativos en las predicciones. Si simplemente queremos estudiar la magnitud de las desviaciones sin importarnos su signo podemos usar el error medio absoluto, que es la media de los errores en valor absoluto:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|Y_i - \hat{Y}_i|, \tag{2.29}$$

O alternativamente la raíz de la media de los errores al cuadrado:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2} \tag{2.30}$$

Para los modelos de clasificación, se acostumbra a representar los resultados obtenidos en una matriz de confusión, que es una tabla que recoge las observaciones de test divididas según su clase real y su predicción a través del modelo. Para el caso de clasificación binaria, con clases $C1$ y $C2$, supongamos que la clase positiva es $C1$. La matriz de confusión recoge los verdaderos positivos y negativos (elementos clasificados correctamente en las clases positiva y negativa, *true positive* y *true negative*, **TP** y **TN**), y los falsos positivos y negativos (elementos clasificados incorrectamente en las

clases positiva y negativa, *false positive* y *false negative*, **FP** y **FN**), como se puede ver en la tabla 2.1, donde R representa las clases reales y P las predicciones.

Cuadro 2.1: Ejemplo de una matriz de confusión

R\P	C1	C2
C1	TP	FN
C2	FP	TN

Sobre esta tabla se pueden calcular una serie de medidas para evaluar las predicciones del modelo. En esta sección veremos la precisión, la sensibilidad, la puntuación $F1$ y la precisión global (Sokolova y Lapalme 2009). La precisión es la proporción de individuos clasificados en la clase positiva que realmente pertenecen a dicha clase, es decir,

$$PPV = \frac{TP}{TP + FP} \tag{2.31}$$

La sensibilidad es la proporción de individuos de la clase positiva que fueron clasificados correctamente,

$$TPR = \frac{TP}{TP + FN} \tag{2.32}$$

La puntuación $F1$ es la media armónica de las dos medidas anteriores,

$$F1 = \frac{2 \cdot PPV \cdot TPR}{PPV + TPR} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \tag{2.33}$$

Finalmente la precisión global es la tasa total de aciertos del modelo, es decir, la proporción de individuos de todo el conjunto que fueron clasificados correctamente,

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.34}$$

La última medida de evaluación de un modelo que mencionaremos en esta sección es la curva ROC (*Receiver Operating Characteristic*). Se puede encontrar información detallada en Hajian-Tilaki (2013), pero la idea básica consiste en representar en una curva la sensibilidad contra la especificidad, que es

$$TNR = \frac{TN}{TN + FP} \tag{2.35}$$

Se utilizará como medida de la capacidad predictiva del modelo el área encerrada bajo dicha curva, que será un valor entre 0 y 1. Cuanto mayor sea este área, mejor es la capacidad predictiva. Un clasificador aleatorio se correspondería con un área bajo la curva ROC de 0,5.

Capítulo 3

Reconocimiento facial

En este capítulo veremos cómo realizar un análisis de las imágenes que forma parte de nuestro conjunto de datos, para obtener información como la edad o el género de sus autores. La propuesta que realizamos es emplear la aplicación de Face++ para realizar este análisis. Sin embargo, previamente intentaremos evaluar la precisión de dicha aplicación, ya que los datos de los que partimos no están etiquetados. Ya existen algunos estudios previos (Jung et al. 2018) que comparan la calidad de Face++ con la de otras aplicaciones de detección y reconocimiento facial. En nuestro caso buscamos realizar un análisis algo más exhaustivo a través de un número mayor de atributos, para lo cual emplearemos el conjunto de datos FairFace presentado en Kärkkäinen y Joo (2019).

FairFace es un conjunto de datos formado por 86744 imágenes de caras, todas ellas con etiquetas de edad, género y raza. La edad no viene dada por un número exacto, sino por un rango, siendo los posibles rangos $0-2$, $3-9$, $10-19$, $20-29$, $30-39$, $40-49$, $50-59$, $60-69$ y 70 o más. En el caso de la raza, hay 7 categorías distintas: White (**W**), Black (**B**), Indian (**In**), East Asian (**EA**), Southeast Asian (**SEA**), Middle Eastern (**ME**) y Latino (**Lat**). Una ventaja que presenta este conjunto de datos es que está relativamente equilibrado para los tres atributos, lo cual contrasta con los sesgos de género y raza que existen en muchos otros conjuntos usados (Kiritchenko y Mohammand 2018, Shankar et al. 2017).

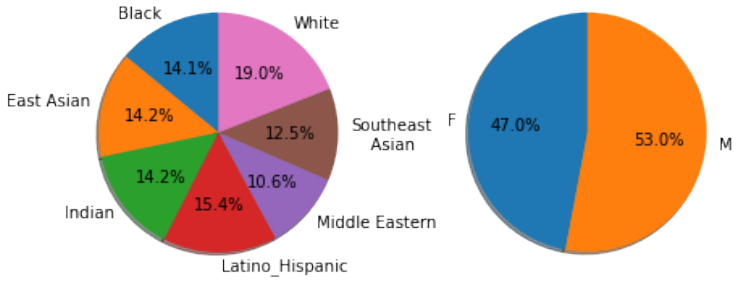

Figura 3.1: División del conjunto de datos FairFace para los atributos de la raza y el género.

En la figura 3.1 podemos ver cómo se divide el conjunto de FairFace con respecto a los atributos de género y raza. Para el género, observamos que hay una diferencia de un 6 % entre la cantidad de hombres y mujeres en las imágenes, en favor de los hombres. Para la raza, tenemos que están infrarrepresentadas las categorías del sudeste asiático, y particularmente la de oriente medio, mientras que la de gente blanca está sobrerrepresentada (19 %, contra el 14,28 % que debería tener bajo igual representación).

En la figura 3.2 vemos, para cada categoría del atributo de la raza, la proporción de hombres y mujeres en el conjunto de datos. Aunque hay un par de categorías para las cuales el número de hombres

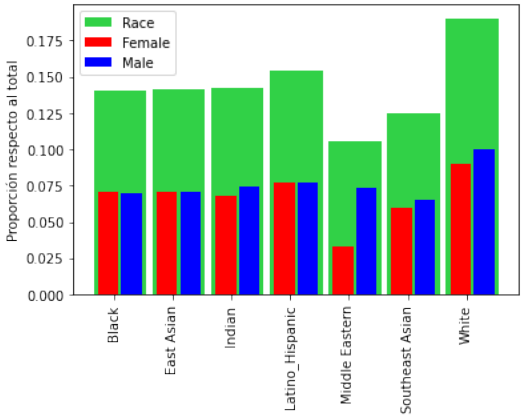

Figura 3.2: Proporción del género para cada raza.

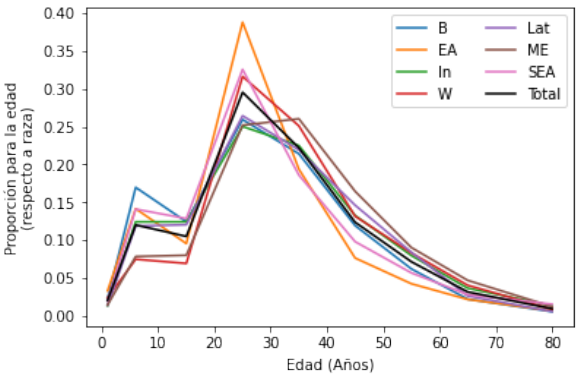

Figura 3.3: Evolución de la edad para cada raza.

es ligeramente mayor que el de mujeres (**In**, **SEA**, **W**), lo más destacable es que dentro de la categoría de oriente medio el número de hombres es más del doble que el de mujeres.

Por último, en la figura 3.3 vemos la distribución de las edades para cada raza. Lo más destacable en este caso es que hay mucha gente de color menor de 10 años, y muy poca blanca y de oriente medio. Por otro lado, hay un pico excesivo de gente del este asiático de entre 20 y 30 años, y poca gente de esta etnia de más de 40 años.

Ahora que hemos estudiado la estructura del conjunto de datos FairFace, pasaremos a analizar el rendimiento de Face++ sobre este. El primer paso que vamos a tomar es analizar su capacidad de reconocimiento facial. Para las 86744 imágenes que componen la base de datos, nos encontramos con que únicamente en 309 de ellas no se detecta ninguna cara, lo cual supone una capacidad de detección superior al 99,6 %. En la figura 3.4 se ha representado en un gráfico de barras la proporción dentro de cada categoría (para las 18 categorías que existen a lo largo de los 3 atributos) de fotografías en las cuales no se ha detectado ninguna cara. Puede verse que en casi todos los casos se obtiene una precisión mayor al 99 %.

La principal excepción ocurre dentro del atributo de la edad. La capacidad de detección va disminuyendo ligeramente a medida que aumenta la edad de la persona representada en la fotografía, hasta que para personas mayores de 60 años hay más de un 1 % de los casos en los cuales no fueron detectados. Sin embargo, se tratan de valores tanto relativos como absolutos (28 fotografías para $60-69$, 9 para más de 70 años) muy pequeños. Con respecto a los demás atributos, tenemos que los hombres se detectan ligeramente mejor que las mujeres; y para la raza los peores resultados se obtienen en la gente blanca, la gente de color, y particularmente para la gente de oriente medio. Sin embargo, la proporción de rostros no detectados sigue siendo lo suficientemente pequeña en todas las categorías como para que estemos satisfechos con la capacidad de detección facial de Face++, y podamos pasar a estudiar su comportamiento a la hora de predecir edad y género.

El siguiente paso que tomaremos será por lo tanto estudiar la capacidad de Face++ para predecir el género. Solo podremos trabajar con las 86435 imágenes en las cuales se detectó una cara. Comenzaremos viendo la matriz de confusión para la predicción del género de todo este conjunto de datos, en términos absolutos y relativos, en la tabla 3.1.

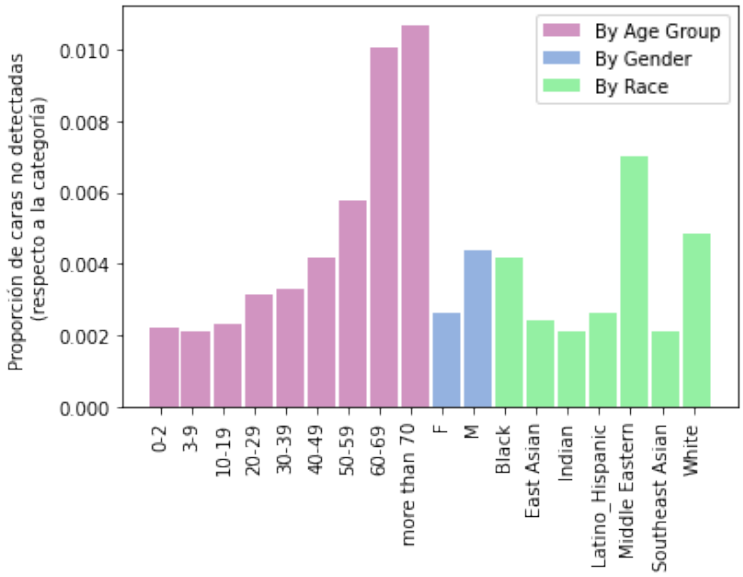

Figura 3.4: Proporción de imágenes en las cuales no se detecta ninguna cara, para cada categoría.

Cuadro 3.1: Matrices de confusión para la predicción de género de Face++

Matriz en términos absolutos

	M	F	Total real
M	39056	6729	45785
F	6556	34094	40650
Total predicción	45612	40823	86435

Matriz en términos relativos

	M	F	Total real
M	0.452	0.078	0.530
F	0.076	0.394	0.470
Total predicción	0.528	0.472	1

La precisión global de aciertos es de 0,846. A partir de la matriz de confusión vamos a poder calcular las medidas para evaluar la clasificación vistas en 2.7, las cuales recogemos en la tabla 3.2.

Cuadro 3.2: Medidas de error para la predicción del género

	Precisión	Sensibilidad	Puntuación F1
M	0.856	0.853	0.854
F	0.835	0.839	0.837
Media	0.845	0.846	0.845
Media ponderada	0.846	0.846	0.846

Vemos que todas las medidas se sitúan entre 0,83 y 0,86, por lo que parece tratarse de un clasificador aceptable. Todos los valores son mayores para el caso de los hombres, lo cual indica que son ligeramente mejor clasificados que las mujeres. Además, para los hombres se tiene que la precisión es muy ligeramente mayor que la sensibilidad, mientras que para las mujeres ocurre lo contrario. Es-

to muestra que el problema de hombres clasificados de manera errónea como mujeres es ligeramente mayor que el de mujeres incorrectamente clasificadas como hombres.

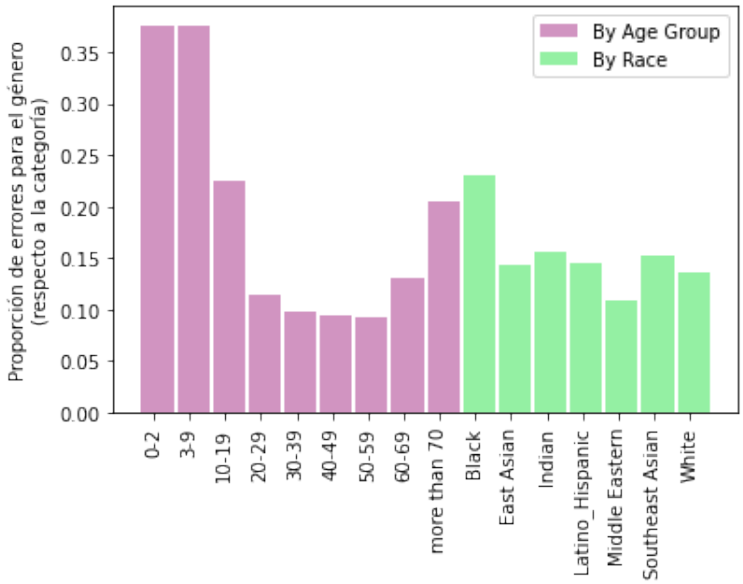

Figura 3.5: Proporción de imágenes en las cuales la predicción del género fue incorrecta, para cada categoría.

En la figura 3.5 podemos ver un desglose de la proporción de los errores de predicción de género para cada categoría de edad y raza, con respecto al total de la categoría. Lo que más destaca en un primer vistazo es que para las personas de menos de 9 años, la proporción de predicciones incorrectas es superior al 35 %. Sin embargo, esto no afecta a nuestro trabajo, ya que nuestro conjunto de datos está formado por usuarios de redes sociales, que no deberían tener edades tan bajas. Los siguientes grupos que peores resultados presentan son el de $10 - 19$ y el de más de 70, con una proporción de errores algo mayor al 20 %. Asumimos que este primer grupo es el que más podría afectar a nuestro estudio, ya que no sería raro encontrar a gente de unos 18 o 19 años en nuestro conjunto de datos. Por lo demás, los grupos de entre 20 y 69 años presentan generalmente buenos resultados, con unos errores que rondan el 10 %.

Con respecto al atributo de la raza, lo más destacable es que hay casi un 10 % más de errores para la categoría correspondiente con la gente de color con respecto a todas las demás. Ignorando este, los otros 6 grupos oscilan muy próximos entre si en torno al 15 %. Sin embargo, esta diferencia no es excesivamente problemática como sí lo sería por ejemplo, como ya dijimos antes, la de la gente menor de 10 años, en caso de que tuviéramos que trabajar con ellos.

El último paso será estudiar la capacidad de predicción de la edad por parte de Face++. La aplicación devuelve la predicción de la edad como un solo número, por lo que consideraremos como error de cada predicción a la distancia de este punto al intervalo correspondiente con el rango de edad recogido en el conjunto de datos de FairFace. Vamos a analizar las predicciones, en primer lugar a través de unas medidas de error. En particular, calculamos para cada grupo de edad el error medio (**ME**), el error absoluto medio (**MAE**) y la raíz del error cuadrático medio (**RMSE**), vistos todos ellos en la sección 2.7, para las predicciones de la edad respecto a las edades reales, y los recogemos en la tabla 3.3.

Cuadro 3.3: Medidas de error para la predicción de la edad en cada grupo

	ME	MAE	RMSE
0-2	11.346	11.346	17.056
3-9	13.056	13.068	16.564
10-19	8.356	8.457	11.670
20-29	4.853	4.960	8.612
30-39	3.155	4.867	8.022
40-49	2.582	5.983	8.746
50-59	2.151	5.777	8.670
60-69	0.494	5.173	8.436
70 o más	-3.753	3.753	8.939

Como ya dijimos antes, en nuestro conjunto de datos no debería haber gente de menos de 10 años, por lo que no prestaremos mucha atención a esas dos categorías. En primer lugar, vemos que casi todos los errores medios son positivos (con excepción del correspondiente al grupo de 70 años o más, que tiene que ser necesariamente negativo). Esto nos indica que Face++ tiende a dar predicciones para la edad mayores que la realidad, y no menores. Sin embargo, esto no supone un problema tan grande para la aplicación sobre fotos de perfil de una red social, ya que no sería raro que una persona tenga una foto de perfil un par de años más joven de lo que es, simplemente por no haberla cambiado en un tiempo. Y más aún, los filtros de corrección facial tienen un uso tan extendido que incluso un ser humano encontraría problemas a la hora de realizar predicciones acertadas.

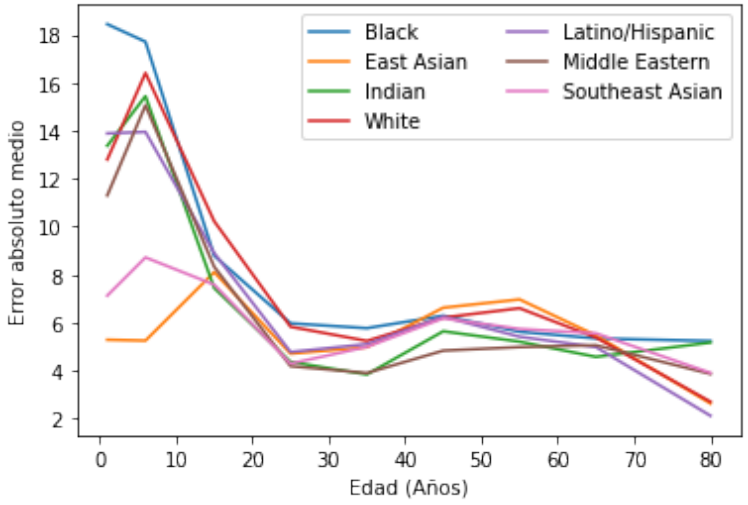

Figura 3.6: Evolución del error absoluto medio (**MAE**) a medida que aumenta la edad, para cada categoría del atributo correspondiente a la raza.

En la columna del error absoluto medio podemos ver la magnitud de los errores cometidos. En general entre los 20 y los 60 años parecen bastante estables, situándose entre 4 y 6. No se tratan de errores excesivamente grandes a simple vista, pero hay que tener en cuenta que el conjunto de datos de FairFace presenta las edades en forma de intervalo. Esto provoca que tengamos el mismo error absoluto si se predice que una persona de 21 años tiene 19, que si predice que tiene 30, ya que en ambos casos la distancia al intervalo de edad $20-29$ es 1. Los errores reales de predicción de la edad serán en realidad mayores de lo que nosotros hemos obtenido.

La última columna nos proporciona información extra acerca de la magnitud de los errores. En particular, en este caso las medidas para los grupos de edad de 20 años o más se encuentran siempre entre 8 y 9. El crecimiento con respecto al **MAE** para el grupo de edad de 70 o más años es con diferencia el mayor de manera relativa, lo cual puede significar que hay número de personas con esta edad y con predicciones muy malas, que causan que aumente el **RMSE** en gran medida.

Lo último que haremos será estudiar si la raza tiene gran efecto sobre los errores de predicción. En particular, hemos representado en la figura 3.6 la evolución del **MAE** para cada una de las razas. Podemos observar que para edades muy tempranas existen diferencias enormes en los errores cometidos (para niños entre 0 y 2 años del este asiático el **MAE** es menor que 6, mientras que para los niños de raza negra es mayor de 18). Sin embargo, esta diferencia se reduce rápidamente al aumentar la edad, y para los grupos de edad que nos interesan en nuestro estudio (a partir de 18 años, ya que trabajamos con usuarios de redes sociales), los errores de todas las categorías se encuentran razonablemente próximos entre si, y son todos razonablemente bajos.

Capítulo 4

Análisis de sentimientos en texto

En este capítulo veremos cómo realizar el análisis de sentimientos para el texto que acompaña las publicaciones de nuestro conjunto de datos. Emplearemos para ello el conjunto de Sentiment Strength presentado en Thelwall et al. (2012). Se trata de una base de datos formada por publicaciones provenientes de distintas redes sociales o foros de internet:

- Foros de la BBC: comentarios publicados en noticias, que constituyen discusiones de temas serios.
- Digg.com: comentarios públicos realizados en noticias, que representan discusiones de noticias de actualidad más generales.
- MySpace: mensajes públicos realizados entre amigos, que representan las comunicaciones en redes sociales.
- Foro de Runners World: mensajes públicos realizados con temática común de carrera de maratón, que representan foros especializados para grupos con intereses comunes.
- Twitter: mensajes públicos que representan la comunicación en forma de microblogging.
- Youtube: comentarios realizados en vídeos que representan comentarios y discusiones asociadas a recursos de información.

