



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA INDUSTRIAL

EFFECTO DE LA CONGESTIÓN DE TRÁFICO EN EL DESEMPEÑO DE MODELOS DE
PREDICCIÓN DE VELOCIDAD A CORTO PLAZO MEDIANTE APRENDIZAJE
AUTOMÁTICO

TESIS PARA OPTAR AL GRADO DE
MAGÍSTER EN GESTIÓN DE OPERACIONES

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL INDUSTRIAL

CRISTIÁN FELIPE LIRA CAMILO

PROFESOR GUÍA:
FERNANDO ORDOÑEZ PIZARRO

MIEMBROS DE LA COMISIÓN:
SEBASTIÁN RÍOS PÉREZ
BASTIÁN VÉJAR TEMER

Este trabajo ha sido parcialmente financiado por ANID

SANTIAGO DE CHILE
2022

RESUMEN DE LA TESIS PARA OPTAR AL GRADO
DE MAGÍSTER EN GESTIÓN DE OPERACIONES
MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL INDUSTRIAL
POR: CRISTIÁN FELIPE LIRA CAMILO
FECHA: 2022
PROF. GUÍA: FERNANDO ORDOÑEZ PIZARRO

EFFECTO DE LA CONGESTIÓN DE TRÁFICO EN EL DESEMPEÑO DE MODELOS DE PREDICCIÓN DE VELOCIDAD A CORTO PLAZO MEDIANTE APRENDIZAJE AUTOMÁTICO

Los tiempos de viaje son un dato valioso para los usuarios del sistema de transporte, pues esta variable les permite discriminar entre los distintos modos y alternativas de su traslado. Para su estimación, la generación de buenos pronósticos de la velocidad futura resultan fundamentales. Recientemente, estos se han visto favorecidos por el crecimiento en la disponibilidad de datos que proveen los Sistemas de Transporte Inteligentes. Algunas aplicaciones consideran a los sistemas de despacho de vehículos o a servicios de navegación.

En esta tesis se propone mejorar la capacidad predictiva de modelos de estimación de velocidad. Para ello, se toma en cuenta al fenómeno de la congestión, el que se caracteriza por un alto flujo que obliga a un desplazamiento más lento de los vehículos. Se desarrolla una metodología para incorporar los estados de tráfico en la predicción de velocidad mediante modelos de aprendizaje automático. Se consideran las fases de congestión y de flujo libre, a partir de las que se emplean enfoques que varían en la forma en que dicha información se utiliza para la predicción. El estudio se enfoca en una autopista urbana que mide las variables de tráfico mediante cuatro sensores en un tramo de 9 km .

En el trabajo se expone un algoritmo de identificación de congestión basado en observaciones recientes de la velocidad y del flujo. Dado que este algoritmo se ejecuta en base a datos recientes, pero también del día completo, se desarrolla un modelo de predicción de congestión que busca emular su funcionamiento con aplicabilidad en tiempo real. El pronóstico del estado de tráfico alimenta a los métodos de estimación de velocidad, tras lo que se constata que la información proporcionada por este procedimiento provee un mejor desempeño predictivo. Se obtiene un error absoluto medio de 4.13 km/h .

El enfoque con mejor desempeño consiste en separar la muestra según el estado de tráfico, con lo que se pueden entrenar dos modelos de forma independiente. Además, los resultados obtienen un menor error en comparación al uso de una regla que identifica el nivel de congestión mediante un umbral de velocidad fijo, que se calcula al agregar toda la información de la muestra. Esto indica que una metodología específica a cada día y basada en observaciones recientes es de mayor utilidad para la predicción de velocidad en el corto plazo.

La principal dificultad es que las predicciones poseen un desfase en el período de transición entre estados de tráfico. Este es el lapso con mayor variabilidad de la velocidad, por lo que se plantean alternativas de trabajo futuro para subsanar este problema.

Agradecimientos

Agradezco a mis padres, a mi hermano, a mis abuelos y abuelas, a mis tíos y tías, a mi primos y a mi prima, que me amaron y formaron. A mis amigos y amigas, que me sostuvieron. A mis *roomies*, que me acompañaron. A mis compañeros y compañeras, a mis profesores y profesoras, a los funcionarios y funcionarias, que hicieron esto posible. A los grupos organizados, comisiones y centros de estudiantes, que me enseñaron fraternidad y colaboración.

Gracias especialmente a Sandra, Cristián, Sebastián, Diana, Carla, Camila, Pía y a todo el CEI 2017.

Agradezco la oportunidad, las dificultades, las alegrías y cada momento cotidiano en Beau-chef, que fue como una pequeña casa.

Tabla de Contenido

1. Introducción	1
1.1. Motivación	1
1.2. Objetivos	2
1.2.1. Objetivo general	2
1.2.2. Objetivos específicos	2
1.3. Alcances	2
1.4. Estructura de la tesis	3
2. Marco teórico	4
2.1. El sistema de transporte	4
2.1.1. Sistemas de Transporte Inteligente (ITS)	5
2.2. Teoría y modelamiento de tráfico	5
2.2.1. Teoría de flujo de tráfico	5
2.2.2. Diagramas fundamentales de tráfico	7
2.2.3. Congestión de tráfico	7
2.3. Predicción mediante aprendizaje automático	9
2.3.1. Predicción de variables de tráfico mediante aprendizaje automático	10
3. Caracterización del caso de estudio	13
3.1. Descripción de los datos	14
3.2. Selección de la muestra	15
4. Metodología	17
4.1. Identificación del estado de congestión de tráfico	18
4.2. Modelos de aprendizaje automático	20
4.2.1. Predicción del estado de congestión futuro	20
4.2.1.1. Naive-Bayes	21
4.2.1.2. Árbol de decisión (clasificación)	22
4.2.1.3. <i>Random forest</i> (clasificación)	24
4.2.1.4. Red neuronal MLP (clasificación)	25
4.2.2. Predicción de velocidad media futura	30
4.2.2.1. Árbol de decisión (regresión)	30
4.2.2.2. <i>Random forest</i> (regresión)	31
4.2.2.3. Red neuronal MLP (regresión)	31
4.3. Medidas de error	31
4.4. Entrenamiento y elección de hiperparámetros	33
4.4.1. Entrenamiento-validación-testeo	33

4.4.2. Sobreajuste	34
4.5. Tratamiento de clases desbalanceadas	35
5. Análisis y Resultados	37
5.1. Preprocesamiento de datos	37
5.1.1. Imputación de datos faltantes	37
5.1.2. Generación de atributos	37
5.2. Exploración de los datos	38
5.3. Identificación del estado de congestión	43
5.3.1. Ejemplo gráfico del etiquetado del estado de tráfico	46
5.4. Comparación y evaluación de modelos de predicción	47
5.4.1. Clasificación de congestión futura	47
5.4.1.1. Importancia de los atributos en la predicción de congestión .	49
5.4.1.2. Ejemplo gráfico para la predicción de congestión	50
5.4.2. Estimación de velocidad media futura	51
5.4.2.1. Importancia de los atributos en la predicción de velocidad .	54
5.4.2.2. Ejemplo gráfico para la predicción de velocidad	55
5.4.2.3. Regla de velocidad fija para identificar congestión	57
6. Conclusiones	58
6.1. Trabajo futuro	59
Bibliografía	60
Anexos	63
Anexo A. Preprocesamiento de los datos	63
A.1. Datos faltantes en la muestra	63
Anexo B. Especificaciones de los modelos de predicción de congestión	65
B.1. Configuración de los modelos y elección de hiperparámetros	65
B.2. Selección de hiperparámetros mediante validación cruzada	66
B.3. Resultados para distintos usos de información pasada mediante rezagos . . .	68
Anexo C. Importancias de los atributos para el modelo de <i>random forest</i>	69
Anexo D. Especificaciones de los modelos de predicción de velocidad	71
D.1. Configuración de los modelos y elección de hiperparámetros	71
D.2. Selección de hiperparámetros mediante validación cruzada	73
D.3. Resultados para distintos usos de información pasada mediante rezagos . . .	78

Capítulo 1

Introducción

1.1. Motivación

Las ciudades acogen una serie de actividades que son necesarias para satisfacer las necesidades de sus habitantes, lo que motiva el desplazamiento de estos hacia diferentes zonas mediante el uso de un sistema de transporte. En este contexto, uno de los aspectos relevantes en la toma de decisiones de los usuarios corresponde a la valoración que estos tienen por disminuir su tiempo de viaje. En el caso en que se opte por modos que involucren vehículos, esta variable guarda una estrecha relación con la velocidad a la que estos pueden desplazarse.

A su vez, las mejoras en la capacidad de capturar y analizar datos han favorecido el estudio de los conceptos anteriores. Los modelos de predicción, por ejemplo, han servido para generar pronósticos sobre el comportamiento futuro de las vías. Ya sea para una empresa que debe decidir la ruta que usarán sus camiones de despacho, para un operador de buses que desea estimar el tiempo de llegada a un paradero o para una compañía de bomberos que busca la manera más rápida de acudir a un siniestro; las herramientas de predicción pueden ser de ayuda para tomar decisiones con base en información actual e histórica.

No obstante lo anterior, una característica típica de los problemas de transporte es su carácter dinámico. Una de las tantas fuentes de incertidumbre corresponde a la congestión, que es un fenómeno que conlleva una disminución de la velocidad y del flujo de tráfico. Aún más, constituye un obstáculo en la elaboración de buenas predicciones dado su origen abrupto y la variabilidad que introduce.

La congestión no cuenta con una definición estandarizada que permita su identificación exacta cuando algunas variables sobrepasen ciertos umbrales. Sin embargo, la existencia de datos y el análisis de estos abre paso a explorar una caracterización que tome en cuenta diferentes factores del tráfico vehicular. Atendiendo lo anterior, esta tesis se ve motivada por dilucidar si una adecuada clasificación del estado de congestión es de ayuda en la elaboración de modelos de predicción de velocidad futura que sean más precisos en sus estimaciones. Cabe mencionar que, a pesar de que dicha inquietud es fundamentalmente metodológica, se ve guiada por la intención de proveer mejores herramientas de planificación a los actores que participan en el sistema de transporte.

1.2. Objetivos

1.2.1. Objetivo general

El objetivo general de la tesis es estudiar el efecto que genera la utilización del estado de congestión de tráfico en la calidad de modelos de predicción de velocidad futura a corto plazo, en el contexto de una autopista urbana que recoge datos mediante sensores fijos.

1.2.2. Objetivos específicos

- Elaborar un procedimiento para identificar el estado de congestión de tráfico para una muestra de datos de una autopista urbana.
- Identificar las principales características de las observaciones clasificadas como congestionadas en cuanto a variables de tráfico y variables temporales.
- Desarrollar y ajustar modelos de aprendizaje automático para predecir el estado de congestión y la velocidad futura.
- Determinar si el uso de información sobre el estado de congestión permite reducir el error de predicción para el caso de estudio.

1.3. Alcances

Dada la amplitud del estudio teórico del tráfico y de las aplicaciones de ciencia de datos, es conveniente explicitar las limitaciones de este trabajo para circunscribir el alcance en torno la generalización de los resultados obtenidos.

En primer lugar, este estudio se desarrolla utilizando datos de una autopista urbana en la que los vehículos pueden circular a alta velocidad y que se encuentra exenta de intersecciones. Si bien se estima que es factible extender la metodología a otras vías de similares características, en ningún caso se puede asegurar que los resultados sean homologables a otros contextos (por ejemplo, a calles en el centro de una ciudad o en sectores residenciales). Con todo, la metodología es susceptible de ser testeada en otras autopistas o calles sin intersecciones.

Por otro lado, más adelante se argumentará sobre que el período relevante para el caso de estudio corresponde a las tardes de los días de semana. Esto se atribuye a las condiciones particulares de las actividades que las personas realizan en la ciudad de Santiago, las que generan un mayor tráfico en el área de estudio durante el período seleccionado. Se debe tener en cuenta que la elaboración y el ajuste de los modelos descansa en las características propias del conjunto de datos a utilizar, el que será descrito detalladamente en la tesis.

Finalmente, este trabajo incorpora el uso de modelos de aprendizaje automático supervisados que han sido ampliamente utilizados en ciencia de datos. El análisis del efecto del estado de congestión se circunscribe a la evaluación mediante estas herramientas, por lo que

el uso de otras metodologías basadas en análisis estadístico, series de tiempo, aprendizaje no supervisado, entre otros; excede los objetivos planteados. Se desarrollan métodos para predecir la velocidad en la autopista para los próximos 5 minutos, por lo que los resultados son descriptivos del desempeño de los modelos en pronósticos a corto plazo.

1.4. Estructura de la tesis

En el Capítulo 2 se aborda el marco teórico acerca de los conceptos relevantes sobre teoría de flujo de tráfico y los modelos de predicción que serán utilizados en el desarrollo de la tesis. Además, se comentan los resultados de otros trabajos que estudian la predicción de variables de tráfico o la relevancia de la congestión, describiendo sus aportes para el desarrollo de este trabajo.

El Capítulo 3 describe el caso de estudio a analizar. En particular, se revisan las características de la vía y de la muestra de datos considerada.

En el Capítulo 4 se presenta la metodología que se utilizará para responder el problema de investigación. Esta incluye un algoritmo para identificar congestión de tráfico, los modelos de predicción de aprendizaje automático, las consideraciones para su ajuste y las métricas de error que se tendrán en cuenta para la evaluación.

El Capítulo 5 enseña los resultados de la tesis obtenidos tras la aplicación de la metodología y de las tareas de preprocesamiento. Se detallan los resultados de las etapas de identificación del estado de tráfico y su predicción, así como las estimación de la velocidad dentro de cinco minutos.

Por último, el Capítulo 6 comenta las conclusiones que se desprenden del trabajo realizado y plantea alternativas de trabajo futuro.

Capítulo 2

Marco teórico

2.1. El sistema de transporte

El sistema de transporte es un asunto central en cualquier ciudad del mundo. La existencia de diferentes espacios y servicios que estas ofrecen suelen ser un atractivo que permite que los habitantes satisfagan sus necesidades. No obstante, es evidente que los puntos de interés se ubican en diversas zonas de las ciudades y exigen que los traslados sean una actividad fundamental en la cotidianidad urbana.

La Figura 2.1 muestra el enfoque analítico desarrollado por Manheim [1]. **T** representa el sistema de transporte (infraestructura, vehículos y su operación), **A** consiste en el sistema de actividades económicas y sociales disponibles, mientras que **F** contempla los patrones de flujo (orígenes, destinos y volúmenes de personas y cargas que se trasladan por el entorno urbano). Tanto el sistema de transporte como de actividades impactan en los flujos que se distribuyen por la red, los que retroalimentan constantemente las características de los primeros.

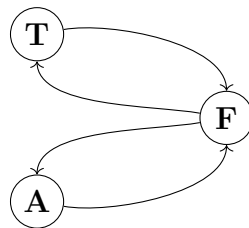


Figura 2.1: Esquema del enfoque Manheim. Elaboración propia a partir de Manheim [1].

Vale la pena considerar el aporte conceptual del enfoque Manheim, pues permite comprender que los fenómenos de tráfico no se ven sujetos solamente al sistema de transporte, sino también a las características socioeconómicas y los patrones de viaje propios de espacio urbano.

2.1.1. Sistemas de Transporte Inteligente (ITS)

El análisis de sistemas de transporte es de ayuda en la elaboración de intervenciones a distintos niveles que pueden mejorar la experiencia de los usuarios y reducir las externalidades negativas que las ciudades deben enfrentar. Bajo este contexto, surgen los llamados Sistemas de Transporte Inteligente (ITS, por sus siglas en inglés) que afrontan el fenómeno incorporando herramientas tecnológicas recientes.

El Consejo de la Unión Europea [2] define a los SIT de la siguiente manera:

Los SIT son aplicaciones avanzadas (...) que tienen el objetivo de proveer servicios innovadores relacionados a los diferentes modos de transporte y gestión del tráfico y permiten a los usuarios estar mejor informados y hacer un uso más seguro, coordinado e “inteligente” de las redes de transporte. Los SIT integran telecomunicaciones, electrónica y tecnologías de la información con ingeniería de transporte para planificar, diseñar, operar, mantener y gestionar de sistemas de transporte.

La implementación de estas nuevas herramientas ha derivado en numerosos proyectos que se basan en el uso de datos, siendo los sensores y el rastreo GPS algunos de los mecanismos que permiten su captura. En consecuencia, la intención de aprovechar las ventajas de los SIT ha traído consigo una mayor disponibilidad de datos en tiempo real que dan cuenta del estado del sistema de transporte. En particular, se abre paso a utilizar herramientas de predicción que permitan a los planificadores tomar decisiones basadas en fuentes de información más confiable.

2.2. Teoría y modelamiento de tráfico

2.2.1. Teoría de flujo de tráfico

En esta sección se toma como referencia el modelo planteado por Greenshields et al. [3], quienes relacionan las variables fundamentales del tráfico. Este modelo se asenta en un supuesto de tráfico ininterrumpido, por lo que no repara en la presencia de intersecciones, paradas o semáforos. Si bien estos son elementos importantes a tener en cuenta, su exclusión del análisis teórico permite comprender las bases del comportamiento de los vehículos al transitar por una vía.

Diversos autores, como Kerner [4], concuerdan en que el tráfico puede estudiarse a nivel macroscópico o microscópico. En esta ocasión, se usa el primer enfoque que corresponde a observar el estado agregado del tráfico mediante promedios u otras métricas, pero no la situación particular de cada vehículo.

Tres variables de gran relevancia para el modelamiento de este fenómeno corresponden al flujo f (cantidad de vehículos que pasan por un segmento de vía por unidad de tiempo), la densidad k (número de vehículos presentes en una vía por unidad de distancia) y la velocidad v . Esta última puede calcularse de dos formas en un enfoque macroscópico:

- La velocidad media espacial v_s corresponde al promedio de las velocidades de los vehículos observados en un segmento de vía.
- La velocidad media temporal v_t se calcula como el promedio de las velocidades de los vehículos que pasan por un punto fijo durante cierto período.

La siguiente ecuación, usualmente denominada como la ecuación fundamental del tráfico y formulada por Greenshields et al. [3], relaciona las variables anteriores con la siguiente expresión:

$$f = v_s k. \quad (2.1)$$

Si bien se debe tener en cuenta que esta relación se cumple en condiciones estacionarias y para mediciones agregadas, Daganzo [5] reafirma su utilidad en el análisis de las dinámicas en el tráfico. Antes de revisar las implicancias de la ecuación 2.1, es conveniente definir dos parámetros relevantes. La velocidad de flujo libre v_l se define como aquella velocidad que es posible alcanzar cuando la vía está despejada y los conductores no se ven obstaculizados, es decir, el flujo es teóricamente nulo. En tanto, la densidad máxima k_j , o *jam density*, consiste en la mayor densidad que puede albergar un segmento de vía que ya no es capaz de admitir más vehículos.

Con lo anterior, Daganzo [5] indica que la ecuación planteada por Greenshields et al. [3] es conveniente para explicar las relaciones fundamentales entre las variables mediante la siguiente función:

$$v(k) = v_l - \frac{v_l}{k_j} k. \quad (2.2)$$

De la ecuación 2.2, es fácil ver que si la densidad toma su valor máximo k_j , entonces la velocidad es cero. Además, se puede concluir que un aumento de la densidad implica una disminución en la velocidad de los vehículos, lo que es consistente con la expresión 2.1. Al reemplazar en esta última, se obtiene que

$$f(k) = vk = v_l k - \frac{v_l}{k_j} k^2. \quad (2.3)$$

También es posible plantear el flujo en función de la velocidad, como indica la ecuación a continuación:

$$f(v) = k_j v - \frac{k_j}{v_l} v^2. \quad (2.4)$$

2.2.2. Diagramas fundamentales de tráfico

El desarrollo de la sección anterior puede representarse gráficamente a través de los diagramas fundamentales del tráfico recopilados por Daganzo [5] a partir del modelo de Greenshields et al. [3]. En la Figura 2.2 se muestran las relaciones densidad-flujo y flujo-velocidad, obtenidas a partir de las ecuaciones revisadas previamente.

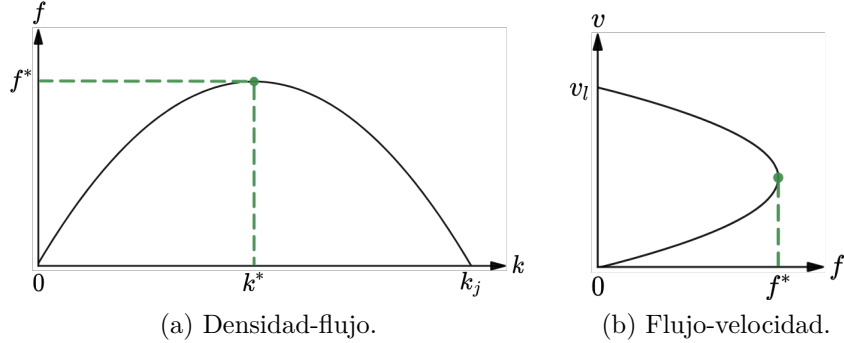


Figura 2.2: Diagramas fundamentales de tráfico. Elaboración propia a partir del modelo de tráfico desarrollado por Greenshields et al. [3].

La curva densidad-flujo en la Figura 2.2.a indica que el flujo es nulo en dos momentos: cuando la densidad es cero y cuando es máxima. El primer caso se obtiene directamente de la ecuación 2.1, mientras el segundo ocurre si el segmento de vía se ve colapsado a un punto tal que se impide el movimiento de los vehículos. Como todos están detenidos, el flujo es cero. Según lo planteado en la ecuación 2.3, el flujo máximo f^* es alcanzado teóricamente cuando la densidad es $k_j/2$.

Por otro lado, el diagrama flujo-velocidad de la Figura 2.2.b muestra que a valores pequeños del flujo se observa una alta velocidad, la que disminuye cuando el flujo se acerca a su valor máximo. A partir de este punto, la sección inferior del diagrama se explica en un contexto de congestión. La alta cantidad de vehículos circulando conlleva a una reducción de la velocidad que merma el flujo total debido a la lentitud del avance en la vía. Si lo anterior persiste, es posible llegar a un escenario de congestión máxima en el que tanto el flujo como la velocidad son cero debido a que ningún conductor se puede mover de sus sitio.

Al igual que las ecuaciones que relacionan las variables estudiadas, los diagramas analizados en esta sección asumen ciertas condiciones ideales como las expuestas en la sección previa. De esta manera, no es de extrañar que datos empíricos generen gráficos distintos según las características del caso estudiado. Sin embargo, el modelo utilizado sigue demostrando su utilidad al establecer un marco general para describir la interacción entre vehículos en una vía, pues incorpora variables claves que logran caracterizar la situación agregada.

2.2.3. Congestión de tráfico

El *Highway Capacity Manual* [6] ha definido el nivel de servicio (o *LOS*, por sus siglas en inglés) en vías de tráfico como una medición cualitativa acerca de la calidad ofrecida a los

usuarios. Por ejemplo, se considera la velocidad a la que se pueden trasladar los vehículos y el tiempo de viaje. La clasificación de estados consiste en un *ranking* desde la letra A hasta la F, donde A representa el mayor nivel de servicio y F el menor.

El nivel A representa el caso en que las maniobras de los conductores no se ven impedidas por otros vehículos y suele denominarse como flujo libre. El manual afirma que, bajo estas condiciones, se observa una velocidad cercana al 90% de la velocidad de flujo libre v_l . En cambio, el nivel F sucede cuando la demanda es mayor a la capacidad de la vía.

Kerner [4] plantea una teoría de tres fases para los estados de tráfico, la que desarrolla a partir de estudios empíricos en autopistas de Alemania basados en datos de sensores [7] [8]. Indica que un aumento de la densidad puede llevar a disminuciones abruptas de la velocidad, es decir, un estado de congestión que define como:

Un estado de tráfico en el que la velocidad media es menor que la velocidad media mínima alcanzada en condiciones de flujo libre.

El paso de un estado de flujo libre a uno de congestión ha sido denominado *traffic breakdown*. Su análisis empírico ha conducido a la conclusión de que su ocurrencia es de naturaleza estocástica, es decir, un mismo valor del flujo de tráfico podría conducir o no a un estado de congestión. Persaud et al. [9] muestra que la probabilidad de *breakdown* es creciente respecto al flujo y es de naturaleza asintótica a medida que se llega a la capacidad máxima de la vía, tal como enseña la Figura 2.3.

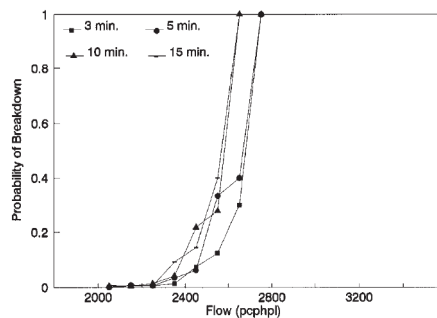


Figura 2.3: Probabilidad de *breakdown* en función del flujo de tráfico medido en vehículos por hora por pista (*pcphpl*) [9].

Esta naturaleza probabilística de la congestión es una de las razones por las que no existe una definición universal que permita identificarla inequívocamente. Por el contrario, parámetros como la densidad máxima, el flujo máximo o la velocidad de flujo libre son dependientes del caso de estudio. Es claro, por ejemplo, que este fenómeno requiere de distintas condiciones para aparecer en una autopista de alta velocidad, en una pista exclusiva de buses o en un camino no asfaltado.

Para este trabajo se distinguirá entre la situación de flujo libre y un período de congestión, caracterizado por disminuciones de la velocidad a causa de un mayor flujo o densidad de vehículos.

2.3. Predicción mediante aprendizaje automático

Tal y como se planteó en el apartado 2.1.1 sobre Sistemas de Transporte Inteligente, existe una creciente disponibilidad de datos en este campo. Esto es un incentivo para el uso de herramientas de predicción, las que pueden ser de ayuda en la gestión y evaluación de las operaciones de transporte.

El aprendizaje automático, o *Machine Learning* en inglés, es un área de la inteligencia artificial que tiene como objetivo desarrollar técnicas que permitan a los computadores “aprender” a partir de la experiencia. Mitchell [10] ofrece la siguiente definición:

Dada una experiencia E , una tarea T y una métrica de desempeño P , se dice que un programa computacional *aprende* si su desempeño realizando T , medido por P , mejora al usar la experiencia E .

Estas herramientas pueden desarrollar una serie de tareas dependiendo de la estructura de los datos o del objetivo de investigación. No obstante, para este caso será relevante considerar los modelos de predicción, cuyo propósito es pronosticar valores futuros a partir de información actual y pasada. Una predicción puede ser de los siguientes dos tipos:

- Clasificación: corresponde a los modelos de aprendizaje automático que determinan la pertenencia de las observaciones a clases significativamente distintas o contrapuestas. Por ejemplo, predecir si hay o no congestión.
- Regresión: el propósito es predecir el valor de una variable continua que caracteriza a las observaciones de algún modo. Por ejemplo, estimar la velocidad futura.

Las dos tareas descritas corresponden a modelos de aprendizaje supervisado, los que se caracterizan por utilizar conjuntos de datos que incorporan la variable o atributo a predecir. El ajuste tiene como objetivo minimizar el error con respecto a un dato observado que es conocido de antemano. Otros enfoques corresponden al aprendizaje no supervisado, semi-supervisado y por refuerzo.

El interés científico por estudiar los SIT mediante estas técnicas ha sido demostrado por numerosos estudios desarrollados en el mundo. Boukerche y Wang [11] realizan una recopilación exhaustiva de modelos de aprendizaje automático aplicados a variables de flujo de tráfico. Por otro lado, Ermagun y Levinson [12] dan cuenta del crecimiento en el uso de estas herramientas que, a pesar de su reciente implementación en comparación a otros métodos, han ganado popularidad en los últimos años.

Dentro de los modelos de aprendizaje automático ha surgido un área llamada aprendizaje profundo (en inglés, *Deep Learning*). Esta se fundamenta en proponer métodos basados en redes neuronales, los que han sido complejizados con el tiempo gracias a la mayor capacidad de procesamiento computacional. De esta forma, es posible explorar más relaciones no lineales al dotar a las redes neuronales de arquitecturas más complejas en cuanto a la cantidad de capas o a las relaciones entre las unidades o celdas.