De manera global, partimos de un total de 11787 publicaciones. Cada una de ellas ha sido etiquetada manualmente con dos puntuaciones enteras: una que abarca del 1 al 5, y la otra del -1 al -5. La puntuación positiva mide el grado de sentimientos positivos contenidos en el texto, y la puntuación negativa mide el grado de sentimientos negativos. Un mayor valor absoluto en una puntuación en particular se identifica con sentimientos más intensos de ese tipo. Por ejemplo, un texto etiquetado con un 5 y un -2 presenta un sentimiento positivo muy intenso junto con un ligero sentimiento negativo. La razón del doble etiquetado es que los autores consideraron que un solo texto podía contener sentimientos tanto positivos como negativos en distintos grados de intensidad.

Comenzaremos estudiando la eficacia de VADER (Hutto y Gilbert 2014) a la hora de evaluar la polaridad de las publicaciones de este conjunto de datos. VADER se trata de un algoritmo de análisis de sentimientos basado en el léxico. Su idea base es partir de un banco de palabras donde cada una tiene asignada una puntuación positiva, negativa y neutral. Sin embargo, va un paso más allá teniendo también en cuenta el contexto de otras palabras para poder por ejemplo identificar la frase "I am not happy" como puramente negativa a pesar de contener la palabra "happy".

Para realizar el análisis emplearemos por un lado las publicaciones sin alterar (lo que llamamos Raw), y por otro lado las publicaciones tras haber eliminado emoticonos, caracteres especiales y mayúsculas (versión Clean). La razón de esto es que muchas veces se trabaja con las versiones preprocesadas de los textos (lo que sería nuestra versión Clean), pero algunos caracteres especiales o

las mayúsculas, pueden ser empleados para realizar énfasis en ciertas palabras. Esto puede alterar la puntuación otorgada por VADER, y por tanto nos interesa saber también si es conveniente realizar el paso de limpieza o no.

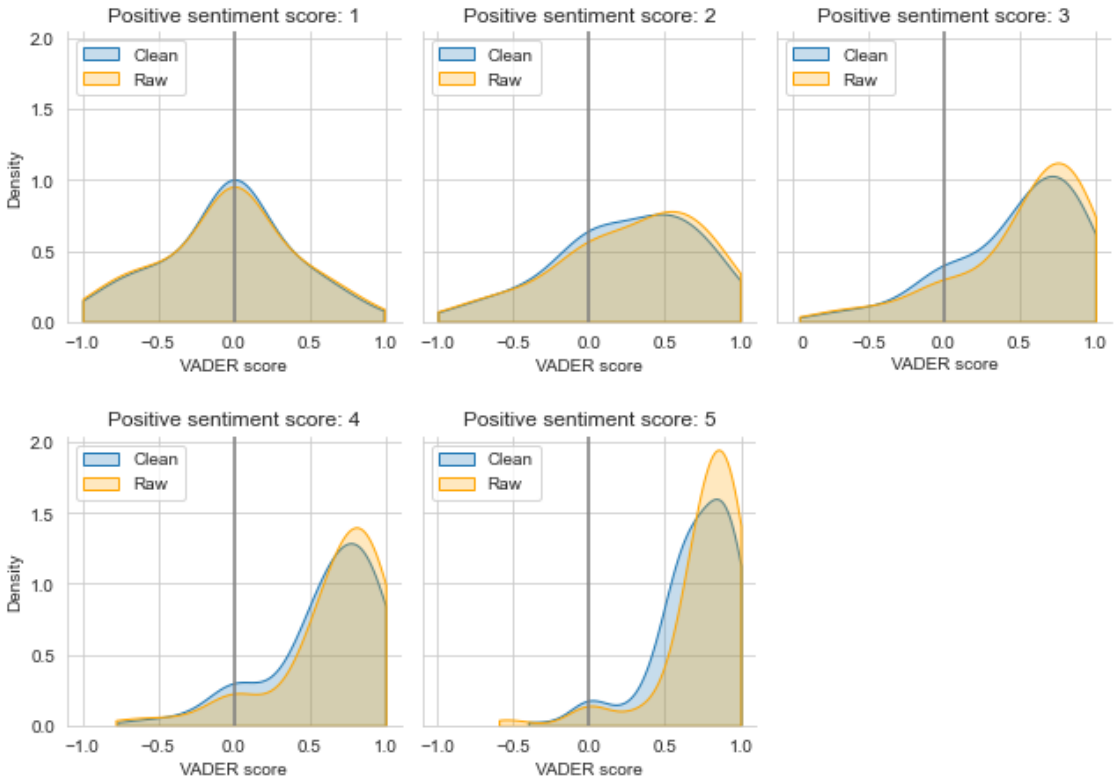

Figura 4.1: Densidad de la puntuación de VADER para cada etiqueta de sentimientos positivos.

En 4.1 hemos representado, para los textos con y sin el paso de limpieza, el estimador núcleo de la densidad de las puntuaciones otorgadas por VADER, para cada una de las 5 etiquetas de sentimientos positivos. En particular, podemos ver que para etiquetas mayores o iguales que 3, la densidad para las puntuaciones de VADER negativas es bastante baja, y hay picos de densidad cada vez más grandes para puntuaciones más positivas. Estos picos no solo van aumentando en tamaño, sino que también se desplazan hacia la derecha, lo cual indica que para las etiquetas más grandes, VADER devuelve puntuaciones cada vez más próximas al 1, que es la máxima puntuación. También puede verse que los picos de densidad son más elevados y se encuentran más a la derecha para los textos que no han pasado por el paso de limpieza, con respecto a los que sí lo han hecho. Esto nos indica que podría ser conveniente no realizar el paso de limpieza, ya que ayuda a VADER a otorgar puntuaciones más altas a los textos muy positivos.

En la figura 4.2 hemos realizado unos gráficos similares a los de 4.1 pero para las puntuaciones negativas. En este caso, tenemos que hasta la etiqueta de sentimientos negativos de 3, VADER ha otorgado más puntuaciones positivas que negativas, y para la etiqueta de 3 está bastante igualado entre puntuaciones positivas y negativas. Para la etiqueta de 4, y sobre todo para la de 5, ya empiezan a aparecer picos importantes de densidad sobre las puntuaciones más negativas, pero aún así la densidad para las puntuaciones positivas es mucho mayor de lo que veíamos en 4.1 para las puntuaciones negativas. Vemos así que VADER es capaz de identificar mejor los sentimientos positivos que los negativos. Además, también vemos que en este caso emplear los textos sin el paso de limpieza disminuye las puntuaciones otorgadas en torno al 0, pero también aumenta ligeramente la densidad

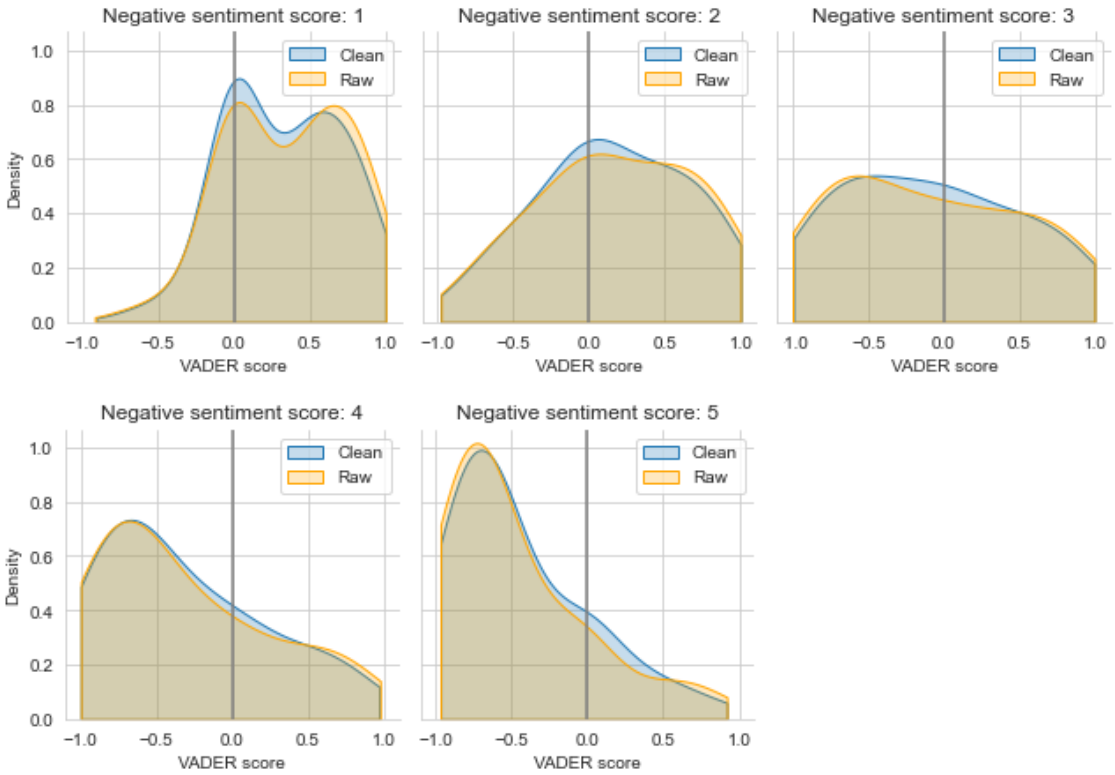

Figura 4.2: Densidad de la puntuación de VADER para cada etiqueta de sentimientos negativos.

para las puntuaciones más positivas, lo cual no debería ocurrir.

Vamos a ver otra manera de visualizar el comportamiento de VADER en este conjunto de datos. En Hutto y Gilbert (2014) se menciona que las puntuaciones de VADER pueden ser empleadas para clasificar un texto en positivo, negativo o neutral en base a unos puntos de corte: si la puntuación es mayor o igual que 0,05 el texto se considera positivo, si es menor o igual que $-0{,}05$ el texto se considera negativo, y si se encuentra entre $-0{,}05$ y 0,05 se considera un texto neutral. En base a esto, estudiaremos las etiquetas de sentimientos negativos y positivos que tenían los textos originalmente, en relación con la categoría en que han sido clasificados por VADER.

En la figura 4.3 representamos los gráficos para los textos clasificados como positivos, negativos y neutrales. Como altura de cada barra hemos considerado la proporción de textos de esa puntuación de sentimientos original que han sido clasificados como positivos (en (a)), o como negativos (en (b)). Podemos ver claramente que VADER tiene tendencia a clasificar los textos como positivos, sobre todo si comparamos los textos según sus sentimientos negativos clasificados como positivos (barras azules en (a)), con los textos según sus sentimientos positivos clasificados como negativos (barras rojas en (b)). Las primeras que mencionamos son considerablemente más altas que las segundas, y llegamos a tener que el 40 % de los textos con puntuación negativa de -3 son clasificados como positivos, contra el 10 % de textos con puntuación positiva de 3 clasificados como negativos. Para que quede claro que esto es un problema del clasificador y no del conjunto de datos (posibilidad de que existan muchos textos con, por ejemplo, puntuación negativa de -3 y positiva de 5 que debieran realmente ser clasificados como negativos), remarcamos que solo el 13 % de los textos con una puntuación negativa de -3 tienen una puntuación positiva de 3 o mayor.

También se ve que los sentimientos negativos tienen que ser mucho más intensos que los positivos para ser clasificados correctamente. El 60 % de los textos con puntuación positiva de 2 son clasificados

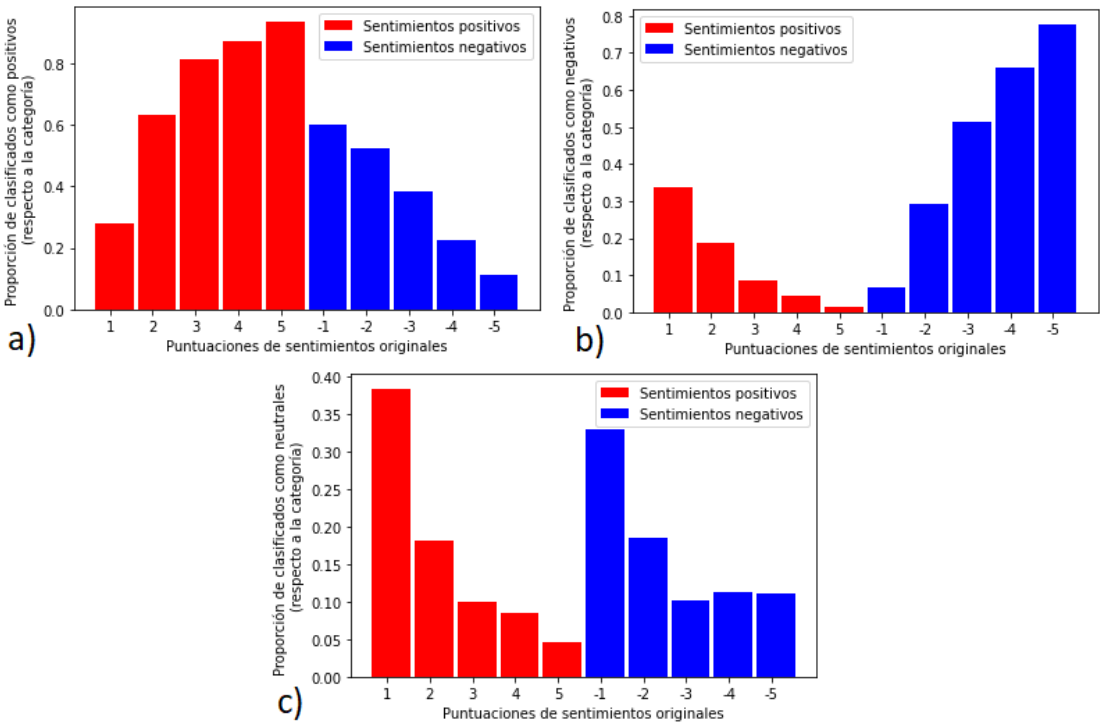

Figura 4.3: Etiquetas de origen para los textos clasificados como positivos (a), negativos (b) y neutrales (c).

como positivos, mientras que tenemos que considerar una etiqueta de -4 para obtener un valor similar para los negativos.

El gráfico (c) es similar a los anteriores, pero para los textos clasificados como neutrales. Para este caso el comportamiento de VADER parece ser bueno, la proporción de publicaciones clasificadas como neutrales comienza siendo alta para puntuaciones bajas de sentimientos tanto positivos como negativos, y disminuye rápidamente para ambos en similar medida. Lo más destacable es que para los sentimientos negativos, esta proporción parece estancarse alrededor del 10 % para etiquetas a partir del -3.

Como ya mencionamos con anterioridad, VADER es en esencia un algoritmo de análisis de sentimientos basado en el léxico. Según el estudio realizado en Kirilenko et al. (2018), estos métodos producen mejores resultados que los basados en aprendizaje automático en caso de que no se realice una calibración del método, entrenándolo en una muestra significativa de los datos. Sin embargo, nos puede interesar saber si un método basado en el aprendizaje automático podría comportarse mejor que VADER con respecto a los sentimientos negativos.

A continuación, entrenaremos un modelo basado en aprendizaje automático que realice una clasificación de los textos en Positivos, Negativos y Neutrales. Ya comentamos con anterioridad cómo se realiza esta clasificación en base a las puntuaciones de VADER: las puntuaciones menores que $-0{,}05$ se corresponden con textos negativos, las superiores a 0,05 con textos positivos, y las que se encuentren entre $-0{,}05$ y 0,05 se clasifican como textos neutrales. Para las etiquetas de origen, consideramos que un texto es neutral si las puntuaciones de sentimientos positivos y negativos son iguales (en valor absoluto). Si, en valor absoluto, la puntuación de sentimientos positivos es mayor que la de sentimientos negativos, etiquetamos el texto como positivos, y en caso de que la puntuación negativa sea mayor, etiquetaremos como texto negativo.

Para realizar la clasificación, emplearemos un modelo de bosque aleatorio. Para obtener las cova-

riables numéricas emplearemos un modelo Word2Vec, introducido en Mikolov et al. (2013). El modelo que creamos será similar al presentado en Basarslan y Kayaalp (2021). Word2Vec se trata de una herramienta de procesado de lenguaje natural no supervisado, que representa palabras que aparecen en un texto en un espacio vectorial del tamaño deseado. Se distinguen dos arquitecturas de aprendizaje diferentes: en la continuous bag of words (CBOW) se realiza una estimación de cada palabra a través de sus palabras colindantes; y en el skip-gram (SG) se estima, para cada palabra, sus palabras colindantes.

El modelo Word2Vec obtiene un vector para cada palabra diferente que aparezca alguna vez a lo largo del conjunto de datos. Para aplicar el modelo de bosque aleatorio, necesitamos obtener vectores correspondientes a cada una de las publicaciones. Consideraremos así, para cada uno de los textos, el vector obtenido al promediar los vectores de cada una de las palabras que componen dicho texto.

Para nuestro caso, hemos aplicado una estructura CBOW con el objetivo de dar una mayor importancia al contexto de cada palabra. Después de eliminar los 22 textos del conjunto de datos que no contenían ninguna palabra (eran solo signos de puntuación), realizamos una división de los datos en conjuntos de entrenamiento, validación y test. Para distintas combinaciones de hiperparámetros (tamaño del espacio vectorial de Word2Vec e hiperparámetro del bosque aleatorio) entrenamos un modelo y lo evaluamos sobre el conjunto de validación. Obtenemos que el modelo óptimo es el que emplea un espacio vectorial para Word2Vec de dimensión 75, y un bosque aleatorio formado por 80 árboles. La precisión final obtenida sobre el conjunto de test es de 49,29 %, contra un 58,1 % obtenido con VADER sobre el mismo conjunto. Representamos en las tablas 4.2 y 4.1 más medidas para evaluar de manera más precisa cada modelo.

Si comparamos ambas tablas podemos notar que el modelo derivado de Word2Vec es mejor que VADER únicamente en la sensibilidad y la puntuación $F1$ de los textos neutrales. Para todas las demás medidas, VADER parece ser superior. También vemos en ambas tablas que la sensibilidad de los textos positivos es considerablemente mayor que su precisión, lo cual nos indica que hay demasiados textos en ambos casos que, sin ser positivos, están siendo clasificados como tal. Para los textos neutrales y negativos, la precisión es mayor que la sensibilidad, lo cual nos dice que muchos textos de esas categorías están siendo clasificados incorrectamente, pero que si un texto es clasificado en esas categorías, entonces es más probable que la clasificación sea correcta. Podemos ver por último que la puntuación $F1$ de los textos negativos para el modelo Word2Vec es particularmente baja, por lo que el modelo se comporta especialmente mal a la hora de clasificar textos negativos.

Cuadro 4.1: Varias medidas de precisión para evaluar el modelo Word2Vec

Word2Vec

	Precisión	Sensibilidad	Puntuación F1
Positivo	0.504	0.606	0.550
Neutral	0.492	**0.455**	**0.473**
Negativo	0.473	0.382	0.423
Media	0.490	0.481	0.482
Media ponderada	0.491	**0.493**	0.489

Cuadro 4.2: Varias medidas de precisión para evaluar el modelo VADER

VADER

	Precisión	Sensibilidad	Puntuación F1
Positivo	**0.564**	**0.780**	**0.655**
Neutral	**0.549**	0.402	0.464
Negativo	**0.647**	**0.513**	**0.572**
Media	**0.587**	**0.565**	**0.564**
Media ponderada	**0.579**	**0.493**	**0.568**

Capítulo 5

Detección de cuentas personales y de negocios

Como ya mencionamos en el apartado 1.1, a la hora de trabajar con datos proveniente de una red social, es probable que no todo el conjunto tenga el mismo valor de estudio. Es necesario realizar un esfuerzo extra para evitar emplear publicaciones de cuentas falsas automatizadas o de spam, que generalmente no tienen interés debido a su naturaleza, pero pueden alterar los resultados obtenidos. Este, por suerte, no es el caso para los datos empleados en este trabajo. A diferencia de lo que pueda ocurrir en otras redes sociales como Twitter, las cuentas automatizadas de Instagram suelen tener el objetivo de inflar el número de interacciones que obtienen otras cuentas, a través de comentarios o likes en sus publicaciones (Omena 2017). Al no ser usual por parte de estas cuentas automatizadas el realizar publicaciones propias, no llegan a formar una parte significativa de nuestro conjunto de datos, y no es necesario realizar un trabajo adicional para intentar eliminarlas.

Sin embargo, aunque no haya cuentas de spam en nuestro conjunto de datos, no todas las publicaciones serán relevantes para la creación del perfil de turismo. Esto es debido a que, aunque muchas publicaciones provienen de cuentas personales, otras tantas pertenecen a negocios como restaurantes u otros comercios. La manera en que ambos tipos de cuentas interactúen con la red social será muy diferente, por lo que en este capítulo centraremos nuestros esfuerzos en la creación de un modelo de clasificación que, a partir de una publicación de Instagram, intente predecir si la cuenta que ha realizado dicha publicación se trata de una cuenta personal o de un negocio.

Para la creación del modelo emplearemos aprendizaje supervisado. Tomaremos una muestra del conjunto de datos para entrenar y validar los posibles modelos. El etiquetado de las cuentas como personales o de negocios fue realizado manualmente. La distinción entre ambas no es tan clara como pueda parecer en un primer momento, debido a la existencia de, por ejemplo, cuentas que se presentan como personales pero basan todas su actividad en ofertar sesiones privadas de nutrición. Para el contexto de este trabajo, hemos decidido considerar a cualquier cuenta que intente vender un producto o servicio como un negocio, lo cual incluye, entre otras, a estas cuentas de nutrición.

En total fueron seleccionadas al azar 434 publicaciones de las 10140 disponibles. De estas 434, 271 fueron etiquetadas como realizadas por cuentas personales, mientras que las otras 163 se consideraron como negocios. Ninguna de las publicaciones consideradas fue sospechosa de haber sido realizada por un bot.

La información que poseemos de cada cuenta presente en el conjunto es limitada. Incluir datos acerca del comportamiento de cada cuenta a lo largo del tiempo mejoraría la clasificación (número de seguidores y gente a la que siguen, frecuencia de las publicaciones, etc), pero casi toda la información con la que contamos es exclusiva de una única publicación que realizan (número de likes y comentarios, título de la publicación, y fecha y hora de la misma), siendo el nombre de usuario la única información extra con la que contamos. Teniendo en cuenta esto, extraemos de manera inicial las siguientes 13

variables:

- Número de comentarios que obtuvo una publicación.
- Número de likes que obtuvo una publicación.
- Día de la semana en el cual se realizó la publicación.
- Hora aproximada (número entero entre 0 y 23) en la cual se realizó la publicación.
- Número de hashtags incluidos en el título de la publicación.
- Número de menciones a otras cuentas incluidas en el título de la publicación.
- Longitud total del título de la publicación, en número de caracteres.
- Cantidad de números presentes en el nombre de usuario de la cuenta.
- Variable indicadora que representa si la publicación es anterior o posterior a la COVID-19.
- Variable indicadora que representa si la publicación es un vídeo.
- Variable indicadora que representa si la publicación está compuesta por una única imagen o vídeo, o por más de una imagen.
- Variable indicadora que representa si el título de la publicación contiene un número de teléfono.
- Variable indicadora que representa si el título de la publicación contiene una url a una página web.

5.1. Análisis de las variables

Antes de entrar en la creación del modelo de clasificación, vamos a realizar un análisis exploratorio de las variables, comenzando por los diagramas de barras de la figura 5.1.

En el gráfico (a) hemos representado el número de publicaciones realizadas antes y después del COVID. Los datos no están completamente balanceados respecto a esta categoría, ya que en la muestra hay más publicaciones realizadas antes que después de la pandemia. Sin embargo, se puede observar que en el año 2021, las publicaciones realizadas por cuentas personales disminuyeron en gran medida en favor de las realizadas por negocios.

En (b) podemos ver la distribución de las publicaciones según el día de la semana en el que se subieron. Los fines de semana parece darse un ligero decrecimiento en cuanto al volumen de publicaciones exclusivamente para las cuentas de negocios. En cambio, las cuentas personales parecen publicar considerablemente menos los viernes, disminuyendo el volumen de publicaciones realizadas en aproximadamente un 40 % con respecto al domingo, que es el segundo día que menos publican.

En (c) y (d) se puede observar, respectivamente, la cantidad de publicaciones que incluyen una pagina web o un número de teléfono en el título. Ambas ocurren con bastante poca frecuencia, siendo la url la mas común de las dos. Aún así, a la vista de los gráficos vemos que, si una publicación incluye alguno de los dos, es altamente probable que se corresponda con un negocio y no con una cuenta personal.

En (e) y (f) está contenida la información acerca de si las publicaciones son vídeos, o múltiples imágenes. Los negocios parecen realizar ambos tipos de publicaciones ligeramente más a menudo que las cuentas personales (proporcionalmente), que optan por subir una sola imagen. Sin embargo, no parece ser una diferencia lo suficientemente grande como para poder extraer alguna conclusión de estas dos variables.

Finalmente, en (g) representamos la cantidad de números contenidos en los nombres de usuario de las cuentas que realizan las publicaciones. Aquí podemos notar claramente que si una cuenta tiene 2 o más números en su nombre, es altamente probable que se trate de una cuenta personal, en comparación con las cuentas de negocios que en casi toda su totalidad no parecen contener números.

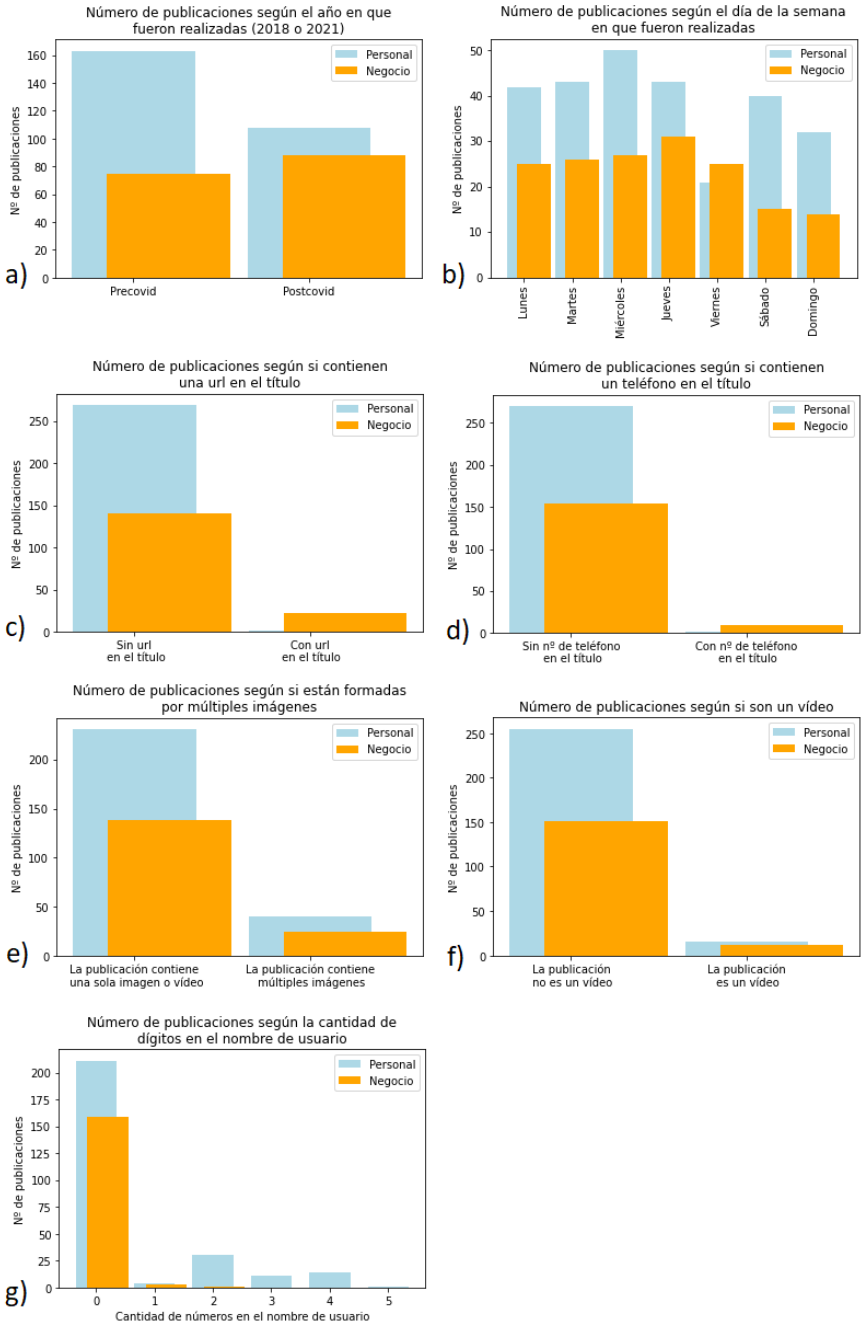

Figura 5.1: Gráficos de barras para distintas variables de la muestra, distinguiendo entre cuentas personales y de negocios.

Para el resto de variables hemos optado por representar sus diagramas de densidad acumulada, distinguiendo una vez más entre las cuentas personales y las de negocios. Los 6 gráficos resultantes están recogidos en la figura 5.2.

En (a) podemos ver la distribución del volumen de publicaciones a lo largo del día para cada tipo de cuenta. Las cuentas de negocios parecen realizar la mayor parte de sus publicaciones en las primeras

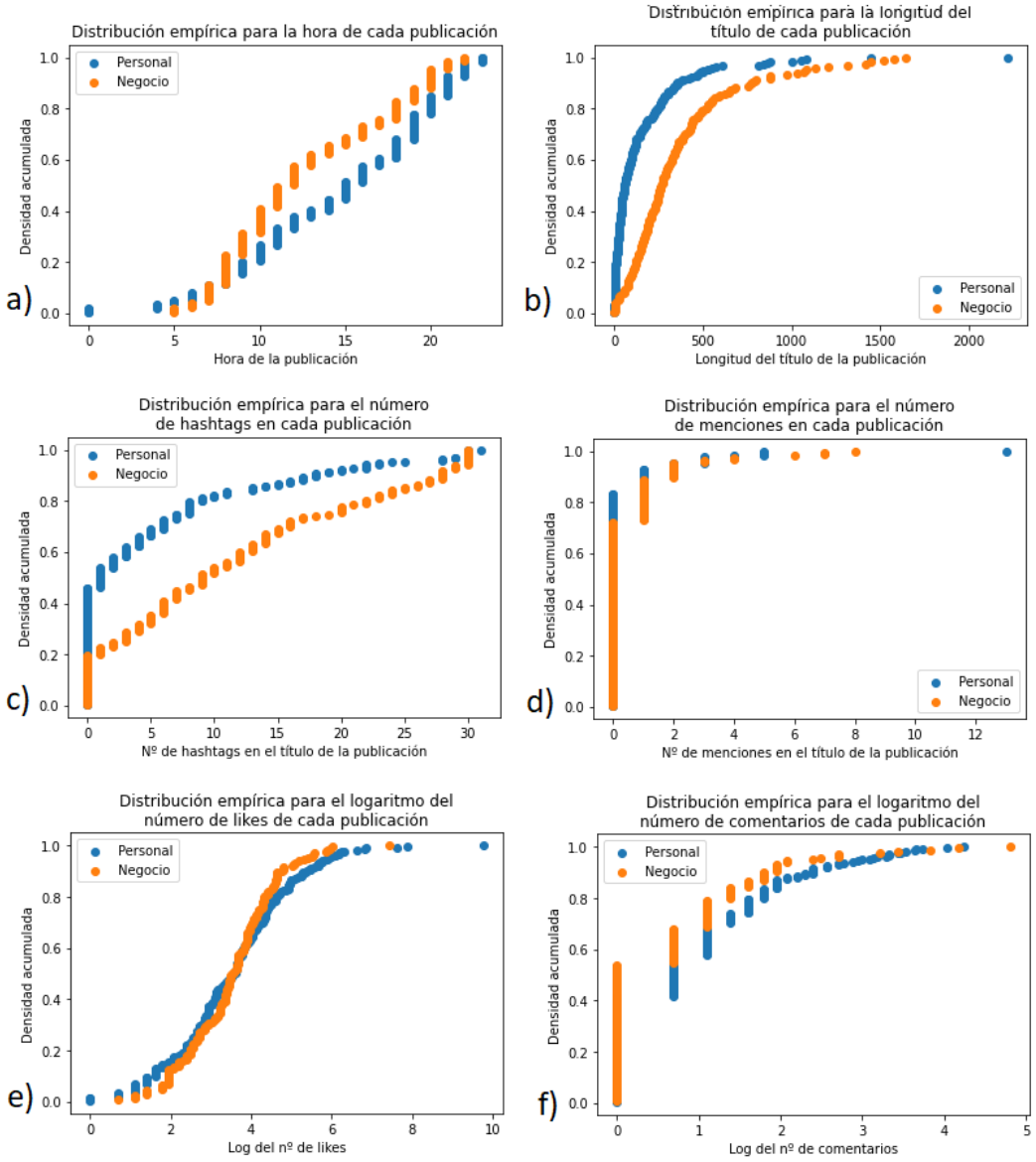

Figura 5.2: Gráficos de la densidad acumulada para distintas variables de la muestra, distinguiendo entre cuentas personales y de negocios.

horas del día, aproximadamente entre las 7 de la mañana y las 12 del mediodía suben el 50 % de las publicaciones diarias. Las cuentas personales tienen su actividad más dispersa a lo largo de todo el día, pero su período de mayor actividad es entre las 6 de la tarde y las 11 de la noche.

En (b) podemos ver como difieren las longitudes de los títulos de las publicaciones para las dos clases de cuentas. Para las cuentas personales, la densidad acumulada aumenta muy rápidamente a medida que lo hace la longitud del título. En contraste con esto, la densidad acumulada de las cuentas de negocios aumenta considerablemente más lento, lo cual nos muestra que es muy poco común para estas cuentas el subir publicaciones sin títulos o con títulos excesivamente cortos.

En (c) representamos la evolución del volumen de publicaciones en función del número de hashtags que tienen en sus títulos. Podemos ver que tan solo 1 de cada 5 publicaciones de negocios no contienen

ningún hashtag, contra más de 2 de cada 5 publicaciones de cuentas personales. Estas cuentas personales siguen la tendencia de usar menos hashtags que las de negocios, y encontramos que el 80 % de las cuentas personales utilizan 8 o menos hashtags, contra menos del 50 % de las cuentas de negocios.

A partir del gráfico (d) podemos realizar un análisis similar, pero esta vez con las menciones en vez de los hashtags. También vemos que los negocios emplean más menciones que las cuentas personales de manera general, pero esta vez la diferencia es mucho menos marcada. Más del 70 % de los negocios no mencionan a nadie en el título, contra un 85 % de las cuentas personales que tampoco lo hacen. Para publicaciones con 3 o más menciones no parece haber diferencias significativas en la cantidad de menciones en función del tipo de cuenta, lo cual podría estar debido a una baja cantidad de datos (tan solo 21 de las 434 publicaciones de la muestra utilizan 3 o más menciones).

En (e) vemos los datos correspondientes con el número de likes recibidos por una publicación, tras haber aplicado un logaritmo. La razón por la que transformamos estos datos (y posteriormente los referentes a los comentarios) es porque un pequeño número de publicaciones de la muestra tenían una cantidad tan elevada de interacciones (likes y comentarios) en comparación con todas las demás, que impedían la visualización correcta de los datos. Con respecto a las diferencias en cuanto a tendencias de cada grupo, podemos ver que hay mas acumulación de densidad en los extremos para publicaciones provenientes de cuentas personales. Es decir, que las publicaciones que obtienen una cantidad muy baja o muy alta de likes tienen más probabilidad de ser personales. Sin embargo, es una diferencia muy pequeña, y ambas curvas de densidad no están lejos de estar superpuestas.

Por último, en (f) vemos los datos del número de comentarios de las publicaciones, tras haber pasado los datos por un logaritmo. Para esta variable fue necesario sumar 1 a la cantidad de comentarios recibidos antes de aplicar la transformación, para evitar evaluaciones del logaritmo en 0 en las numerosas publicaciones que no recibieron ningún comentario. Podemos ver que la densidad acumulada para recibir 2 comentarios o menos es considerablemente mayor para las cuentas de negocios. En cambio, hay más cuentas personales que reciben una cantidad muy elevada de comentarios en la publicación.

El siguiente paso que tomaremos será comprobar si la variable indicadora del tipo de cuenta (personal o de negocio) puede considerarse independiente de cada una de las demás variables. Comenzaremos por las variables categóricas: situación de la pandemia (**covid**), día de la semana (**semana**), presencia de url (**url**) o de número de teléfono (**teléfono**), y si la publicación es un vídeo (**vídeo**) o si contiene múltiples imágenes (**multi**). Añadimos también una variable extra (**viernes**) que tan solo indica si el día de la semana es viernes.

Para cada una de estas variables, aplicamos el test chi-cuadrado de Pearson (Agresti 2007). Este test está diseñado para contrastar la independencia entre dos variables categóricas. Sea n_{ij} el número de individuos pertenecientes a la categoría i de la primera variable y j de la segunda, $n_{i\bullet}$ el número de total de individuos de la categoría i de la primera variable, y $n_{\bullet j}$ el de individuos de la categoría j de la segunda variable. Bajo la hipótesis de independencia, la distribución conjunta para ambas variables es el producto de sus distribuciones marginales. Es decir, que el valor $n_{ij}/n - n_{i\bullet}n_{\bullet j}/n^2$ debería ser próximo a 0, siendo n el número total de observaciones. Suponiendo que la primera variable tiene I categorías diferentes, y la segunda tiene J, el test chi-cuadrado de Pearson contrasta la hipótesis nula de independencia entre las dos variables a partir del estadístico:

$$\chi^2 = \sum_{i=1}^{I}\sum_{j=1}^{J} \frac{(n_{ij} - n_{i\bullet}n_{\bullet j}/n)}{n_{i\bullet}n_{\bullet j}/n} \tag{5.1}$$

Bajo la hipótesis de independencia, el estadístico χ^2 de 5.1 sigue una distribución chi-cuadrado con $(I-1)\cdot(J-1)$ grados de libertad.

En nuestro caso aplicaremos este test para contrastar la independencia de cada una de las variables predictoras categóricas con la respuesta. Para ello, fijamos un nivel de significación del 0,05. Recogemos en la tabla 5.1 los estadísticos (**est**) y p-valores (**pval**) obtenidos.

Cuadro 5.1: Resultados del test chi-cuadrado de independencia de la respuesta con distintos predictores

	est	pval
covid	7.651	0.006
semana	9.804	0.133
url	29.321	6.133e-08
teléfono	9.824	0.002
vídeo	0.158	0.691
multi	5.915e-04	0.981
viernes	5.410	0.020

Bajo el nivel de significación fijado al 0,05, rechazamos la hipótesis nula de independencia con el tipo de cuenta de Instagram (y por tanto, asumimos que existe dependencia) para las variables **covid**, **url** y **teléfono**. Es decir, que el test encuentra evidencias significativas de que la situación de la pandemia y la presencia de url o número de teléfono influyen en el tipo de cuenta que realiza una publicación.

También bajo el mismo nivel de significación no se encuentran evidencias significativas para rechazar la hipótesis nula de independencia para las variables **vídeo** y **multi**, lo cual significa que el hecho de que una publicación sea un vídeo o contenga múltiples imágenes no parece tener relación con si quien la ha subido sea una cuenta personal o un negocio.

Por último, debemos mencionar las variables **semana** y **viernes**. Aunque los resultados del test muestran que el hecho de que sea viernes o no es claramente significativo sobre la respuesta, no se rechaza la hipótesis nula de independencia parala variable global del día de la semana. En consecuencia, no consideramos que el día de la semana parezca afectar por sí solo a la respuesta en gran medida.

Veremos lo que ocurre también con las demás variables, que son la cantidad de números en el nombre de usuario (**números**), la hora de la publicación (**hora**), la longitud del título (**longitud**), el número de **hashtags** y **menciones** que usan, y los **likes** y **comentarios** que recibieron. Para cada una de estas variables, aplicaremos un ANOVA para comparar las medias de cada uno de los tipos de cuenta. Es necesario destacar que los resultados que obtengamos deberán ser interpretados únicamente como una aproximación, ya que no podemos garantizar las hipótesis de normalidad o igualdad de varianzas. Sin embargo, se ha demostrado (Glass et al. 1972) que los resultados del ANOVA no son afectados en gran medida por el incumplimiento en particular de la hipótesis de normalidad. En consecuencia, emplearemos el ANOVA como una herramienta exploratoria más, y no como un método de preselección de variables. En la tabla 5.2 recogemos los resultados obtenidos, donde incluimos la proporción de varianza total asociada con cada variable (η^2).

A la vista de los resultados obtenidos, las variables que representan una mayor proporción de la variabilidad global son la longitud del título y los hashtags que contiene, seguidas por la cantidad de números en el nombre de usuario y la hora de publicación. En cambio, con el mismo nivel de significación del 0,05 de antes, no se rechaza la hipótesis nula para la igualdad de medias entre cuentas personales y negocios de las 3 variables restantes. Todo esto concuerda con lo que intuíamos a partir de los gráficos de densidad acumulada.

Cuadro 5.2: Resultados del ANOVA entre la respuesta y distintos predictores

	est	pval	η^2
números	31.834	3.043e-08	0.069
hora	12.741	3.980e-04	0.029
longitud	59.973	6.905e-14	0.122
hashtags	50.580	4.776e-12	0.105
menciones	2.565	0.110	0.006
likes	1.185	0.276	0.003
comentarios	0.561	0.454	0.001

Finalmente, estudiaremos la correlación entre las variables con el objetivo de detectar un posible caso de colinealidad. En particular, calcularemos para cada par de variables el coeficiente de correlación de Pearson. Si suponemos que tenemos una muestra de n individuos, para los cuales sus valores para dos variables diferentes son $x_{1,1}, \ldots, x_{1,n}$ y $x_{2,1}, \ldots, x_{2,n}$, y sean $\overline{x}_1$ y $\overline{x}_2$ las medias muestrales de dichos valores. El coeficiente de correlación de Pearson de la muestra para estas dos variables viene dado por

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_{1,i} - \overline{x}_1)(x_{2,i} - \overline{x}_2)}{\sqrt{\sum_{i=1}^{n}(x_{1,i} - \overline{x}_1)^2}\sqrt{\sum_{i=1}^{n}(x_{2,i} - \overline{x}_2)^2}} \tag{5.2}$$

El coeficiente de correlación de Pearson es aplicable en caso de que alguna de las dos variables sea binaria (o incluso en caso de que ambas lo sean). En particular, si una de las dos lo es, la correlación de Pearson es matemáticamente equivalente a la correlación point biserial (Kornbrot 2005), mientras que si ambas variables son binarias es equivalente al al coeficiente de correlación de Matthews (Matthews 1975). Por tanto, tiene sentido calcular la correlación de Pearson sobre cualquiera de nuestros pares de variables.