2.3.1. Predicción de variables de tráfico mediante aprendizaje automático

En el sentido de las aplicaciones a los ITS, el aprendizaje automático habilita una mejor comprensión de las características del sistema de transporte, por ejemplo mediante la anticipación de escenarios o fenómenos que son detectados por los modelos con cierta precisión. En este sentido, Highaghat et al. [13] explican que ha existido un creciente interés durante la última década por utilizar redes neuronales y aprendizaje profundo en esta área, siendo algunas aplicaciones la predicción de incidentes o el reconocimiento visual a partir de imágenes de video. No obstante, destaca también la amplia literatura en torno a lo que denomina características (o variables) de tráfico, que son aquellas magnitudes medibles que determinan el estado del sistema y consisten, principalmente, en el flujo, la velocidad o la densidad. Por otro lado, Boukerche [11] también realza la popularidad de utilizar modelos de redes neuronales para esta tarea, así como otros enfoques de aprendizaje automático que incluyen máquinas vectores de soporte, K-vecinos más cercanos o modelos híbridos.

El uso de métodos basados en árboles también ha sido considerado, por ejemplo por Prasad et al. [14] que utilizan árboles de clasificación para identificar horas punta, en el sentido de observar picos en el flujo. Utiliza como entrada imágenes aéreas que capturan información georreferenciada sobre los flujos de tráfico mediante una escala cromática, demostrando la utilidad de este tipo de modelos para identificar patrones espaciales de tráfico. Hamner [15] propone el uso de un modelo de *random forest* para la predicción de la velocidad media futura a partir de datos GPS, destacando que su implementación y ajuste son relativamente sencillos, sin perjuicio de que podrían ser mejorados mediante la inclusión de información sobre las variaciones a los que se ve sujeto el flujo de los vehículos.

El uso de *random forest* también es impulsado por Basso et al. [16] con el objetivo de determinar la importancia relativa de cada atributo en un problema de predicción de accidentes. Si bien este problema no pronostica el flujo o la velocidad, sí utiliza estas variables de tráfico como entrada al modelo. Los datos se obtienen mediante sensores ubicados en la Autopista Central de Santiago de Chile, lo que también corresponde al caso de estudio de este trabajo. Los resultados están a la par del estado del arte, dando cuenta de la factibilidad de aplicar modelos de predicción con éxito en este tipo de contextos, así como de emplear métodos basados en árboles para *rankear* la contribución de cada atributo.

Rzeszótka y Hoa [17] proponen la construcción una red neuronal con una capa oculta para predecir la velocidad media de segmentos de vía, demostrando su competitividad ante otros enfoques. Su principal comentario metodológico refiere al cuidado en probar y contraponer diferentes arquitecturas, así como en la selección de atributos a considerar. Esto ya que argumentan que la inclusión o remoción de alguna variable puede influir en los resultados del error de manera sistemática.

Karlaftis y Vlahogianni [18] realizan una recopilación de literatura para comparar métodos estadísticos (como regresiones, modelos *logit*, autorregresivos, etc.) y modelos de redes neuronales. Se destaca la ventaja de estos últimos para la captura de relaciones más complejas en los datos, siendo además flexibles debido a las variaciones que pueden incorporar en su arquitectura. No obstante, una desventaja corresponde a que carecen de las ventajas de los modelos estadísticos en cuanto a su capacidad explicativa, pues su foco está en la precisión de

los pronósticos. Además, se debe tener especial cuidado en evitar su sobreajuste al escoger los hiperparámetros, ya que existe una amplia variedad de combinaciones de estos que pueden ser considerados al construir un modelo generalizable a datos no observados.

La capacidad predictiva de las redes neuronales también es comprobada por Vlahogianni [19] al comparar con un modelo autorregresivo integrado de media móvil (ARIMA, por sus siglas en inglés). Se identifican diferentes regímenes de flujo mediante *clusters*, estableciendo una red neuronal de una capa oculta para cada uno de estos grupos y que se encarga de predecir el volumen de tráfico futuro. Se muestra la importancia en distinguir patrones en el flujo de tráfico, pero no se aísla el efecto que es atribuible al uso de estos patrones porque no se explicita un contrafactual, así como tampoco es comentado el impacto sobre el atributo de velocidad.

En Cai et. al. [20] se aplica el método de k-vecinos más cercanos para predecir velocidades en una red de calles, tras lo que se indica que vale la pena ajustar hiperparámetros específicos del modelo ante cada estado de tráfico. Sin embargo, no se comenta cuánto del error se atribuye a esta diferenciación, pues no se identifica el efecto asociado a la clasificación en distintos estados.

Mihaita et al. [21] desarrollan modelos de aprendizaje profundo que incluyen redes neuronales *feed forward*, convolucionales, recurrentes (*Long Short-Term Memory* o LSTM) e híbridas que combinan los últimos dos casos. Como modelos base, se comprueba el uso de la media histórica y ARIMA. El trabajo busca predecir el flujo de tráfico en puntos fijos de una autopista a partir de datos recogidos con sensores fijos. Los modelos de aprendizaje profundo presentan un desempeño bastante similar entre sí en el corto plazo (predicciones de hasta dentro de 10 minutos) y en cualquier caso vencen a los modelos base. En predicciones de largo plazo, se nota una ventaja de la LSTM incluso ante el enfoque híbrido que la combina con el modelo convolucional.

En el trabajo de Polson y Sokolov [22] se predice velocidad futura ante la presencia de fenómenos no lineales, como la congestión, días con nevazones o eventos urbanos importantes como un partido de fútbol. Se implementa una red neuronal *feed forward* que muestra competitividad por identificar más acertadamente este tipo de fenómenos en comparación a modelos lineales. Además, se indica que las redes con una única capa oculta pierden poder predictivo y se sugiere incluir atributos que entreguen información sobre los fenómenos que inducen no-linealidad, por ejemplo un pronóstico del tiempo. El cambio de fase producto de la congestión es reconocido como un asunto a tratar, pero no es evaluado en su particularidad.

El presente documento está fuertemente inspirada en la Tesis de Magíster de Bastián Véjar [23], donde se trabaja con la misma autopista que será analizada en los capítulos posteriores. El objetivo es predecir velocidad futura, para lo que se testean los modelos de *random forest*, máquina de vectores de soporte y redes neuronal *feed forward* y recurrente (implementada mediante una LSTM). Se introduce una metodología para entrenar modelos por separado dependiendo del nivel de congestión de cada observación, para lo que se distingue según si la velocidad está por sobre o bajo cierto límite. Este es calculado como la velocidad promedio de los datos que poseen un flujo en el percentil 99.5, que equivale a alrededor de 80 km/h .

Véjar concluye que las redes neuronales proveen un mejor desempeño. La red *feed forward* es preferible a la LSTM ante predicciones de corto plazo (5 o 10 minutos), lo que se revierte

para el largo plazo (15 o 30 minutos). Usar la congestión como variable de entrada mejora los resultados, aunque queda pendiente diseñar un método de clasificación de congestión más complejo que se adapte a los datos disponibles. También se discute cuál es la mejor manera mediante la que debe utilizarse dicho dato en los diferentes modelos de predicción, dando preferencia a la separación de la muestra según el estado de congestión de la vía.

Capítulo 3

Caracterización del caso de estudio

El conjunto de datos a utilizar fue obtenido a una tabla en formato *csv* disponibilizada por la Autopista Central, la cual entrega información sobre las variables de tráfico medidas mediante sensores para el caso de estudio. Este consiste en un tramo de la Autopista Central en su eje General Velásquez, el que se emplaza en la ciudad de Santiago de Chile y atraviesa varias comunas de esta en dirección norte-sur.

La Figura 3.1 muestra un diagrama del segmento de la autopista a considerar, la que tiene una longitud de alrededor de 9 km . En ella también se indica la ubicación de los cuatro pódicos en los que se realizan las mediciones de las variables de tráfico, denominados PA26, PA24, PA22 y PA20 desde el norte hacia el sur. Se puede observar que existen entradas y salidas a la autopista entre estos pódicos, por lo que pueden existir variaciones importantes en la cantidad o las características de los vehículos que atraviesan cada punto de medición.

Una característica relevante del tramo considerado tiene que ver con la estructura de actividades en torno a la ciudad de Santiago. Esta concentra una gran cantidad de servicios que inducen la generación de viajes por diferentes motivos (trabajo, estudio, trámites, ocio, etc.), atrayendo a población de zonas aledañas. En consecuencia, existen momentos con un mayor flujo de tráfico y que son coincidentes con los horarios punta de la mañana y de la tarde. Dado que el tramo estudiado inicia desde el centro de Santiago y va en dirección sur, como se indica en la Figura 3.2, se prevé que es más probable observar un estado de congestión durante la tarde por motivo del retorno de los viajeros a sus hogares.

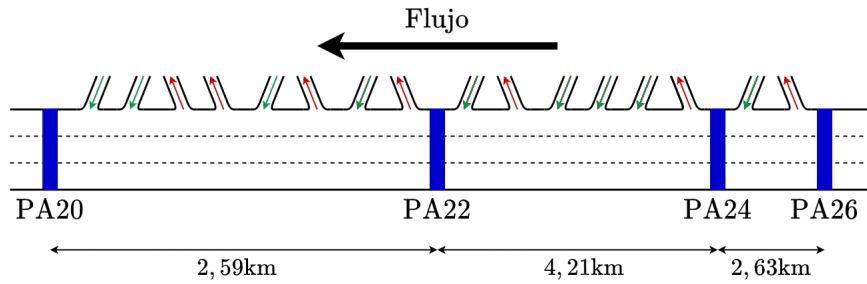


Figura 3.1: Diagrama del tramo de la autopista estudiado y ubicación de los puntos de medición.

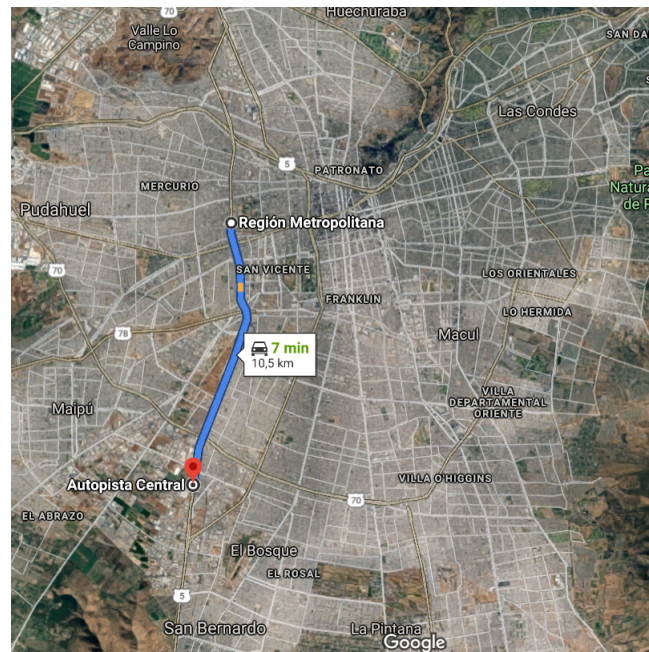


Figura 3.2: Ubicación del tramo estudiado sobre la ciudad de Santiago

3.1. Descripción de los datos

Cada uno de los pórtilos realiza mediciones mediante sensores de manera ininterrumpida durante el día. Para este caso se cuenta con un conjunto de datos cuya agregación temporal es de un minuto, reportándose cantidades como la velocidad promedio o la cantidad total de vehículos que fueron observados durante ese lapso. La muestra contempla desde el 1 de octubre de 2016 hasta el 27 de marzo de 2018, equivalente a 542 días y determinada por 781.063 entradas o filas.

El conjunto de datos dispone de la fecha en formato día-mes-año y la hora en formato hora-minuto-segundo, aunque la agregación temporal minutil implica que las mediciones de segundos sean irrelevantes. También existen variables *dummy* que indican si cada día es hábil

o fin de semana, así como si es o no un feriado.

Se distinguen tres clases de vehículos en el conjunto de datos (livianos, pesados y motocicletas) y para cada uno de ellos se miden los siguientes atributos en los cuatro pódicos:

- Flujo de vehículos livianos [veh/min].
- Velocidad media temporal de vehículos livianos [km/h].
- Velocidad media espacial de vehículos livianos [km/h].
- Desviación estándar de la velocidad media temporal de vehículos livianos [km/h].

Como resultado, se cuenta con 52 atributos. Los datos serán limpiados y enriquecidos con nuevos atributos durante el procesamiento, el cual se detalla en la Sección 5.1.

3.2. Selección de la muestra

Se ha indicado que uno de los principales eventos de congestión se asocia a la punta tarde, pues hay un flujo importante de usuarios a raíz de que dejan la ciudad para dirigirse a zonas aledañas. Este trabajo se enfocará en este fenómeno, por lo que se buscará aumentar su presencia relativa en los datos.

Se propone utilizar el subconjunto de datos correspondientes al período entre las 14 y las 23 horas, pues se espera que los eventos de congestión se concentren en esta franja horaria. Esto podría reducir un futuro desbalance entre observaciones congestionadas y descongestionadas, pues se deja fuera a la madrugada y a la mañana. En estos dos escenarios es esperable que una amplia proporción de los datos ocurran bajo un régimen de flujo libre, lo que penalizaría la presencia relativa de la clase congestionada. Por las mismas razones, se considerarán solo los datos correspondientes a días de semana, pues se observa un menor flujo los días sábado y domingo que mantiene estable el valor de la velocidad.

La Tabla 3.1 muestra los promedios y desviaciones estándar de la velocidad y el flujo para varios períodos del día. En esta se observa que el lapso de mayor flujo ocurre entre las 17 y las 20 horas, en el que este atributo también posee una menor velocidad promedio y mayor variabilidad. Todas estas son características de la congestión, ya sea por una disminución de la velocidad a causa del alto flujo o bien por los cambios bruscos que ocurren durante la transición con el flujo libre. Por otro lado, la madrugada (entre las 23 y las 6 horas) posee un bajo flujo que es más de 7 veces menor al observado en la punta tarde. Si bien posee la segunda mayor desviación estándar de la velocidad, se cree que esto ocurre debido a conductores que transitan a mayor rapidez durante la noche.

La extensión del análisis a la etapa entre las 14 y las 23 horas se justifica por la inclusión de períodos de referencia que contrasten con las fases de congestión. En la Tabla 3.1 se muestra que la desviación estándar del flujo y de la velocidad en este espacio de tiempo es mayor a la observada durante la mañana.

Tabla 3.1: Métricas de la velocidad y el flujo para distintos períodos del día.
 La velocidad se encuentra medida en km/h y el flujo en veh/min .

Período	Velocidad media	Desv. est. velocidad	Flujo medio	Desv. est. flujo
06:00 - 09:59	96.65	7.45	19.48	12.52
10:00 - 13:59	94.59	7.88	21.26	7.1
14:00 - 16:59	93.6	9.03	23	8.08
17:00 - 19:59	88.63	18.61	29.76	11.21
20:00 - 22:59	94.42	9.77	17.81	7.41
23:00 - 05:59	95.78	11.34	4.32	4

Las situaciones descritas anteriormente se evidencian también en la Figura 3.3. Por un lado, se indica que el aumento en el flujo conduce a una disminución abrupta en la velocidad se observa sólo en las tardes, en torno a las 19 horas. Además, se aprecia un flujo notoriamente menor durante los fines de semana que no provoca disminuciones en la velocidad.

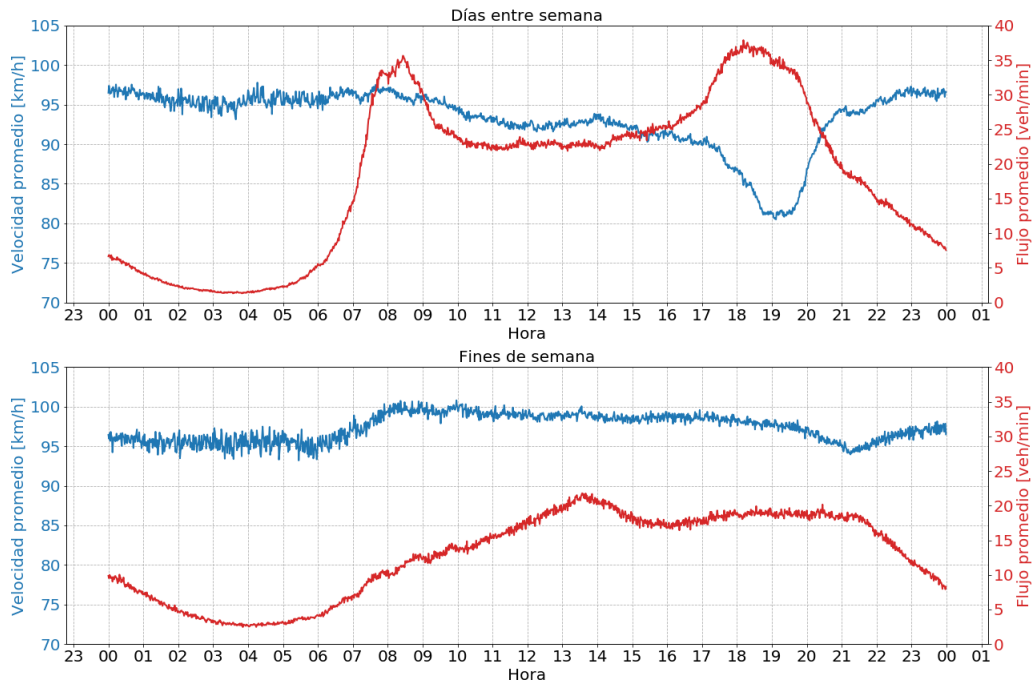


Figura 3.3: Velocidad y flujo promedio por hora para días de semana y fines de semana en el pórtico PA20.

Finalmente, también se excluyen algunos días que contienen errores de medición (identificados, por ejemplo, por la presencia de valores faltantes por un lapso prolongado). El conjunto de datos resultantes tras estas consideraciones contiene 207.900 observaciones o filas, equivalentes a 385 días.

Capítulo 4

Metodología

Dado que se busca dilucidar si la predicción de velocidad pueden mejorarse a partir del uso del estado de congestión de tráfico, se debe establecer una metodología que permita construir los atributos y modelos necesarios para comparar entre diferentes enfoques. Como se mencionó en la Sección 2.2.3, no existe una definición pre-establecida para identificar congestión, sino que esta depende de características propias del caso de estudio o del fenómeno que se quiera analizar. Por ejemplo, las consideraciones pueden variar si se contemplan las condiciones climáticas o la ocurrencia de eventos especiales, tal como indican por Polson y Sokolov [22].

En esta oportunidad, se utilizará la información que es provista por las variables físicas de tráfico como la velocidad o el flujo, además de otros atributos a detallar en las secciones siguientes. El primer paso se discute en la Sección 4.1 y consiste en construir un método para etiquetar los instantes en los que se aprecia un régimen de congestión. Su aplicación a cada una de las observaciones se alimentará de todos los datos disponibles para el día correspondiente, por lo que el método no será aplicable en tiempo real dado el uso de información futura.

En la Sección 4.2 se describen los métodos de aprendizaje automático a utilizar. En principio, el inconveniente de aplicar una clasificación del estado de tráfico en tiempo real se resolverá en la Sección 4.2.1. En esta se propone la construcción de modelos de aprendizaje automático que utilicen las etiquetas generadas en el paso previo como un insumo para su ajuste. Su propósito es contar con un modelo que sí sea capaz de identificar congestión en tiempo real a partir de un subconjunto de datos de entrenamiento, lo que será justificado en detalle.

La última etapa consiste en la elaboración de modelos de predicción de velocidad que puedan incluir o no al atributo de congestión que se obtiene a partir de los métodos descritos previamente. Para ello pueden existir diferentes enfoques respecto a la manera en que se utiliza dicho dato. Esta etapa será descrita en la Sección 4.2.2.

Finalmente, los detalles metodológicos acerca del ajuste de los modelos de aprendizaje automático y de las métricas de error serán enunciadas en las Secciones 4.3 y 4.4.

4.1. Identificación del estado de congestión de tráfico

Con la intención de identificar de la forma más clara posible el efecto de la congestión de tráfico en la predicción de velocidad, se considerarán dos estados de tráfico: congestionado y des congestionado (flujo libre). El primero consiste en el caso en que un aumento en el flujo de vehículos copa la capacidad de la vía, forzando una reducción de velocidad de los vehículos. A esto se contraponen un escenario de flujo libre, en el que los conductores pueden decidir la velocidad de viaje con menos restricciones dadas por el resto de vehículos.

Se busca generar un atributo que indique la presencia o ausencia de congestión con los valores 1 o 0, respectivamente. El enfoque a adoptar consiste en exigir a cada observación el cumplimiento de una serie de requisitos, en cuyo caso serán clasificadas como “congestionadas”. El resto de observaciones serán, por defecto “des congestionadas”.

Como se vio en la Sección 3.1, la información sobre la velocidad promedio se encuentra desagregada en vehículos livianos, pesados y motocicletas. Estos son cualitativamente diferentes, pues existen condiciones que hacen dispares sus comportamientos en la vía. Por ejemplo, en ocasiones las motocicletas se mueven por fuera o entre las pistas de la vía, mientras que los vehículos pesados poseen restricciones propias para la velocidad máxima. En esta ocasión, se opta por identificar los estados de tráfico para los vehículos livianos. Más adelante se revisarán cuáles son las características descriptivas de este subconjunto (ver Sección 5.2), pero se debe tener en cuenta que estos representan la mayor parte de la muestra.

El algoritmo a desarrollar se construye de forma tal que pueda ser aplicado por separado a cada p rtico y a cada d a en la muestra, requiriendo de las mediciones de velocidad promedio y del flujo en cada minuto. Se pretende que su dependencia de los valores observados para fijar una regla de clasificaci n lo haga m s generalizable a diferentes contextos. Se utilizan los siguientes par metros que deben ser ajustados para el correcto funcionamiento del procedimiento propuesto:

- v^* : velocidad m xima para un per odo de congesti n. Se usar  $v^* = 80 \text{ km/h}$ y se asume que por sobre este valor siempre hay flujo libre.
- \mathcal{F} : cota superior del flujo de veh culos para un d a sin tr fico congestionado. Si el flujo m ximo observado en alg n d a es menor a \mathcal{F} , se asume que no hay eventos de congesti n. Se usar  $\mathcal{F} = 40 \text{ veh/min}$.
- τ_i^{ref} : per odo de referencia de flujo libre del d a i , el que se define como un lapso de dicho d a en el que no se observa congesti n. Se usa por defecto el lapso entre las 14:00 y las 15:00, aunque este se ajusta en caso de que alg n d a presente congesti n entre dichas horas. Por ejemplo, para el 24 de agosto de 2017 se fija $\tau_i^{\text{ref}} = [20:30, 21:30]$.
- $v_{i,\mathcal{P}}^{\text{P10}}$: percentil 10 de las velocidades de veh culos livianos observadas durante el per odo de referencia del d a i . Se emplea para conocer cu les son los valores que podr an servir de cota superior para las velocidades que se alcanzar an bajo un estado de congesti n.

El m todo se basa en establecer un valor l mite de la velocidad v^{lim} que permita discriminar a los per odos con congesti n, suponiendo que esta variable disminuye ante este fen meno. El l mite es calculado para cada d a por separado y depende de los valores observados en el

período de referencia τ^{ref} . Se considera la velocidad del percentil 10 del intervalo, sin perjuicio de que v^{lim} valga a lo más v^* para que las velocidades altas no queden rotuladas como congestionadas. Si bien podría seleccionarse el mínimo del período de referencia, se usa un percentil para evitar que observaciones puntuales afecten el funcionamiento del mecanismo.

Se exige que un día deba contar con al menos una observación cuyo flujo sea mayor a \mathcal{F} para analizar si cuenta con períodos congestionados, mientras que se asume que en caso contrario hay flujo libre. Con esta consideración se identifican los días que representan una menor actividad vehicular, tales como los feriados. En caso de que efectivamente se supere un flujo \mathcal{F} , el hallazgo de una observación congestionada supone que la velocidad de los vehículos livianos se mantiene bajo v^{lim} por al menos 5 minutos. La intención es eludir *outliers* que se deban al paso fortuito de vehículos a baja velocidad por razones ajenas a las condiciones de tráfico.

El Algoritmo 1 resume lo indicado previamente e indica los datos que deben ser considerados para su aplicación.

Algoritmo 1: Identificación de estados de tráfico para el etiquetado de los datos

Resultado: Nuevo atributo que etiqueta a cada observación como “congestionada” o “descongestionada” mediante los valores 1 y 0, respectivamente.

Sean D la lista de días en el conjunto de datos
 T la lista de minutos de la muestra $\{14:00, \dots, 23:00\}$
 $v_{i,t,\mathcal{P}}$ la velocidad promedio de los vehículos livianos en el pórtico \mathcal{P} durante el minuto t del día i ($[km/h]$)
 $f_{i,t,\mathcal{P}}$ el flujo de vehículos de todo tipo en el pórtico \mathcal{P} durante el minuto t del día i ($[veh/min]$)

para $\mathcal{P} \in \{PA20, PA22, PA24, PA26\}$ **hacer**

para $i \in D$ **hacer**

 Calcular el flujo máximo de cada día $f_{i,\mathcal{P}}^{\text{max}} = \max\{f_{i,t,\mathcal{P}} : t \in T\}$;

si $f_{i,\mathcal{P}}^{\text{max}} < \mathcal{F}$ **entonces**

 Etiquetar todas las observaciones del día i como “descongestionadas” con el valor 0;

en otro caso

 Calcular $v_{i,\mathcal{P}}^{\text{P10}}$ y definir $v_{i,\mathcal{P}}^{\text{lim}} = \min\{v^*, v_{i,\mathcal{P}}^{\text{P10}}\}$;

para $t \in T$ **hacer**

si $v_{i,t+k,\mathcal{P}} < v_{i,\mathcal{P}}^{\text{lim}} \ \forall k = 0, \dots, 4$ **entonces**

 Etiquetar las observaciones de los minutos $t, \dots, t + 4$ como “congestionadas” con el valor 1;

en otro caso

 La observación del minuto t se etiqueta como “descongestionada” con el valor 0;

fin

fin

fin

fin

fin

Una de las características principales del método de identificación de congestión es que padece de *target leakage* (también llamado *data leakage* o simplemente *leakage*), el que se define como el uso de información futura para la determinación de un estado actual. En este caso en particular, se requiere conocer la velocidad de hasta los cuatro minutos siguientes para clasificar a cada observación. Este fenómeno provoca severos perjuicios en los modelos predictivos, pues puede inflar artificialmente su desempeño sin que estos puedan ser aplicados en tiempo real. Esta situación ha sido estudiada ampliamente, por ejemplo por Kaufman et al. [24] o por Nisbet et al. [25] que la ha calificado como uno de los 10 errores más comunes en minería de datos.

En definitiva, no es apropiado predecir velocidad usando el atributo generado con el método de etiquetado que se detalla en esta sección. Una alternativa es revisar las velocidades en el intervalo $\{t - 4, \dots, t\}$ en lugar de $\{t, \dots, t + 4\}$, lo que provocaría un retraso de al menos cinco minutos en la identificación de un período congestionado. Se propone, entonces, construir un modelo de aprendizaje automático que tome como entrada al atributo que resulta del Algoritmo 1 para predecir la congestión futura, el que se considerará como el dato “real” u observado. Este modelo puede implementarse luego sobre nuevos datos. Los modelos de predicción de congestión se detallan en la Sección 4.2.1 y su uso para predecir velocidad en la Sección 4.2.2.

4.2. Modelos de aprendizaje automático

Ante la tarea de predecir velocidad en el corto plazo, se opta por emplear métodos de aprendizaje automático. Estos algoritmos cuentan con una amplia variedad de enfoques, aunque tienen en común el uso de parámetros numéricos para ejecutar una tarea a partir de un conjunto de datos. En esta sección se detallarán los modelos necesarios para contar con un dato de congestión que corrija el problema de *target leakage*, con el fin de obtener dicha información en tiempo real. Además, se señalarán aquellos métodos a utilizar para realizar pronósticos de velocidad en el corto plazo.