En la figura 5.3 hemos representado por tanto la matriz de correlaciones de Pearson de las 13 variables originales, junto con la variable respuesta **entidad**, indicadora del tipo de cuenta de Instagram (personal o de negocio). Podemos ver que la correlación más alta con diferencia es la correspondiente con las variables **longitud** y **hashtags**. Esto es debido a la naturaleza de las variables, ya que si el texto que acompaña a la publicación contiene muchos hashtags (que forman parte del texto), entonces naturalmente la longitud del propio texto deberá ser lo suficientemente elevada como para que quepan esos hashtags. Sin embargo, no es una correlación que nos haga sospechar de colinealidad, y es que un texto puede ser largo por si solo sin necesariamente contener ningún hashtag.

La matriz también nos es útil para poder ver la correlación entre la variable respuesta **entidad** y todas las demás variables. Para entender estas correlaciones, es necesario mencionar que codificamos que una cuenta fuese personal con un 1, y que fuese un negocio con un 2. Podemos ver entonces que las correlaciones más positivas de **entidad** se corresponden con las variables **hashtags**, **url** y **longitud**. Esto significa que, cuando el valor de una de estas 3 variables aumenta, la variable **entidad** tiende a aumentar su valor. Es decir, que si una publicación utiliza muchos hashtags, contiene una url o está acompañada por un título muy largo, es más probable que pertenezca a un negocio. En cambio, las correlaciones más negativas se corresponden con las variables **números** y **hora**. Esto quiere decir que, cuantos más números contenga el nombre de la cuenta, o más tarde se realice una publicación a lo largo del día, es más probable que provenga de una cuenta personal.

Antes de comenzar la búsqueda del modelo de clasificación supervisado, entrenaremos a partir de

	comentarios	likes	semana	hora	covid	vídeo	hashtags	menciones	números	teléfono	url	longitud	multi	entidad
comentarios	1.000	0.396	-0.029	0.013	0.051	-0.035	0.160	0.077	-0.061	-0.014	-0.020	0.302	0.119	-0.036
likes	0.396	1.000	-0.026	-0.023	-0.047	-0.020	0.063	0.039	-0.040	-0.016	-0.019	0.098	0.011	-0.052
semana	-0.029	-0.026	1.000	0.062	-0.034	0.024	-0.050	0.012	0.094	0.014	-0.043	-0.114	-0.023	-0.028
hora	0.013	-0.023	0.062	1.000	-0.022	0.003	-0.079	0.028	0.041	-0.022	-0.015	-0.061	0.054	-0.169
covid	0.051	-0.047	-0.034	-0.022	1.000	0.044	0.138	0.013	-0.023	0.138	0.044	0.218	0.190	0.138
vídeo	-0.035	-0.020	0.024	0.003	0.044	1.000	-0.089	0.109	-0.007	0.085	0.060	-0.013	-0.110	0.029
hashtags	0.160	0.063	-0.050	-0.079	0.138	-0.089	1.000	0.126	-0.074	0.073	0.133	0.567	-0.000	0.324
menciones	0.077	0.039	0.012	0.028	0.013	0.109	0.126	1.000	0.063	-0.003	0.081	0.174	0.082	0.077
números	-0.061	-0.040	0.094	0.041	-0.023	-0.007	-0.074	0.063	1.000	-0.058	-0.092	-0.132	0.136	-0.262
teléfono	-0.014	-0.016	0.014	-0.022	0.138	0.085	0.073	-0.003	-0.058	1.000	0.232	0.160	0.065	0.166
url	-0.020	-0.019	-0.043	-0.015	0.044	0.060	0.133	0.081	-0.092	0.232	1.000	0.224	-0.017	0.270
longitud	0.302	0.098	-0.114	-0.061	0.218	-0.013	0.567	0.174	-0.132	0.160	0.224	1.000	0.089	0.349
multi	0.119	0.011	-0.023	0.054	0.190	-0.110	-0.000	0.082	0.136	0.065	-0.017	0.089	1.000	0.008
entidad	-0.036	-0.052	-0.028	-0.169	0.138	0.029	0.324	0.077	-0.262	0.166	0.270	0.349	0.008	1.000

Figura 5.3: Matriz de correlaciones de Pearson de las variables.

la muestra un modelo de k-medias. Nuestro objetivo es realizar un análisis de los clusters obtenidos para entender mejor los datos. Elegimos dividir la muestra en $k = 8$ grupos, ya que consideramos que produce resultados interesantes. En la tabla 5.3 incluimos la proporción de la muestra total según el tipo de cuenta (personal o negocio) y el cluster al que pertenece (del 1 al 8).

Cuadro 5.3: Distribución de la muestra en clusters, distinguiendo entre cuentas personales y de negocios.

	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Personal	0.1014	0.0023	0.0046	0.0046	0.4194
Negocio	0.1521	0.0000	0.0023	0.0115	0.0968

	Cluster 6	Cluster 7	Cluster 8
Personal	0.0138	0.0438	0.0346
Negocio	0.0299	0.0092	0.0737

En caso de emplear este modelo para la clasificación, agruparíamos los clusters 2, 3, 5 y 7 como cuentas personales, y los otros 4 como cuentas de negocios, obteniendo una precisión global de 0,7373. Pero más que la precisión, nos interesa saber en base a qué criterio se han formado los grupos. Para ello representaremos diagramas de densidad acumulada similares a los de 5.2, pero con la muestra dividida en los 8 clusters. La mayoría de los 13 gráficos que podríamos realizar no aportan demasiada información visual. Hemos optado por incluir los gráficos correspondientes con las variables **longitud**, **likes**, **comentarios** y **hashtags**, los cuales están representados en 5.4.

En la gráfica 5.4 (a) figura el gráfico para la longitud del título de la publicación. Aquí podemos ver claramente el criterio empleado para la creación de los clusters 5, 1, 8, 6 y 4. Por ejemplo, el cluster 5 está formado por publicaciones que no tienen título, o con un título muy corto. En el extremo contrario tenemos al cluster 4, formado por publicaciones con títulos de más de 1200 caracteres. Vemos que, a medida que aumenta la longitud media de los títulos de las publicaciones pertenecientes a un cluster, su tamaño disminuye. Particularmente, el cluster 5 contiene a más del 50 % de los datos totales, mientras que el cluster 5 contiene únicamente 7 datos.

Si ahora observamos el gráfico (b), en el cual se representa el número de likes que han recibido las

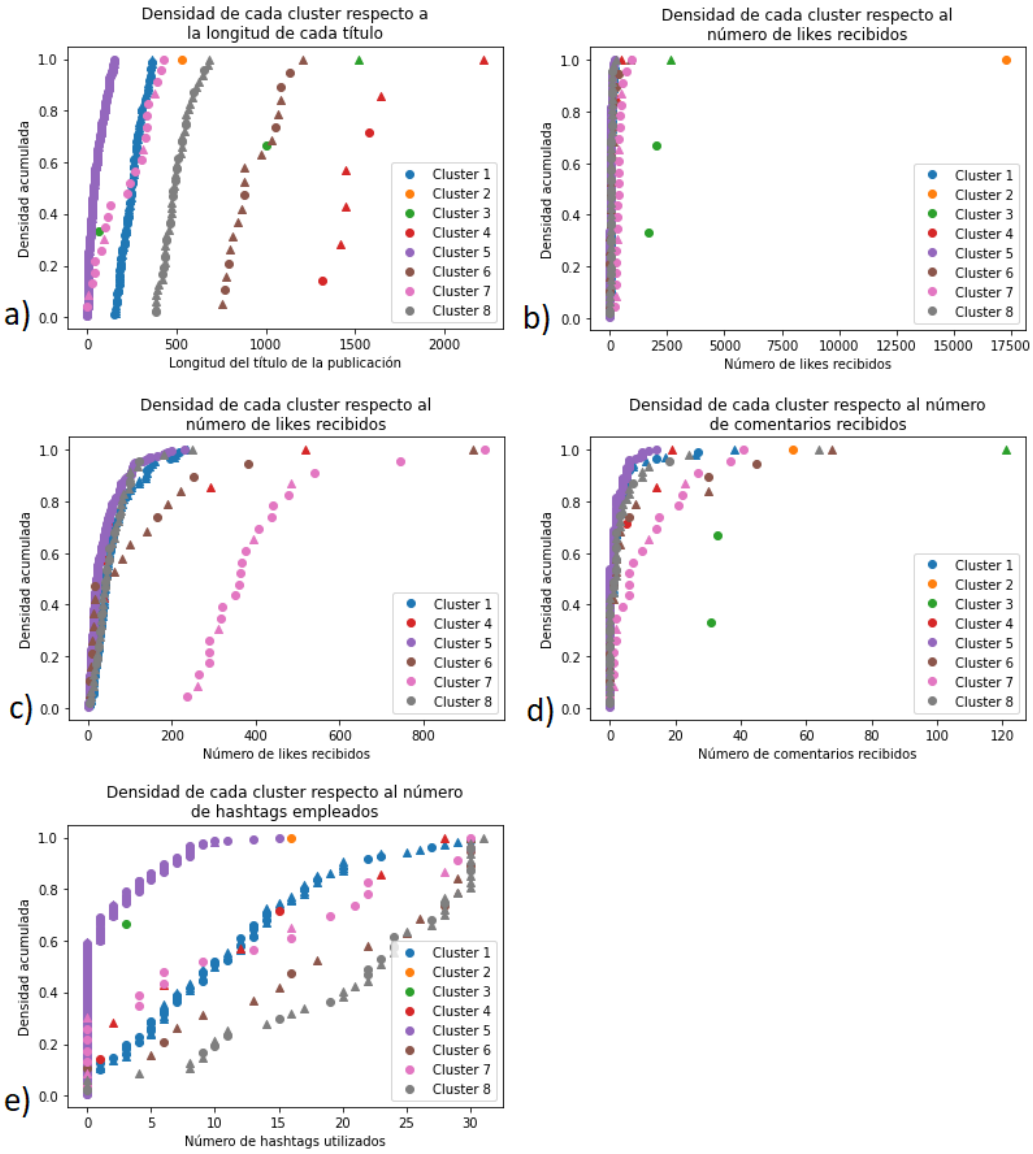

Figura 5.4: Diagramas de densidad acumulada para la muestra dividida en 8 clusters, para las variables **longitud** (a), **likes** (b) y (c), **comentarios** (d) y **hashtags** (e), y distinguiendo entre cuentas de negocios (△) y personales (◯).

publicaciones, podemos entender la creación de los clusters 2 y 3. El cluster 2 está formado por una única publicación que recibió una cantidad inusualmente grande de likes (más de 17000, mientras que la segunda publicación que más recibió no alcanzó los 3000). Al cluster 3 por su parte lo componen las otras 3 publicaciones que más likes recibieron, sin contar a la que acabamos de mencionar.

Para poder ver lo que ocurre con los demás clusters, hemos representado la misma gráfica en (c) pero excluyendo a estos 2 grupos. Vemos que, aunque en (a) había una distinción clara entre los clusters 5, 1 y 8, parecen ser idénticos en cuanto a los likes que recibieron. Las publicaciones de los grupos 6 y 4, que empleaban títulos más largos, parecen haber recibido de media más likes que los 3 anteriores, pero sin demasiada diferencia. Lo realmente interesante de este gráfico es que nos permite ver de donde

surge la creación del cluster 7. En conjunto con el gráfico (a), podemos decir que el cluster 7 está formado por publicaciones que recibieron una cantidad considerable de likes (gráficamente, parece que la publicación de este grupo que menos likes obtuvo tiene más likes que cualquier publicación de los grupos 5, 1 y 8) y que a la vez tiene títulos de no demasiada longitud (centrándonos solo en la variable **longitud**, los datos de este custer formarían parte de los clusters 5 y 1, que son los que menos longitud media tienen).

El gráfico (d) nos permite analizar el número de comentarios recibidos. La tendencia general es la misma que observamos para los likes: los clusters que reciben de media menos likes también recibirán menos comentarios, y lo mismo es cierto en el sentido inverso. Lo más sorprendente es que la cantidad tan elevada de likes de la una publicación que componía el cluster 2 no se traduce en una cantidad igualmente elevada de comentarios, sino que se encuentra por detrás ciertas publicaciones de los clusters 8, 6 y 3.

Por último, en (e) se representa la cantidad de hashtags que utilizan las publicaciones. En relación con el gráfico (a), la principal sorpresa que vemos es que las publicaciones del cluster 8 emplean de media más hashtags que las del 6 y el 4, y aún así sus títulos son considerablemente más cortos. Podemos suponer que estos dos últimos grupos tienen títulos orgánicamente más largos, mientras que una cantidad importante de publicaciones del grupo 8 posiblemente abusa del sistema de hashtags.

En 5.5 hemos representado los diagramas de dispersión de unos pocos pares de variables, con el objetivo de visualizar mejor las interacciones entre estas. En los gráficos (a) y (b) se representa la variable **likes** en el eje x y la variable **longitud** en el y. La diferencia entre ambos gráficos es que en el segundo de ellos hemos suprimido los grupos 2 y 3 para poder visualizar mejor a los demás datos. Estos gráficos no nos aportan información nueva, pero aún así resultan particularmente interesantes ya que nos permiten ver que los clusters formados parecen totalmente separables para estas dos variables.

En el gráfico (c) podemos ver la forma en que varía la longitud de los títulos de las publicaciones en función de la hora a la que han sido realizadas. En general vemos que nadie publica nada muy de madrugada, entre la 1 y las 3. Sin embargo, resulta notable ver que los grupos que utilizan títulos más largos (clusters 4 y 6), o los que reciben una alta cantidad de interacciones (clusters 2, 3 y 7) publican casi exclusivamente entre las 7 y las 11 de la mañana, y entre las 5 de la tarde y las 10 de la noche, dejando una franja al mediodía en la que no suben nada.

En el gráfico (d) se ve cómo depende el número de comentarios recibidos por una publicación en función de la hora en que ha sido realizada. Aunque casi a todas horas hay publicaciones sueltas que reciben bastantes más comentarios que la media, entre las 5 y las 7 de la tarde parece haber un cambio de tendencia general que hace que más publicaciones reciban una cantidad considerable de comentarios, lo cual podría sugerir que a esas horas es cuando hay más actividad en la aplicación.

En el gráfico (e) se representa la variación del número de hashtags empleados en una publicación en función de la longitud de su título en caracteres. Este gráfico tampoco nos aporta información buena, pero es una buena manera de visualizar la cantidad inusualmente elevada de hashtags que utilizan las publicaciones del grupo 8, sobre todo en comparación con las de los grupos 4 y 5 que tienen títulos más largos.

Finalmente, en (f) representamos la longitud de los títulos de las publicaciones en función de la cantidad de números en el nombre de usuario de la cuenta. Este gráfico es interesante únicamente porque nos permite ver que prácticamente todas las cuentas que tienen algún número en su nombre suben publicaciones con títulos cortos, y por tanto acaban siendo clasificadas en los grupos 1 y 5.

Podemos describir cada uno de los 8 clusters formados de la siguiente manera:

- El **Cluster 1** está formado por publicaciones con títulos no muy largos, de entre 148 y 363 caracteres. Reciben una cantidad de interacciones baja en relación con la muestra total, obteniendo de media 58 likes y 2 comentarios (contra la media global de 124 likes y 3 comentarios). Prácticamente todas contienen hashtags, y la cantidad que contienen parece estar repartida uniformemente entre los 0 y los 20, con un pequeño grupo que sobrepasa esa última cifra. Aproximadamente 3 de cada 5 cuentas pertenecientes a este cluster son negocios.

- El **Cluster 2** está formado por una única publicación realizada por una cuenta personal que

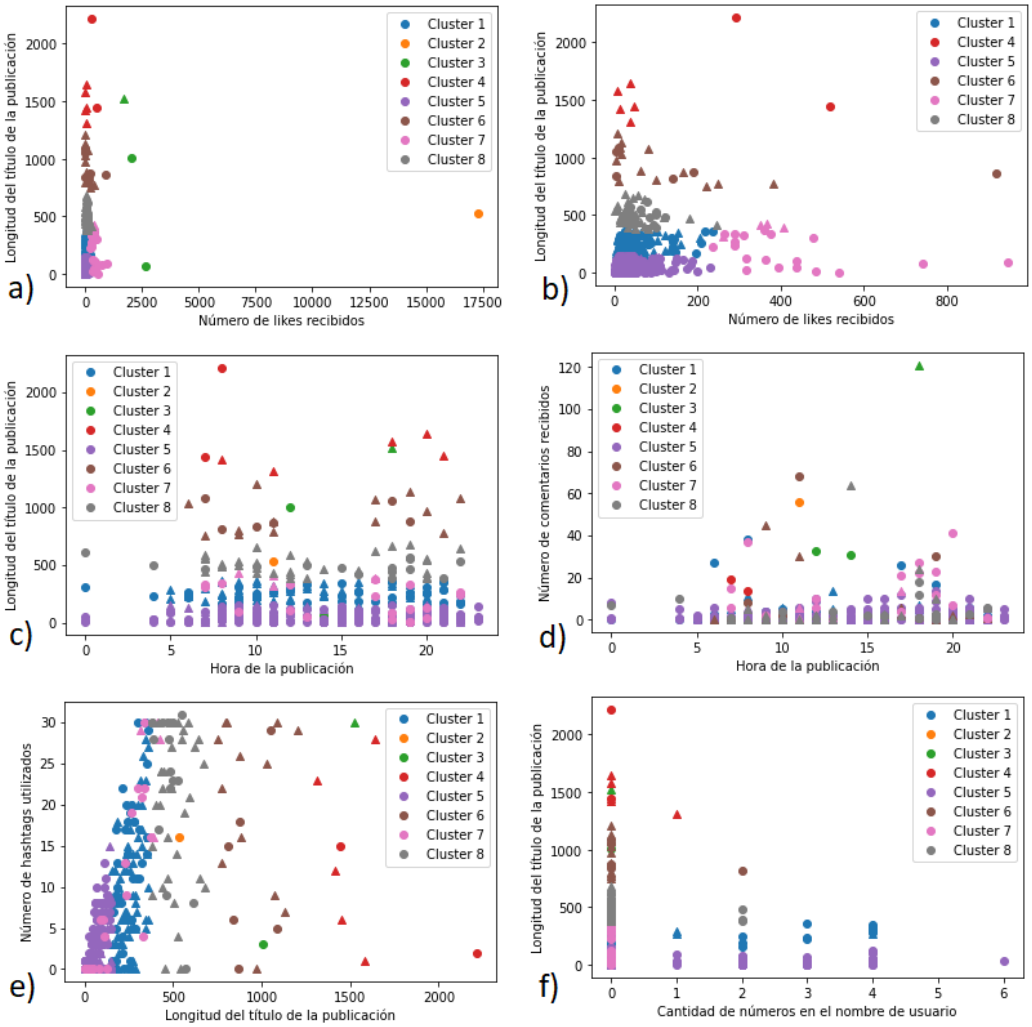

Figura 5.5: Diagramas de dispersión para varios pares de variables, distinguiendo entre cuentas de negocios (△) y personales (◯).

recibió 17264 likes (la segunda publicación de la muestra con más likes recibió 2651) y 56 comentarios.

- El **Cluster 3** está formado por 3 publicaciones con una cantidad muy elevada de interacciones (2651, 2047 y 1699 likes, 31, 33 y 121 comentarios), con longitudes de títulos altamente variables (de 63 a 1522 caracteres). Tan solo 1 de las 3 cuentas es un negocio.

- El **Cluster 4** está formado por publicaciones con títulos considerablemente largos, de entre 1312 y 2215 caracteres, y con una tendencia de uso de hashtags similar a la vista en el cluster 1. La mitad de las publicaciones obtuvieron muy pocas interacciones (similar una vez más a las del cluster 1), pero otras llegaron a alcanzar los 517 likes y 19 comentarios. De las 7 cuentas representadas en este cluster, 5 de ellas son de negocios.

- El **Cluster 5** está formado por publicaciones con títulos muy cortos, de menos de 152 caracteres. Obtuvieron en general muy pocas interacciones (de media 37 likes y 1 comentario, incluso menos

que el cluster 1), y apenas utilizan hashtags (más de la mitad de las publicaciones no contienen ni uno). Aproximadamente 4 de cada 5 cuentas de este cluster son personales.

- El **Cluster 6** está formado por publicaciones con títulos de longitud elevada, entre 754 y 1207 caracteres. Al igual que ocurre con el cluster 4, la mitad de las publicaciones de este grupo obtuvo muy pocas interacciones, pero otras alcanzan los 945 likes y 41 comentarios (la media está en 402 likes y 11 comentarios, muy por encima de la media global de 124 y 3). Además, emplean una cantidad elevada de hashtags, 12 de media contra la media global de 7. Aproximadamente el 70 % de las cuentas de este cluster son de negocios.

- El **Cluster 7** está formado por publicaciones con títulos de longitud no muy elevada, de no más de 418 caracteres, pero que obtuvieron una cantidad bastante elevada de interacciones (un mínimo de 235 likes, superior a la media global, y una media de 402 likes y 11 comentarios). Aproximadamente 4 de cada 5 cuentas de este grupo con tantas interacciones son cuentas de negocios.

- El **Cluster 8** está formado por publicaciones con títulos moderadamente largos (longitudes entre 382 y 680 caracteres, situándolo entre los clusters 1 y 6). Sus likes recibidos son similares a los que vimos para el cluster 1, con una media de 55 (menos de la mitad de la media global de 124). En cambio, obtienen 4 comentarios de media, lo cual está 1 por encima de la media global. Además, este grupo emplea una cantidad increíblemente elevada de hashtags, con una media de 20 por publicación. Aproximadamente 2 de cada 3 cuentas de este grupo pertenecen a negocios.