Los modelos a desarrollar corresponden a una rama del aprendizaje automático llamada aprendizaje supervisado. Esta se distingue por ajustar un método mediante datos de entrada o *inputs* para que se entregue una salida cuyos valores esperados se conocen de antemano en una muestra de testeo. Otros tipos de modelos corresponden al aprendizaje no supervisado o al aprendizaje por refuerzo, los que no forman parte del alcance de esta tesis.

4.2.1. Predicción del estado de congestión futuro

Una vez ejecutados los procedimientos para identificar el estado de tráfico, se cuenta con una variable que indica si existe o no congestión en cada pórtico. Sin embargo, fue explicado en la Sección 4.1 que dicho procedimiento requiere de información futura. Por ejemplo, si se desea conocer si hay congestión en $t = 16:00$, el Algoritmo 1 debe observar la velocidad entre ese instante y las 16:04. Ante esto se presenta la alternativa de desarrollar un modelo de aprendizaje automático para predecir la ocurrencia de congestión futura.

En términos prácticos, la variable a predecir será el estado de tráfico dentro de cinco minutos. Siguiendo el ejemplo, se puede determinar el estado de congestión en $t = 15:55$ mediante el Algoritmo 1, lo que requiere información hasta $t + 4 = 15:59$. Con esto, este atributo se puede emplear como una variable de entrada para pronosticar el estado de congestión en $t + 5 = 16:00$.

Los métodos de aprendizaje automático a detallar corresponden a modelos de clasificación, pues la variable a predecir es categórica y vale 0 o 1 según se observe flujo libre o congestión en los próximos cinco minutos, respectivamente. El estudio se enfocará en la predicción sobre lo que ocurre en el pódico PA20, en vista que es el último en la autopista estudiada dado el sentido del tráfico (ver Figura 3.1) y permite el uso de información de los puntos de medición ubicados “aguas arriba” en la vía. El detalle de las variables a utilizar como *inputs* será abordado en la Sección 5.1, una vez sea discutido el preprocesamiento que permite generar el conjunto de datos de entrada de los modelos.

4.2.1.1. Naive-Bayes

Este modelo toma sus fundamentos en el teorema de Bayes y en un supuesto “ingenuo” sobre los datos de entrada al algoritmo, asumiendo que los atributos son variables independientes entre sí. Como lo puntualiza Murty [26], este es un supuesto fuerte y no necesariamente cierto, pero aún así los modelos de Naive-Bayes han demostrado un buen desempeño en tareas de clasificación como la detección de correos *spam*.

Sean $x = (x_1, x_2, \dots, x_n)$ los n atributos de entrada e y la variable a predecir. En este caso, esta última será una variable binaria que indique la presencia de congestión de los próximos cinco minutos. Interesa la probabilidad de que, dados los datos observados, se esté en el estado c , lo que puede expresarse como $\mathbb{P}(y = c | x) = \mathbb{P}(y = c | x_1, \dots, x_n)$, con $c \in \{0, 1\}$.

Usando el teorema de Bayes, se obtiene que

$$\mathbb{P}(y = c | x) = \frac{\mathbb{P}(y = c)\mathbb{P}(x | y = c)}{\mathbb{P}(x)} \quad (4.1)$$

y teniendo en cuenta el supuesto de independencia,

$$\begin{aligned} \mathbb{P}(x | y = c) &= \mathbb{P}(x_1, x_2, \dots, x_n | y = c) \\ &= \mathbb{P}(x_1 | y = c) \mathbb{P}(x_2 | y = c) \dots \mathbb{P}(x_n | y = c) \\ &= \prod_{i=1}^n \mathbb{P}(x_i | y = c). \end{aligned} \quad (4.2)$$

Con esto, se puede mostrar que la Ecuación 4.1 corresponde a lo siguiente (por ejemplo, ver en Murty [26]):

$$\mathbb{P}(y = c | x) = \frac{\mathbb{P}(y = c) \prod_{i=1}^n \mathbb{P}(x_i | y = c)}{\mathbb{P}(x)}. \quad (4.3)$$

A partir de esta expresión, se escoge a la clase \hat{c} que sea más probable de observar a partir de los datos los datos, es decir, que maximice la expresión 4.3. Ya que el denominador es

constante, esto es equivalente a computar

$$\hat{c} = \arg \max_{c \in \{0,1\}} \mathbb{P}(y = c) \prod_{i=1}^n \mathbb{P}(x_i | y = c). \quad (4.4)$$

El desarrollo anterior es consistente con la implementación disponible en la librería de acceso abierto *Scikit-Learn* y documentada por Pedregosa et al. [27], la que es utilizada durante esta tesis.

Las diferentes implementaciones de los modelos de Naive-Bayes se distinguen según la distribución de probabilidad *a priori* que se asuma para los atributos en los datos de entrenamiento.

- La implementación **Gaussiana** supone que $\mathbb{P}(x_i | y = c)$ distribuye normal, es decir,

$$\mathbb{P}(x_i | y = c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} \exp\left(-\frac{(x_i - \mu_c)^2}{2\sigma_c^2}\right), \quad (4.5)$$

donde los parámetros μ_c y σ_c asociados a la clase c se estiman mediante el método de máxima verosimilitud a partir de los datos de entrenamiento.

- Por otro lado, se puede asumir una distribución de **Bernoulli** para cada atributo. Se considera que todos estos son binarios y, en caso de que no fuese así, se establece un umbral (usualmente 0 debido a la popularidad de la estandarización normal de las variables durante el preprocesamiento). En este caso,

$$\mathbb{P}(x_i | y = c) = p_i x_i + (1 - p_i)(1 - x_i), \quad (4.6)$$

con p_i un parámetro dependiente de cada atributo y que indica su probabilidad de “aparecer” o tomar el valor 1.

4.2.1.2. Árbol de decisión (clasificación)

En términos generales, un árbol de decisión es un algoritmo que identifica diferentes estados de un fenómeno a partir de la ocurrencia de ciertas condiciones observables. En el caso del aprendizaje automático, corresponde a un modelo de clasificación que particiona el conjunto de observaciones en función de patrones identificados mediante rangos sobre las variables de entrada.

Gráficamente, este método se caracteriza por su fácil comprensión. Se puede representar como un grafo de árbol binario cuyo nodo raíz contiene al total del conjunto de datos. Cada ramificación del árbol se define a partir de un atributo sobre el que se define un umbral, dividiendo en dos conjuntos disjuntos a los datos en el nodo precedente. A continuación, se dará el contexto teórico del modelo a partir de la documentación en *Scikit-Learn* realizada por Pedregosa et al. [27] y basada en Breiman et al. [28].

Sea $x = (x_1, x_2, \dots, x_n)$ el vector de los n atributos e $y \in \{0, 1\}$ la clase a predecir, que vale 1 si hay congestión y 0 en caso contrario. Dado un nodo m arbitrario, sea Ω_m el conjunto de

datos respectivo y $|\Omega_m| = N_m$ la cantidad de observaciones que contiene. Sea $\theta := \theta(x_j, t_m)$ el criterio de ramificación asociado al umbral t_m sobre el atributo x_j , el que permite generar dos nuevos nodos $\Omega_m^l(\theta)$ y $\Omega_m^r(\theta)$ definidos como

$$\Omega_m^l(\theta) = \{(x, y) \mid x_j \leq t_m\} \quad y \quad \Omega_m^r(\theta) = \Omega_m \setminus \Omega_m^l(\theta). \quad (4.7)$$

Es decir, es una partición disjunta de los datos del nodo precedente m . Sea $H : \Omega \rightarrow \mathbb{R}$ una función de pérdida o impureza que asigna un valor real a un conjunto de observaciones. A partir de esta, en la implementación descrita en Pedregosa et al. [27], se define la impureza de la ramificación G como la siguiente cantidad:

$$G(\Omega_m, \theta) := \frac{N_m^l}{N_m} H(\Omega_m^l(\theta)) + \frac{N_m^r}{N_m} H(\Omega_m^r(\theta)). \quad (4.8)$$

El criterio de ramificación a considerar se define como aquel que minimiza la impureza G , esto es,

$$\theta^* = \arg \min_{\theta} G(\Omega_m, \theta). \quad (4.9)$$

Si bien la definición de la función de pérdida H puede variar, en este caso se emplea el criterio de impureza de Gini. Dado un nodo m ante un problema de clasificación, este se expresa como

$$H(\Omega_m) = 2p_m^0 p_m^1, \quad (4.10)$$

con p_m^i la proporción de observaciones de Ω_m que pertenecen a la clase i , es decir,

$$p_m^i = \frac{1}{N_m} \sum_{(x,y) \in \Omega_m} \mathbb{1}_{\{y=i\}}, \quad \forall i \in \{0, 1\}. \quad (4.11)$$

El término $\mathbb{1}_{\{y=i\}}$ es una indicatriz que toma el valor 1 si y corresponde a la clase i y 0 en caso contrario. Dado que el problema de clasificación es binario, la entropía también puede expresarse por

$$H(\Omega_m) = 2p_m^1(1 - p_m^1). \quad (4.12)$$

En la Figura 4.1 se enseñan parte de los elementos descritos previamente, mostrando el mecanismo que permite ejecutar una ramificación a partir de un nodo arbitrario.

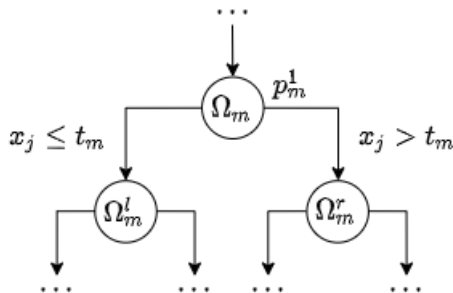


Figura 4.1: Ramificación en un árbol de decisión para un nodo arbitrario m considerando un criterio $\theta(x_j, t_m)$. Elaboración propia a partir de Pedregosa et al. [27] y Breiman et al. [28].

El proceso anterior se repite sucesivamente mediante la generación de nuevos nodos del árbol hasta que se cumpla al menos uno de los siguientes criterios:

- Todas las hojas son puras, es decir, contienen observaciones pertenecientes a una única clase.
- No hay más atributos que puedan usarse como criterio para ramificar.
- Volver a ramificar supondría superar la profundidad máxima del árbol que está configurada de antemano. La profundidad se define como la distancia entre una hoja y el nodo raíz, medida en cantidad de ramificaciones.

Una vez acabada la construcción del grafo, el modelo es capaz de ejecutar la clasificación de datos nuevos. Para una observación asociada a los atributos x , el árbol es recorrido hasta llegar al nodo hoja correspondiente, digamos a , según los criterios de ramificación. Si $p_a^1 \geq 0.5$, se dirá que la clase de la observación es 1, mientras que en caso contrario será 0.

4.2.1.3. *Random forest* (clasificación)

El aprendizaje conjunto (del inglés *ensemble learning*) es una clase de modelos de aprendizaje automático que se fundamentan en el entrenamiento de una colección de métodos independientes, los que luego se combinan bajo algún criterio de agregación. Opitz y Maclin [29] apuntan que el clasificador resultante de esta combinación tiende a ser más exacto en sus predicciones que los métodos individuales que lo componen. Además, indican que es deseable que estos últimos no sean similares entre sí en el sentido de que no entreguen la misma predicción. Esta idea descansa en que cada parte del modelo conjunto pueda identificar patrones distintos en los datos, de manera tal que se abarque un mayor espectro de escenarios posibles que no podrían capturarse con un único modelo.

En el caso en que los clasificadores individuales sean árboles de decisión, el modelo conjunto se denomina *Random Forest*. Un algoritmo popular para la construcción de este método corresponde a la Agregación de *Bootstrap*, también denominada *Bagging*. Esta se basa en que, dado un conjunto de datos, cada árbol se entrena a partir de una submuestra seleccionada aleatoriamente y con reposición. El *bagging* fue descrito por Breiman [30], quien indica que el procedimiento ayuda a mejorar el desempeño predictivo de este tipo de modelos basados en árboles.

Cada árbol de decisión se asocia a una submuestra y se entrena por separado. La implementación de *random forest* en *Scikit-Learn* [27] agrega los modelos mediante el promedio de la probabilidad de pertenecer a cada clase que es entregada por cada árbol. Esto es, si se considera emplear A árboles de decisión, cada $a \in \{1, \dots, A\}$ entrega una probabilidad empírica p_a^1 de pertenecer a la clase $y = 1$ y la predicción generada corresponde a

$$p_{RF}^1 = \frac{1}{A} \sum_{a=1}^A p_a^1. \quad (4.13)$$

Si $p_{RF}^1 \geq 0.5$, se predice que la observación pertenece a la clase 1. La Figura 4.2 diagrama el

procedimiento de *Bagging* en la construcción de un modelo de *random forest*, considerando una cantidad A de árboles de decisión para entregar una predicción.

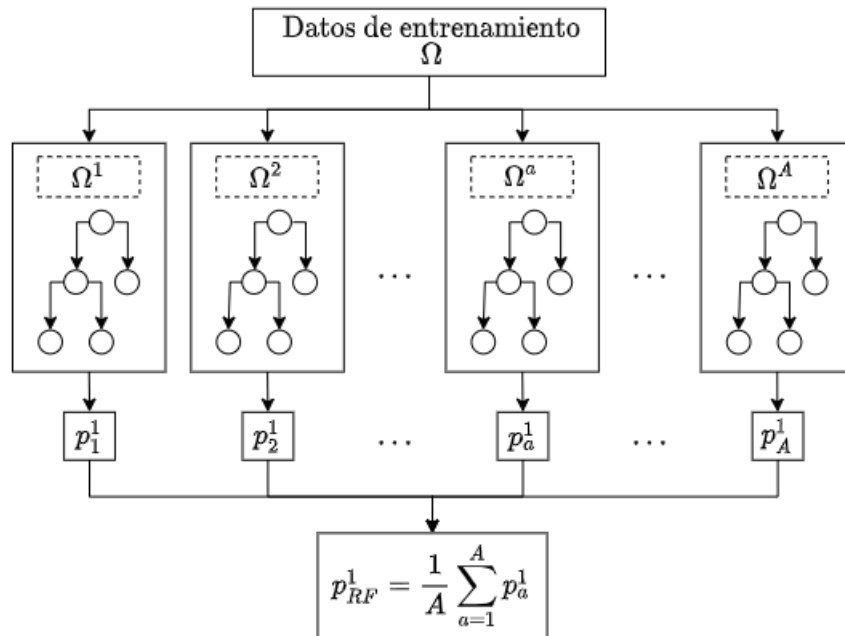


Figura 4.2: Ramificación en un árbol de decisión para un nodo arbitrario m y considerando un criterio $\theta(x_j, t_m)$. A partir del conjunto de datos Ω , se generan submuestras aleatorias y con reposición denominadas Ω^a para $a \in \{1, \dots, A\}$. Elaboración propia a partir de Breiman [30].

Los modelos de este tipo se caracterizan por ser más generalizables tras la aplicación de la agregación de *bootstrap*. Breiman [31] justifica mediante la Ley de los Grandes Números que los modelos de *random forest* son una manera de enfrentar el problema de sobreajuste (que será discutido en profundidad en la Sección 4.4).

4.2.1.4. Red neuronal MLP (clasificación)

Las redes neuronales son un modelo de aprendizaje automático que han dado paso al área del aprendizaje profundo o *deep learning*, logrando popularidad en tareas de clasificación y regresión. Su origen proviene de una “imitación” de la estructura de las neuronas dentro del sistema nervioso, a pesar de que su desarrollo actual dista de circunscribirse únicamente a este concepto.

Es importante realizar la distinción entre parámetros e hiperparámetros bajo este contexto. Los primeros se refieren a los valores que caracterizan el procesamiento matemático realizado por el modelo para producir una predicción (como la matriz W a definir más adelante), mientras que los segundos son las características que determinan la arquitectura o “la forma” de las redes neuronales (por ejemplo, la cantidad de capas). Una diferencia fundamental es que los hiperparámetros se fijan de antemano, mientras que los parámetros se calculan durante el entrenamiento según un procedimiento de minimización de una métrica de error.

La arquitectura más simple, que a su vez es la unidad básica de los modelos de redes neuronales, ha adquirido varias denominaciones: unidad, neurona o perceptrón. Dado un problema de clasificación con F atributos de entrada $x = (x_1, x_2, \dots, x_F)$ e y la variable a predecir, la neurona cuenta con tres parámetros fundamentales que determinan la salida del modelo. Estos corresponden a un vector de pesos $w \in \mathbb{R}^F$ asignado a cada elemento de x , un valor de sesgo $b \in \mathbb{R}$ y una función de activación $f: \mathbb{R} \rightarrow \mathbb{R}$.

La neurona agrega a los atributos de entrada ponderados por sus pesos. A este resultado se le suma el término de sesgo para finalmente aplicar la función de activación y entregar la salida \hat{y} correspondiente a la clasificación ejecutada por el método. La situación anterior se resume en la expresión

$$\hat{y} = f(w^\top x + b). \quad (4.14)$$

Existen varias alternativas de funciones de activación en la implementación de este tipo de clasificadores, siendo lo más clásico imitar a la función de Heaviside o de paso que toma los valores 1 y 0 dependiendo del signo del valor de entrada. Algunas aproximaciones continuas son la función sigmoide o la tangente hiperbólica. El rol del sesgo b es ajustar al término $w^\top x$ para que tome un signo u otro, dependiendo de lo que sea más adecuado para predecir la clase de manera más certera. Como resultado, se cuenta con un clasificador binario caracterizado por $n+1$ parámetros w y b . En la Figura 4.3 se enseña un esquema que resume los componentes de una neurona.

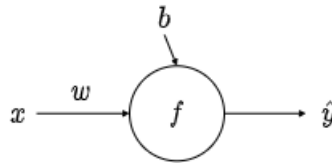


Figura 4.3: Diagrama de una neurona simple y sus componentes. Elaboración propia a partir de Géron [32].

El perceptrón fue introducido en 1958 por Frank Rosenblatt [33], aunque su desarrollo ha sufrido vaivenes hasta su estado actual. Diversos autores —por ejemplo Géron [32]— han indicado que ciertos factores retrasaron la difusión masiva en el uso de redes neuronales, tales como la falta de capacidad de procesamiento o la aparición de otros métodos competitivos de aprendizaje automático. También indica que hoy se han sorteado algunas de estas dificultades, así como se ha perfeccionado la implementación al haber mayor disponibilidad de datos masivos y mejores algoritmos de entrenamiento.

Una complejización del método anterior consiste en una serie de neuronas que funcionan en paralelo dentro de una *capa*. Para una mejor conceptualización, se distinguen tres tipos de capas:

- La capa de entrada, constituida por los atributos de entrada que son enviados a las neuronas que siguen en la red.
- La capa oculta o escondida, que reúne a la serie de neuronas que procesan los atributos recibidos desde la capa de entrada y que alimentan a la capa de salida.

- La capa de salida, formada por una única neurona que recibe los valores entregados por los perceptrones anteriores.

Se dice que una capa es “densa” o “totalmente conectada” si está conectada a cada unidad de la capa que la antecede, lo que se cumple en este tipo de estructuras. Esta característica provoca que algunos autores se refieran a este tipo de modelos como redes neuronales totalmente conectadas o *fully-connected neural network* (FFNN). La Figura 4.4 enseña esta arquitectura e indica sus componentes fundamentales.

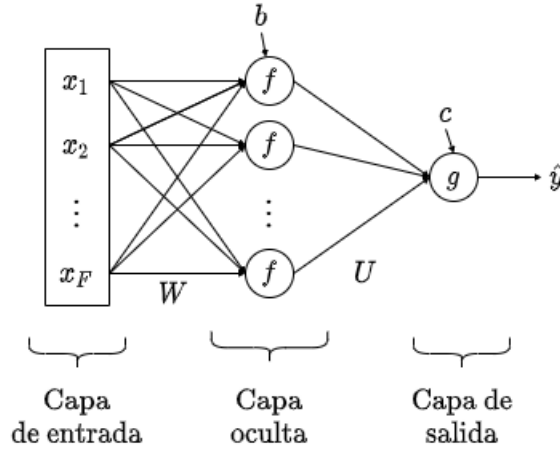


Figura 4.4: Diagrama de la arquitectura del perceptrón y sus componentes. Elaboración propia a partir de Géron [32].

Suponiendo que la capa oculta se compone de d unidades, los pesos corresponden a una matriz $W \in \mathbb{R}^{d \times F}$ cuyas componentes w_{ij} son el ponderador de la conexión entre el atributo x_i y la j -ésima neurona. Además, cada unidad cuenta con su propio término de sesgo, agrupados en el vector $b = (b_1, b_2, \dots, b_d) \in \mathbb{R}^d$. En general, todas las neuronas de la misma capa comparten su función de activación. Con esto, es posible establecer la salida parcial h que produce la capa oculta dada por

$$h = f(Wx + b) \quad \forall i \in \{1, \dots, d\}, \quad (4.15)$$

donde la función f se aplica a cada componente para producir el vector $h = (h_1, h_2, \dots, h_d)$. La capa de salida recibe estos valores y computa \hat{y} de manera análoga a las unidades anteriores según la expresión

$$\hat{y} = g(U^T h + c), \quad (4.16)$$

donde $U \in \mathbb{R}^d$ es el vector de pesos de la capa de salida, g la función de activación y c el término de sesgo.

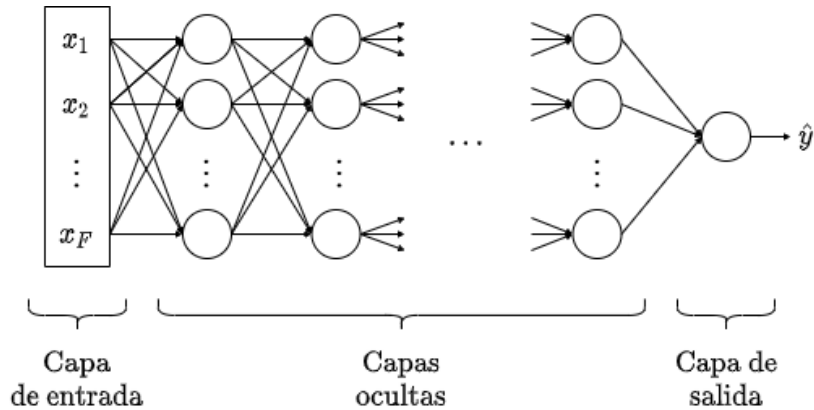
Un hiperparámetro que define a una red neuronal de una sola capa oculta es la cantidad de unidades que componen a esta última. Aunque la capa de salida puede contar con más de una neurona, este caso no es relevante para el presente estudio en vista que el resultado deseado requiere clasificar sólo en dos estados. Aquello puede modelarse con el resultado \hat{y} como una variable binaria.

La extensión natural a un modelo más complejo consiste en introducir más capas ocultas, todas totalmente conectadas a sus predecesoras. Esto suele recibir el nombre de Perceptrón Multicapa (o MLP, por sus siglas del inglés *Multi-Layer Perceptron*), aunque también ha sido denominada *feedforward neural network* debido a que todas sus conexiones entre neuronas van “hacia adelante” y no retroalimentan a capas anteriores. La Figura 4.5.a muestra la forma genérica de este tipo de arquitecturas.

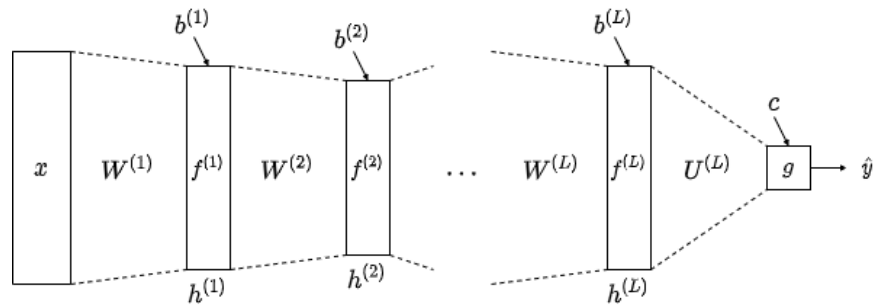
Suponiendo una red multicapa con L capas ocultas, como la diagramada en la Figura 4.5.b, cada una de ellas cuenta con sus propios parámetros que se ajustan independientemente con el objetivo de mejorar la predicción \hat{y} . Así, cada capa l cuenta con una función de activación $f^{(l)}$, una matriz de pesos $W^{(l)}$ y un vector de sesgo $b^{(l)}$. Con ello es posible explicitar la salida de las neuronas de cada capa oculta, dada por

$$h^{(l)} = f^{(l)}(W^{(l)}h^{(l-1)} + b^{(l)}) \quad \forall l \in \{1, \dots, L\}, \quad (4.17)$$

considerando que la capa de entrada es la capa 0, es decir, $h^{(0)} = x$.



(a) Conexiones entre capas en una MLP



(b) Componentes de una MLP

Figura 4.5: Diagramas de una red neuronal MLP de L capas ocultas. Elaboración propia a partir de Géron [32].

Para una capa l arbitraria, los parámetros verifican que $W^{(l)} \in \mathbb{R}^{d_l \times d_{l-1}}$ y $b \in \mathbb{R}^{d_l}$. Por otro lado, la capa de salida genera la predicción según la fórmula

$$\hat{y} = g(U^T h^{(L)} + c). \quad (4.18)$$

Los hiperparámetros básicos que definen a una red neuronal MLP consisten en la cantidad de capas ocultas y en las unidades contenidas en cada una de ellas. Por la misma razón que en el caso previo, basta considerar que la capa de salida se conforma de una sola neurona.

En la práctica, los modelos de aprendizaje automático recogen una cantidad arbitraria de instancias. Suponiendo que hay m observaciones $\{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$, cada una con F atributos, y una variable objetivo y , estas se pueden agrupar en la matriz $X \in \mathbb{R}^{m \times F}$. Sean $y \in \mathbb{R}^m$ los resultados observados e $\hat{y} \in \mathbb{R}^m$ las predicciones realizadas por la red neuronal.

$$X = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \cdots & x_F^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \cdots & x_F^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{(m)} & x_2^{(m)} & \cdots & x_F^{(m)} \end{bmatrix} \quad y = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{pmatrix} \quad \hat{y} = \begin{pmatrix} \hat{y}^{(1)} \\ \hat{y}^{(2)} \\ \vdots \\ \hat{y}^{(m)} \end{pmatrix} \quad (4.19)$$

Dada una métrica de error $L: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$, se define la función de pérdida \mathcal{L} como el promedio de los errores sobre las observaciones disponibles:

$$\mathcal{L} = \frac{1}{m} \sum_{i=1}^m L(y^{(i)}, \hat{y}^{(i)}). \quad (4.20)$$

El propósito de esta cantidad es medir qué tan correctas son las predicciones ejecutadas por la red y se mencionan algunos ejemplos en la Sección 4.3. El objetivo es que la función de pérdida sea lo más pequeña posible. En general, el conjunto de datos está dado. Además, los hiperparámetros que definen la arquitectura del modelo se fijan de antemano (para este trabajo se usa validación cruzada, ver Sección 4.4.2). Con esto, la función de pérdida sólo depende de los parámetros $\theta = (W, b, U, c)$ y el problema se puede expresar como uno de minimización:

$$\min_{\theta} \mathcal{L} = \min_{\theta} \mathcal{L}(\theta). \quad (4.21)$$

Una de las tareas más relevantes en el campo de las redes neuronales consiste en cómo escoger los mejores parámetros para reducir el valor de la función de pérdida. Un trabajo que significó un punto de inflexión consistió en el desarrollo de una metodología propuesta en 1986 por Rumelhart, Hinton y Williams [34]. Esta se conoce como el algoritmo de *backpropagation* y se basa en la idea de que el error de la red neuronal se “propaga” hacia adelante a partir del procesamiento que realiza cada componente de la red. De esta manera, el error final es atribuible a alguna neurona en la primera capa, así como al resto de componentes de la red. El método de *backpropagation* traza estos errores hacia atrás, lo que se modelará sucintamente a continuación.

El algoritmo se repite iterativamente, por lo que esta descripción se fija en el paso t . Se utiliza el método de descenso de gradiente y algoritmos de optimización derivados de él, pero en términos simples los parámetros se actualizan según la siguiente regla:

$$\hat{\theta}^{(t+1)} := \hat{\theta}^{(t)} + \eta \left(\frac{d\mathcal{L}}{d\theta} \right)_{\theta=\hat{\theta}^{(t)}}, \quad (4.22)$$

donde $\hat{\theta}^{(t)}$ representa a los valores actuales de los parámetros en el paso t y η es un valor real denominado tasa de aprendizaje.

La innovación del algoritmo de *backpropagation* consiste en el cálculo de la derivada en la expresión 4.22. Rumelhart et al. [34] proponen un uso inteligente de la regla de la cadena que permita reutilizar información. Un ejemplo sencillo consiste en considerar que, usando esta regla, el error propagado por la salida h de una capa sirve para computar el error que proviene de todas las capas anteriores. El método de *backpropagation* calcula las derivadas parciales hasta llegar a la capa de entrada. Para su correcto funcionamiento, requiere que la arquitectura de la red esté dada por un grafo acíclico.

Es usual que la red neuronal no procese todas las observaciones al mismo tiempo, sino que las procesa por lotes pequeños denominados *batches*. Cada vez que un *batch* es procesado por la red neuronal, se obtiene el valor de la función de pérdida y se ejecuta el método de descenso del gradiente (una iteración de la ecuación 4.22). Cuando todos los datos de entrada se han utilizado, que equivale a que todos los *batches* han pasado por la red, se dice que esta se ha entrenado durante un *epoch* (o época). El entrenamiento suele considerar más de un *epoch*, lo que implica repetir el procedimiento anterior para perfeccionar los valores de los parámetros mediante la reutilización de la base de entrenamiento.

4.2.2. Predicción de velocidad media futura

Los modelos de clasificación detallados anteriormente proveen una predicción sobre el estado de congestión en los próximos cinco minutos. El paso siguiente corresponde a generar los pronósticos de la velocidad y , ya que la velocidad un valor continuo, requiere el entrenamiento de modelos de regresión.

4.2.2.1. Árbol de decisión (regresión)

El árbol de decisión usado como método de regresión cuenta con un funcionamiento similar a su versión de clasificador. La principal diferencia consiste en el criterio de ramificación usado, pues la noción de probabilidad asociada a pertenecer a una clase ya no es aplicable. Para estos efectos, la métrica de error en la implementación de *Scikit-Learn* [27] es el error cuadrático medio.

Sea un nodo m que alberga un conjunto Ω_m de observaciones de cardinalidad N_m . Dado \bar{y}_m , correspondiente al valor promedio de las variables a predecir en dicho nodo, el criterio del error cuadrático medio se calcula mediante la expresión

$$H(\Omega_m) = \frac{1}{N_m} \sum_{y | (x,y) \in \Omega_m} (y - \bar{y}_m)^2. \quad (4.23)$$

Las ramificaciones ocurren de manera idéntica a como se vio en la Sección 4.2.1.2, es decir, escogiendo los umbrales θ de forma tal que se minimice la impureza $G(\Omega_m, \theta)$. Para predecir el valor de una nueva observación, esta se asigna a un nodo hoja, digamos a , y se entrega el promedio \bar{y}_a de los valores de la variable a predecir contenidas en él.

4.2.2.2. *Random forest* (regresión)

En este caso, los modelos individuales que se entrenan mediante la agregación de *bootstrap* corresponden a árboles de regresión. El resto de procedimientos del *bagging* son análogos. Si se emplean A árboles, donde cada uno de estos genera una predicción y_a , el *random forest* entrega como pronóstico al siguiente valor:

$$y_{RF} = \frac{1}{A} \sum_{a=1}^A y_a. \quad (4.24)$$

4.2.2.3. Red neuronal MLP (regresión)

La arquitectura de una red neuronal usada para predecir valores continuos no difiere en lo grueso de un clasificador binario, es decir, en términos de unidades y capas. No obstante, es importante asegurar que la salida del modelo entregue un valor continuo.

En primer lugar, la función de activación de la capa de salida no debe restringir a valores binarios, como ocurre con el uso funciones como la sigmoide o la tangente hiperbólica. Lo más usual es usar la función identidad $f(x) = x$, aunque también se recomienda el uso de la función ReLU (del inglés *rectified linear unit*) cuando no se desea retornar valores negativos. Esta se define como la función $f(x) = \max\{0, x\}$.

Una consecuencia de lo anterior es que se debe contar con una métrica de error adecuada en la función de pérdida \mathcal{L} , las que se detallarán en la siguiente sección.

4.3. Medidas de error

Según la definición dada por Mitchell [10], un elemento fundamental de los métodos de aprendizaje automático es la métrica de desempeño, la que permite evaluar el funcionamiento de estos para discriminar cuáles son aquellos que proveen los mejores resultados. Dado el entrenamiento de un modelo, en el proceso de validación y testeo es posible calcular un conjunto de predicciones \hat{y} cuyos valores observados se recogen en el vector y . Las medidas de error permiten comparar a y con \hat{y} , ambos de tamaño N , de manera cuantitativa y sistemática.

En primer lugar, cabe destacar que las redes neuronales se entrenan iterativamente con la intención de reducir el error de predicción de forma, por lo que las métricas de este tipo son fundamentales para su funcionamiento. En el caso de las redes neuronales clasificadoras, se considera la entropía cruzada binaria (o *BCE* por las siglas de *binary cross-entropy*) al existir dos clases a predecir. Para el caso de regresión, se utiliza el error cuadrático medio (o

MSE por las siglas de *mean squared error*).

$$BCE = -\frac{1}{N} \sum_{i=1}^N y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \quad (4.25)$$

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (4.26)$$

Para los modelos de clasificación, es usual reportar los valores de la matriz de confusión. Esta realiza una comparación entre los valores reales y predichos, mostrando cuando coinciden o difieren entre sí. En la Tabla 4.1 se enseña una matriz de confusión con sus cuatro componentes para un problema de clasificación binario: la cantidad de verdaderos negativos (TN), verdaderos positivos (TP), falsos negativos (FN) y falsos positivos (FP).

$$\begin{aligned} TN &= \sum_{i=1}^N \mathbb{1}_{\{y_i=0 \wedge \hat{y}_i=0\}} & TP &= \sum_{i=1}^N \mathbb{1}_{\{y_i=1 \wedge \hat{y}_i=1\}} \\ FN &= \sum_{i=1}^N \mathbb{1}_{\{y_i=1 \wedge \hat{y}_i=0\}} & FP &= \sum_{i=1}^N \mathbb{1}_{\{y_i=0 \wedge \hat{y}_i=1\}} \end{aligned} \quad (4.27)$$

Tabla 4.1: Matriz de confusión

		Valor predicho	
		0	1
Valor real	0	TN	FP
	1	FN	TP

A partir de estos valores se pueden reportar otras métricas más específicas que también serán utilizadas y que se definen a continuación.

$$Accuracy(y, \hat{y}) = \frac{TP + FP}{TP + TN + FP + FN} \quad (4.28)$$

$$Balanced\ accuracy(y, \hat{y}) = \frac{1}{2} \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right) \quad (4.29)$$

$$Recall(y, \hat{y}) = \frac{TP}{TP + FN} \quad (4.30)$$

$$Precision(y, \hat{y}) = \frac{TP}{TP + FP} \quad (4.31)$$

$$F1 - Score(y, \hat{y}) = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4.32)$$

El *accuracy* (o exactitud) corresponde a la proporción de observaciones en las que la predicción acertó, mientras que el *accuracy* balanceado es el promedio de la proporción de observaciones de cada clase que el modelo predijo correctamente. A partir de la precisión y del *recall* (o exhaustividad), se calcula el *F1 - Score* como la media armónica de estas dos cantidades.

En el caso de los modelos de regresión, se obtendrá el error absoluto medio (*MAE*) y el

error porcentual absoluto medio ($MAPE$).

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (4.33)$$

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100 \quad (4.34)$$

4.4. Entrenamiento y elección de hiperparámetros

Los modelos de aprendizaje automático se caracterizan por el uso de hiperparámetros que afectan el resultado de una tarea de clasificación o regresión. A diferencia de los parámetros de un modelo, los hiperparámetros se escogen de antemano al entrenamiento y no mediante un algoritmo de optimización que tenga en cuenta una función de pérdida. Por ejemplo, la profundidad de un árbol de decisión se establece antes de entrenarlo. Sin embargo, no existe un método para obtener la configuración óptima de un modelo *a priori*, sino que deben evaluarse diferentes combinaciones de hiperparámetros y observar el desempeño en cada caso.

4.4.1. Entrenamiento-validación-testeo

La elección de hiperparámetros requiere entrenar varios modelos, cada uno de ellos con una configuración diferente, y luego poner a prueba el desempeño de cada uno. Una regla fundamental en los modelos de aprendizaje automático es que la evaluación de modelos debe realizarse con datos que no hayan sido “vistos” por el modelo, es decir, que no se hayan usado durante su entrenamiento.

Para este trabajo, los datos se dividirán en tres subconjuntos: los datos de entrenamiento, los datos de validación y los datos de testeo. Los datos de entrenamiento corresponden a aquellos que, dada cierta configuración de hiperparámetros Φ , serán usados para ajustar los parámetros del modelo que se obtienen mediante el algoritmo de minimización del error.

Una vez se complete dicha etapa, se emplean los datos de validación para evaluar el desempeño de los distintos modelos y escoger la combinación de hiperparámetros Φ^* cuya predicción provea el menor error. En seguida, con la intención de realizar el ajuste con la mayor cantidad de datos posibles, se puede re-entrenar el modelo usando tanto los datos de entrenamiento y de validación, según la configuración Φ^* de hiperparámetros. Este puede ser evaluado sobre los datos de testeo que aún no se han empleado durante el proceso.

En resumen, los datos de validación permiten comparar diferentes configuraciones de un mismo modelo de aprendizaje automático, mientras que los datos de testeo asumen que dicho problema ya está resuelto y sirven para discriminar *entre* modelos (un árbol de decisión, una red neuronal, etc.). Durante este trabajo, se considera el 60 %, 20 % y 20 % de las observaciones para los subconjuntos de entrenamiento, validación y testeo; respectivamente.

4.4.2. Sobreajuste

Considerando que la validación y el testeo se ejecuta sobre datos que no participan del ajuste, se cuestiona qué tan eficaz puede ser un modelo puesto a prueba ante información nueva. Una preocupación importante en el aprendizaje automático es el fenómeno del sobreajuste. Este se presenta cuando el método es un buen predictor de los datos de entrenamiento, pero muestra un pobre desempeño ante datos de validación o testeo. Su consecuencia más evidente es que se obtendrá un método de predicción deficiente en términos de capacidad de generalización.

En este contexto, la elección de las observaciones a usar en el entrenamiento, la validación y el testeo también juega un rol. La determinación arbitraria de estos subconjuntos podría perjudicar el desempeño de los modelos. Por ejemplo, si un modelo de predicción de ventas de helados se entrena sólo con datos de días calurosos, sus resultados serán muy diferentes en comparación a usar una muestra escogida aleatoriamente.

La validación cruzada (o CV por las siglas de *cross validation*) es un método de entrenamiento que permite reducir el sobreajuste y consiste en la evaluación de una misma configuración de hiperparámetros sobre diferentes conjuntos de validación. Se implementará de la siguiente manera.

1. La muestra se ordena cronológicamente y se divide en cinco conjuntos del mismo tamaño A, B, C, D y E. El subconjunto E, formado por las observaciones más recientes, constituye la base de testeo.

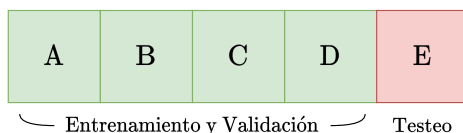


Figura 4.6: División del conjunto de datos para la validación cruzada. Elaboración propia.

2. Se generan cuatro escenarios para la validación cruzada a partir de los subconjuntos A, B, C y D. El modelo se entrena en cada escenario i para cada configuración de hiperparámetros Φ , obteniendo una métrica de error $\mathcal{L}_i(\Phi)$.

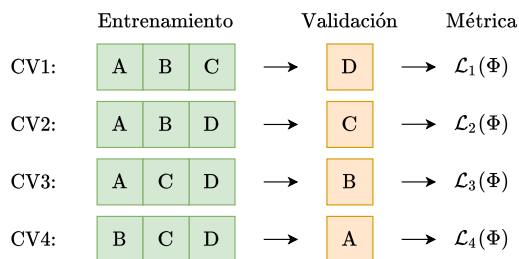


Figura 4.7: Escenarios de entrenamiento y validación. Elaboración propia.

3. La métrica de error que caracterizará a cada configuración de hiperparámetros se calcula

como el promedio sobre los cuatro escenarios de la validación cruzada, es decir,

$$\bar{\mathcal{L}}(\Phi) = \frac{\sum_{i=1}^4 \mathcal{L}_i(\Phi)}{4}. \quad (4.35)$$

4. La mejor configuración de hiperparámetros es aquella con la menor métrica de error promedio, dada por $\Phi^* = \arg \min_{\Phi} \{\bar{\mathcal{L}}(\Phi)\}$. La implementación concluye con el re-entrenamiento del modelo sobre los subconjuntos A, B, C y D con la configuración Φ^* . Las métricas de testeo $\mathcal{L}(\Phi^*)$ se obtienen a partir del subconjunto E.

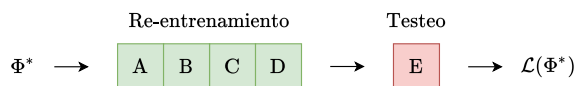


Figura 4.8: Esquema para el re-entrenamiento y testeo del modelo con los hiperparámetros ajustados Φ^* .

El proceso anterior también se denomina *k-fold cross validation*, usando $k = 5$ en este caso. Según Nisbet [25], esta metodología permite escoger los hiperparámetros de un modelo de forma más robusta, pues la media del error sobre varios escenarios es más exacta que el que podría obtenerse a partir de un solo experimento.

4.5. Tratamiento de clases desbalanceadas

Los modelos de clasificación son susceptibles de enfrentar datos desbalanceados, lo que corresponde a que existe una o más clases con una baja presencia relativa respecto a otras que son más numerosas en la muestra. De esta forma, el algoritmo puede llegar a obtener buenas métricas de desempeño atendiendo sólo a la clase dominante. Por ejemplo, una muestra con dos clases que se observan en una proporción 99 a 1 puede tener un exactitud del 99% si siempre pronostica la ocurrencia de la clase dominante.

Nisbet [25] señala que los métodos de árbol y las redes neuronales son sensibles a datos desbalanceados, pero también indica que existen métodos que permiten reducir el efecto de este problema. En particular, señala el uso de pesos sobre cada clase que actúen como ponderadores en la función de pérdida. Su adecuada utilización resulta en que la predicción de las clases con menor frecuencia tenga una mayor relevancia en la minimización del error.

El tratamiento anterior será aplicado en los modelos de predicción de congestión, lo que se justificará más adelante debido a los resultados de la aplicación del algoritmo que etiqueta la muestra con estados de tráfico. El peso o ponderador de cada clase se calcula como un valor inversamente proporcional a la presencia de este en el conjunto de datos.

$$w_i = \frac{n}{num_clases \times n_i} \quad (4.36)$$

La ecuación 4.36 indica la expresión a utilizar en este trabajo para la obtención del peso w_i asociado a la clase i . Este se obtiene considerando la cantidad total de observaciones (n),

la cantidad de clases en la muestra (num_clases) y las veces que se observa a la clase i (n_i).

De acuerdo con la documentación de *Scikit-Learn* provista por Pedregosa et al. [27], esta metodología se puede implementar mediante la siguiente modificación del cálculo de la entropía señalado en la ecuación 4.11:

$$p_m^i = \frac{1}{N_m} \sum_{(x,y) \in \Omega_m} w_i \mathbb{1}_{\{y=i\}}. \quad (4.37)$$

Esta expresión también permite el balanceo de clases en *random forests*. Análogamente, la implementación en redes neuronales sólo exige ponderar la contribución de la predicción de cada instancia j disponible en los datos de entrenamiento por el peso de la clase correspondiente:

$$\mathcal{L} = \frac{1}{m} \sum_{j=1}^m \sum_{i \in \{0,1\}} \mathbb{1}_{\{\hat{y}^{(j)}=i\}} w_i L(y^{(j)}, \hat{y}^{(j)}). \quad (4.38)$$

Capítulo 5

Análisis y Resultados

5.1. Preprocesamiento de datos

5.1.1. Imputación de datos faltantes

La muestra de datos de la autopista posee algunos campos en los que no se informa el valor del atributo respectivo, lo que puede deberse a fallos en la medición o a que no existe información suficiente. Por ejemplo, cuando en alguna observación ningún vehículo pasa por el pódico, es imposible obtener un valor empírico de la velocidad promedio.

Las únicas variables que poseen valores faltantes son la velocidad media y su desviación estándar, ya sea considerando los cuatro pódicos o los tres tipos de vehículos (livianos, pesados y motocicletas). En el Anexo A se indica la proporción de datos faltantes para cada variable respecto al total de la muestra. La variable a predecir, que consiste en la velocidad promedio de los vehículos livianos en el pódico PA20, posee un 0.022 % de valores faltantes.

Dada una variable cuya observación x_t debe ser imputada, un primer método utilizado fue calcular el promedio de las observaciones inmediatamente anterior y posterior. Esto es,

$$x_t = \frac{x_{t-1} + x_{t+1}}{2}. \quad (5.1)$$

No obstante, aún persisten algunos valores faltantes debido a que las observaciones x_{t-1} o x_{t+1} también deben ser imputadas. Para subsanar aquello, se considera el promedio de la variable x en otros días del conjunto de datos y para la misma hora de observación.

5.1.2. Generación de atributos

Algunas variables son generadas para obtener mayor información sobre las observaciones. A partir de la fecha y hora, que son separados en atributos independientes, también se obtiene el día de la semana: lunes, martes, miércoles, jueves o viernes.

Además de lo anterior, se genera el atributo de densidad de tráfico. Esta magnitud fue discutida en la Sección 2.2 y representa a la cantidad de vehículos por unidad de distancia. Según la ecuación 2.1, la densidad de tráfico puede calcularse como

$$k_{i,t,\mathcal{P}} = \frac{f_{i,t,\mathcal{P}}}{v_{i,t,\mathcal{P}}}, \quad (5.2)$$

siendo $k_{i,t,\mathcal{P}}$, $f_{i,t,\mathcal{P}}$ y $v_{i,t,\mathcal{P}}$ la densidad, el flujo y la velocidad promedio del pórtico \mathcal{P} durante el minuto t del día i , respectivamente. Por otro lado, se crean atributos que indican la composición del flujo en relación al tipo de vehículo. Estos se calculan con la expresión

$$Comp_{i,t,\mathcal{P}}^j = \frac{f_{i,t,\mathcal{P}}^j}{f_{i,t,\mathcal{P}}}, \quad (5.3)$$

donde f^j representa al flujo de los vehículos de tipo j (livianos, pesados o motocicletas).

Adicionalmente, se construyen variables de rezago para las mediciones obtenidas en los pórticos con el objetivo de considerar tendencias en los datos a partir de información pasada. Dada una observación x_t que da cuenta del estado de la variable x en el instante t , su rezago se asocia al valor de esta magnitud en intervalos de tiempo anteriores.

En específico, los rezagos serán calculados como el promedio móvil de 5 minutos entre los instantes $t - \Delta$ y $t - \Delta + 4$. El rezago de la variable x_t con horizonte Δ , denotada por \tilde{x}_t^Δ , se puede caracterizar mediante la fórmula

$$\tilde{x}_t^\Delta = \frac{1}{5} \sum_{i=0}^4 x_{t-\Delta+i}. \quad (5.4)$$

En el caso de este estudio, se considerarán los horizontes de tiempo $\Delta \in \{5, 10, 15, 30\}$.

5.2. Exploración de los datos

Como fue mencionado en la Sección 3.2, la muestra contiene 385 días de semana y mediciones entre las 14 y las 23 horas, registradas cada minuto. Existe la misma cantidad de días lunes, martes, miércoles, jueves y viernes, siendo 77 de cada uno de estos. Una minoría de estos corresponden a días feriados, los que representan al 4.68% del total.

Para el caso analizado, el flujo en un pórtico corresponde a la cantidad de vehículos que transitan por él durante un minuto. Por otro lado, la velocidad se obtiene como el promedio de esta variable sobre todos los vehículos que cruzan un pórtico en el lapso de un minuto.

La Figura 5.1 muestra que la velocidad disminuye entre las 17:30 y las 20:30, aproximadamente, lo que representa a la hora punta de la tarde. Los pórticos donde este fenómeno es más pronunciado corresponden al PA24 y al PA26 que están ubicados en el sector céntrico de la ciudad, y se alcanza un promedio en torno a los 55 km/h . Por otra parte, los pórticos PA20 y PA22 muestran una congestión menos severa y, en promedio, registran una mayor velocidad fuera de punta. Durante la semana, el caso más notable es el del día viernes, pues

en él se acentúa la hora punta y su inicio se adelanta en los casos de los pórticos PA24 y PA26.

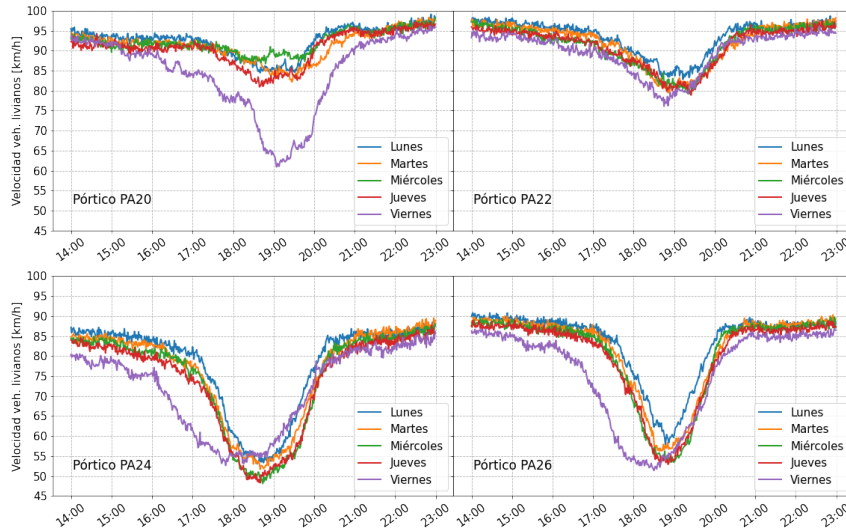


Figura 5.1: Velocidad promedio de vehículos livianos para cada día de la semana y para cada pórtico.

Al distinguir entre vehículos livianos, pesados y motocicletas, se observa que los patrones descritos se mantienen, tal como se enseña en la Figura 5.2. En promedio, los vehículos pesados transitan a una menor velocidad, aunque esta coincide con los vehículos livianos en la hora punta del pórtico PA24. Por otro lado, las motocicletas son menos sensibles a la congestión en los pórticos PA24 y PA26, pero presentan un comportamiento similar fuera de punta y en los demás puntos de medición.



Figura 5.2: Velocidad promedio para cada tipo de vehículo y para cada pórtico.

El análisis del flujo promedio en cada pórtico es consistente con lo visto respecto a la

velocidad. La Figura 5.3 da cuenta de una mayor cantidad de vehículos en la hora punta, siendo el pórtico PA26 aquel que presenta los valores más altos y el pórtico PA20 el de menor variación. El flujo es creciente desde las 14 horas y decreciente luego de la etapa de congestión. El día viernes es el que registra un mayor flujo promedio.

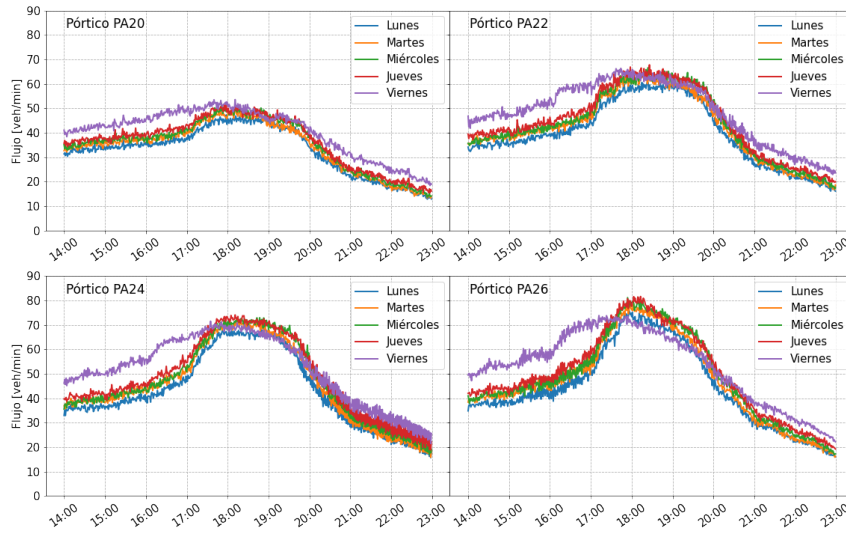


Figura 5.3: Flujo promedio de vehículos livianos para cada día de la semana y cada pórtico.

Según la Figura 5.4, los vehículos livianos componen mayoritariamente a la muestra y poseen un comportamiento similar al promedio general. Los vehículos pesados tienen una presencia constante hasta las 18 horas, tras lo que su flujo disminuye. La aparición de motocicletas es marginal, aunque mayor en la hora punta. Al comparar entre pórticos, el flujo de camiones y motocicletas es similar en todos ellos, a diferencia del caso de los automóviles livianos que es claramente mayor en los pórticos PA24 y PA26.

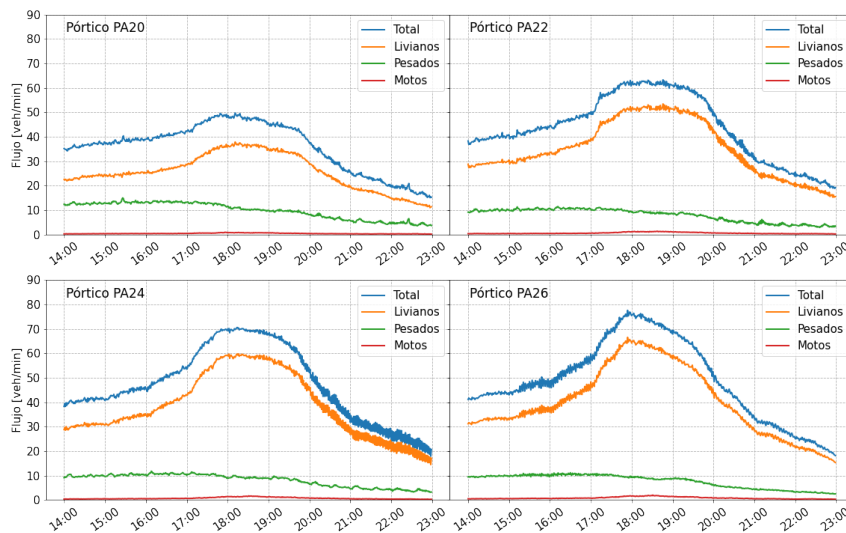


Figura 5.4: Flujo promedio para cada tipo de vehículo y cada pórtico

La Figura 5.5 muestra el coeficiente de variación del flujo y la velocidad, el que se calcula como la razón entre la desviación estándar de una variable y su promedio. La dispersión de la velocidad es similar para cada tipo de vehículo, aunque ligeramente mayor para las motocicletas y menor en el pódico PA22. Por otro lado, el flujo tiene una dispersión similar entre pódicos y es más estable en los vehículos livianos. Las motocicletas presentan un alto coeficiente de variación, lo que se explica por ser esta una medida relativa al flujo promedio que es especialmente bajo en este caso.