5.2. Creación del modelo de clasificación supervisado

El objetivo de esta sección consiste en entrenar un modelo de aprendizaje supervisado que permita clasificar a nuestros datos como cuentas personales o negocios. Como variables predictoras consideraremos a las 13 variables que hemos estado explorando: **covid**, **semana**, **url**, **teléfono**, **vídeo**, **multi**, **números**, **hora**, **longitud**, **hashtags**, **menciones**, **likes** y **comentarios**.

Como posibles modelos que vamos a entrenar consideraremos un bosque aleatorio, un clasificador de soporte vectorial, un método de k-vecinos más próximos y un perceptrón multicapa. En todos los casos hemos realizado la búsqueda de los hiperparámetros óptimos mediante un método de validación cruzada. Una descripción más detallada de estos modelos y sus hiperparámetros se encuentra en el capítulo 2. Además, hemos optado por no suprimir ninguna variable predictora, ya que no notamos mejoras sobre la capacidad predictiva de los modelos, y en algunos casos llegaba a empeorar eliminando las variables menos importantes. La versión de validación cruzada que emplearemos será la validación cruzada en k iteraciones, o *k-fold cross-validation*, descrita en la sección 2.6.

Como ya dijimos antes, nuestra muestra está formada por 434 publicaciones, siendo 271 de ellas de cuentas personales y las otras 163 de negocios. A la hora de aplicar validación cruzada para buscar los hiperparámetros de cada modelo, decidimos dividir la muestra en $k = 10$ grupos. Además, para todos los modelos hemos tomado el 80 % de los datos como conjunto de entrenamiento, y el otro 20 % como conjunto de test. Para la división de la muestra que se usa en todos los modelos, en el conjunto de entrenamiento hay 221 cuentas personales y 126 negocios, mientras que en el de test hay 50 personales y 37 negocios.

A la hora de evaluar los modelos mediante validación cruzada, hemos considerado como clase positiva a las cuentas personales. La selección del modelo ha sido realizada en base a la precisión global (la tasa global de aciertos en ambas clases), el valor predictivo positivo (cociente entre el número de cuentas personales clasificadas correctamente y número de cuentas clasificadas como personales), la sensibilidad (cociente entre el número de cuentas personales clasificadas correctamente y número de cuentas personales totales), y el área bajo la curva ROC (sección 2.7).

Para el bosque aleatorio, el hiperparámetro es el número de árboles usados por el modelo. En este caso probamos varias rejillas que se movían entre los 80 y los 350 árboles. Encontramos que tomar

135 árboles produce los mejores resultados para todas las medidas consideradas, con excepción de la sensibilidad, para la cual sigue encontrándose muy cerca del modelo óptimo con 80 árboles. Por tanto, nos quedamos con el bosque aleatorio con 135 árboles, que devuelve una tasa de aciertos sobre el conjunto de test de 0,747.

El siguiente modelo que entrenamos es la máquina de soporte vectorial. Hemos considerado núcleos radiales y polinomiales. En ambos casos debemos buscar los hiperparámetros C y γ, que son respectivamente la penalización por clasificación incorrecta y el inverso del radio de influencia de cada observación. En el caso del núcleo polinomial habrá que hallar adicionalmente el coeficiente independiente del núcleo, c_0, y el grado del polinomio, d. Para el parámetro γ consideraremos dos posibles valores: el inverso del número de predictores, y el inverso del producto entre el número de predictores y la varianza del conjunto de entrenamiento. Para ambos núcleos hemos considerado rejillas para C entre los órdenes de 0,01 y 100.

En primer lugar realizamos pruebas con el núcleo radial. Necesitamos tomar valores de γ muy bajos para obtener un modelo aceptable, de entre 10^{-5} y 10^{-8}, y aún así los resultados son peores que los que encontramos para el núcleo polinomial. Para este segundo núcleo, probamos grados d del polinomio de hasta 8, y coeficientes independientes c_0 de hasta 20. No pudimos emplear rejillas de parámetros muy densas debido a los altos tiempos computacionales del modelo, que en algunos casos sobrepasaban 1 hora para rejillas con solo 2 valores diferentes para cada parámetro. Logramos encontrar aún así que, para una penalización de $C = 0{,}1$, grado $d = 7$ y coeficiente independiente $c_0 = 8$, el modelo es el óptimo para la tasa global de aciertos y el valor predictivo positivo, y proporciona buenos resultados para las otras dos medidas. La proporción de aciertos en el conjunto de test que obtenemos con este modelo es de 0,747.

A continuación entrenamos el modelo de k vecinos más próximos. Consideramos para ello la distancia euclídea, y pesos tanto uniformes como proporcionales a la distancia. Hemos considerado rejillas de valores de k de entre 5 y 100. En este caso el modelo óptimo es diferente para cada una de las 4 medidas consideradas, pero encontramos que el modelo con mejores resultados en general es el óptimo para el área bajo la curva ROC, que toma $k = 60$ con pesos basados en la distancia. La tasa de aciertos de este modelo sobre el conjunto de test es de 0,759.

Por último, entrenamos un perceptrón multicapa. Además del número de capas ocultas, y la cantidad de nodos en cada capa, también habrá que determinar la función de activación empleada, y el método empleado para obtener los coeficientes del modelo de manera iterativa. En total consideramos emplear entre 1 y 5 capas ocultas. Además, probamos con distintos tamaños de capas ocultas, de entre 30 y 150 nodos cada una. Al final obtuvimos el mejor resultado para 4 capas ocultas, de tamaños 70, 120, 130 y 80 respectivamente, utilizando un rectificador como función de activación, y con un optimizador estocástico basado en el gradiente. Este modelo tiene una precisión global sobre el conjunto de test de 0,759.

En las tablas 5.4, 5.5, 5.6 y 5.7 recogemos las medidas de evaluación de la clasificación sobre el conjunto de test para los 4 modelos que acabamos de seleccionar, resaltando el máximo de cada medida en negrita. Vemos que el bosque aleatorio es el que menos cuentas personales clasifica como negocios, pero esto viene a costa de que muchos negocios sean clasificados como personales (prácticamente la mitad). Si hiciéramos un estudio de la cuentas personales, podríamos acabar con demasiadas cuentas de negocios entre ellas, por lo que en principio podría no interesarnos emplear este modelo.

Los otros 3 modelos son más equilibrados. El clasificador de soporte vectorial por ejemplo es el que mejor se comporta con las cuentas de negocio a cambio de clasificar peor a las cuentas personales. En este trabajo proponemos emplear un modelo de clasificación que use estos otros 3 clasificadores, de manera que su predicción sea la moda de las predicciones de los 3 modelos. Esto es lo que se conoce como modelo *ensemble*.

En la tabla 5.8 recogemos medidas acerca de la capacidad predictiva sobre el conjunto de test de este nuevo modelo. Hemos resaltado las medidas que sean superiores a todas las medidas correspondientes de los modelos que lo forman. Podemos ver que este modelo tiene una mayor precisión global, de 0,77, y que las medias tanto normales como ponderadas de la precisión, la sensibilidad y la puntuación $F1$ son mejores. Además, se comporta mejor para las cuentas de negocios que los modelos anteriores (0,73

de puntuación $F1$, contra el 0,704 del perceptrón multicapa), sin perjudicar a cambio en gran medida su comportamiento sobre las cuentas personales (0,8 de puntuación $F1$, contra el 0,804 del modelo de k-vecinos).

Una vez construido el modelo de clasificación, solo queda aplicarlo sobre el conjunto total de los 10140 datos. Obtenemos así que 6293 de las publicaciones han sido clasificadas como provenientes de cuentas personales, mientras que las otras 3847 fueron consideradas como cuentas de negocios. Dedicaremos el siguiente capítulo a realizar un análisis más profundo de las publicaciones clasificadas como personales.

Cuadro 5.4: Varias medidas de precisión para el modelo de bosque aleatorio

Bosque aleatorio			
Precisión global: 0.747			
	Precisión	Sensibilidad	Puntuación F1
Personal	0.719	**0.920**	**0.807**
Negocio	**0.826**	0.514	0.634
Media	**0.773**	0.717	0.721
Media ponderada	**0.765**	0.747	0.733

Cuadro 5.5: Varias medidas de precisión para el modelo de clasificador de soporte vectorial

Clasificador de soporte vectorial			
Precisión global: 0.747			
	Precisión	Sensibilidad	Puntuación F1
Personal	**0.780**	0.780	0.780
Negocio	0.703	**0.703**	0.703
Media	0.742	0.742	0.742
Media ponderada	0.747	0.747	0.747

Cuadro 5.6: Varias medidas de precisión para el modelo de K-vecinos más próximos

K-vecinos más próximos

Precisión global: 0.759

	Precisión	Sensibilidad	Puntuación F1
Personal	0.754	0.860	0.804
Negocio	0.767	0.622	0.687
Media	0.761	0.741	0.741
Media ponderada	0.760	**0.759**	0.750

Cuadro 5.7: Varias medidas de precisión para el perceptrón multicapa

Perceptrón multicapa

Precisión global: 0.759

	Precisión	Sensibilidad	Puntuación F1
Personal	0.774	0.820	0.796
Negocio	0.735	0.676	**0.704**
Media	0.755	**0.748**	**0.750**
Media ponderada	0.757	**0.759**	**0.757**

Cuadro 5.8: Varias medidas de precisión para el modelo de clasificación ensemble

Modelo ensemble

Precisión global: 0.770

	Precisión	Sensibilidad	Puntuación F1
Personal	**0.800**	0.800	0.800
Negocio	0.730	**0.730**	**0.730**
Media	**0.765**	**0.765**	**0.765**
Media ponderada	**0.770**	**0.770**	**0.770**

Capítulo 6

Análisis de los datos

En el capítulo anterior entrenamos un modelo de clasificación que dividiera las cuentas presentes en el conjunto de datos en personales o de negocios. El objetivo de este capítulo será realizar un análisis de las 6293 cuentas clasificadas como personales, con especial interés en estudiar las posibles diferencias entre los datos de antes y después de la pandemia de 2020.

6.1. Creación de grupos en los datos

El primer paso que tomaremos será aplicar un algoritmo de k-medias, explicado en la sección 2.5, a los datos de cada año (2018 y 2021) de manera independiente. Realizaremos las comparaciones entre pares de grupos, en lugar de entre todos los datos de cada año, con el objetivo de obtener mejores resultados.

En el capítulo anterior ya empleamos un modelo de este tipo para realizar un análisis de las variables que iban a tomar parte en el entrenamiento del modelo de clasificación. Sin embargo, en este caso la mayor parte de esas variables ya no nos interesan (por ejemplo, no es muy relevante ahora la cantidad de números que tenga el nombre de usuario, o si la publicación es un vídeo, a pesar de que sí que lo fueran para distinguir entre tipos de cuenta). Seleccionamos finalmente 5 variables que consideramos interesantes para la creación de grupos: el número de likes y comentarios que recibió una publicación (nos indica el alcance que tuvo cada publicación según sus interacciones), el número de hashtags y menciones que incluyó el autor (que indica el alcance que buscó tener el autor, ya que por ejemplo cuantos más hashtags se utilicen, más probable es que una persona ajena que navega por ellos se encuentre con la publicación), y el índice de lecturabilidad de Flesch del título que la acompaña (publicado en Flesch (1948), visto ya en la sección 1.3). Para aplicar su fórmula a nuestros datos, hemos eliminado los hashtags, menciones y direcciones de páginas web de los textos, y los hemos traducido al inglés.

Una vez seleccionadas las 5 variables, el siguiente paso es transformarlas para poder trabajar mejor con ellas. Si se intenta aplicar el algoritmo de k-medias sobre los datos sin transformar se obtienen grupos de tamaños muy poco balanceados, que no aportan mucha información. Por ejemplo, si en particular se toma $k = 8$, se obtiene que la mitad de los grupos no suman en total 50 datos, mientas que otro de ellos contiene a casi todas las publicaciones. Uno de los principales causantes de esto son las publicaciones que tuvieron una cantidad de interacciones altamente superior a la de la media, por lo que la primera transformación que haremos será aplicar un logaritmo al número de likes y comentarios recibidos. Finalmente, también reescalaremos cada una de las variables para que estén contenidas en el intervalo $[0, 1]$.

Por último, debemos seleccionar el valor de k, es decir, encontrar el número de grupos que mejor pueda representar a nuestros datos. Como ahora ya no tenemos ninguna variable respuesta, debemos emplear un método alternativo a la validación cruzada. Para esto existen varios métodos, dos de los

cuales describiremos a continuación.

El primero que usaremos será el valor de la silueta, o *silhouette*. Se puede encontrar en detalle en Rousseeuw (1987), pero la idea básica es que la silueta es un valor entre -1 y 1 que mide lo bien que se ajusta una observación a su grupo, y lo mal que se ajusta a grupos vecinos. Una puntuación elevada indica que la observación está cerca del centroide de su grupo, y alejada de los de los demás grupos. Por otro lado, una puntuación negativa podría sugerir que la observación está mal clasificada.

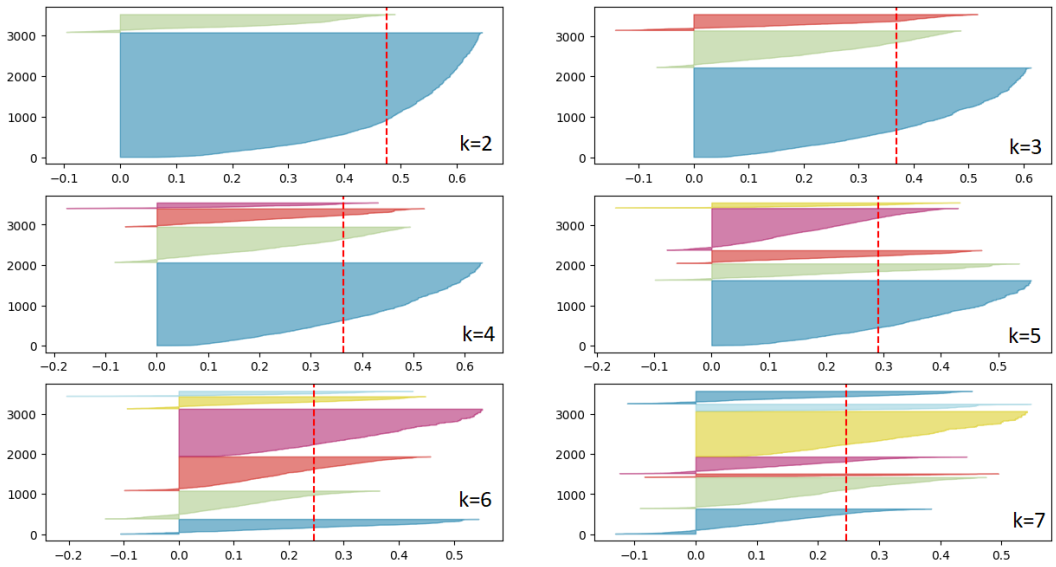

Figura 6.1: Gráficos de siluetas para valores de k entre 2 y 7, para los datos de 2018.

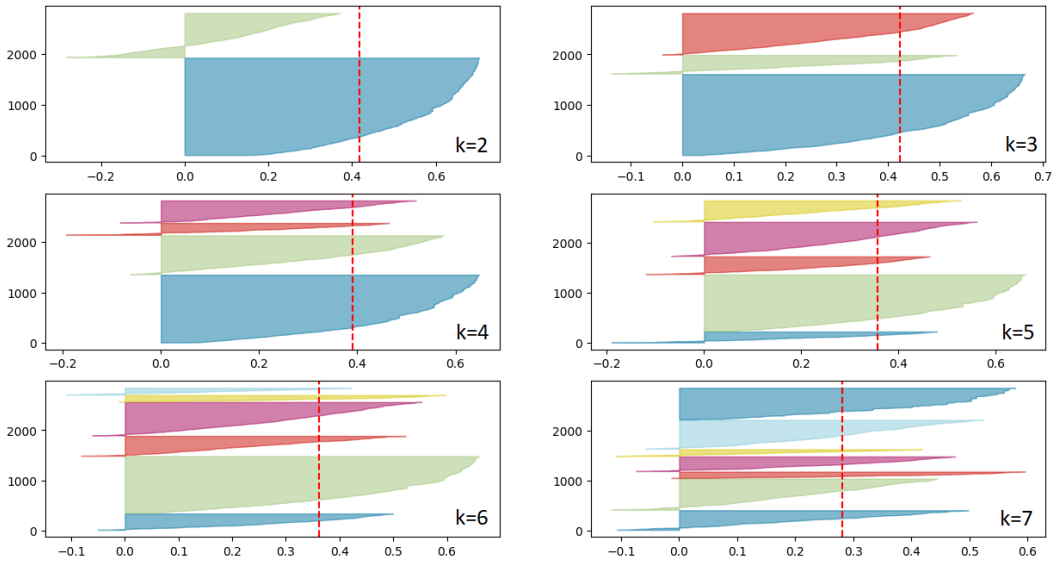

Figura 6.2: Gráficos de siluetas para valores de k entre 2 y 7, para los datos de 2021.

En la figura 6.1 se han representado los gráficos de siluetas para los modelos de k-medias con valores de k entre 2 y 7 para los datos de antes de la pandemia, y en 6.2 para los datos de después de la misma. En este tipo de gráficos, cada región de un color se corresponde con uno de los clusters

formados. La altura de cada región en el eje y representa la cantidad de observaciones que hay en su grupo correspondiente, mientras que su longitud en el eje x representa la evolución de los valores de las siluetas de los elementos que lo componen. La línea roja vertical de cada gráfico representa la media de las siluetas de todos los datos. En general buscamos que ninguna región sea demasiado estrecha (ya que entonces su grupo tendrá muy pocos datos), que no tomen valores muy negativos en exceso y que todas las regiones alcancen la media (el cluster que se corresponda con una región que no la alcance será particularmente malo). Teniendo en cuenta todo esto, por ahora nos inclinamos por tomar $k = 3$, ya que parece ser el que mejor cumple todas las condiciones para ambos grupos de datos.

Otra manera más simple de trabajar con el valor de la silueta es a partir de las medias de las siluetas de los datos. Será preferible en este caso el modelo que tenga mayor valor, ya que es el que tiene grupos que mejor se ajustan a los datos que los componen, y que mejor se distinguen entre si. En la tabla 6.1 hemos representado las medias de las siluetas para valores de k entre 2 y 7, para cada uno de los dos conjuntos. Viendo solo esto probablemente nos quedaríamos con $k = 2$ que tiene mejor media. Sin embargo, en 6.1 vemos que para este valor de k, los grupos que se generan para los datos de 2018 están demasiado desbalanceados, y en 6.2 vemos que no se realiza una clasificación muy buena para el grupo más pequeño de los datos de 2021 (la región verde no alcanza la media, y tiene demasiados valores negativos).

Cuadro 6.1: Media de las siluetas de los datos.

	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$
Pre-COVID	0.475	0.369	0.363	0.291	0.246	0.246
Post-COVID	0.418	0.422	0.390	0.357	0.362	0.280

El otro método de selección de k que emplearemos será el método del codo (*elbow method*, ver por ejemplo Ketchen y Shook 1996). Para cada valor de k considerado, se calcula la distancia de cada observación al centroide respecto al cual ha sido clasificada por el modelo de k medias. Después, se calcula para cada k la suma de estas distancias al cuadrado, y se representan en una función. Se selecciona como k el punto donde la función tenga un codo, es decir, donde cambie visiblemente de pendiente.

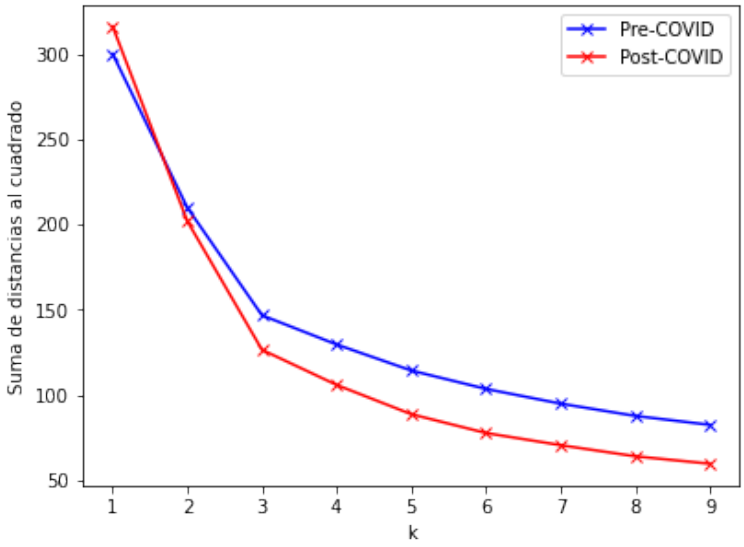

Figura 6.3: Método del codo para la selección de k en el modelo de k-medias.

En nuestro caso hemos realizado dicha representación en 6.3 para ambos conjuntos de datos. En ambos casos vemos que las sumas de las distancias al cuadrado decrecen constantemente a medida que se aumenta el valor de k. Sin embargo, este decrecimiento se vuelve considerablemente más lento a partir de $k = 3$, por lo que por el método del codo seleccionaríamos este valor como hiperparámetro.

En vista de los resultados obtenidos en ambos métodos, dividimos cada uno de los dos conjuntos de datos en 3 grupos a través del algoritmo de k-medias. En la tabla 6.2 incluimos el número de datos que fueron clasificados en cada grupo. Además, hemos representado en 6.4 y 6.5 unas funciones de distribución empírica y diagramas de dispersión para los datos de 2018 y 2021 respectivamente.

Cuadro 6.2: División de los datos en grupos.