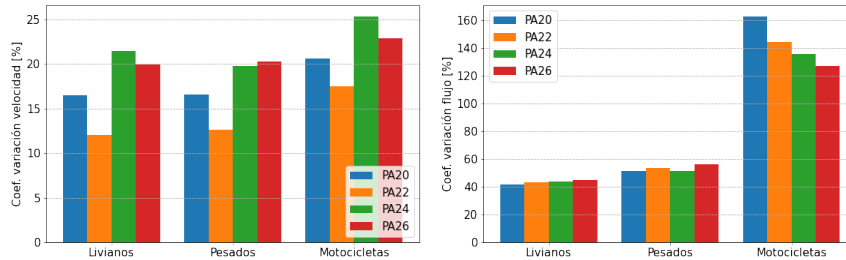


Figura 5.5: Coeficiente de variación de la velocidad y el flujo promedio por pódico y tipo de vehículo.

La Figura 5.6 muestra el diagrama velocidad-flujo empírico. En el pódico PA20 existe una región superior de observaciones en las que hay altas velocidades y otra inferior en las que ocurre el caso contrario. A diferencia del diagrama clásico y de lo que ocurre en PA22, PA24 y PA26, existe una discontinuidad evidente entre ambos estados de tráfico. Esto puede dar cuenta de que la transición a la congestión ocurre de forma abrupta, por lo que las velocidades intermedias de 40 o 60 km/h son más escasas en la muestra. En consecuencia, este pódico puede ser uno en los que sea más claro distinguir dos estados del tráfico según las características físicas observadas. El pódico PA26 representa el caso contrario, lo que da cuenta de fases intermedias entre una alta congestión y un flujo libre, las que han sido denominadas por Kerner [7] como *moving jam*. Estas se caracterizan por un avance sincronizado de un alto flujo de vehículos a velocidades menores a la de flujo libre.

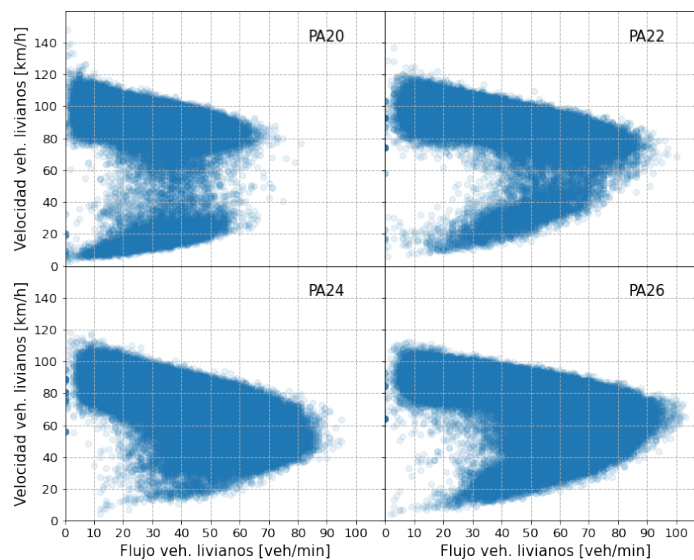


Figura 5.6: Diagrama flujo-velocidad en los diferentes pódicos

El p rtico PA20 concentra en los d as viernes la mayor parte de las observaciones con baja velocidad, como se muestra en la Figura 5.7. Para este d a hay una mayor presencia de observaciones con velocidades medias, lo que puede atribuirse a que ocurren m s cambios de fase. Al menos en lo que respecta a este diagrama, cada fase por separado cuenta con caracter sticas similares en cuanto a los valores del flujo y la velocidad, siendo esto consistente con el hecho de que representan a un mismo segmento de v a.

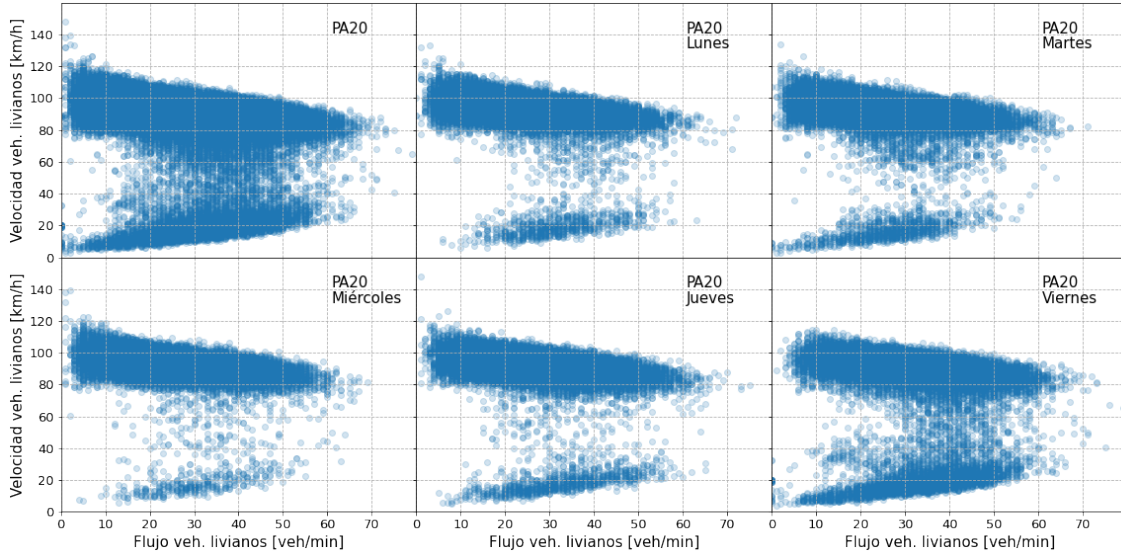


Figura 5.7: Diagrama flujo-velocidad en el p rtico PA20 para la muestra completa y seg n el d a de la semana.

La Figura 5.8.a muestra las correlaciones entre la velocidad y el flujo desde el punto de vista espacial, es decir, analizando los diferentes p rticos. En primer lugar, se evidencia una relaci n negativa entre las dos variables de tr fico que son analizadas. Como es natural, las correlaciones m s altas de cada p rtico se dan respecto a aquellos que se encuentran m s cerca, siendo los p rticos PA24 y PA26 los m s destacables por ubicarse a una menor distancia entre s . Por otro lado, el p rtico PA20 es el que se encuentra menos influenciado por el resto.

Finalmente, la Figura 5.8.b da cuenta de las correlaciones temporales en el p rtico PA20 al comparar con la velocidad y el flujo hace 5, 10, 15 y 30 minutos. La velocidad es la variable que se encuentra m s influenciada por informaci n pasada, en particular con una alta correlaci n entre su valor actual y los observados hace 5 y 10 minutos. Sus valores hace 30 minutos, as  como todos los del flujo, presentan d biles correlaciones temporales.

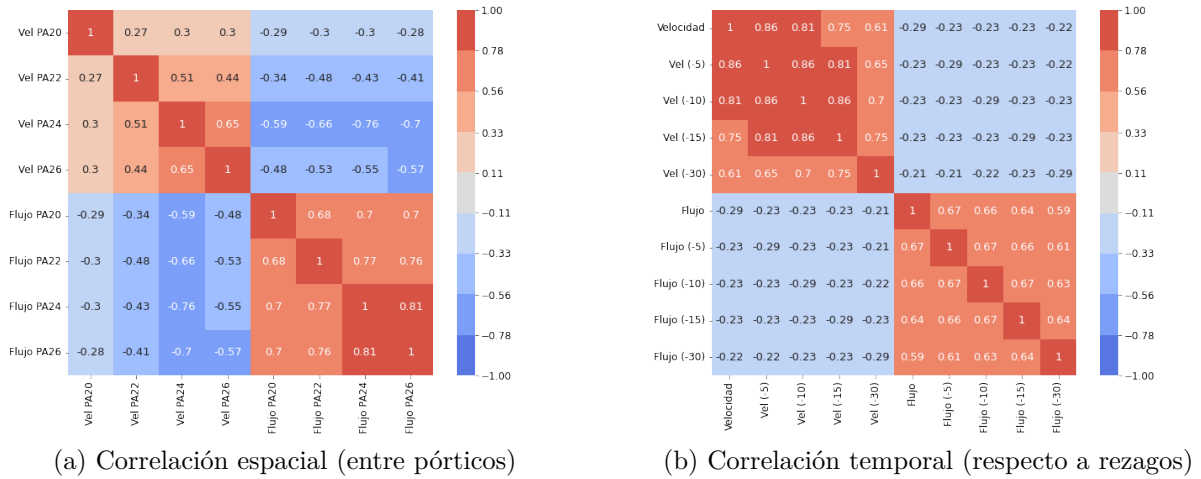


Figura 5.8: Correlaciones espaciales y temporales del flujo y la velocidad

5.3. Identificación del estado de congestión

El proceso de asignar un estado de tráfico al conjunto de datos fue desarrollado con el Algoritmo 1, descrito en la Sección 4.1, sobre 207,900 observaciones. Los cuatro pódicos fueron tratados separadamente durante el proceso.

En términos generales, la Tabla 5.1 muestra que la mayoría de las veces se observa un régimen de flujo libre, aunque con algunos diferencias según el día de la semana. En general, el algoritmo registra congestión en menos del 5% de las instancias para el pódico PA20 que será analizado más adelante, lo que supone el uso del tratamiento para el balance de clases descrito en la Sección [SECCIÓN] en los modelos de predicción.

El día viernes es el que presenta una mayor congestión, mientras que este fenómeno tiene una presencia más débil los días lunes en los puntos de medición PA22, PA24 y PA26. Los pódicos más congestionados son el PA24 y el PA26 (25.9% y 17.8%, respectivamente), mientras que en PA20 y PA22 no se alcanza a superar el 5% de los casos. Lo anterior da cuenta de que estos últimos se encuentran en una zona suburbana de la autopista y en dirección hacia afuera de la ciudad.

La Tabla 5.2 señala de que en la mayoría de los días en la muestra se registra al menos un evento de congestión, lo que ocurre cerca de un 90% de las veces para los pódicos PA24 y PA26. Para los otros dos puntos de medición, a pesar de que esta situación ocurre un tercio de las veces, es claro que hay una gran cantidad de días en los que no se registran eventos de congestión.

En el caso del pódico PA20, que es el que será utilizado para las predicciones, se aprecia un comportamiento uniforme en los días lunes, martes, miércoles y jueves. El día viernes ocurren más eventos de congestión, pues el 41.8% de los 123 días que presentan este fenómeno corresponden a este día de la semana. Sin perjuicio de lo anterior, se puede anticipar la

necesidad de establecer mecanismos para solucionar el desbalance en la proporción de cada clase.

Tabla 5.1: Porcentaje de observaciones etiquetadas bajo cada estado de tráfico para los diferentes pódicos y días de la semana.

	Lunes	Martes	Miércoles	Jueves	Viernes	Total
Cantidad de obs.	41580	41580	41580	41580	41580	207900
PA20 sin congestión	97.7 %	96.6 %	98.8 %	97.1 %	88.2 %	95.7 %
PA20 con congestión	2.3 %	3.4 %	1.2 %	2.9 %	11.8 %	4.3 %
PA22 sin congestión	97.8 %	96.8 %	97 %	97 %	95.3 %	96.8 %
PA22 con congestión	2.2 %	3.2 %	3 %	3 %	4.7 %	3.2 %
PA24 sin congestión	77.8 %	76.1 %	73.2 %	71.9 %	71.5 %	74.1 %
PA24 con congestión	22.2 %	23.9 %	26.8 %	28.1 %	28.5 %	25.9 %
PA26 sin congestión	87.6 %	85.1 %	82.2 %	81.2 %	74.9 %	82.2 %
PA26 con congestión	12.4 %	14.9 %	17.8 %	18.8 %	25.1 %	17.8 %

Tabla 5.2: Cantidad de días con al menos una observación congestionada y sin observaciones congestionadas para diferentes pódicos y días de la semana.

	Lunes	Martes	Miércoles	Jueves	Viernes	Total
Cantidad de días	77	77	77	77	77	385
PA20 con congestión	17	20	17	18	51	123
PA20 sin congestión	60	57	60	59	26	262
PA22 con congestión	27	25	31	34	37	154
PA22 sin congestión	50	52	46	43	40	231
PA24 con congestión	68	74	75	76	71	364
PA24 sin congestión	9	3	2	1	6	21
PA26 con congestión	64	69	72	73	71	349
PA26 sin congestión	13	8	5	4	6	36

Debido a que el algoritmo utilizado exige que las observaciones congestionadas se encuentren bajo un umbral de velocidad, es esperable que existan diferencias notorias en el comportamiento de esta variable según el estado de tráfico. En la Tabla 5.3 se aprecia que la velocidad promedio es de 92.89 km/h bajo el régimen de flujo libre, mientras que dicho valor es de 28.61 km/h cuando hay congestión. Además, se observa que en este último caso existe un mayor flujo promedio que alcanza los 33.09 veh/min vs. 25.46 veh/min en flujo libre.

Al revisar las métricas de dispersión, se observa que el flujo mantiene niveles similares de variabilidad en ambos estados de tráfico, pero la desviación estándar de la velocidad es cuatro veces mayor cuando hay congestión. Este hecho se interpreta como una consecuencia de las fuertes disminuciones en el valor de esta variable. La composición del tráfico según el tipo de vehículo da cuenta de que en este crecimiento participan más vehículos livianos que pesados.

Estos comportamientos indican que el algoritmo permite separar la muestra de datos en dos clases claramente distinguibles a partir de sus variables de tráfico. En consecuencia,

puede resultar útil considerar esta clasificación al momento de predecir la velocidad en el corto plazo, puesto que contienen información respecto al comportamiento de los vehículos en el segmento estudiado.

Tabla 5.3: Promedio y desviación estándar de variables de tráfico según el estado de congestión.

	Flujo Libre		Congestión	
	Promedio	Desv. Estándar	Promedio	Desv. Estándar
Velocidad [km/h]	92.89	5.82	28.61	19.6
Flujo [veh/min]	25.46	10.68	33.09	10.88
Veh. livianos [%]	71.5	10.7	74.9	9.4
Veh. pesados [%]	27.3	10.6	23.4	8.2

La Figura 5.9 muestra cómo se distribuye la densidad de tráfico en la muestra, distinguiendo entre cada estado identificado. En primer lugar, se observa claramente que los mayores valores de esta variable se logran durante el período de congestión, lo que es consistente con que este atributo mide la cantidad de vehículos por unidad de longitud. Se observa que la densidad se encuentra distribuida como una campana durante el período de flujo libre, con una cola más larga hacia la derecha. Por otro lado, la distribución durante el período de congestión es bimodal, con una predominancia de los valores cercanos a los $100 \text{ veh}/\text{km}$. Este resultado indica que futuras investigaciones podrían considerar una adaptación del Algoritmo 1 para incorporar un tercer estado de tráfico. Se estima que este correspondería a la transición entre el flujo libre y la congestión, pues consisten en aquellos que, estando catalogados como congestionados, revelan una menor saturación de las vías.

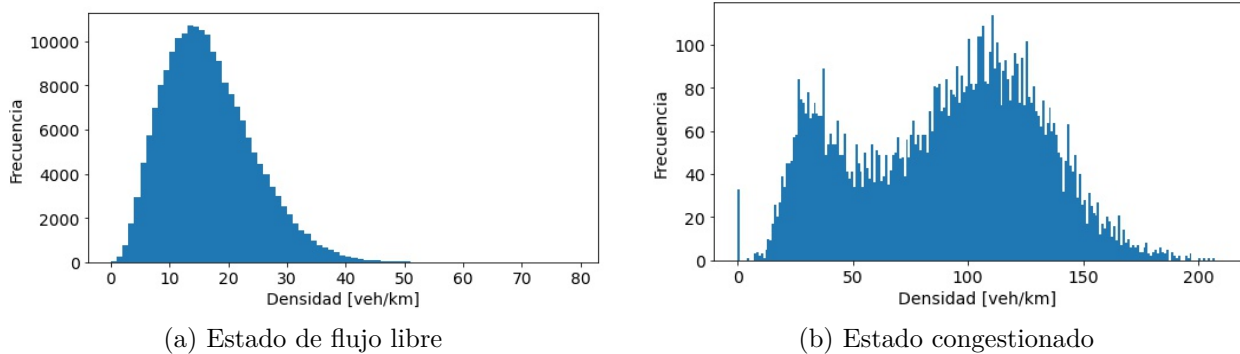
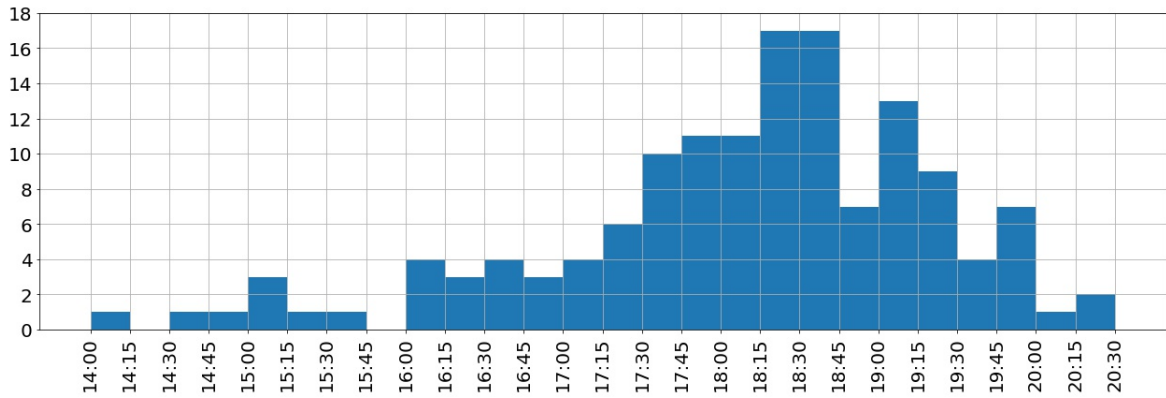
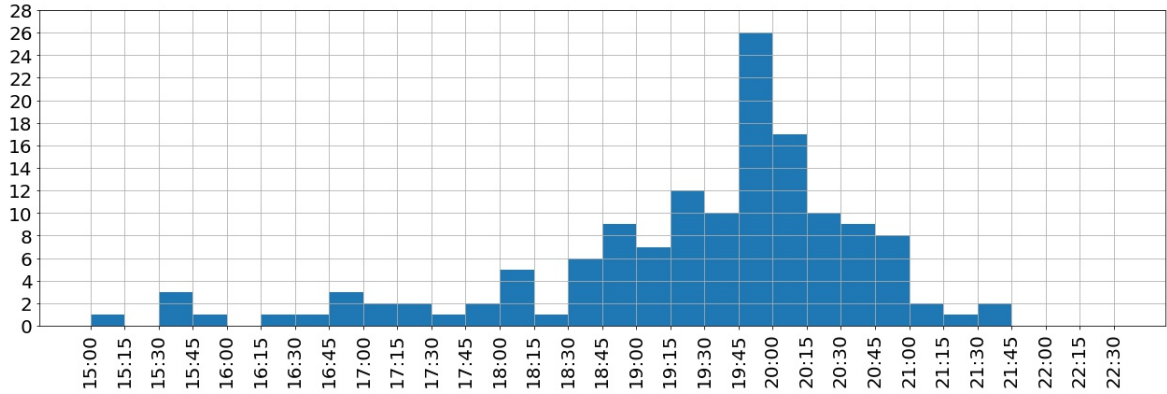


Figura 5.9: Distribución de la densidad de tráfico según el estado de congestión identificado.

El período de congestión suele concentrarse en la hora punta de la tarde, como se aprecia en la Figura 5.10. Este comportamiento es el esperado, pues la vía analizada corresponde al tráfico que se dirige hacia las afueras de la ciudad en dirección al sur. No obstante, aunque la congestión suele iniciar en torno a las 18:30 y finalizar cerca de las 20:00, la muestra recoge casos en los que este estado de tráfico ocurre en otras horas de la tarde.



(a) Distribución inicio de congestión



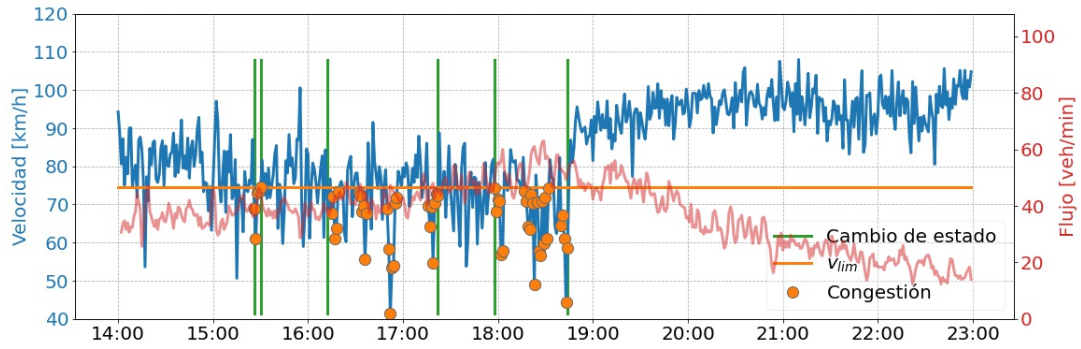
(b) Distribución término de congestión

Figura 5.10: Distribución de la hora de inicio y de término del período de congestión de cada día en el pórtico PA20.

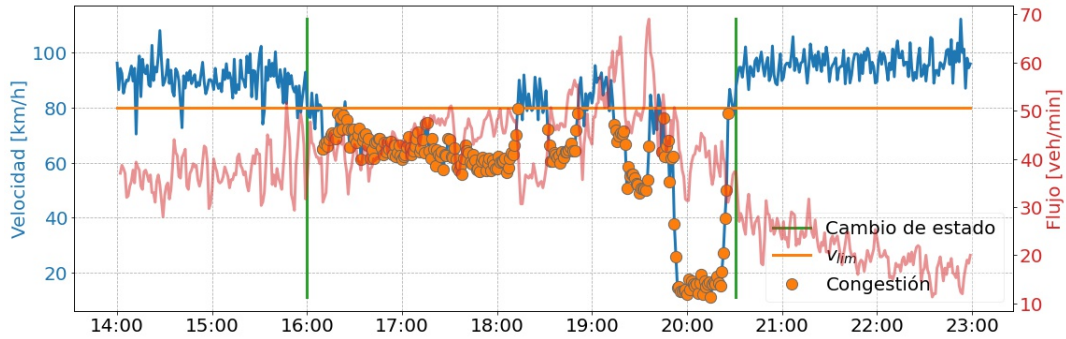
5.3.1. Ejemplo gráfico del etiquetado del estado de tráfico

En la Figura 5.11 se muestran ejemplos del resultado de la clasificación de estados de tráfico. El umbral de velocidad utilizado como límite en el algoritmo empleado ronda los 80 km/h y permite identificar diferentes tipos de eventos de congestión. En el caso de la Figura 5.11.a, existen pequeños episodios en los que hay una disminución de la velocidad por debajo de los 75 km/h . En la Figura 5.11.b se aprecian lapsos con una congestión más persistente, aunque son distintos respecto a la velocidad alcanzada. Uno de ellos alcanza valores en torno a los 20 km/h , lo que es el caso general en la Figura 5.11.c durante el período de mayor tráfico. De esta manera, la metodología desarrollada permite distinguir una pluralidad de interrupciones en el tráfico de flujo libre en cuanto a su prolongación e intensidad.

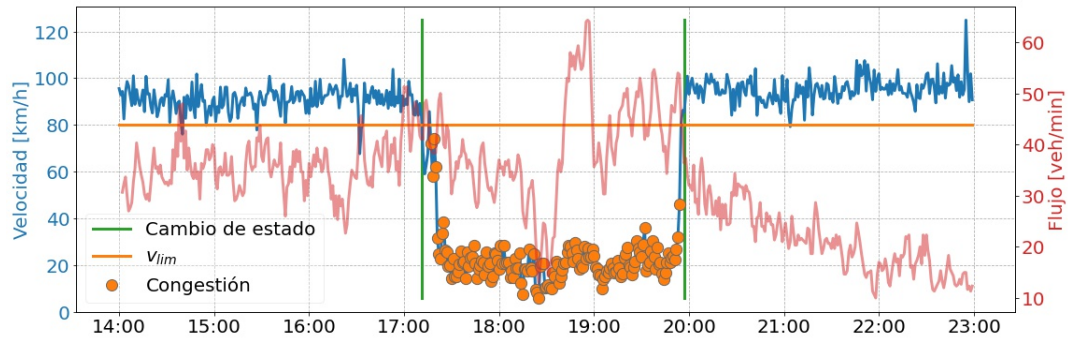
Si bien es posible notar una correlación negativa entre el flujo y la velocidad, no es claro que la medición de la cantidad de vehículos en la vía permita identificar por sí sola un período de congestión. Aún más, en ocasiones el flujo disminuye durante la hora de mayor tráfico. Este es el caso del 12 de marzo de 2018 en la Figura 5.11.c alrededor de las 18 horas, por lo que la velocidad sigue siendo la variable de mayor interés al momento de realizar una búsqueda empírica de períodos de congestión.



(a) 27 de diciembre de 2016



(b) 21 de noviembre de 2017



(c) 12 de marzo de 2018

Figura 5.11: Ejemplos del resultado del proceso de identificación de estados de tráfico. Se muestran las observaciones catalogadas como congestionadas, los instantes en los que ocurre el cambio de estado y el umbral v_{lim} empleado en el Algoritmo 1.

5.4. Comparación y evaluación de modelos de predicción

5.4.1. Clasificación de congestión futura

La primera tarea de predicción corresponde a pronosticar el estado de tráfico que se observará dentro de cinco minutos en el pódico PA20. Las clases consideradas corresponden al

flujo libre y a la congestión, por lo que este procedimiento constituye una tarea de clasificación binaria. Los modelos a considerar se detallan en la Sección 4.2 y sus resultados se muestran en las Tablas 5.4 y 5.5.

En primer lugar, la Tabla 5.4 señala un resultado crucial respecto al uso de información pasada para la predicción de congestión. Todos los modelos revisados obtienen su mejor resultado cuando no se utiliza ningún rezago de las variables, lo que sugiere que este tipo de atributos es poco útil para predecir congestión en el corto plazo. Se considera que este hecho puede explicarse por el carácter abrupto de la transición entre el flujo libre y la congestión, siendo más valiosa la información presente que aquella registrada en el pasado.

Los modelos que obtienen los mejores resultados son el *random forest* y la red neuronal. La diferencia fundamental entre ambos se aprecia en la Tabla 5.5, en la que se reporta que el *random forest* cuenta con más verdaderos negativos y menos verdaderos positivos. Es decir, este método etiqueta más observaciones bajo el estado de flujo libre, a diferencia de la red neuronal que hace lo contrario al predecir más eventos de congestión. Se optará por usar el modelo de *random forest* para la predicción de congestión, dado que sus cantidades de falsos positivos y negativos se encuentra más equilibrada y, en consecuencia, se obtiene un mejor desempeño respecto al F1-Score. La falencia principal de la red neuronal es que cuenta con una mayor proporción de falsos positivos, por lo que disminuye su precisión y esto penaliza al F1-Score.

El árbol de decisión tiene un comportamiento similar a los métodos anteriormente revisados, pero cuenta con un peor desempeño en todas las métricas. Los modelos de Naive-Bayes son notoriamente peores y se caracterizan por sobrerrepresentar el estado de tráfico congestionado. En consecuencia, pronostican una alta cantidad de falsos positivos y se afecta tanto a la precisión como al F1-Score de los modelos.

Tabla 5.4: Métricas de desempeño de los modelos de predicción de congestión considerando el uso de rezagos con mejores resultados. El 0 indica que no se usan rezagos durante la predicción.

	Naive-Bayes (Gaussiano)	Naive-Bayes (Bernoulli)	Árbol de decisión	<i>Random forest</i>	Red neuronal
Mejor rezago	0	0	0	0	0
F1-Score	0.543	0.359	0.823	0.905	0.857
Precisión	0.38	0.226	0.784	0.904	0.79
Recall	0.949	0.873	0.867	0.907	0.936
Accuracy bal.	0.943	0.876	0.929	0.952	0.963

Tabla 5.5: Matrices de confusión (en porcentaje) de los modelos de predicción de congestión. La clase 0 corresponde a observaciones catalogadas bajo flujo libre y la clase 1 a aquellas congestionadas.

Clase	Naive-Bayes (Gaussiano)		Naive-Bayes (Bernoulli)		Árbol de decisión		<i>Random forest</i>		Red neuronal	
	0	1	0	1	0	1	0	1	0	1
0	95.05	6.04	84.4	11.69	95.16	0.93	95.72	0.38	95.12	0.97
1	0.2	3.71	0.5	3.41	0.52	3.39	0.36	3.54	0.25	3.66

Los detalles de la especificación de cada modelo pueden ser consultados en el Anexo B.1. Algunos hiperparámetros fueron elegidos mediante validación cruzada según los procedimientos descritos en la Sección 4.4.2, tras lo que resultan las siguientes configuraciones:

- La profundidad del árbol de decisión se fijó en 20 niveles.
- Los árboles entrenados por el *random forest* tendrán una profundidad de 12 niveles. La diferencia respecto al modelo del punto anterior da cuenta de que el proceso de *bagging* exige menos reglas de clasificación.
- La red neuronal contará con 8 capas y 150 unidades en cada una de ellas.

Cabe mencionar que los valores aquí reportados corresponden al caso en que no se usan rezagos, dado que dicha configuración es la que entrega un mejor desempeño. Los resultados de la validación cruzada ante distintos hiperparámetros pueden consultarse en el Anexo B.2. Las métricas resultantes de utilizar información pasada se reportan en el Anexo B.3.

Como fue visto en la Sección 5.3, existe un desbalance en la cantidad de observaciones de cada estado presentes en la muestra tras la aplicación del algoritmo de etiquetado del estado de congestión. En consecuencia, se opta por asignar pesos a cada clase para equilibrar la importancia de la que tiene menor presencia (estado congestionado) en relación a la clase mayoritaria (estado de flujo libre). Este procedimiento es descrito metodológicamente en la Sección 4.5 y es aplicado sobre los modelos de árbol de decisión, *random forest* y red neuronal.

Los ponderadores usados corresponden a 0.523 para el flujo libre y 11.277 para las observaciones congestionadas, fijando una proporción 1: 20 a partir de que el estado congestionado representa a, aproximadamente, el 5% de los datos del pórtico PA20 en la muestra. Los resultados de los modelos que recibieron este tratamiento dan cuenta de su eficacia al contar con una alta *accuracy* balanceada (o exactitud balanceada).

5.4.1.1. Importancia de los atributos en la predicción de congestión

El modelo seleccionado consiste en un *random forest* y una de las características de este tipo de métodos es que permiten obtener la contribución exacta de cada atributo en la reducción de la función de pérdida, la que en este caso corresponde a la impureza de Gini. Una vez normalizado, este valor suele denominarse importancia de Gini.

La Tabla 5.6 muestra los 10 atributos más influyentes en el entrenamiento del modelo. Las variables más relevantes son las que entregan información del tráfico, como la velocidad o la densidad. Además, la información del pórtico para el que se pronostica el estado de tráfico es la más relevante, pues otros como el PA22 o PA24 tienen una importancia de Gini de menos del 2%.

Destaca la importancia de la velocidad de los vehículos pesados y de las motocicletas, a pesar de que la predicción se enfoca en los vehículos livianos. Esto puede ser un indicador de que la presencia de estos otros dos tipos de vehículos es incidente en el estado de tráfico de las vías, a pesar de su menor presencia. En particular, los vehículos pesados como camiones

o buses utilizan un mayor espacio en cada pista y suelen transitar a menor velocidad según lo visto en la Sección 5.2.

En general, los atributos que se refieren al día de la semana no cobran relevancia en el modelo al contar todos estos con una importancia del 0.1 % o menos. La única excepción la constituye el día viernes, que aporta en un 0.7 % a la reducción de la impureza a raíz de su comportamiento único por su mayor flujo y ocurrencia de eventos de congestión. Además, la hora de medición tiene una importancia de Gini del 0.3 %, por lo que se estima que este atributo pierde relevancia ante el uso de las variables de tráfico que reportan información de la situación actual. Los resultados para el resto de atributos pueden ser consultados en el Anexo C.

Tabla 5.6: Importancia de Gini de los 10 atributos que más aportan a la reducción de la impureza del modelo de *random forest* sin rezagos.

Importancia	Atributo
20.2 %	Velocidad vehículos pesados (PA20)
16.4 %	Estado de tráfico actual (PA20)
12 %	Velocidad vehículos livianos (PA20)
11.6 %	Velocidad motocicletas (PA20)
9.6 %	Densidad vehículos livianos (PA20)
5.7 %	Densidad vehículos pesados (PA20)
4.2 %	Desviación estándar velocidad vehículos livianos (PA20)
2 %	Velocidad vehículos livianos (PA24)
1.8 %	Desviación estándar velocidad vehículos pesados (PA20)
1.8 %	Densidad vehículos livianos (PA22)

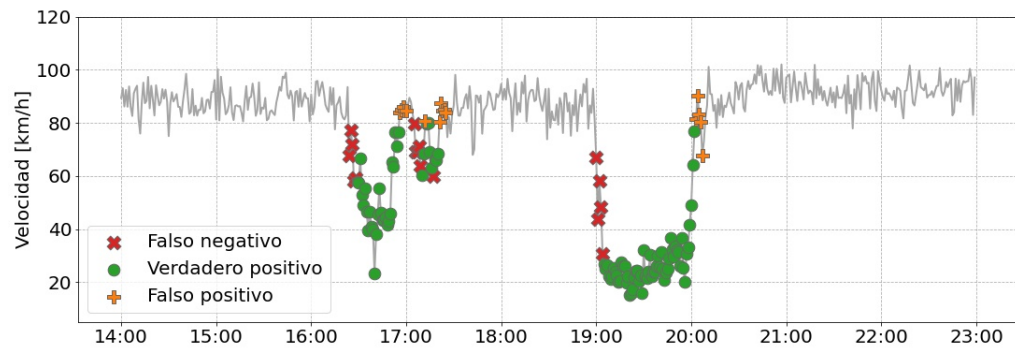
Los resultados expuestos indican que las correlaciones temporales y espaciales no adquieren mayor relevancia en la predicción de congestión, al menos para el caso estudiado. Por otro lado, la baja importancia de los pódicos anteriores se puede explicar por la amplia distancia que existe entre ellos (el pódico PA22 está a 2.6 km, el PA24 a 6.6 km y el PA26 a 9.2 km), lo que se agrava por la existencia de entradas y salidas a la autopista que alteran el flujo que esta mantiene.

5.4.1.2. Ejemplo gráfico para la predicción de congestión

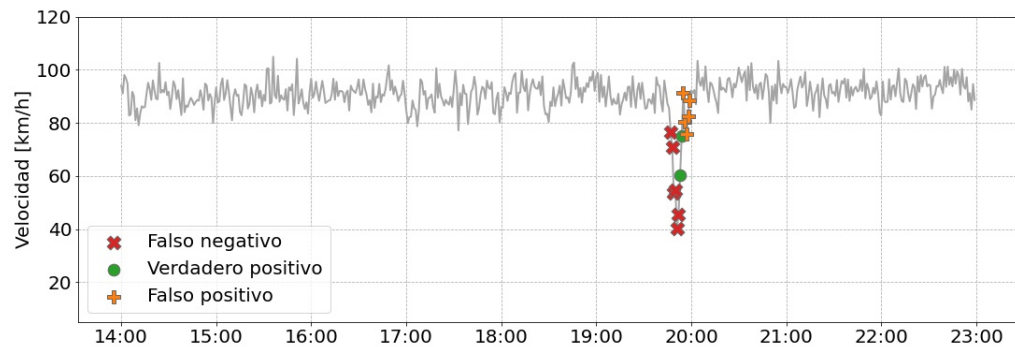
La Figura 5.12 da cuenta de algunos resultados obtenidos al predecir el estado de congestión sobre una base de testeo y utilizando el modelo de *random forest* sin rezagos. En todos los casos, es posible observar un desfase del método en el reconocimiento del inicio y del término del período congestionado. Esta tardanza es de alrededor de 5 minutos, similar al horizonte de predicción, y puede deberse a la naturaleza abrupta de los cambios en la velocidad.

En línea con lo anterior, el modelo entrenado enfrenta dificultades frente a casos en los que la disminución de la velocidad es breve, como el presentado en el 9 de febrero de 2018 (Figura 5.12.b). Su mayor fortaleza la tiene en períodos prolongados de congestión, como el observado el 12 de marzo de 2018 (Figura 5.12.c), lo que indica que la predicción es confiable

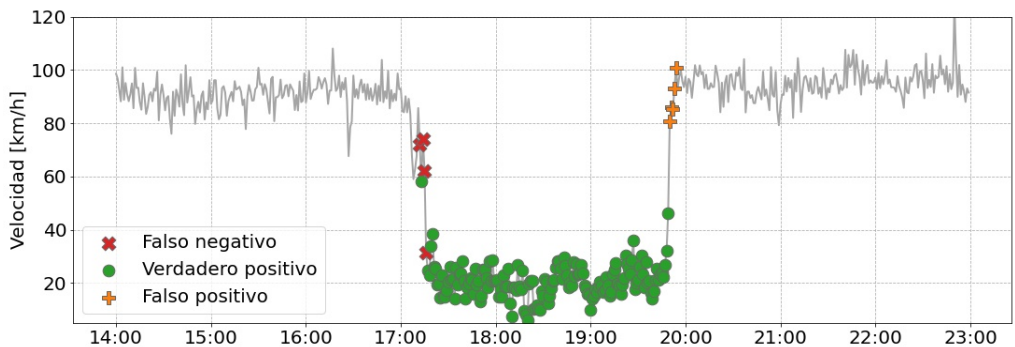
una vez que se estabiliza la transición entre estados de tráfico.



(a) 20 de diciembre de 2017



(b) 9 de febrero de 2018



(c) 12 de marzo de 2018

Figura 5.12: Ejemplos del desempeño del mejor método de predicción de congestión (*random forest* sin rezagos).

5.4.2. Estimación de velocidad media futura

Los procedimientos previos para predecir el estado de congestión permiten obtener un atributo que sirve de entrada a los pronósticos de velocidad, prescindiendo de utilizar el Algoritmo 1 referido en la Sección 4.1. El método de *random forest* para determinar el estado de tráfico puede ser entrenado a partir de información pasada, evitando el problema de *target leakage* que se definió en dicha sección.

Los modelos de aprendizaje automático tienen como objetivo la predicción de la velocidad en el p rtico PA20 con un horizonte de tiempo de cinco minutos. Estos corresponden a un  rbol de regresi n, un *random forest* basado en este tipo de  rboles y una red neuronal MLP. Cada uno de los m todos anteriores ser  entrenado en cuatro enfoques distintos, los que dependen de el uso que le es dado al dato de congesti n obtenido en la secci n anterior.

El primer enfoque, ‘‘Sin congesti n’’, prescinde de utilizar la predicci n de congesti n de velocidad y sirve como caso base para notar el efecto que tiene su uso. Por el contrario, el enfoque ‘‘Con congesti n’’ emplea dicho dato como una variable de entrada m s en el pron stico de velocidad. La tercera estrategia, denominada ‘‘Separaci n por estado de tr fico’’, funciona en dos etapas. En primer lugar, se obtiene la predicci n del estado de tr fico y la muestra se separa en dos subconjuntos seg n el resultado. Posteriormente, se entrena un modelo para cada subconjunto y se obtiene un m todo para obtener predicciones de velocidad durante la etapa de flujo libre y otro para estados congestionados.

El  ltimo enfoque, se alado como ‘‘Con congesti n (algoritmo)’’ utiliza la etiqueta de estado de tr fico obtenida mediante el Algoritmo 1. Es importante recalcar que este enfoque no admite un uso pr ctico, ya que adolece del problema de *target leakage* descrito anteriormente. No obstante, se justifica su an lisis te rico al servir de *benchmark* respecto al enfoque ‘‘Con congesti n’’, pues las diferencias entre ambos dan cuenta del efecto que tiene la calidad de los modelos que pronostican el estado de tr fico (Secci n 5.4.1) sobre la predicci n de velocidad.

Tabla 5.7: M tricas de desempe o de los modelos de predicci n de velocidad con el uso de rezagos que provee mejores resultados (ver Tabla 5.8). El MAE est  medido en km/h y el MAPE es un porcentaje adimensional.

Enfoque	�rbol de regresi�n		<i>Random forest</i>		Red neuronal MLP	
	MAE	MAPE	MAE	MAPE	MAE	MAPE
Sin congesti�n	4.42	5.8	4.31	5.62	4.23	5.55
Con congesti�n	4.33	5.74	4.22	5.53	4.17	5.64
Separaci�n	4.3	5.74	4.21	5.57	4.13	5.46
Con cong. (algoritmo)	4.13	5.24	4.04	5.15	3.97	5.12

Tabla 5.8: Configuraciones de uso de rezagos con mejores desempe os

Enfoque	�rbol de regresi�n	<i>Random Forest</i>	Red neuronal MLP
Sin congesti�n	10	5	30
Con congesti�n	30	30	5
Separaci�n	15	15	15
Con cong. (algoritmo)	30	30	15

La Tabla 5.7 da cuenta de los resultados obtenidos para los diferentes enfoques ante la utilizaci n de los m todos propuestos. La m trica a utilizar para comparar los resultados consistir  en el MAE. Las especificaciones de los modelos de predicci n de velocidad se indican en el Anexo D.1, mientras que el desempe o ante el uso de distintos hiperpar metros se indican en el anexo D.2.

Por otro lado, la Tabla 5.8 indica el horizonte temporal de rezagos que provee mejores

resultados para cada modelo y enfoque. El desempeño para otras configuraciones del uso de información pasada se reporta en el Anexo D.3.

Uno de los resultados más claros es que el enfoque que no incorpora información sobre el estado de tráfico (“Sin congestión”) tiene los peores resultados al compararse con el resto de estrategias. Esta situación se da en todos los modelos de aprendizaje automático, por lo que se evidencia que el conocimiento del nivel de congestión es un insumo útil para la predicción de velocidad. Al comparar los enfoques “Con congestión” y “Separación”, se observa que este último es el que entrega un error menor.

El uso del estado de congestión obtenido a través del Algoritmo 1 no puede usarse para predecir en tiempo real, pero su uso teórico representa los mejores resultados al emplearlo como *input* en la predicción de velocidad. El desarrollo de mejores modelos de predicción de congestión permitiría cerrar la brecha entre esta configuración y el enfoque “Con congestión”. De la misma manera, es posible que un método de predicción de congestión con menor error también favorezca el desempeño del enfoque de separación según el estado de tráfico.

Con respecto a la comparación entre los modelos de aprendizaje automático usados para el pronóstico de velocidad, la red neuronal es la que presenta un mejor desempeño ante todos los enfoques estudiados. Cabe destacar que, al igual que en los resultados de la predicción de congestión, el *random forest* supera al árbol de regresión.

Los resultados descritos previamente se mantienen, en general, al observar el MAPE. La excepción ocurre en el enfoque “Con congestión”, pues esta métrica indica un mejor desempeño del *random forest* por sobre la red neuronal. Por el carácter relativo del MAPE, esto implica que el *random forest* concentra su error en observaciones de menor velocidad, es decir, con mayor probabilidad de estar en un período de congestión.

Según la Tabla 5.8 se recomienda el uso de rezagos en todos los casos, pero no existe un patrón claro para cada enfoque. Por ejemplo, los árboles de decisión y *random forest* poseen valores similares y requieren de menos información pasada cuando no se utiliza ninguna información sobre el estado de congestión, pero la red neuronal no mantiene el mismo comportamiento. En cualquier caso, la incorporación de datos anteriores sí ayuda a obtener mejores predicciones, a diferencia de lo ocurrido en la clasificación de congestión mediante estos mismos métodos de aprendizaje automático.

Adicionalmente, se calcula el desempeño de modelos de referencia con enfoques más sencillos, cuyos resultados se informan en la Tabla 5.9. La predicción ingenua asume que la predicción en $t + 5$ es idéntica a la observada en el instante t , mientras que el promedio histórico calcula la media de las velocidades pasadas en cada minuto y lo utiliza como pronóstico. Estos dos *benchmark* tienen peores resultados respecto a los obtenidos mediante los métodos basados en árboles o redes neuronales, es decir, poseen más dificultades para identificar las fluctuaciones de corto plazo y la variabilidad diaria de la velocidad.

Otro modelo de referencia consiste en el promedio de velocidad por estado de tráfico. Se utiliza la predicción del nivel de congestión al estimar que la velocidad será 92.89 km/h si se pronostica que el estado será de flujo libre o 28.61 km/h si habrá congestión. Este modelo utiliza parcialmente la información de los modelos detallados en la Sección 5.4.1 y, a pesar de ser superado claramente, se cree que este hecho contribuye a que sea el *benchmark* más

cercano a los resultados de las técnicas de aprendizaje automático.

Tabla 5.9: Métricas de desempeño de enfoques de referencia. El MAE está medido en km/h y el MAPE es un porcentaje adimensional.

Enfoque	MAE	MAPE
Predicción ingenua	5.49	6.87
Promedio histórico	7.13	14.49
Promedio por estado de tráfico	4.9	6.54

Por último, la Figura 5.13 muestra el promedio por hora de los residuos obtenidos al predecir la velocidad futura mediante la red neuronal. Se aprecia que el error promedio es más estable en períodos fuera de punta, pues el lapso entre las 17:30 y las 21 horas cuenta con mayor variabilidad debido a los cambios abruptos en la velocidad de los vehículos y además muestra los promedios de residuos más altos del día. La presencia de cambios en el estado de tráfico resulta ser la mayor fuente de variabilidad en la muestra, lo que se refleja en los errores de los modelos de predicción.

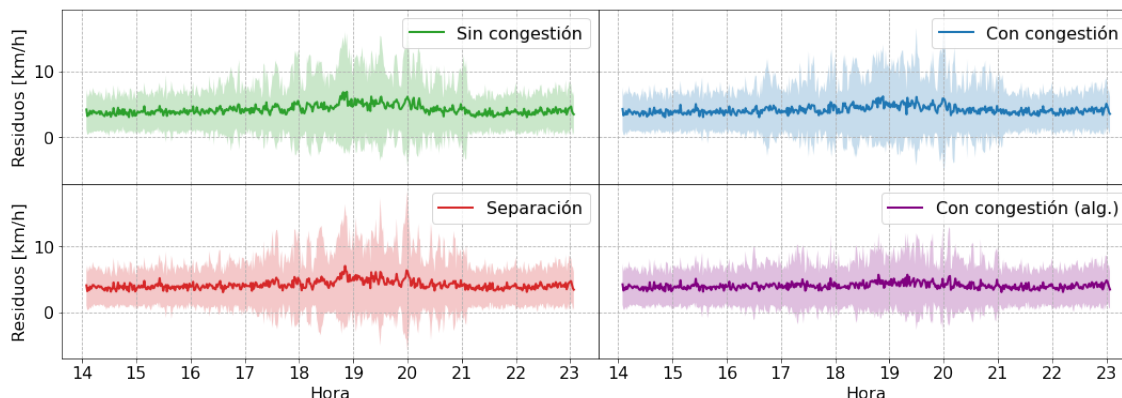


Figura 5.13: Promedio del valor absoluto de los residuos por hora para la predicción realizada mediante una red neuronal MLP. El área sombreada corresponde al rango de dos desviaciones estándar centrado en el promedio.

5.4.2.1. Importancia de los atributos en la predicción de velocidad

Si bien el método de *random forest* no es el que presenta un mejor desempeño, este modelo puede considerarse para estudiar las importancias relativas de cada atributo en la predicción. Este procedimiento se ha empleado en la literatura, por ejemplo, por Basso et al. [16] en la predicción de accidentes en una autopista.

Se revisarán las importancias dadas por este método de aprendizaje automático para el caso de “Separación”, en vista de las ventajas provistas por este enfoque. La Tabla 5.10 muestra las importancias de las variables en el modelo entrenado con las observaciones congestionadas y la Tabla 5.11 las asociadas al flujo libre.

La principal diferencia radica en que la velocidad de los vehículos livianos concentra casi toda la importancia al predecir bajo un estado de congestión, superando el 80%. Debido a la

alta variabilidad durante este contexto, lo observado en el mismo instante en que se ejecuta la predicción cobra mayor relevancia que la información pasada y la situación de los otros pódicos.

Por otro lado, las importancias de las variables están más equilibradas ante un escenario de flujo libre. Se valora más el uso de información pasada y de pódicos anteriores en la autopista, lo que señala una mayor relevancia de las correlaciones espacio-temporales cuando la velocidad mantiene valores estables durante un estado estacionario. La ausencia de cambios abruptos durante este contexto le resta poder explicativo a la información contingente. Por el contrario, se estima que el conocimiento de la situación del tráfico “aguas arriba” o en intervalos de tiempo anteriores cobra importancia debido a que son más útiles en la determinación de tendencias de mediano plazo.

Tabla 5.10: Importancia de Gini de los atributos del modelo de *random forest* para la predicción de velocidad en la submuestra congestionada. Se reportan aquellos con una importancia mayor al 1 %.

Importancia	Atributo
80.3 %	Velocidad vehículos livianos (PA20)
5 %	Velocidad vehículos livianos hace 10 minutos (PA20)
3.1 %	Velocidad motocicletas (PA20)
1.6 %	Velocidad vehículos pesados (PA20)
1.1 %	Velocidad vehículos livianos hace 15 minutos (PA20)

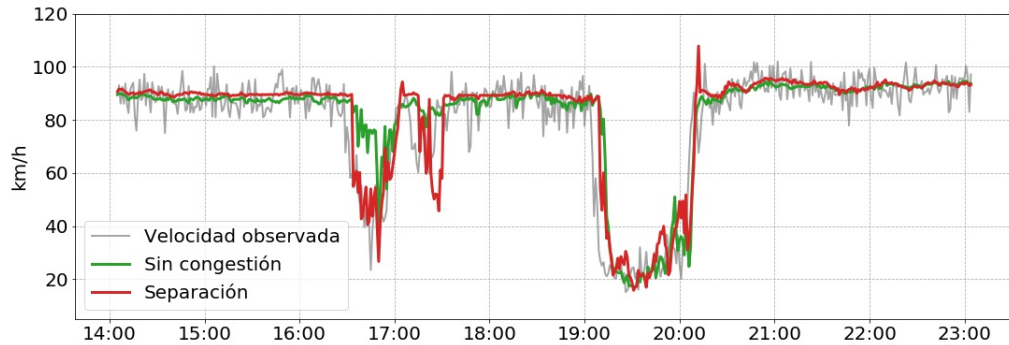
Tabla 5.11: Importancia de Gini de los atributos del modelo de *random forest* para la predicción de velocidad en la submuestra de flujo libre. Se reportan aquellos con una importancia mayor al 1 %.

Importancia	Atributo
37 %	Densidad vehículos pesados hace 5 minutos (PA24)
20.4 %	Velocidad vehículos livianos hace 5 minutos (PA20)
4.7 %	Velocidad vehículos livianos hace 10 minutos (PA20)
3.1 %	Velocidad vehículos livianos hace 5 minutos (PA26)
1.8 %	Velocidad vehículos livianos hace 5 minutos (PA24)
1.7 %	Velocidad vehículos livianos (PA20)
1.6 %	Velocidad promedio vehículos livianos hace 15 minutos (PA20)

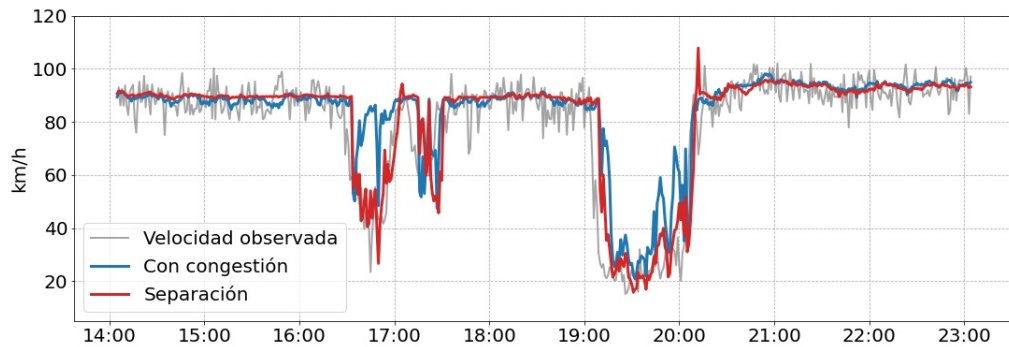
5.4.2.2. Ejemplo gráfico para la predicción de velocidad

En la Figura 5.14 se muestra un ejemplo gráfico de la aplicación de la red neuronal, que es el modelo de aprendizaje automático con mejor desempeño. El resultado es similar para todos los enfoques en el régimen de flujo libre, por lo que las principales diferencias surgen en el estado congestionado. Las Figs. 5.14.a y 5.14.b indican que el enfoque que separa la muestra según el estado de tráfico realiza una mejor estimación en períodos de congestión en los que la velocidad disminuye de forma moderada, como el observado entre las 16:30 y las 17 horas. Al revisar el período de congestión entre las 19 y las 20 horas, la Figura 5.14.b señala que el desarrollo de métodos específicos para cada estado de tráfico da resultados más certeros.

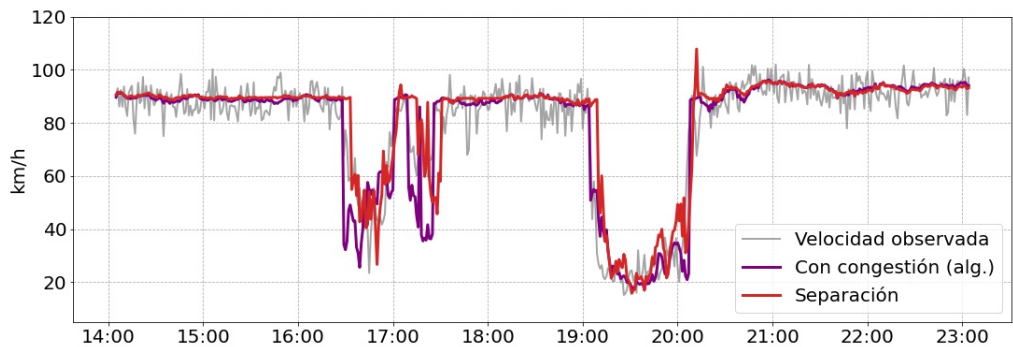
Cabe destacar que todos los enfoques muestran un desfase en su predicción al cambiar el estado de tráfico, a excepción del que utiliza la etiqueta que resulta de la aplicación del Algoritmo 1. En la Figura 5.14.c se aprecia que este enfoque es más preciso ante las transiciones de estado, lo que permite inferir que la precisión en la detección precisa de los cambios de fase es un factor relevante a la hora de predecir la velocidad de los vehículos con menor error.



(a) Sin información sobre el estado de tráfico



(b) Uso del atributo del estado de tráfico obtenido con el clasificador de la Sección 5.4.1



(c) Uso del atributo del estado de tráfico obtenido mediante el Algoritmo 1

Figura 5.14: Ejemplo del desempeño de la red neuronal en la predicción de velocidad. Se muestran los resultados para el día 20 de diciembre de 2017 y se compara al mejor enfoque (“Separación”) ante al resto de estrategias.

5.4.2.3. Regla de velocidad fija para identificar congestión

El mecanismo para identificar las situaciones de congestión utilizado por Véjar [23] contempla el establecimiento de un umbral fijo que discierne entre los estados de tráfico. De esta manera, si la velocidad media observada supera al umbral se asume que hay flujo libre y, en caso contrario, se establece que se está bajo congestión. El umbral se establece como la velocidad promedio de las observaciones que poseen un flujo por sobre el percentil 95 en la muestra.

Para los datos utilizados, la regla anterior supone un umbral de 77.87 km/h . La Tabla 5.12 indica el desempeño de los enfoques al considerar este límite fijo para clasificar los estados de tráfico. Los resultados son consistentes en mostrar un mejor desempeño de la separación de modelos según la presencia de congestión, como fue indicado por Véjar [23].

Al contrastar con la Tabla 5.7, se aprecia que los resultados mejoran al incorporar el mecanismo de identificación y predicción de congestión propuesto en este trabajo. El desempeño al emplear un umbral fijo es similar al enfoque “Sin congestión” reportado previamente.