	Cluster 1	Cluster 2	Cluster 3
Pre-COVID	2208	905	393
Post-COVID	1601	819	367

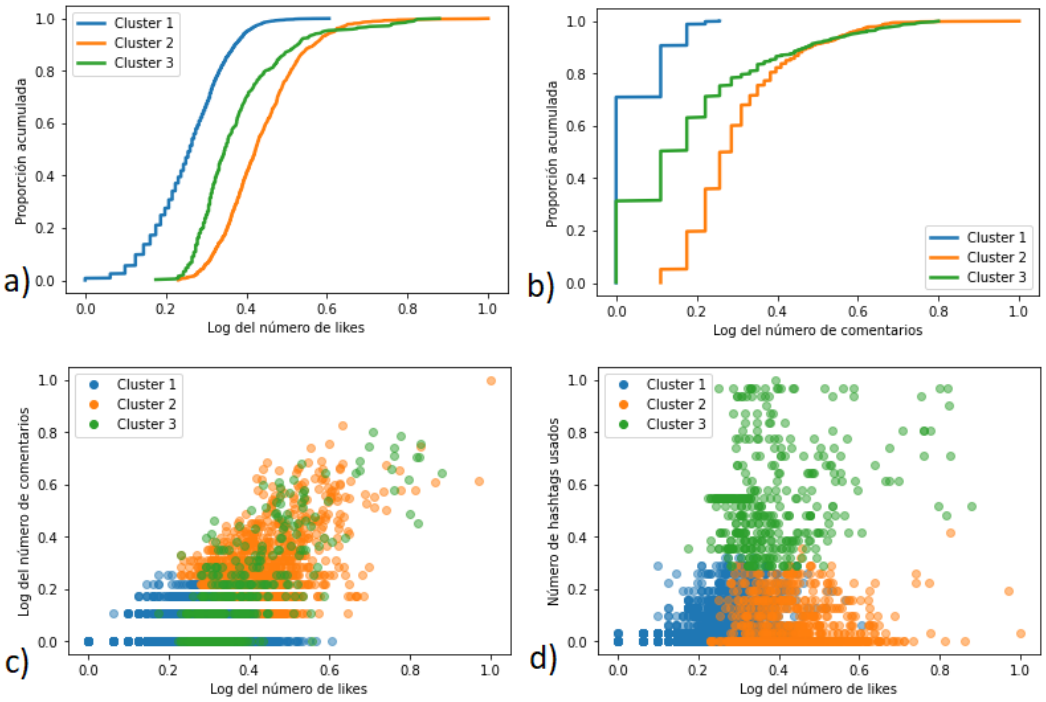

Figura 6.4: Funciones de distribución empírica y diagramas de dispersión para algunas variables de los datos de 2018, distinguiendo entre clusters.

Mirando a las gráficas 6.4 y 6.5 podemos intuir cómo se componen los diferentes grupos de manera aproximada. Ya que las características de los clusters son similares para ambos años, los comentaremos centrándonos en la figura 6.4 de los datos de 2018. En las gráficas (a) y (b) hemos representado las funciones de densidad acumulada de los logaritmos del número de likes y de comentarios recibidos, respectivamente. Vemos que los miembros del cluster 1 son los que de media reciben menos interacciones (particularmente muy pocos comentarios), mientras que los que más reciben son los del cluster 2.

También puede verse que hay una mayor proporción de miembros del cluser 3 que reciben pocas interacciones, que del cluster 2. Sin embargo, la proporción para ambos clusters de individuos que reciben muchas interacciones es casi igual. Esto puede verse también en el gráfico (c), en el que se

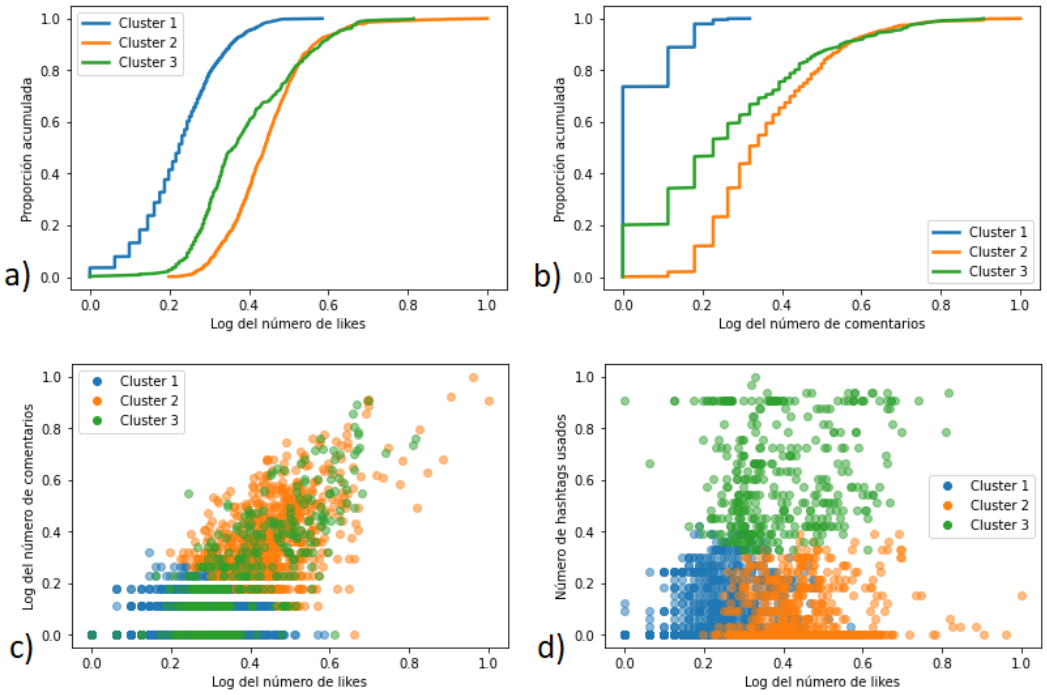

Figura 6.5: Funciones de distribución empírica y diagramas de dispersión para algunas variables de los datos de 2021, distinguiendo entre clusters.

representa un diagrama de dispersión entre el logaritmo del número de likes y el de comentarios. Vemos que las nubes de puntos de los grupos 1 y 2 están relativamente separadas, con la primera de ellas situándose más cerca de la esquina inferior izquierda, que se corresponde con la zona de menos interacciones, y la segunda estando ya más arriba a la derecha. Sin embargo, la nube del tercer grupo es mas difícil de describir, ya que se encuentra parcialmente superpuesta con las otras dos, conteniendo a la vez a individuos con muy pocas y con muchas interacciones.

En el gráfico (d), que se corresponde con el diagrama de dispersión entre el logaritmo del número de likes y el número de hashtags de la publicación, se puede identificar mejor la diferencia entre los dos primeros grupos y el tercero. En particular, se ve que los individuos del cluster 3 son los que emplearon una mayor cantidad de hashtags de todo el conjunto de datos, con resultados variables en cuanto a interacciones obtenidas.

Por tanto, podemos describir a grandes rasgos los clusters de la siguiente manera:

- El **Cluster 1** está compuesto por los individuos que no intentan llegar a una gran audiencia, y por tanto utilizan pocos hashtags. En consecuencia, obtienen una cantidad relativamente baja de interacciones.

- El **Cluster 2** está formado por los individuos que no necesitan usar una cantidad elevada de hashtags para alcanzar a una gran audiencia, sino que consiguen de manera natural una cantidad elevada de interacciones.

- Al **Cluster 3** lo componen las publicaciones que contienen un gran número de hashtags, posiblemente para alcanzar a una mayor cantidad de público, independientemente de la cantidad de interacciones que obtuvieran finalmente.

Para algo más de información acerca de estos clusters, puede verse el apéndice A, en el cual hemos

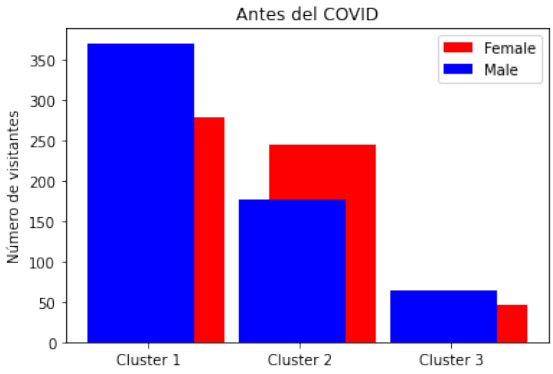

Figura 6.6: Publicaciones de cada género y grupo en 2018.

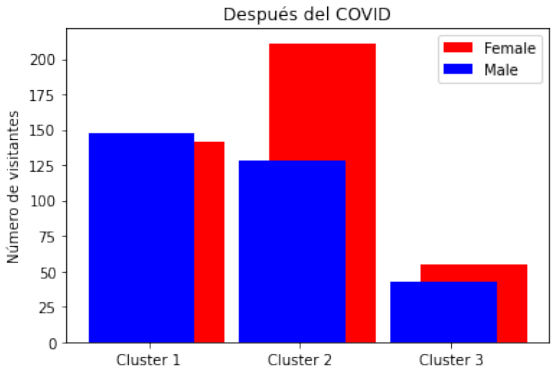

Figura 6.7: Publicaciones de cada género y grupo en 2021.

incluido algunas gráficas similares a las de las figuras 6.4 y 6.5 para otras variables involucradas en el entrenamiento del modelo de k-medias, y que contienen algo de información extra.

Ahora que finalmente tenemos los conjuntos de datos divididos en grupos, dedicaremos la siguiente sección a analizar características de los grupos más allá de las 5 variables empleadas en esta sección, y también estudiaremos las diferencias entre los datos de los años 2018 y 2021 para cada par de grupos equivalentes.

6.2. Estudio de los grupos

En este sección compararemos los grupos en los que hemos dividido a los datos de los años 2018 y 2021, con el objetivo de detectar como ha afectado la pandemia de 2020 al perfil de los turistas que visitan la ciudad de Vigo.

Comenzaremos nuestro estudio por el género de los visitantes. Ya que no se trata de un dato que tengamos, debemos usar las predicciones sobre las imágenes de las publicaciones que realice Face++ (visto en el capítulo 3). Estamos por tanto limitados a estudiar únicamente aquellas publicaciones que sean imágenes (descartando así todos los vídeos), y en las cuales el programa detecte un rostro. De las 6293 publicaciones con las que estamos trabajando, nos encontramos con que 1910 de ellas cumplen estas condiciones. En particular son 1184 publicaciones de 2018 y 726 de 2021.

En la figura 6.6 hemos representado la cantidad de personas de cada género que detectó Face++ en cada grupo, para los datos de 2018, mientras que en 6.7 hemos hecho lo mismo para los datos de 2021. Para el primer cluster, podemos ver en 6.6 que en el año 2018 hubo un 32,5 % más de hombres que de mujeres. En cambio, en 2021 ambos géneros tienen presencia similar dentro de su primer grupo. Ya sabíamos gracias a la tabla 6.2 que el tamaño del primer grupo se redujo en gran medida sobre todo con respecto a los otros dos (se redujeron en 607, 86 y 26 individuos, lo cual suponía el 27,5 %, 9,5 % y 6,6 % de su tamaño original respectivamente). Estos gráficos nos dan la información adicional de que para los 3 clusters disminuyó en mayor medida la cantidad de hombres que de mujeres.

Para establecer si el efecto de la COVID-19 fue realmente significativo sobre la proporción del género de los visitantes, construiremos una tabla de contingencia para estas dos variables (consultar por ejemplo Fienberg 1977). Una tabla de contingencia es una matriz que recoge las frecuencias conjuntas de dos variables categóricas. En la tabla 6.3 hemos representado las tablas de contingencia para los 3 clusters en los que se han agrupado los datos. A simple vista llama la atención que el único caso en el que la cantidad de visitantes en nuestro conjunto de datos aumentó pasada la pandemia es para las mujeres pertenecientes al cluster 3, que pasan de ser 46 en 2018 a 55 en 2021.

Cuadro 6.3: Tablas de contingencia de los 3 grupos, para las variables del género y el año.

Cluster 1

	Mujer	Hombre	Total
2018	280	371	651
2021	141	148	289
Total	421	519	940

Cluster 2

	Mujer	Hombre	Total
2018	245	177	422
2021	211	128	339
Total	456	305	761

Cluster 3

	Mujer	Hombre	Total
2018	46	65	111
2021	55	43	98
Total	101	108	209

Para contrastar si ambas variables (género y año) son independientes entre sí, o si en cambio el año tiene un efecto sobre la proporción de visitantes de cada género, emplearemos el test chi-cuadrado de Pearson (Agresti 2007), en el cual ya entramos en detalle en la sección 5.1. Para un nivel de significación fijado al 0,05, contrastaremos la independencia entre el año y el género dentro de cada uno de los 3 grupos. Los resultados obtenidos están recogidos en la tabla 6.4. Vemos que no existen evidencias significativas para descartar la hipótesis de independencia para los dos primeros grupos, pero sí para el tercero. Por tanto, afirmamos bajo un nivel de significación del 0,05 que la proporción de hombres y mujeres pertenecientes al cluster 3 que visitó Vigo cambió tras la pandemia de 2020, aumentando la proporción de mujeres y disminuyendo la de los hombres, mientras que no tenemos evidencia suficiente para afirmar que la proporción también se vio alterada dentro de los otros dos grupos. Recordamos que las publicaciones de este tercer grupo se caracterizaban por emplear una gran cantidad de hashtags, sin ninguna restricción en cuanto al número de interacciones que reciben.

Cuadro 6.4: Resultados del test chi-cuadrado de independencia entre el año y el género, para cada grupo

	est	pval
Cluster 1	2.474	0.116
Cluster 2	1.202	0.273
Cluster 3	3.924	0.048

El resto de variables para las cuales nos interesa estudiar su independencia con respecto al año en el que se realizó la publicación son la edad detectada por Face++, la puntuación de VADER del título, y su índice de Flesch, todas ellas variables continuas. Por tanto, haremos un pequeño inciso para ver que tests podremos aplicar.

Sean F_1^i y F_2^i las funciones de distribución de alguna variable continua antes y después de la COVID-19 respectivamente, para el cluster i. Queremos realizar el contraste de hipótesis nula $H_0^i : F_1^i = F_2^i$, $i = 1, 2, 3$, contra la alternativa bilateral. Para ello, emplearemos una serie de tests no paramétricos,

que solo requieran como hipótesis la independencia entre las muestras de los años 2018 y 2021.

Los primeros tests que veremos serán el de Kolmogorov-Smirnov para dos muestras, y el de Cramér-von Mises (consultar Pratt y Gibbons 1981b para una descripción detallada). Estos tests utilizan como estadísticos de contraste las distancias entre las funciones de distribución empíricas de las dos muestras, que bajo la hipótesis nula deberían estar próximas entre sí. Sean $\hat{F}_{n_i}$ y $\hat{F}_{m_i}$ las funciones de distribución empíricas de la variable continua antes y después de la COVID-19 respectivamente, para el cluster i, donde n_i y m_i son el número de observaciones en el cluster i en 2018 y 2021 respectivamente. El estadístico de contraste de Kolmogorov-Smirnov viene dado por:

$$D_{n_i,m_i} = \sup_x \left| \hat{F}_{n_i}(x) - \hat{F}_{m_i}(x) \right| \tag{6.1}$$

El test de Crámer-von Mises se basa en cambio las distancias en base a la media cuadrática. Sean $X^i_{1,1}, \ldots, X^i_{1,n_i}$ y $X^i_{2,1}, \ldots, X^i_{2,m_i}$ las observaciones del cluster i para los años 2018 y 2021 respectivamente. El estadístico de Crámer-von Mises viene dado por la expresión:

$$Q^2_{n_i,m_i} = \frac{n_i m_i}{n_i + m_i} \left[\sum_{j=1}^{n_i} \left(\hat{F}_{n_i}(X^i_{1,j}) - \hat{F}_{m_i}(X^i_{1,j}) \right)^2 + \sum_{k=1}^{m_i} \left(\hat{F}_{n_i}(X^i_{2,k}) - \hat{F}_{m_i}(X^i_{2,k}) \right)^2 \right] \tag{6.2}$$

El último test que veremos es el de Mann-Whitney (Pratt y Gibbons 1981a). Este test se basa en suponer que, bajo la hipótesis nula de igualdad de distribuciones, si se mezclan las dos muestras y se ordenan en base a los valores de la variable continua, entonces ambas deberían estar lo suficientemente entremezcladas, y no debería haber una secuencia muy larga de elementos consecutivos de la misma muestra.

Continuando con la notación de antes, supongamos que para cada cluster se han juntado los datos de antes y después de 2020, y se han ordenado de manera ascendente. Para cada $j = 1, \ldots, n_i$ y cada $k = 1, \ldots, m_i$ se define $Z_{jk} = \mathbb{I}(X^i_{1,j} < X^i_{2,k})$, es decir, la variable indicador de si el elemento j de antes de la COVID en el cluster i es menor que el elemento k de después de la COVID en ese mismo cluster. El estadístico de Mann-Whitney vendrá dado entonces por

$$U_{n_i,m_i} = \sum_{j=1}^{n_i} \sum_{k=1}^{m_i} Z_{jk} \tag{6.3}$$

Comenzaremos entonces estudiando la variable de la edad, limitándonos una vez más a las 1910 publicaciones en las cuales Face++ logró detectar un rostro. En las figuras 6.8, 6.9 y 6.10 hemos representado la proporción de las edades detectadas en cada año y dentro de cada uno de los 3 clusters. En los 3 casos parece que la proporción de gente de entre 20 y 25 años ha aumentado. En particular, para el tercer grupo hay bastante más gente de entre 20 y 30 en el año 2021, en favor de la gente entre 30 y 50.

Nos interesa contrastar si, dentro de cada cluster las distribuciones de las edades de los años 2018 y 2021 son similares. Es importante destacar que los 3 tests que acabamos de describir están pensados para ser aplicados sobre variables con distribuciones continuas, y los tests de Kolmogorov-Smirnov y Cramér-von Mises no son válidos en caso de que haya empates, es decir, varias observaciones con los mismos valores de las variables. Por tanto, como los datos de la edad está agrupados por año, aplicaremos únicamente el test de Mann-Whitney, que permite correcciones por empates. Lo haremos con un nivel de significación fijado al 0,05. Recogemos los resultados obtenidos en la tabla 6.5.

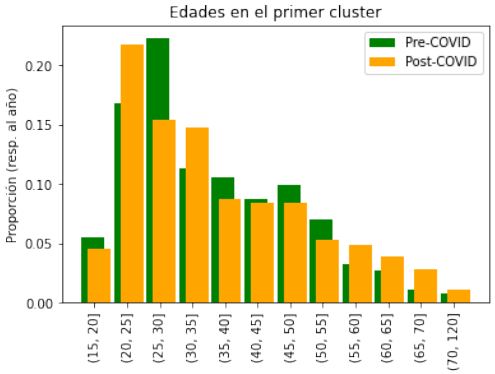

Figura 6.8: Distribución de las edades dentro del primer cluster.

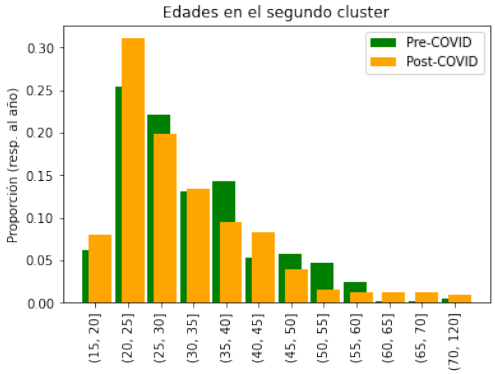

Figura 6.9: Distribución de las edades dentro del segundo cluster.

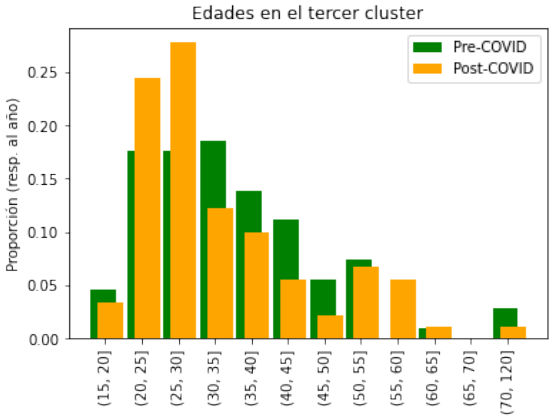

Figura 6.10: Distribución de las edades dentro del tercer cluster.

Cuadro 6.5: Resultados del test de Mann-Whitney para las distribuciones de la edad respecto al año, para cada grupo.

	est	pval
Cluster 1	91007	0.425
Cluster 2	78601	0.019
Cluster 3	6214	0.076

Para los datos de los clusters 1 y 3 (figuras 6.8 y 6.10 respectivamente) el test no encuentra evidencias significativas para rechazar la hipótesis nula, y por tanto no podemos afirmar que la pandemia supuso una variación significativa en cuando a las edades de los visitantes. En cambio, el test rechaza la hipótesis nula para los datos del segundo cluster. Por tanto, bajo un nivel de significación de 0,05, encontramos evidencias significativas de que la proporción de las edades de las personas que visitaron Vigo pertenecientes al segundo grupo se vio afectada con la pandemia de 2020.

Recordamos que las publicaciones pertenecientes al segundo cluster se caracterizaban por recibir un número elevado de interacciones sin emplear muchos hashtags. Viendo la gráfica 6.9 y con los

resultados del test, ahora sabemos que tras la pandemia la gente que pertenece a este grupo es más joven, y en particular se concentra más entre los 20 y 25 años.