Tabla 5.12: Métricas de desempeño de los modelos de predicción de velocidad con una regla fija para identificar los períodos de congestión.

Enfoque	Árbol de regresión		<i>Random forest</i>		Red neuronal MLP	
	MAE	MAPE	MAE	MAPE	MAE	MAPE
Con congestión	4.45	5.87	4.34	5.66	4.26	5.72
Separación	4.39	5.8	4.3	5.66	4.22	5.55

La comparación de los métodos de identificación del estado de tráfico indica que no sólo es recomendable considerar esta información, sino que metodologías más certeras o refinadas ofrecen un mayor aporte a la precisión en la predicción de velocidad. En particular, la aplicación de un método que basa la clasificación en información reciente, como el del Algoritmo 1, entrega mejores resultados que un umbral calculado a partir de información histórica de las variables de tráfico.

Capítulo 6

Conclusiones

El principal hallazgo de este trabajo consiste en que la metodología de identificación de congestión propuesta provee una mejora en el desempeño de los modelos de predicción de velocidad. Las comparaciones efectuadas son consistentes en mostrar las ventajas del algoritmo diseñado ante distintos enfoques y métodos de aprendizaje automático.

La implementación de la identificación de estados de tráfico es exitosa en distinguir al período congestionado del de flujo libre, pues se muestran claras diferencias en las variables que caracterizan a cada uno. En efecto, su aplicación es de utilidad para el entrenamiento de dos modelos de predicción independientes, el que provee mejores resultados respecto a la incorporación de la clasificación como un atributo de entrada binario. En cualquier caso, emplear del estado de tráfico entrega resultados más acertados respecto al caso base en que esta información no se toma en cuenta.

En atención a los algoritmos de aprendizaje automático, se observa que lo más conveniente es utilizar un *random forest* para pronosticar congestión, sin incorporar rezagos. Para la estimación de la velocidad futura, la red neuronal provee una mejor predicción y su mejor desempeño se da al considerar información pasada. Se difiere en el horizonte de tiempo óptimo de los rezagos, pero el trabajo desarrollado sugiere su uso en todos los casos.

La separación de los modelos de aprendizaje automático también permite observar las importancias que cada uno asigna a los atributos durante la predicción. Las observaciones congestionadas dependen en gran medida de la información reciente (sobre un 80%), descartando el uso intensivo de correlaciones espaciales o temporales. En cambio, la menor variabilidad de la velocidad en el flujo libre provoca que los métodos se valgan de las tendencias de mediano plazo que pueden rescatarse de la información pasada o de otros pórticos “aguas arriba”.

Por otro lado, los resultados indican que el procedimiento de clasificación de estados de tráfico supera al uso de una regla que establece un umbral fijo de la velocidad para todos los días en la muestra. La metodología utilizada en esta tesis calcula este umbral de forma específica para cada día y privilegia la información reciente de la velocidad observada, siendo más precisa en el hallazgo de los períodos congestionados y de los cambios de fase. Lo anterior se mantiene aún cuando se habilite su uso en tiempo real mediante un método de predicción

de congestión. Si bien esto se testea en el pronóstico de velocidad, la utilización del algoritmo puede extenderse a otras aplicaciones de esta naturaleza.

La estrategia seguida fue ejecutada con más de 380 días de datos y sin emplear información concerniente a accidentes de tráfico, eventos masivos o condiciones climáticas. En conclusión, el método es replicable sin requerir de datos ajenos a los medidos por un sensor ubicado en una vía. El contexto contempla una amplia distancia entre los pódicos con entradas o salidas de la autopista, por lo que no se requiere de un seguimiento exhaustivo vehículo a vehículo y puede ser aplicado sobre métricas agregadas del tráfico.

6.1. Trabajo futuro

A pesar de que la predicción de congestión es exitosa con una exactitud balanceada del 96 %, resulta imprecisa durante las transiciones entre los estados de tráfico, sobre todo cuando estos ocurren en un lapso breve. Esto se traduce en un desfase cercano a los cinco minutos en la identificación de la congestión futura. El uso de pronósticos más exactos sobre cambios de fase puede entregar información de mayor calidad sobre la situación de la vía, lo que repercutiría en mejores desempeños de los métodos de predicción de velocidad.

Una alternativa consiste en el empleo más exhaustivo de rezagos de corto plazo (menores a cinco minutos), pero también se puede considerar la evaluación de modelos de aprendizaje automático más refinados que están presentes en la literatura. Entre ellos, destaca el uso de las redes neuronales recurrentes mediante la implementación de celdas *Long Short-Term Memory* (LSTM) o *Gated Recurrent Units* (GRU).

También se puede plantear la adopción de un modelo de tres fases de tráfico, teniendo en cuenta la naturaleza bimodal que se observa en la distribución de la densidad bajo el estado de congestión. Esta diferenciación permitiría separar a las observaciones correspondientes a la transición entre estados, las que podrían ser estudiadas en su especificidad.

Finalmente, este trabajo se desarrolla en el contexto de una autopista urbana en la que no se registran detenciones de los vehículos atribuibles a intersecciones de vías. No es claro que la metodología propuesta conserve sus buenos resultados ante otro escenario, como el que ocurre en una red de calles urbanas interconectadas. El análisis del efecto de la congestión sobre las predicciones de velocidad puede ser explorado en aquel escenario, con el cuidado de incorporar la compleja naturaleza de las correlaciones espaciales y temporales entre las calles.

Bibliografía

- [1] M. L. Manheim, *Fundamentals of Transportation System Analysis*, vol. 1. Boston: MIT Press, 1979.
- [2] Parlamento Europeo, Consejo de la Unión Europea, “Directive 2010/40/eu of the european parliament and of the council,” Julio 2010.
<https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX%3A32010L0040>.
- [3] B. D. Greenshields, D. Schapiro, and E. L. Ericksen, *Traffic Performance at Urban Street Intersections*. Technichal Report N° 1, Bureau of Highway Traffic, Yale University, 1947.
- [4] B. S. Kerner, *Introduction to Moder Traffic Flow Theory and Control: The Long Road to Three-Phase Traffic Theory*. Springer, 2009.
- [5] C. F. Daganzo, *Fundamentals of Transportation and Traffic Operations*. Pergamon, 1997.
- [6] Transportation Research Board, *Highway Capacity Manual*. National Research Council, 6 ed., 2010.
- [7] B. S. Kerner, “Experimental characteristics of traffic flow for evaluation of traffic modelling,” *Physical Review E*, vol. 53, pp. R1297–R1300, February 1996.
- [8] B. S. Kerner and H. Rehborn, “Experimental properties of complexity in traffic flow,” *Physical Review E*, vol. 53, pp. R4275–R4278, May 1996.
- [9] B. Persaud, S. Yagar, and R. Brownlee, “Exploration of the breakdown phenomenon in freeway traffic,” *Transportation Research Record*, vol. 1634, no. 1, pp. 64–69, 1998.
- [10] T. M. Mitchell, *Machine Learning*. USA: McGraw-Hill, Inc., 1 ed., 1997.
- [11] A. Boukerche and J. Wang, “Machine learning-based traffic prediction models for intelligent transportation systems,” *Computer Networks*, vol. 181, p. 107530, 2020.
- [12] A. Ermagun and D. Levinson, “Spatiotemporal traffic forecasting: Review and proposed directions,” *Transport Reviews*, vol. 38, no. 6, pp. 786–814, 2018.
- [13] A. K. Haghighat, V. R. Mouli, P. Chakraborty, Y. Esfandiari, S. Arabi, and A. Sharma, “Applications of deep learning in intelligent transportation systems,” *Journal of Big Data Analytics in Transportation*, vol. 2, pp. 115–145, 08 2020.
- [14] K. Prasad and S. Ramakrishna, “An efficient traffic forecasting system based on spatial data and decision trees,” *International Arab Journal of Information Technology*, vol. 11, pp. 186–194, 03 2014.

- [15] B. Hamner, “Predicting travel times with context-dependent random forests by modeling local and aggregate traffic flow,” in *2010 IEEE International Conference on Data Mining Workshops*, pp. 1357–1359, 2010.
- [16] F. Basso, L. J. Basso, F. Bravo, and R. Pezoa, “Real-time crash prediction in an urban expressway using disaggregated data,” *Transportation Research Part C: Emerging Technologies*, vol. 86, pp. 202–219, 2018.
- [17] J. Rzeszótko and N. Hoa, “Machine learning for traffic prediction,” *Fundamenta Informaticae*, vol. 119, pp. 407–420, 08 2012.
- [18] M. Karlaftis and E. Vlahogianni, “Statistical methods versus neural networks in transportation research: Differences, similarities and some insights,” *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 3, pp. 387–399, 2011.
- [19] E. I. Vlahogianni, “Enhancing predictions in signalized arterials with information on short-term traffic flow dynamics,” *Journal of Intelligent Transportation Systems*, vol. 13, no. 2, pp. 73–84, 2009.
- [20] P. Cai, Y. Wang, G. Lu, P. Chen, C. Ding, and J. Sun, “A spatiotemporal correlative k-nearest neighbor model for short-term traffic multistep forecasting,” *Transportation Research Part C Emerging Technologies*, vol. 62, pp. 21–34, 01 2016.
- [21] A. S. Mihaita, L. Haowen, H. Zongyang, and M.-A. Rizoiu, “Motorway traffic flow prediction using advanced deep learning,” pp. 1683–1690, 10 2019.
- [22] N. G. Polson and V. O. Sokolov, “Deep learning for short-term traffic flow prediction,” *Transportation Research Part C: Emerging Technologies*, vol. 79, pp. 1–17, 2017.
- [23] B. Véjar, “Predicción de velocidad en una autopista urbana utilizando estados de tráfico y modelos de inteligencia artificial,” Master’s thesis, Departamento de Ingeniería Industrial, Universidad de Chile, Santiago, Chile, 2019.
- [24] S. Kaufman, S. Rosset, and C. Perlich, “Leakage in data mining: Formulation, detection, and avoidance,” in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (New York, NY, USA), p. 556–563, Association for Computing Machinery, 2011.
- [25] G. M. Robert Nisbet, John Elder IV, *Handbook of Statistical Analysis and Data Mining Applications*. Academic Press/Elsevier, 2009.
- [26] M. N. Murty and V. S. Devi, *Pattern Recognition: An Algorithmic Approach*. Springer Science, 2011.
- [27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [28] L. Breiman, J. Friedman, R. Olshen, and C. J. Stone, *Classification and regression trees*. Wadsworth International Group, 1984.
- [29] D. Opitz and R. Maclin, “Popular ensemble methods: An empirical study,” *Journal of*

Artificial Intelligence Research, vol. 11, pp. 169–198, December 1999.

- [30] L. Breiman, “Bagging predictors,” *Machine Learning*, vol. 24, pp. 123–140, August 1996.
- [31] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, pp. 5–32, October 2001.
- [32] A. Géron, *Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. Sebastopol, CA: O’Reilly Media, 2017.
- [33] F. Rosenblatt, “The perceptron: A probabilistic model for information storage and organization in the brain,” *Psychological Review*, vol. 65, no. 6, pp. 386–408, 1958.
- [34] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, pp. 533–536, 1986.

Anexo A

Preprocesamiento de los datos

A.1. Datos faltantes en la muestra

Tabla A.1: Porcentaje de valores faltantes por atributo

Valores faltantes	Atributo
0 %	Fecha y hora
0 %	Día de la semana
0 %	<i>Dummy</i> que indica si el día es feriado
0 %	Flujo de vehículos (PA20)
0.02 %	Velocidad vehículos livianos (PA20)
0.05 %	Desviación estándar velocidad vehículos livianos (PA20)
3.18 %	Velocidad vehículos pesados (PA20)
9.82 %	Desviación estándar velocidad vehículos pesados (PA20)
70.94 %	Velocidad motocicletas (PA20)
93.62 %	Desviación estándar velocidad motocicletas (PA20)
0 %	Flujo de vehículos (PA22)
0.16 %	Velocidad vehículos livianos (PA22)
0.17 %	Desviación estándar velocidad vehículos livianos (PA22)
4.6 %	Velocidad vehículos pesados (PA22)
13.16 %	Desviación estándar velocidad vehículos pesados (PA22)
64.22 %	Velocidad motocicletas (PA22)
89.45 %	Desviación estándar velocidad motocicletas (PA22)
0 %	Flujo de vehículos (PA24)
0.28 %	Velocidad vehículos livianos (PA24)
0.29 %	Desviación estándar velocidad vehículos livianos (PA24)
4.22 %	Velocidad vehículos pesados (PA24)
12.2 %	Desviación estándar velocidad vehículos pesados (PA24)
60.32 %	Velocidad motocicletas (PA24)
86.72 %	Desviación estándar velocidad motocicletas (PA24)
0 %	Flujo de vehículos (PA26)
0.18 %	Velocidad vehículos livianos (PA26)
0.19 %	Desviación estándar velocidad vehículos livianos (PA26)
7.15 %	Velocidad vehículos pesados (PA26)

Tabla A.1: Continuación

18.34 %	Desviación estándar velocidad vehículos pesados (PA26)
56.23 %	Velocidad motocicletas (PA26)
83.95 %	Desviación estándar velocidad motocicletas (PA26)

Anexo B

Especificaciones de los modelos de predicción de congestión

B.1. Configuración de los modelos y elección de hiperparámetros

Tabla B.1: Configuración de hiperparámetros para los árboles de decisión entrenados en la tarea de predecir congestión.

Hiperparámetro	Configuración
Profundidad	Fijada con validación cruzada
Criterio de ramificación	Impureza de Gini
Nº de tributos empleados	Todos los disponibles
Pesos de clases	Balanceados (0.523 para flujo libre y 11.277 para congestión)

Tabla B.2: Configuración de hiperparámetros para los modelos de *random forest* entrenados en la tarea de predecir congestión.

Hiperparámetro	Configuración
Profundidad de cada árbol	Fijada con validación cruzada
Número de árboles	100
Criterio de ramificación	Impureza de Gini
Nº de atributos empleados	Raíz cuadrada del total de los disponibles ^a
Pesos de clases	Balanceados (0.523 para flujo libre y 11.277 para congestión)

^a Para el entrenamiento cada árbol se escoge aleatoriamente una cantidad de atributos igual a la raíz cuadrada del total disponible.

Tabla B.3: Configuración de hiperparámetros para las redes neuronales MLP entrenadas en la tarea de predecir congestión.

Hiperparámetro	Configuración
N° de capas ocultas	Fijada con validación cruzada
N° de unidades por capa	Fijada con validación cruzada
Pesos de clases	Balanceados (0.523 para flujo libre y 11.277 para congestión)
N° máximo de <i>epochs</i>	30
Tamaño de los <i>batches</i>	32
Paciencia ^a	3
Optimizador	Adam con tasa de aprendizaje de 10^{-3}
Función de pérdida	Entropía cruzada binaria
Función de activación	ReLU en capas ocultas y sigmoide en la capa de salida
Inicialización de los pesos	Aleatorio según distribución uniforme $U(-0.05, 0.05)$
Inicialización de los sesgos	Constantes e iguales a 0

^a Dada una paciencia k , se interrumpe el entrenamiento de la red neuronal si alguna métrica de desempeño no mejora durante k *epochs*. En este caso, se monitorea la función de pérdida evaluada en el conjunto de validación.

B.2. Selección de hiperparámetros mediante validación cruzada

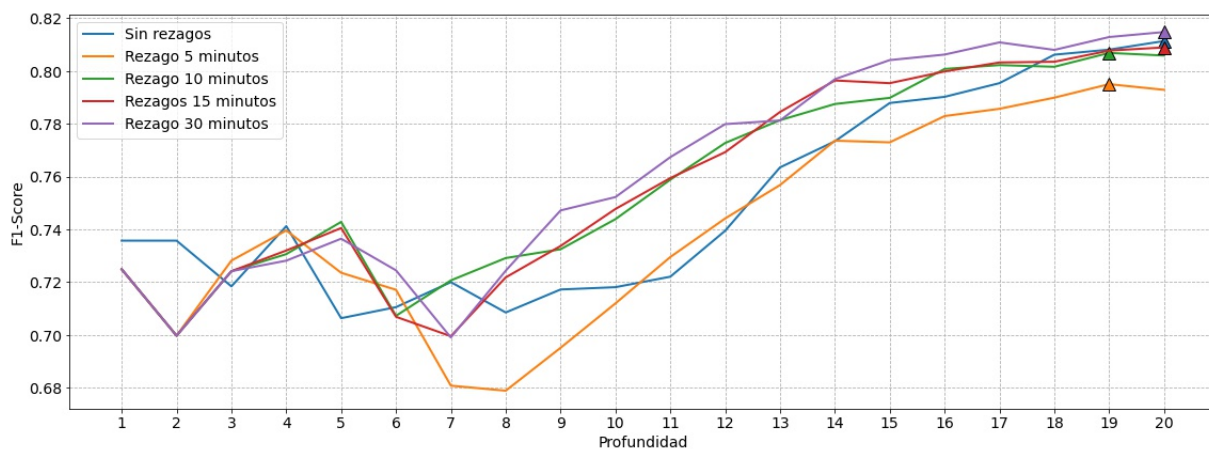


Figura B.1: F1-Score de los árboles de decisión para diferentes valores de la profundidad. Los valores corresponden al promedio de los resultados obtenidos en cada *fold* de la validación cruzada.

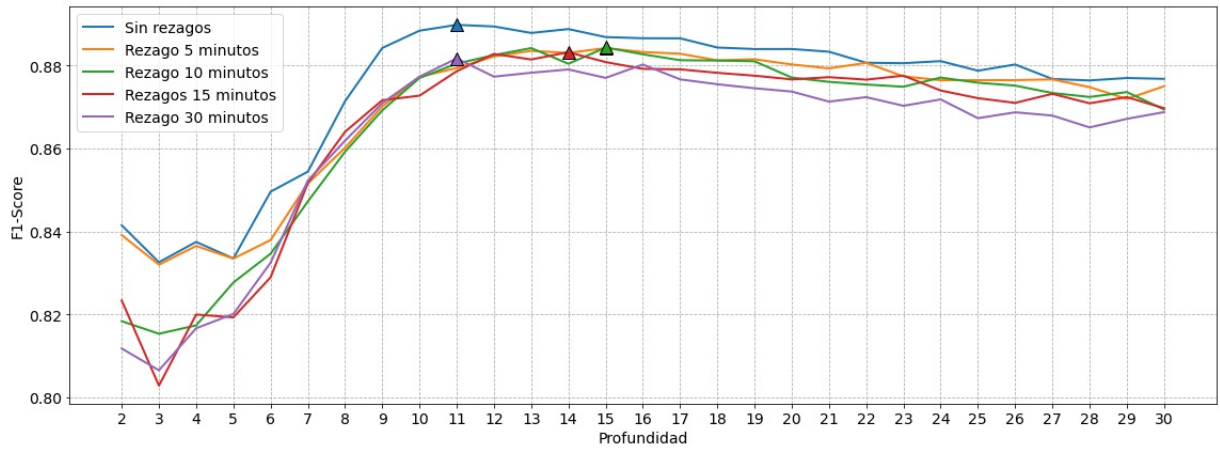


Figura B.2: F1-Score de los *random forest* para diferentes valores de la profundidad. Los valores corresponden al promedio de los resultados obtenidos en cada *fold* de la validación cruzada.

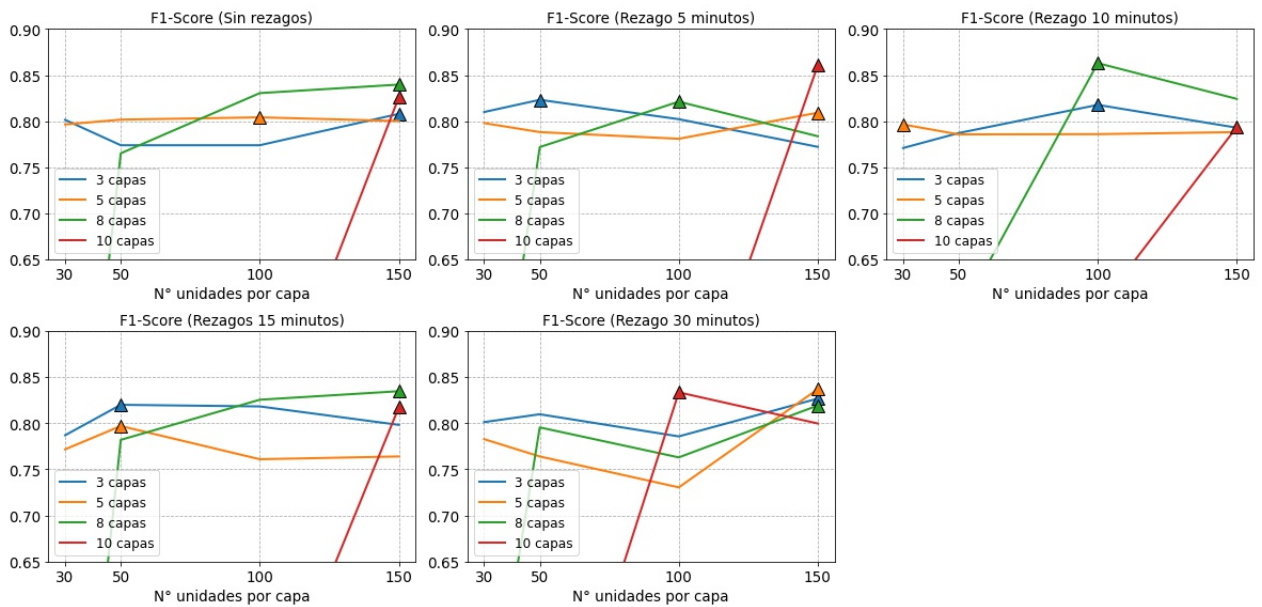
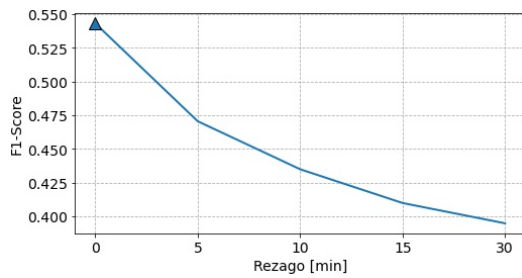
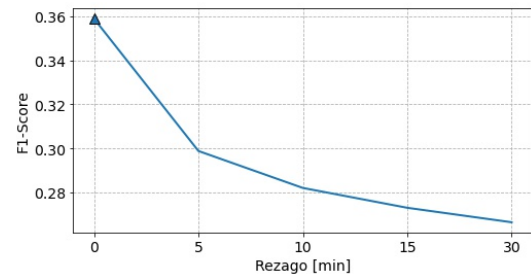


Figura B.3: F1-Score de las redes neuronales para los diferentes valores de la cantidad de capas ocultas y de unidades en cada una de ellas. Los valores corresponden al promedio de los resultados obtenidos en cada *fold* de la validación cruzada. Sólo se muestran los resultados mayores a 0.65.

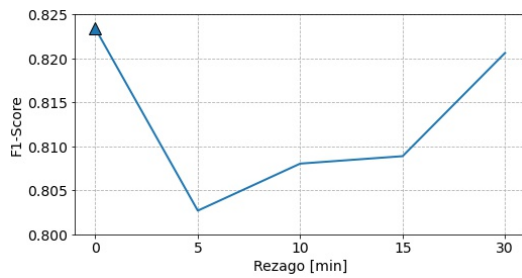
B.3. Resultados para distintos usos de información pasada mediante rezagos



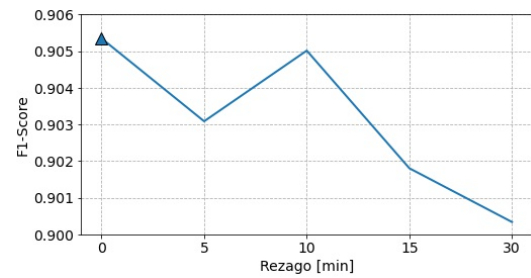
(a) Naive-Bayes Gaussiano



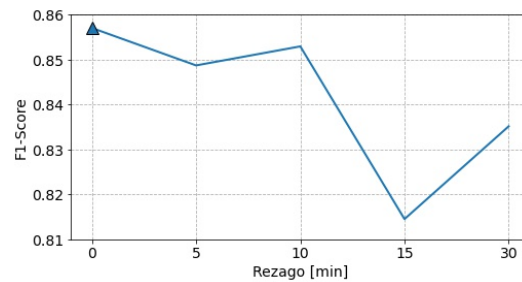
(b) Naive-Bayes Bernoulli



(c) Árbol de decisión



(d) *Random forest*



(e) Red neuronal

Figura B.4: Resultados del F1-Score para las diferentes configuraciones del uso de rezagos.

Anexo C

Importancias de los atributos para el modelo de *random forest*

Tabla C.1: Importancia de Gini de los atributos utilizados para entrenar el modelo de *random forest* que predice el estado de tráfico. Se muestran las importancias del modelo que no emplea rezagos por ser el que posee un mejor desempeño.

Importancia	Atributo
20.2 %	Velocidad vehículos pesados (PA20)
16.4 %	Estado de tráfico (PA20)
12 %	Velocidad vehículos livianos (PA20)
11.6 %	Velocidad motocicletas (PA20)
9.6 %	Densidad vehículos livianos (PA20)
5.7 %	Densidad vehículos pesados (PA20)
4.2 %	Desviación estándar velocidad vehículos livianos (PA20)
2 %	Velocidad vehículos livianos (PA24)
1.8 %	Desviación estándar velocidad vehículos pesados (PA20)
1.8 %	Densidad vehículos livianos (PA22)
1.6 %	Densidad vehículos livianos (PA26)
1.3 %	Densidad vehículos livianos (PA24)
1.1 %	Velocidad vehículos livianos (PA26)
1.1 %	Velocidad vehículos livianos (PA22)
0.8 %	Flujo de vehículos (PA24)
0.8 %	Velocidad vehículos pesados (PA26)
0.7 %	Estado de tráfico (PA24)
0.7 %	<i>Dummy</i> que indica si el día es un viernes
0.3 %	Flujo de vehículos (PA20)
0.3 %	Desviación estándar velocidad motocicletas (PA26)
0.3 %	Hora en que se registra la observación
0.3 %	Estado de tráfico (PA26)
0.2 %	Estado de tráfico (PA22)
0.2 %	Flujo de vehículos (PA22)
0.2 %	Velocidad vehículos pesados (PA24)
0.2 %	Desviación estándar velocidad motocicletas (PA20)

Tabla C.1: Continuación

0.2 %	Desviación estándar velocidad vehículos livianos (PA22)
0.2 %	Velocidad motocicletas (PA22)
0.2 %	Velocidad vehículos pesados (PA22)
0.2 %	Velocidad motocicletas (PA26)
0.2 %	Flujo de vehículos (PA26)
0.2 %	Desviación estándar velocidad vehículos livianos (PA26)
0.2 %	Desviación estándar velocidad motocicletas (PA24)
0.2 %	Proporción de vehículos livianos en el flujo (PA20)
0.2 %	Desviación estándar velocidad motocicletas (PA22)
0.2 %	Desviación estándar velocidad vehículos pesados (PA26)
0.1 %	Desviación estándar velocidad vehículos livianos (PA24)
0.1 %	Densidad vehículos pesados (PA24)
0.1 %	Densidad motocicletas (PA20)
0.1 %	Desviación estándar velocidad vehículos pesados (PA22)
0.1 %	Proporción de vehículos pesados en el flujo (PA20)
0.1 %	Desviación estándar velocidad vehículos pesados (PA24)
0.1 %	Densidad vehículos pesados (PA22)
0.1 %	Densidad vehículos pesados (PA26)
0.1 %	Proporción de vehículos livianos en el flujo (PA24)
0.1 %	Velocidad motocicletas (PA24)
0.1 %	Proporción de vehículos livianos en el flujo (PA26)
0.1 %	Proporción de vehículos pesados en el flujo (PA22)
0.1 %	Proporción de vehículos livianos en el flujo (PA22)
0.1 %	Proporción de vehículos pesados en el flujo (PA26)
0.1 %	<i>Dummy</i> que indica si el día es un miércoles
0.1 %	Proporción de vehículos pesados en el flujo (PA24)
0.1 %	Densidad motocicletas (PA26)
0.1 %	Densidad motocicletas (PA24)
0.1 %	Densidad motocicletas (PA22)
<0.1 %	<i>Dummy</i> que indica si el día es un martes
<0.1 %	<i>Dummy</i> que indica si el día es un jueves

Anexo D

Especificaciones de los modelos de predicción de velocidad

D.1. Configuración de los modelos y elección de hiperparámetros

Tabla D.1: Configuración de hiperparámetros para los árboles de regresión entrenados en la tarea de predecir velocidad.