También sabíamos gracias a la tabla 6.2 que el primer cluster era el que más integrantes perdió tras la pandemia en proporción con los otros dos. Ahora sabemos que esta pérdida de gente fue más o menos equivalente a lo largo de todas las edades, y que no hubo un grupo de edad que estuviese menos representado en comparación.

A continuación realizaremos un estudio similar sobre las posibles diferencias dentro del índice de lecturabilidad de Flesch de los títulos que acompañan a las publicaciones. Para calcular los índices de los títulos de las publicaciones, habíamos eliminado previamente todos los hashtags y las menciones que contenían. Una vez hecho esto, nos quedamos con que 3878 publicaciones de las 6293 tienen algún texto que analizar, por lo que trabajaremos con estas. Hemos representado sus índices para cada uno de los 3 clusters antes y después de la COVID-19 en las gráficas 6.11,6.12 y 6.13.

Antes de intentar extraer algún tipo de conclusión, le aplicaremos a los datos los tests de Kolmogorov-Smirnov, Cramér-von Mises y Mann-Whitney, con el mismo nivel de significación fijado en 0,05. Hemos recogido los resultados en la tabla 6.6. Podemos ver que ningún test encuentra pruebas significativas al 0,05 de que la distribución de los índices de Flesch de los dos últimos clusters se vio alterada con la pandemia. En cambio, los tres tests también encuentran evidencias significativas como para rechazar la hipótesis nula en el primer cluster, y afirmar que los índices de Flesch de las publicaciones de este grupo presentan diferencias en cuanto a su distribución entre 2018 y 2021.

El primer cluster era el compuesto por publicaciones con pocos hashtags y que recibían pocas interacciones. En vista de la gráfica 6.11 y de los resultados de la tabla 6.6, podemos afirmar que parece haberse reducido la proporción de visitantes a Vigo de este grupo que utilizaban títulos más simples en sus publicaciones, con índices de Flesch de entre 90 y 130 aproximadamente. En cambio, aumenta la proporción de visitantes cuyas publicaciones presentan índices de Flesch menores a 70.

Cuadro 6.6: Resultados de los tests de comparación de las distribuciones del índice de Flesch respecto al año, para cada grupo

	Kolmogorov-Smirnov		**Cramér-von Mises**		**Mann-Whitney**	
	est	pval	est	pval	est	pval
Cluster 1	0.116	1.898e-6	3.442	7.986e-9	612863	5.588e-9
Cluster 2	0.050	0.360	0.152	0.384	210312	0.480
Cluster 3	0.075	0.463	0.171	0.333	31253	0.298

Puede realizarse finalmente un estudio similar basado esta vez en el análisis de sentimientos en los textos. En nuestro caso, lo hemos hecho mediante la herramienta de VADER (capítulo 4). Para cada publicación del conjunto de datos, traducimos su título al inglés y pasamos esta traducción por VADER para obtener una medida entre -1 y 1 de las emociones detectadas en los textos. Al hacerlo, encontramos que en mucho textos VADER no logra encontrar ningún indicio de emociones positivas o negativas, por lo que devuelve una puntuación perfectamente neutra de 0. Para trabajar con esta variable, hemos optado por no considerar esos datos, quedándonos así con 2801 publicaciones de las 6293 totales. Los gráficos de las puntuaciones obtenidas para cada cluster están representados en las figuras 6.14, 6.15 y 6.16.

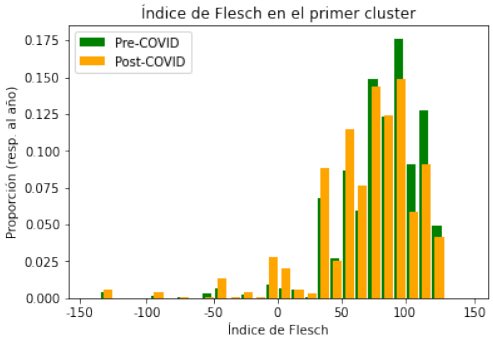

Figura 6.11: Distribución de los índices de Flesch dentro del primer cluster.

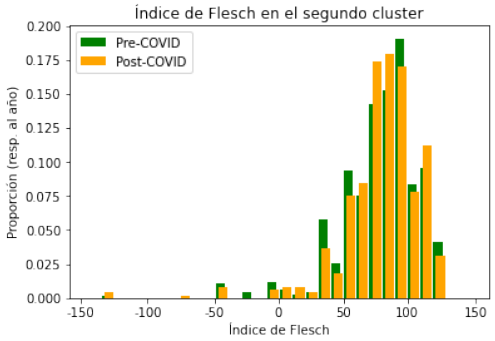

Figura 6.12: Distribución de los índices de Flesch dentro del segundo cluster.

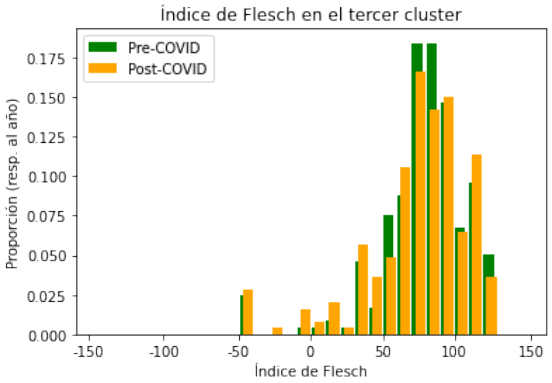

Figura 6.13: Distribución de los índices de Flesch dentro del tercer cluster.

Nos interesa realizar el mismo contraste que en los casos anteriores, que tiene como hipótesis nula para cada grupo la igualdad de distribuciones en 2018 y 2021, contra la alternativa bilateral. En la tabla 6.7 recogemos los resultados obtenidos para los tests de Kolmogorov-Smirnov, Cramér-von Mises y Mann-Whitney.

Cuadro 6.7: Resultados de los tests de comparación de las distribuciones de la puntuación de VADER respecto al año, para cada grupo

	Kolmogorov-Smirnov		**Cramér-von Mises**		**Mann-Whitney**	
	est	pval	est	pval	est	pval
Cluster 1	0.081	0.028	0.551	0.030	193655	0.037
Cluster 2	0.093	0.020	0.746	0.010	124731	0.009
Cluster 3	0.086	0.385	0.192	0.284	23481	0.590

Fijando como hasta ahora un nivel de significación de 0,05, tenemos que todos los tests rechazan la hipótesis nula para los dos primeros clusters, mientras que no encuentran evidencia significativa para rechazarla en el tercero. Es decir, que existen evidencias significativas bajo un nivel de significación

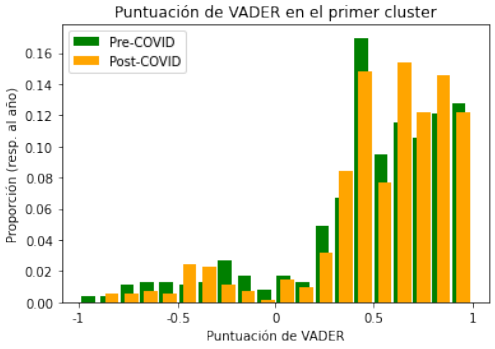

Figura 6.14: Distribución de las puntuaciones de VADER dentro del primer cluster.

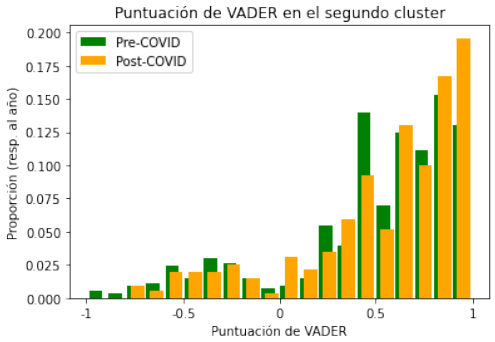

Figura 6.15: Distribución de las puntuaciones de VADER dentro del segundo cluster.

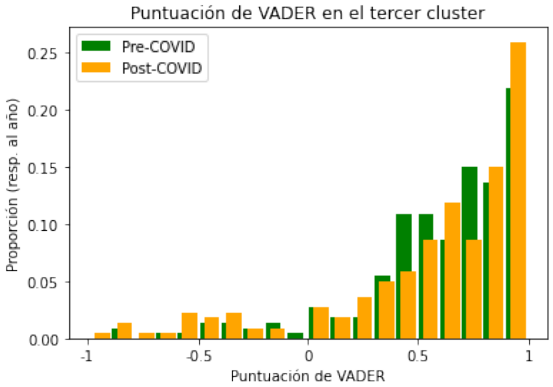

Figura 6.16: Distribución de las puntuaciones de VADER dentro del tercer cluster.

del 0,05 de que los sentimientos plasmados en los textos de las publicaciones pertenecientes a los dos primeros clusters experimentaron una variación entre 2018 y 2021.

En general parece que las puntuaciones de VADER están más concentradas en los valores más altos en el año 2021 que en el 2018, así que parece que pasada la COVID-19, los textos que acompañan a las publicaciones son generalmente más positivos de lo que eran antes.

Finalmente, haremos un último análisis de las imágenes de las publicaciones. Hasta ahora tan solo hemos trabajado con las imágenes de 1910 publicaciones, que son aquellas en las que la herramienta de Face++ detectó un rostro. Sin embargo, como parte del conjunto de datos contamos con 5148 imágenes de las 6293 publicaciones clasificadas como personales, por lo que no estamos utilizando la información de más de 3000 imágenes.

Nuestro objetivo es por tanto extraer algo más de información de todas las imágenes del conjunto de datos. Para ello, emplearemos la herramienta de ResNeXT (Xie et al. 2017). ResNeXT es un modelo de deep learning que permite la detección de objetos en imágenes. Utilizaremos en particular su implementación sobre el conjunto de datos Imagenet-1K (Russakovsky et al. 2014), que distingue hasta 1000 clases diferentes, que van desde múltiples razas de perro hasta distintas prendas de ropa o herramientas. Sin embargo, que haya tantas clases diferentes puede dificultar el análisis debido a estar los datos demasiado dispersos. Por ello usaremos una agrupación de las 1000 etiquetas en 67 categorías[1], y realizaremos el análisis sobre estas.

En las figuras 6.17, 6.18 y 6.19 hemos representado la proporción de imágenes clasificadas por

[1] https://github.com/noameshed/novelty-detection/blob/master/imagenet_categories.csv

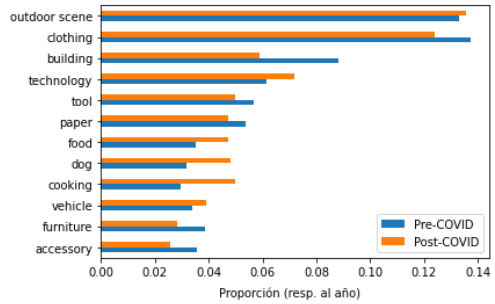

Figura 6.17: Categorías detectadas en las imágenes por ResNeXt en el primer cluster.

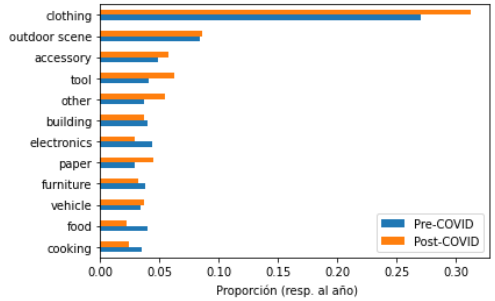

Figura 6.18: Categorías detectadas en las imágenes por ResNeXt en el segundo cluster.

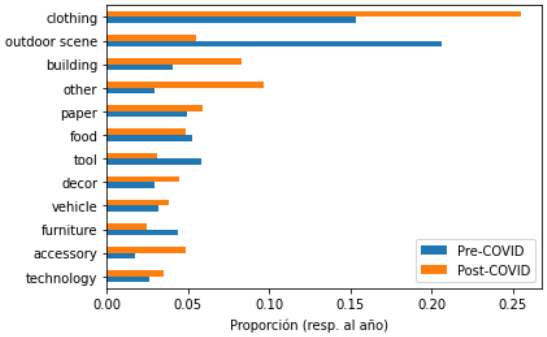

Figura 6.19: Categorías detectadas en las imágenes por ResNeXt en el tercer cluster.

ResNeXt en cada grupo, para cada cluster y distinguiendo entre 2018 y 2021. Comenzamos viendo lo que ocurre en el primer cluster, en la figura 6.17. Vemos que las clases más representadas con diferencia son las de escenas al aire libre y las de prendas de ropa, seguidas de edificios y aparatos tecnológicos. En particular, las etiquetas más comunes dentro de los escenarios al aire libre eran todas relacionadas con el agua, como "playa", "puerto", o "rompeolas". Además, las etiquetas de ropa predominantes son "vaqueros", "jersey" o "gafas de sol", por lo que no parece que los integrantes de este cluster sacasen muchas fotos de sí mismos en la playa. En cuanto a las demás categorías de imágenes, para los edificios destacan las etiquetas de "restaurante" y "cine", y en tecnología predomina "páginas web", por lo que una cantidad importante del cluster se compone de capturas de sitios web, en lugar de fotografías tomadas por el dueño de la cuenta.

A continuación pasamos al segundo cluster, que se corresponde con la figura 6.18. Podemos ver como las fotografías en las que se detectan prendas de vestir dominan completamente a todas las demás categorías, representando un 30 % de las imágenes totales del cluster. Algunas de las etiquetas con mayor presencia dentro de las prendas de vestir en 2018 fueron "vaqueros", "gafas de sol", "maillot", "minifalda", "traje" o "jersey", mientras que en 2021 destacamos "bikini", "gafas de sol", "maillot" o "bañador". Esto parece sugerir que antes de la pandemia la gente publicaba menos imágenes con ropa de ir a la playa que después, y algunos integrantes del cluster visitaron Vigo en ambos años con el objetivo de hacer ciclismo.

Finalmente vemos en 6.19 que para el cluster 3 las clases más predominantes vuelven a ser ropa e imágenes al aire libre. Aquí llama la atención sin embargo que la proporción de publicaciones con imágenes en las que se detectó ropa aumentó en más de un 50 % para 2021 con respecto a 2018, mientas que aquellas en las que se detectan escenarios al aire libre disminuyeron hasta un cuarto del original. En particular, las etiquetas más comunes para las publicaciones de 2018 donde se detectó un escenario al aire libre fueron "playa" y "banco de arena", mientras que para 2021 destacamos la etiqueta de

“puerto” . Para las prendas de vestir, en 2018 destacamos “gafas de sol”, “bañador” y “maillot”, y en 2021 “pareo” y “bikini”. Estas etiquetas concuerdan con la tendencia vista en las gráficas 6.6 y 6.7 y el test de chi-cuadrado con resultados recogidos en la tabla 6.4 de que, dentro del tercer cluster, la proporción de hombres en 2018 era mayor que la de mujeres, mientras que en 2021 se invirtió esa tendencia.

Capítulo 7

Conclusiones y trabajo futuro

En este trabajo hemos podido realizar un análisis del perfil de turismo de los visitantes de Vigo empleando únicamente publicaciones recogidas de Instagram. Las múltiples herramientas de análisis de imágenes, sentimientos, etc., de acceso público que existen a día de hoy dan la posibilidad de recabar información fuera del alcance del esfuerzo humano en un tiempo razonable.

Al agrupar a las publicaciones integrantes de nuestro conjunto de datos según su comportamiento dentro de la red social (a través del alcance que intentaban tener en base al número de hashtags que empleaban, y del que lograban tener en base al número de interacciones), logramos detectar diferencias significativas en distintos atributos de los turistas pertenecientes a determinados grupos. Por ejemplo, identificamos que el grupo de personas que obtiene menos interacciones y tampoco utiliza muchos hashtags ha reducido en gran medida su tamaño, particularmente en proporción con los otros dos grupos. Para este mismo grupo también detectamos que los títulos de sus publicaciones se han vuelto considerablemente más complejos, en cuanto a su facilidad de lectura. Por otro lado, para el grupo de visitantes que emplea más hashtags, detectamos que su proporción de mujeres y de gente joven aumentó en 2021 en comparación con 2018.

Una vez estudiadas estas diferencias, el modelo de ResNeXt nos permite tener una mayor información del contenido de las imágenes que forman parte de las publicaciones. Detectamos así por ejemplo que el primer grupo anteriormente mencionado con una menor cantidad de interacciones es además el único sin una proporción significativa de imágenes en las cuales se detecte ropa de playa, a diferencia de los otros dos grupos (lo cual no debería ser achacable a condiciones meteorológicas, ya que no existe una separación temporal entre las publicaciones de distintos grupos dentro de cada uno de los dos años considerados).

Una posible continuación del análisis realizado en este trabajo podría partir del uso de un nuevo conjunto de datos que, en vez de recoger únicamente publicaciones con localización en Vigo, esté formado por las propias cuentas que en múltiples ocasiones hayan marcado esa ciudad como ubicación de sus publicaciones. Esto permitiría llevar a cabo un análisis adicional de los patrones de visitas de la gente que viaja a la ciudad de manera reiterada. Así, podríamos distinguir posibles efectos de la pandemia de 2020 en los patrones visita de estos visitantes reiterados.

También podría de especial interés para un futuro análisis el estudio de las zonas de la ciudad que más visitan distintos grupos demográficos. El modelo de ResNeXt ya posibilita un acercamiento a esta tarea, al dar la posibilidad de detectar imágenes características de distintas zonas como podrían ser un puerto, una fuente o una iglesia. Pero profundizar más en esta tarea resulta de gran interés a comercios relacionados con el sector turístico, ya que les permite conocer en mayor profundidad a su público, lo cual supone a su vez la posibilidad de ofrecer un mejor servicio.

Otra posibilidad sería la de realizar un estudio más exhaustivo de los títulos que acompañan a las publicaciones. La división de grupos podría realizarse por ejemplo a través de un modelo Word2Vec aplicado sobre estos títulos. También pueden resultar de utilidad en el análisis posterior, para realizar por ejemplo un estudio de las palabras más comunes en los diferentes grupos y acompañar a las

predicciones obtenidas a partir de ResNeXt.

Finalmente, podría extenderse este análisis a otras redes sociales como puedan ser Facebook o Twitter. El método de estudio debería ser adaptado para trabajar con estas plataformas diferentes, debido a sus diferencias en cuanto a diseño y comportamiento de sus usuarios. Por ejemplo, a diferencia de Instagram que requiere de una imagen o vídeo para realizar un publicación, Twitter está mucho más basado en el texto, y por tanto requeriría centrarse más en modelos de análisis de textos como Word2Vec, y dejar más de lado los de análisis facial o de imágenes en general. Las conclusiones obtenidas de estas otras plataformas podrían aportar nueva información y complementar a los resultados de este trabajo.

Apéndice A

Información adicional de los clusters

En la sección 6.1 dividimos la parte de nuestro conjunto de datos etiquetada como cuentas personales en 3 grupos, en función de 5 variables: el número de likes y comentarios recibidos por la publicación, el número de hashtags y menciones que empleó, y el índice de Flesch de su título. También en esa sección incluimos unos gráficos que nos aportaban información acerca de en base a qué variables se había producido la creación de grupos por el algoritmo de k-medias. En esta sección incluiremos por completitud unos gráficos extra que aportan menos información pero también pueden resultar interesantes.

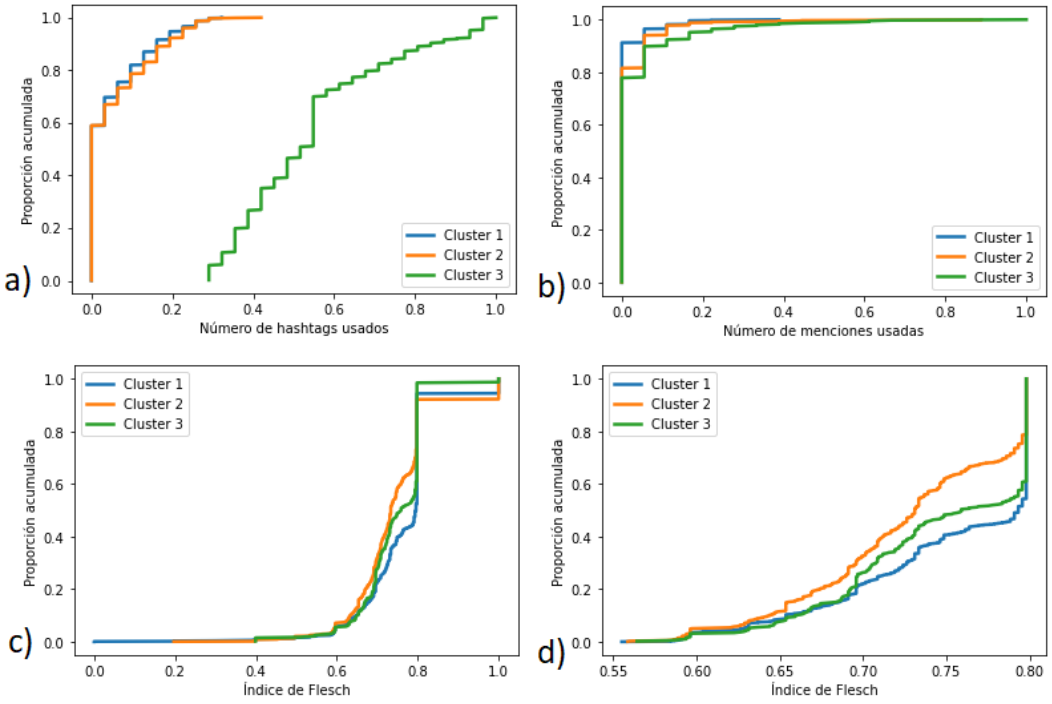

Figura A.1: Varias funciones de distribución empírica para algunas variables de los datos de 2018, distinguiendo entre clusters.