Hiperparámetro	Configuración
Profundidad	Fijada con validación cruzada
Criterio de ramificación	Error cuadrático medio
Nº de atributos empleados	Todos los disponibles

Tabla D.2: Configuración de hiperparámetros para el entrenamiento de los modelos de regresión de *random forest* en la tarea de predecir velocidad.

Hiperparámetro	Configuración
Profundidad de cada árbol	Fijada con validación cruzada
Número de árboles	100
Criterio de ramificación	Error cuadrático medio
Nº de atributos empleados	Todos los disponibles

Tabla D.3: Configuración de hiperparámetros para las redes neuronales MLP entrenadas en la tarea de predecir velocidad.

Hiperparámetro	Configuración
Nº de capas ocultas	Fijada con validación cruzada
Nº de unidades por capa	Fijada con validación cruzada
Nº máximo de <i>epochs</i>	30
Tamaño de los <i>batches</i>	32
Paciencia ^a	3
Optimizador	Adam con tasa de aprendizaje de 10^{-3}
Función de pérdida	Error cuadrático medio
Función de activación	ReLU en capas ocultas e identidad en la capa de salida
Inicialización de los pesos	Aleatorio según distribución uniforme $U(-0.05, 0.05)$
Inicialización de los sesgos	Constantes e iguales a 0

^a Dada una paciencia k , se interrumpe el entrenamiento de la red neuronal si alguna métrica de desempeño no mejora durante k *epochs*. En este caso, se monitorea la función de pérdida evaluada en el conjunto de validación.

D.2. Selección de hiperparámetros mediante validación cruzada

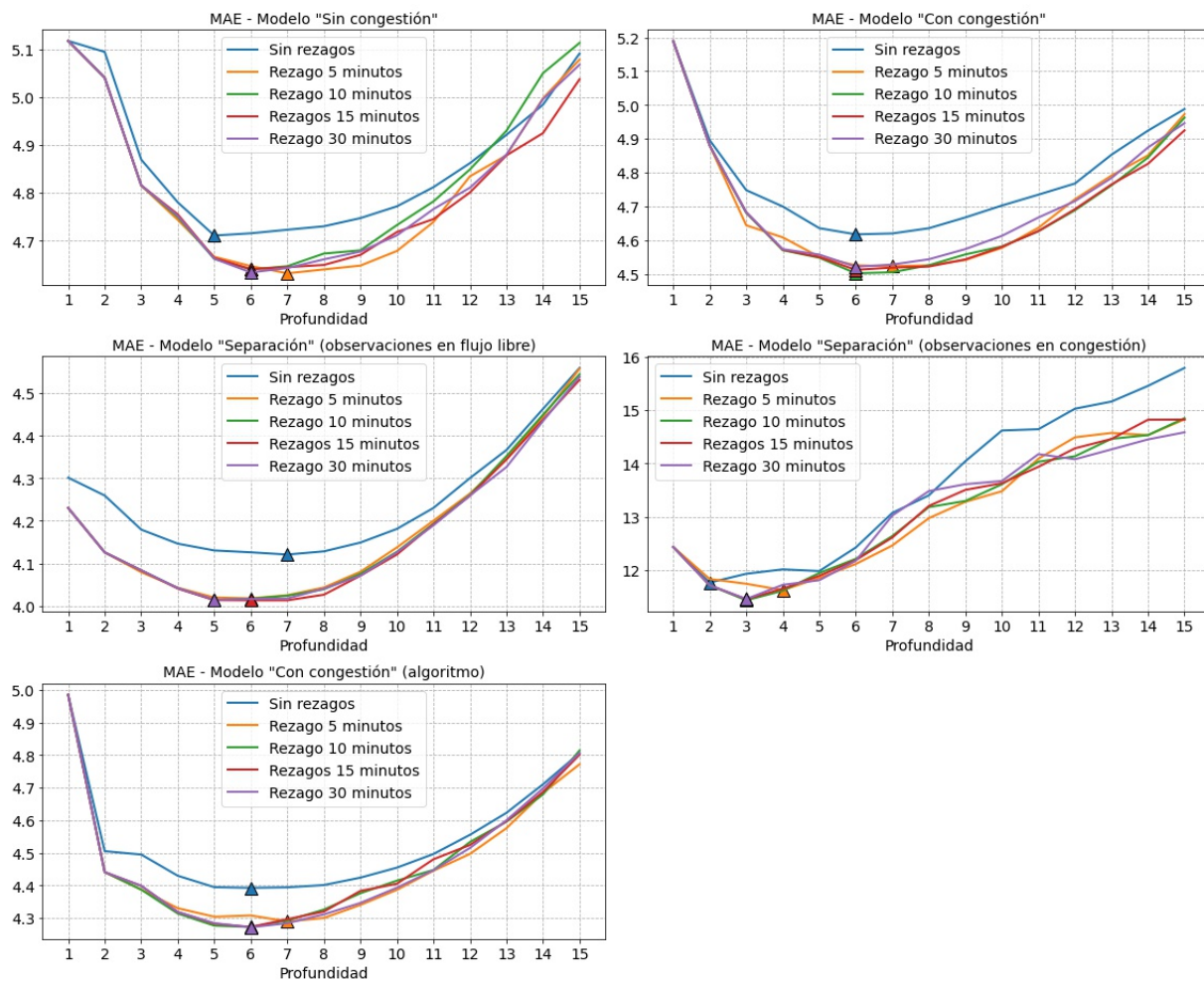


Figura D.1: Error absoluto medio (MAE) de los árboles de regresión para diferentes enfoques y valores de la profundidad. Se reporta el promedio de los resultados obtenidos en cada *fold* de la validación cruzada.

Tabla D.4: Profundidad óptima de los árboles de regresión según rezago y enfoque a partir de lo obtenido mediante validación cruzada.

Rezago	“Sin congestión”	“Con congestión”	“Separación” (flujo libre)	“Separación” (congestionado)	“Con congestión” (algoritmo)
0	5	6	7	2	6
5	7	7	6	4	7
10	6	6	5	3	6
15	6	6	6	3	6
30	6	6	5	3	6

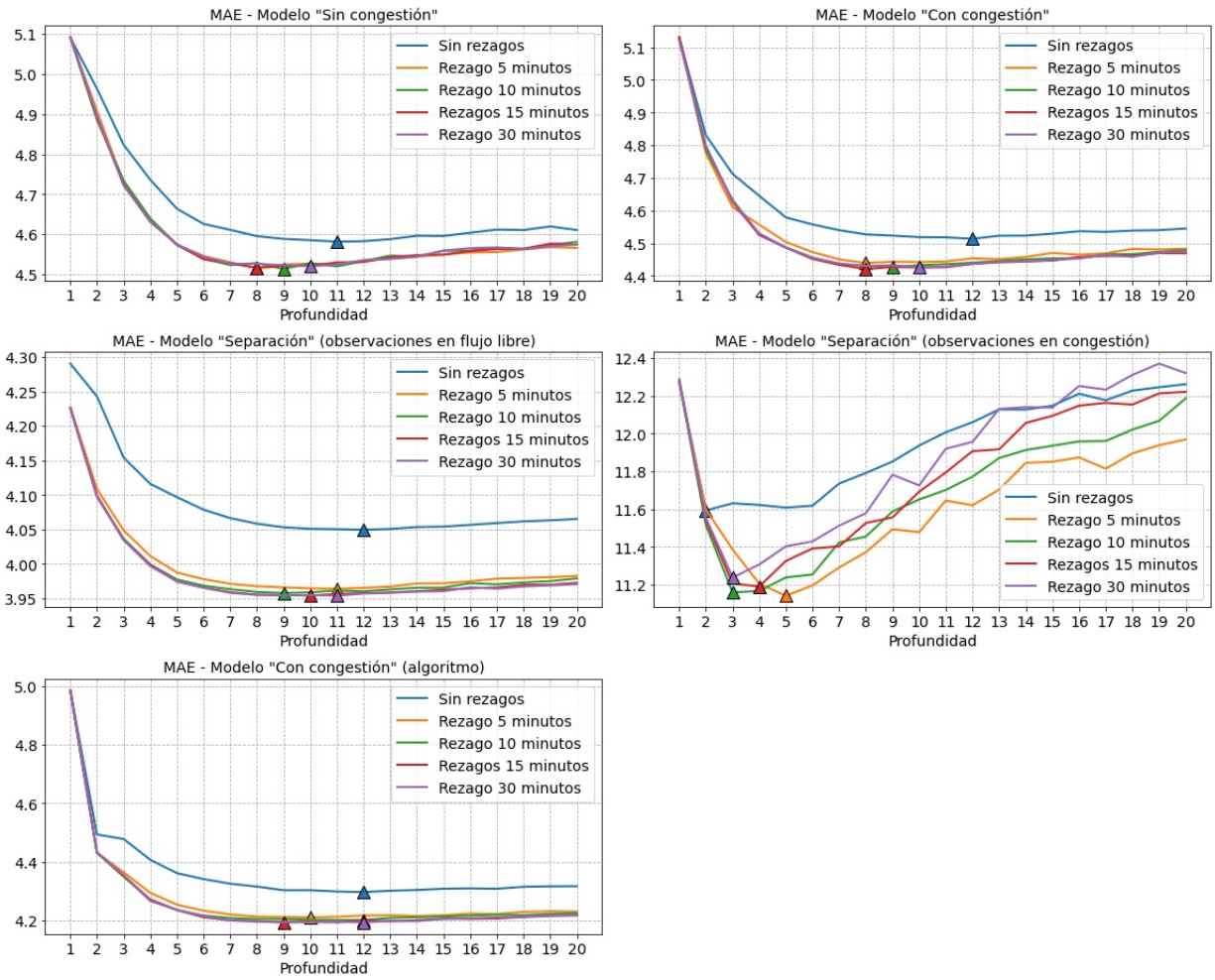


Figura D.2: Error absoluto medio (MAE) de los modelos de *random forest* para diferentes enfoques y valores de la profundidad. Se reporta el promedio de los resultados obtenidos en cada *fold* de la validación cruzada.

Tabla D.5: Profundidad óptima de los árboles de regresión del *random forest* según rezago y enfoque, a partir de lo obtenido mediante validación cruzada.

Rezago	“Sin congestión”	“Con congestión”	“Separación” (flujo libre)	“Separación” (congestionado)	“Con congestión” (algoritmo)
0	11	12	12	2	12
5	8	8	11	5	10
10	9	9	9	3	12
15	8	8	10	4	9
30	10	10	11	3	12

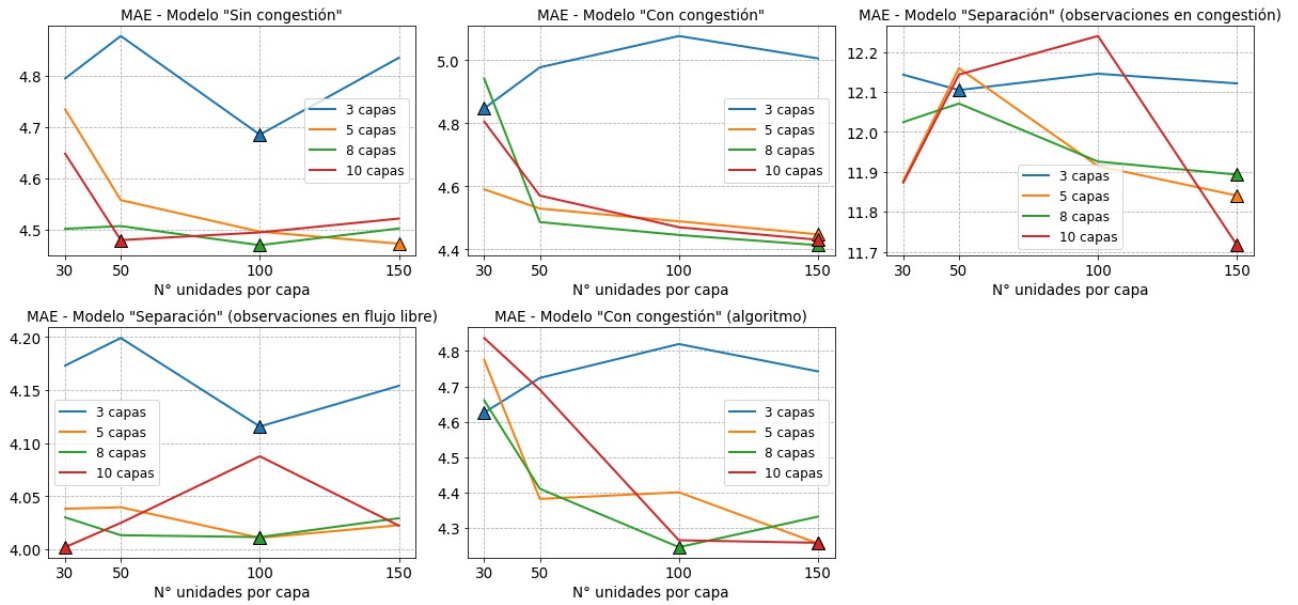


Figura D.3: Error absoluto medio (MAE) de las redes neuronales **sin rezagos** para los diferentes enfoques, valores de la cantidad de capas ocultas y de unidades en cada una de ellas. Se reporta el promedio de los resultados obtenidos en cada *fold* de la validación cruzada.

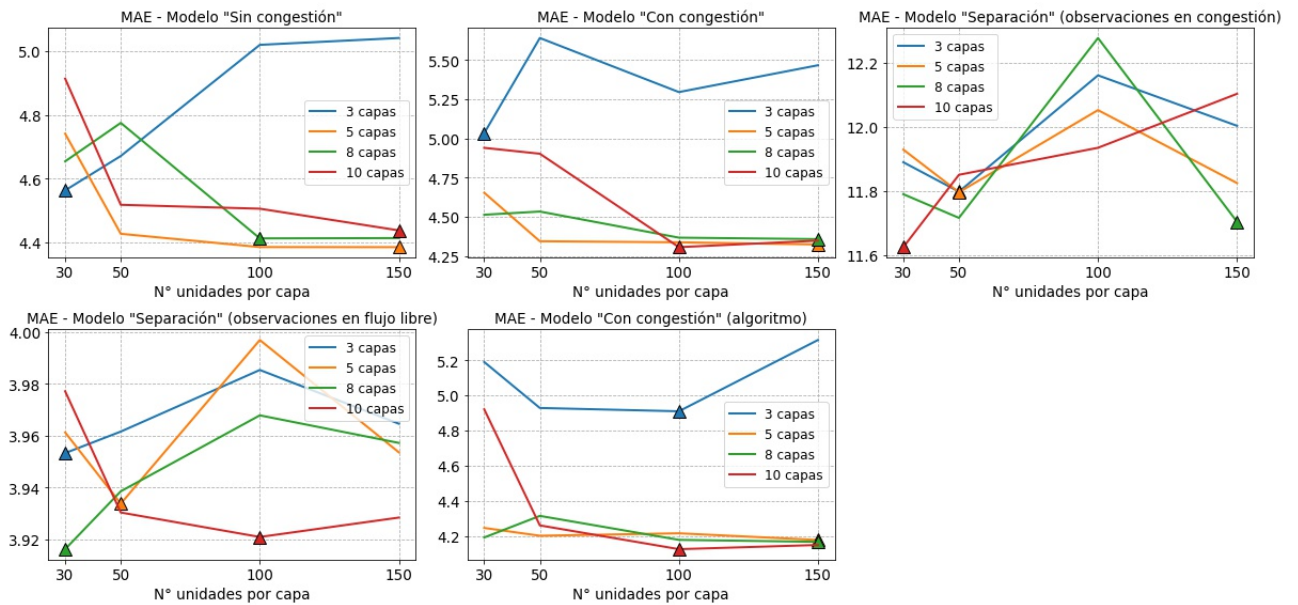


Figura D.4: Error absoluto medio (MAE) de las redes neuronales con rezago de **5 minutos** para los diferentes enfoques, valores de la cantidad de capas ocultas y de unidades en cada una de ellas. Se reporta el promedio de los resultados obtenidos en cada *fold* de la validación cruzada.

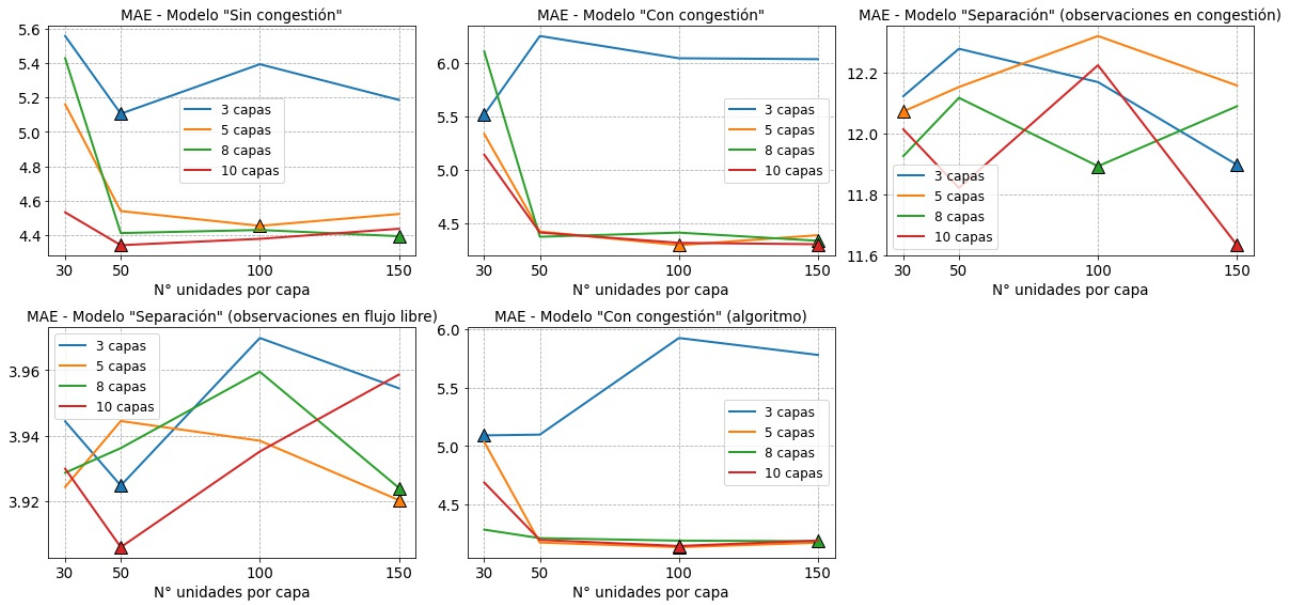


Figura D.5: Error absoluto medio (MAE) de las redes neuronales con rezago de **10 minutos** para los diferentes enfoques, valores de la cantidad de capas ocultas y de unidades en cada una de ellas. Se reporta el promedio de los resultados obtenidos en cada *fold* de la validación cruzada.

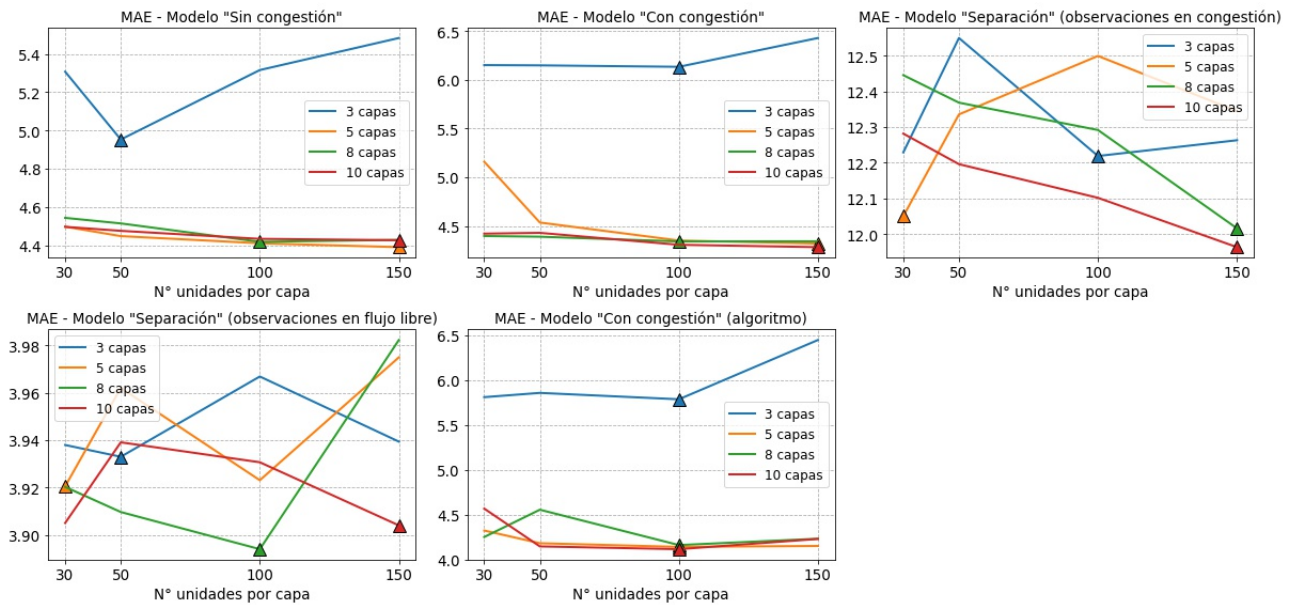


Figura D.6: Error absoluto medio (MAE) de las redes neuronales con rezago de **15 minutos** para los diferentes enfoques, valores de la cantidad de capas ocultas y de unidades en cada una de ellas. Se reporta el promedio de los resultados obtenidos en cada *fold* de la validación cruzada.

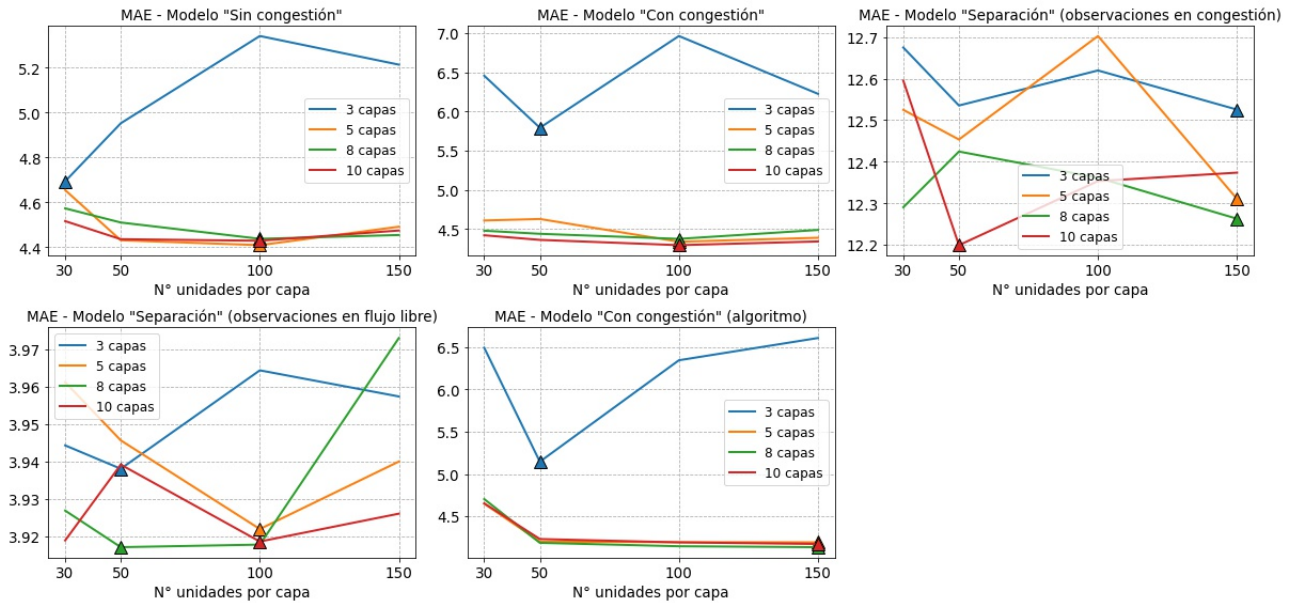


Figura D.7: Error absoluto medio (MAE) de las redes neuronales con rezago de **30 minutos** para los diferentes enfoques, valores de la cantidad de capas ocultas y de unidades en cada una de ellas. Se reporta el promedio de los resultados obtenidos en cada *fold* de la validación cruzada.

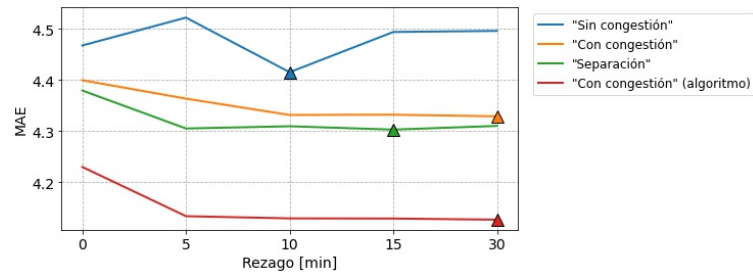
Tabla D.6: Cantidad de capas óptima de las redes neuronales según rezago y enfoque, a partir de lo obtenido mediante validación cruzada.

Rezago	“Sin congestión”	“Con congestión”	“Separación” (flujo libre)	“Separación” (congestionado)	“Con congestión” (algoritmo)
0	8	8	10	10	8
5	5	10	8	10	10
10	10	5	10	10	5
15	5	10	8	10	10
30	5	10	8	10	8

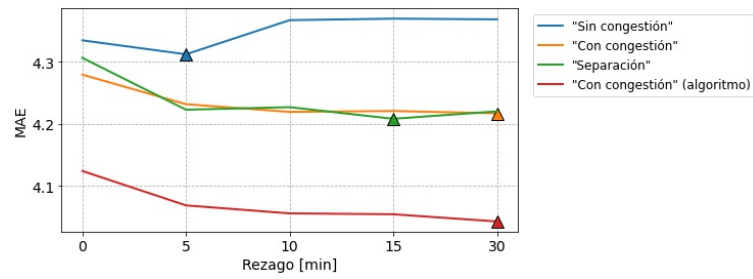
Tabla D.7: Cantidad óptima de unidades por capa oculta de las redes neuronales según rezago y enfoque, a partir de lo obtenido mediante validación cruzada.

Rezago	“Sin congestión”	“Con congestión”	“Separación” (flujo libre)	“Separación” (congestionado)	“Con congestión” (algoritmo)
0	100	150	30	150	100
5	150	100	30	30	100
10	50	100	50	150	100
15	150	150	100	150	100
30	100	100	50	50	150

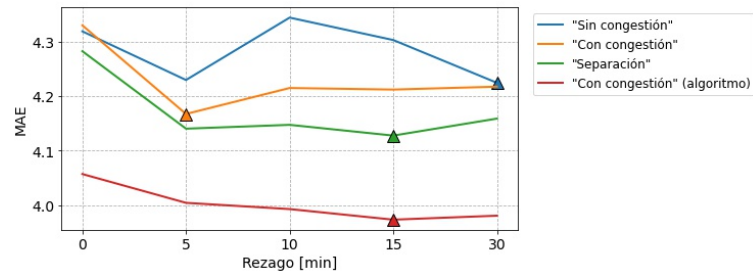
D.3. Resultados para distintos usos de información pasada mediante rezagos



(a) Árbol de regresión



(b) *Random forest*



(c) Red neuronal

Figura D.8: Resultados del error absoluto medio (MAE) para las diferentes configuraciones del uso de rezagos, una vez fijados los hiperparámetros a utilizar