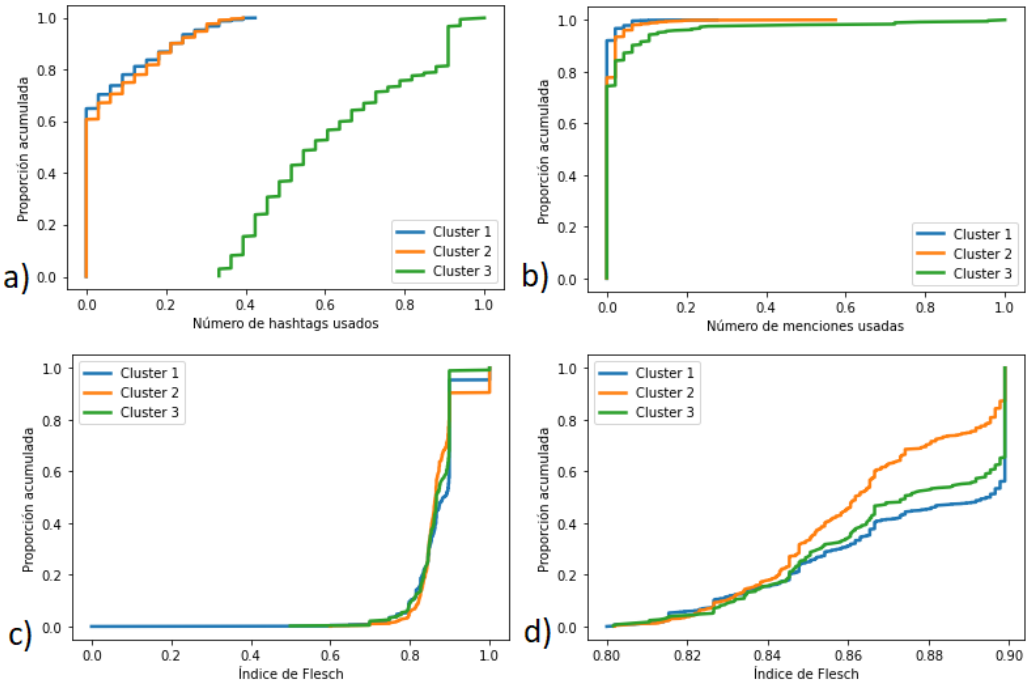

Figura A.2: Varias funciones de distribución empírica para algunas variables de los datos de 2021, distinguiendo entre clusters.

Comenzamos representando las funciones de distribución empíricas de las 3 variables que formaron parte del modelo de k-medias y no representamos en la sección 6.1. En particular, tenemos las de los datos de 2018 en A.1, y las de 2021 en A.2. Las conclusiones que podemos extraer son similares para ambos años, por lo que los comentarios a continuación pueden aplicarse a ambas figuras. En ambos casos, la gráfica (a) se corresponde con la distribución empírica del número de hashtags usados. La diferencia entre los clusters 1 y 2 es mínima, mientras que el cluster 3 destaca claramente por usar muchos más hashtags. Sin embargo, es notable en estas gráficas que la división entre los clusters 1 y 2 y el 3 no es perfecta en base a únicamente esta variable, ya que hay publicaciones de los dos primeros grupos que emplean más hashtags que ciertas publicaciones del tercer grupo.

En (b) se representa la distribución empírica para el número de menciones. Aquí el tercer grupo parece destacar una vez más por emplear de media más menciones que el resto, pero se trata de una diferencia muy pequeña como para extraer conclusiones significativas.

Finalmente en (c) representamos la distribución empírica para el índice de Flesch de los títulos de las publicaciones, mientras que en (d) representamos las secciones de esas gráficas en las cuales se concentra mayor proporción de datos, para eliminar atípicos. Para ambos años se ve en estas secciones que las publicaciones del cluster 2 tienen generalmente un índice de Flesch menor, por lo que sus títulos son más complejos que la media. En cambio, las del cluster 1 son también para ambos años las que usan títulos más simples.

Finalmente, incluiremos también unos diagramas de dispersión extras en las figuras A.3 y A.4 para los datos de 2018 y 2021 respectivamente. En (a) representamos el logaritmo del número de likes recibidos contra el número de menciones empleadas. Se ve en A.3 que los datos del cluster 3 antes de la COVID-19 parecían estar más concentrados en un punto medio sobre todo respecto al número de likes, sin alcanzar valores tan bajos como los del cluster 1. En cambio, observando A.4 parece que la situación cambió en 2021, y hay una mayor proporción de publicaciones de este tercer cluster que reciben muy pocas interacciones.

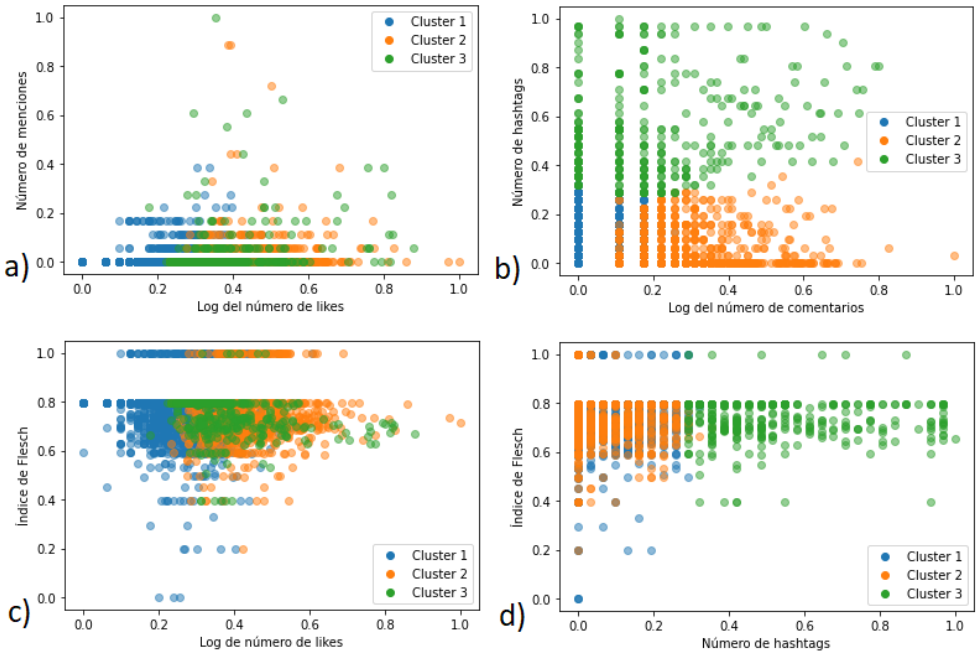

Figura A.3: Varios diagramas de dispersión para algunas variables de los datos de 2018, distinguiendo entre clusters.

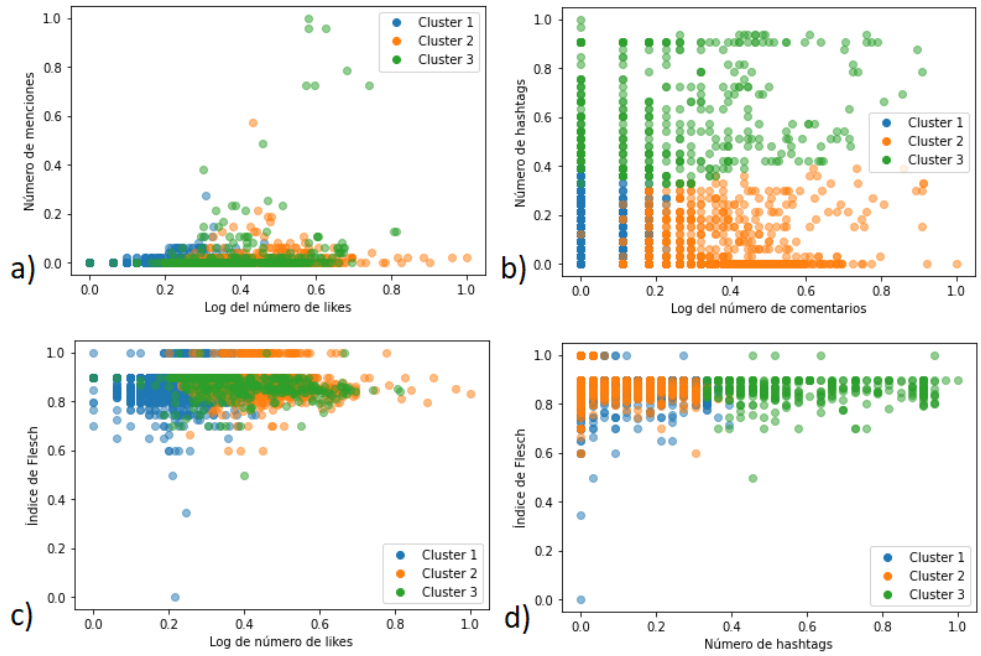

Figura A.4: Varios diagramas de dispersión para algunas variables de los datos de 2021, distinguiendo entre clusters.

En (b) representamos el logaritmo del número de comentarios recibidos contra la cantidad de hashtags empleados. Estos gráficos proporcionan una información similar a los (d) de las figuras 6.4 y 6.5 en el sentido de que permiten visualizar de manera separada los 3 clusters. Los gráficos que representamos en este apéndice tienen la pega de que el número de comentarios es generalmente mucho menor que el de likes, por lo que para valores bajos de comentarios, estos nuevos gráficos tienen aspecto de líneas verticales, en lugar de una nube más homogénea.

En los dos gráficos restantes representamos nubes de puntos que incluyen al índice de Flesch de los títulos de las publicaciones. En particular, en (c) lo representamos contra el logaritmo del número de likes recibidos, y en (d) contra el número de hashtags empleados. Es posible que en A.3 pueda apreciarse una ligera tendencia a que las publicaciones que emplean menos hashtags tengan un menor índice del Flesch que la media, lo cual quiere decir que emplean títulos más complejos. Pero incluimos estos gráficos principalmente para mostrar que no parece haber una relación entre el índice de Flesch y las demás variables, y que simplemente parece haber una mayor variación en cuanto a este índice en las zonas de mayor densidad de datos como es lógico.

Bibliografía

[1] Agresti A (2007). An Introduction to Categorical Data Analysis, John Wiley & Sons Hoboken, NJ, pp 34-41.

[2] Altman NS (1992). An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician. 46 (3): 175-185.

[3] Basarslan M, Kayaalp F (2021). Sentiment Analysis with Machine Learning Methods on Social Media. Advances in Distributed Computing and Artificial Intelligence Journal, 9, 5-15. 10.14201/ADCAIJ202093515.

[4] Belson WA (1959). Matching and Prediction on the Principle of Biological Classification. Journal of the Royal Statistical Society. Series C (Applied Statistics), 8(2), 65-75.

[5] Benevenuto F, Magno G, Rodrigues T, Almeida V (2010). Detecting spammers on twitter. Collaboration, electronic messaging, anti-abuse and spam conference (CEAS). Vol. 6.

[6] Boser B, Guyon I, Vapnik V (1996). A Training Algorithm for Optimal Margin Classifier. Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory, 144-152.

[7] Bradley PS, Fayyad U (1998). Refining Initial Points for K-Means Clustering. Proc. of the 15th Int. Conf. on Machine Learning, 91-99.

[8] Breiman L, Friedman JH, Olshen RA, Stone CJ (1984). Classification And Regression Trees (1st ed.). Routledge.

[9] Breiman L (1996). Bagging Predictors. Machine Learning, 123-140.

[10] Celebi ME, Kingravi H, Vela P (2013). A Comparative Study of Efficient Initialization Methods for the K-Means Clustering Algorithm. Expert Systems with Applications, 40, 200-210.

[11] Chervonenkis AY (2013). Early History of Support Vector Machines. In: Schölkopf, B., Luo, Z., Vovk, V. (eds) Empirical Inference. Springer, Berlin, Heidelberg.

[12] Cortes C, Vapnik V (1995). Support-vector networks. Machine Learning, 20(3), 273-297.

[13] Ersahin B, Aktas O, Kilinc D, Akyol C (2017). Twitter Fake Account Detection. International Conference on Computer Science and Engineering (UBMK).

[14] Fernández J (1959). Medidas sencillas de lecturabilidad. Consigna, 214, 29-32.

[15] Fienberg SE (1977). The Analysis of Cross-Classified Categorical Data. Cambridge, MA: MIT Press.

[16] Fix E, Hodges JL (1989). Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties. International Statistical Review / Revue Internationale de Statistique, 57(3), 238-247.

[17] Flesch R (1948). A new readability yardstick. Journal of Applied Psychology, 32(3), 221-233.

[18] Glass GV, Peckham PD, Sanders JR (1972). Consequences of failure to meet assumptions underlying fixed effects analyses of variance and covariance. Rev. Educ. Res. 42: 237-288.

[19] González-Rodríguez M, Díaz-Fernández MC, Gómez C (2020). Facial-Expression Recognition: an emergent approach to the measurement of tourist satisfaction through emotions. Telematics and Informatics. 51. 101404.

[20] Gunning R (1952). The Technique of Clear Writing. McGraw-Hill.

[21] Guo G, Zhang N (2019). A survey on deep learning based face recognition. Comput. Vis. Image Underst., 189.

[22] Hajian-Tilaki K (2013). Receiver Operating Characteristic (ROC) Curve Analysis for Medical Diagnostic Test Evaluation. Caspian journal of internal medicine, 4(2), 627-635.

[23] Ho TK (1995) Random Decision Forest. Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, 14-16 August 1995, 278-282.

[24] Hu G, Yang Y, Yi D, Kittler J, Christmas W, Li SZ, Hospedales T (2015). When Face Recognition Meets with Deep Learning: an Evaluation of Convolutional Neural Networks for Face Recognition. arXiv:1504.02351

[25] Hutto C, Gilbert E (2014). VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. Proceedings of the International AAAI Conference on Web and Social Media, 8(1), 216-225.

[26] Insaf A, Ouahabi A, Benzaoui A, Taleb-Ahmed A (2020). Past, Present, and Future of Face Recognition: A Review. Electronics. 9. 1188. 10.3390/electronics9081188.

[27] Jung SG, An J, Kwak H, Salminen J, Jansen BJ (2018). Assessing the Accuracy of Four Popular Face Recognition Tools for Inferring Gender, Age, and Race. Proceedings of the 12th International AAAI Conference on Web and Social Media, ICWSM 2018, Palo Alto, California USA, June 25-28. 624-627. Research Collection School Of Information Systems.

[28] Kalra G, Yu M, Lee D, Cha M, Kim D (2018). Ballparking the Urban Placeness: A Case Study of Analyzing Starbucks Posts on Instagram. In International Conference on Social Informatics, 291-307.

[29] Kanade T (1977). Computer recognition of human faces.

[30] Kärkkäinen K, Joo J (2019). FairFace: Face Attribute Dataset for Balanced Race, Gender, and Age for Bias Measurement and Mitigation. arXiv:1908.04913

[31] Kaufman L, Rousseeuw P (1990) Finding Groups in Data: An Introduction to Cluster Analysis, Wiley-Interscience.

[32] Ketchen DJ, Shook CL (1996). The Application of Cluster Analysis in Strategic Management Research: An Analysis and Critique. Strategic Management Journal, 17(6), 441-458.

[33] Kirby M, Sirovich L (1990). Application of the Karhunen-Lokve Procedure for the Characterization of Human Faces. Transactions On Pattern Analysis And Machine Intelligence. Vol. 12, No. I, 103-108.

[34] Kirilenko AP, Stepchenkova SO, Kim H, Li XR. (2018). Automated Sentiment Analysis in Tourism: Comparison of Approaches. Journal of Travel Research, 57(8), 1012-1025.

[35] Kiritchenko S, Mohammad SM (2018). Examining gender and race bias in two hundred sentiment analysis systems. arXiv:1805.04508

[36] Kornbrot D (2005). Point Biserial Correlation. 10.1002/0470013192.bsa485.

[37] Li J, Cardie C, Li S (2013). TopicSpam: a Topic-Model-Based Approach for Spam Detection. Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 217-221.

[38] Lloyd SP (1957). Least square quantization in PCM. Bell Telephone Laboratories Paper.

[39] Lloyd SP (1982). Least squares quantization in PCM (PDF). IEEE Transactions on Information Theory. 28 (2): 129-137.

[40] MacQueen J (1967) Some Methods for Classification and Analysis of Multivariate Observations. Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, 1, 281-297.

[41] Mäntylä MV, Graziotin D, Kuutila M (2018). The evolution of sentiment analysis:A review of research topics, venues, and top cited papers. Computer Science Review, Volume 27, 16-32.

[42] Matthews BW (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme, Biochimica et Biophysica Acta (BBA) - Protein Structure, Volume 405, Issue 2, 442-451.

[43] McHugh ML (2013). The chi-square test of independence. Biochemia medica, 23(2), 143-149.

[44] Mikolov T, Chen K, Corrado G, Dean J (2013). Effcient estimation of word representations in vectorspace. arXiv:1301.3781

[45] Mitchell T. (2006). The Discipline of Machine Learning.

[46] Nilsson NJ (2009). The Quest for Artificial Intelligence. A History of Ideas and Achievements, 172-173.

[47] Omena JJ (2017). Insta Bots and the black market of social media engagement. 10.13140/RG.2.2.19685.42722.

[48] Pang B, Lee L, Vaithyanathan S (2002). Thumbs up? Sentiment Classification using Machine Learning Techniques. Proceedings of the ACL-02 conference on Empirical methods in natural language processing. Volume 10, 79-86.

[49] Park SB, Ok CM, Chae BK (2016). Using Twitter data for cruise tourism marketing and research. Journal of Travel & Tourism Marketing, 33(6), 885-898.

[50] Phillips PJ, Grother P, Micheals R, Blackburn D, Tabassi E, Bone J (2003). Facial Recognition Vendor Test 2002. Evaluation Report.

[51] Phillips PJ, Flynn PJ, Scruggs T Bowyer K, Worek W (2006). Preliminary Face Recognition Grand Challenge Results, 15-24. 10.1109/FGR.2006.87.

[52] Pratt JW, Gibbons JD (1981a). Concepts of Nonparametric Theory. Springer-Verlag New York Inc, pp 249-264.

[53] Pratt JW, Gibbons JD (1981b). Concepts of Nonparametric Theory. Springer-Verlag New York Inc, pp 318-344.

[54] Quinlan JR (1986). Induction of Decision Trees. MACH. LEARN vol 1, 81-106.

[55] Rauss PJ, Phillips J, Hamilton MK, DePersia A (1997). The FERET (FacE REcognition Technology) Program. Proc. SPIE 2962, 25th AIPR Workshop: Emerging Applications of Computer Vision.

[56] Rosenblatt F (1957). The Perceptron: a perceiving and recognizing automaton. Report 85-460-1. Cornell Aeronautical Laboratory.

[57] Rousseeuw PJ (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, Volume 20, 53-65.

[58] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, Berg AC, Fei-Fei L (2014). ImageNet Large Scale Visual Recognition Challenge. arXiv:1409.0575.

[59] Ryu K, Lee M (2016). A Study on Smart Tourism Based on Face Recognition Using Smartphone. International Journal of Internet, Broadcasting and Communication Vol.8 No.4, 39-47.

[60] Shankar S, Halpern Y, Breck E, Atwood J, Wilson J, Sculley D (2017). No classification without representation: Assessing geodiversity issues in open data sets for the developing world. arXiv preprint arXiv:1711.08536

[61] Sheela K, Deepa SN (2013). Review on Methods to Fix Number of Hidden Neurons in Neural Networks. Mathematical Problems in Engineering. 10.1155/2013/425740.

[62] Sokolova M, Lapalme G (2009). A systematic analysis of performance measures for classification tasks, Information Processing & Management, Volume 45, Issue 4, 427-437.

[63] Spaulding S (1956). A Spanish Readability Formula. The Modern Language Journal, 40(8), 433-441.

[64] Stone M (1974). Cross-Validatory Choice and Assessment of Statistical Predictions, Journal of the Royal Statistical Society. Series B (Methodological), Vol. 36, No. 2 (1974), 111-147.

[65] Stringhini G, Kruegel C, Vigna G (2010). Detecting spammers on social networks. Proceedings of the 26th Annual Computer Security Applications Conference, 1-9.

[66] Tenkanen H, Di Minin E, Heikinheimo V, Hausmann A, Herbst M, Kajala L, Toivonen T (2017). Instagram, Flickr, or Twitter: Assessing the usability of social media data for visitor monitoring in protected areas. Scientific reports, 7(1), 1-11.

[67] Thelwall M, Buckley K, Paltoglou G (2012). Sentiment strength detection for the social web. Journal of the American Society for Information Science and Technology 63(1), 163-173.

[68] Turk M, Pentland A (1991). Eigenfaces for Recognition. Journal of Cognitive Neuroscience, 3(1), 71-86.

[69] Turney PD (2003). Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. Proceedings of the 40th annual meeting on association for computational linguistics, 417-424.

[70] Vapnik VN, Chervonenkis AYa (1964). A class of algorithms for pattern recognition learning, Avtomat. i Telemekh., 25:6, 937?945.

[71] Wiebe JM, Bruce RF, O'Hara TP (1999). Development and Use of a Gold-Standard Data Set for Subjectivity Classifications. Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics, 246-253.

[72] Xie S, Girshick R, Dollar P, Tu Z, He K (2017). Aggregated Residual Transformations for Deep Neural Networks. arXiv:1611.05431.

[73] Yoo K, Gretzel U (2009). Comparison of Deceptive and Truthful Travel Reviews. Information and Communication Technologies in Tourism, 37-47.