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

DESARROLLO DE UN PROTOTIPO DE MODELO PREDICTOR DE TIEMPOS DE LLEGADA DE BUSES PARA UNA APLICACIÓN QUE BRINDA SERVICIOS DE INFORMACIÓN SOBRE EL TRANSPORTE PÚBLICO

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL INDUSTRIAL

DIEGO ANTONIO JORQUERA LÓPEZ

PROFESOR GUÍA:
MARCOS ORCHARD CONCHA

MIEMBROS DE LA COMISIÓN:
PABLO MARÍN VICUÑA
CHARLES THRAVES CORTÉS-MONROY

SANTIAGO DE CHILE
2020

RESUMEN DE LA MEMORIA PARA OPTAR
AL TÍTULO DE INGENIERO CIVIL INDUSTRIAL
POR: DIEGO ANTONIO JORQUERA LÓPEZ
FECHA: 2020
PROF. GUÍA: MARCOS ORCHARD C.

DESARROLLO DE UN PROTOTIPO DE MODELO PREDICTOR DE TIEMPOS DE LLEGADA DE BUSES PARA UNA APLICACIÓN QUE BRINDA SERVICIOS DE INFORMACIÓN SOBRE EL TRANSPORTE PÚBLICO

La estimación del tiempo de llegada de un bus a un paradero en el transporte público representa un aspecto importante para aplicaciones de teléfonos móviles que disponen servicios de transporte, ya que genera una impresión de calidad para el usuario.

El presente trabajo genera un prototipo de modelo que predice tiempos de llegada de un conjunto de buses del transporte público de la Región Metropolitana para una aplicación que brinda servicios de información sobre el sistema de transporte, desarrollando dos modelos distintos que resuelven el problema de distinta manera. El desempeño de estos modelos se mide bajo la métrica *RMSE.*

Para el desarrollo de este trabajo, se estudiaron los recorridos 301, 315e y 506, y dos horarios, horario Valle y Punta. El primer modelo es una red neuronal artificial *LSTM*, que se usa para predecir el tiempo de llegada del bus a todos los paraderos de la ruta. Se observó que la predicción mejora a medida que el bus se acerca al paradero. Se calculó el desempeño para distinto tiempo de lejanía, y para un bus de 5 minutos de distancia, el error promedio es de 3 minutos para los tres recorridos estudiados. A medida que aumenta la lejanía, disminuye la precisión de la predicción.

El segundo modelo es uno mixto, donde un Filtro de Partículas complementa su funcionamiento con datos obtenidos de una red neuronal Bayesiana. La ruta de un bus se divide en tramos, y se calcula el tiempo de recorrido en cada tramo. El valor obtenido por este modelo consta de un valor discreto del tiempo de llegada y una probabilidad asociada a dicho valor. Se calculó que este modelo, posee un error de máximo 2 minutos para los tres recorridos de buses, para ambos horarios.

Se concluye bajo la comparación de ambos modelos que el modelo mixto de Filtro de Partículas con red neuronal Bayesiana posee mejor predicción, con un error más bajo. Se recomienda a la empresa la implementación de este modelo, diseñando un prototipo en que los tramos de ruta son los que comprenden paraderos consecutivos.

A mi padres.
Todo lo que soy es gracias a ellos.

Agradecimientos

Agradezco a mi maravillosa familia, que me ha apoyado en todo lo que me he propuesto, y me ha dado un amor incondicional. A mi madre y mi padre, Silvia y Marco, que han hecho hasta lo imposible por ayudarme a cumplir mis metas, que han reído y llorado conmigo, que siempre han estado junto a mí y me han dado valores de los que me siento sumamente orgulloso. Soy afortunado de ser su hijo.

Agradezco a mi hermana Camila, que es la mejor hermana del mundo, desde pequeños nos hemos complementado y ayudado en todo, siempre con amor y alegría. Gracias por todo.

Gracias a Thor y Paty por su cariño y compañía, no son mascotas, son familia.

Agradezco y mando besos al cielo a mi Mami y mi Manina, que me llenaron de amor y caricias desde pequeño. Les agradezco infinitamente y las amo mucho.

Agradezco a mis tíos Julio y Rosa, y a mis primos Claudio y Gabriel, por todo el cariño, apoyo y los momentos felices que me han dado.

Agradezco todo el cariño y apoyo del Piño: Víctor, Alonso, Pablo, Rorro, Edgar, Marce, Paz, Momo y Nacho. Gracias por todos los momentos, asados y almuerzos con grata compañía. Harto estudiamos, pero las risas nunca faltaron.

Agradezco a mi profesor guía, Marcos Orchard, por el apoyo incondicional en este trabajo, por la paciencia y la disposición a enseñarme, sin eso esto no hubiese sido posible.

Agradezco a Francisco Jaramillo, que me brindó ayuda incondicional en este trabajo, con toda la disposición del mundo y mucho entusiasmo.

Finalmente, agradezco a mi polola Analía, por creer en mí, por su apoyo, su comprensión y amor infinito. Gracias a la vida por conocerte, gracias a ti por existir. Estoy seguro la vida nos tiene preparado un futuro juntos, un futuro hermoso.

Tabla de Contenido

1. Antecedentes Generales **1**
1.1. Empresa: Antecedentes Generales 1

2. Descripción del proyecto **5**
2.1. Descripción del proyecto y justificación 5
2.2. Objetivos 8
2.2.1. Objetivo General 8
2.2.2. Objetivos Específicos 8
2.3. Alcances 8
2.4. Resultados esperados 9

3. Marco Teórico **10**
3.1. Redes Neuronales Artificiales 10
3.1.1. *LSTM* (*Long Short Term Memory*) 11
3.1.2. Red Neuronal Bayesiana 13
3.2. Métodos de Monte Carlo 15
3.2.1. Filtro de Partículas 16
3.3. Métricas para evaluar desempeño de modelos 17
3.3.1. *MSE* (Mean Squared Error) 18
3.3.2. *RMSE* (Root Mean Squared Error) 18
3.3.3. Correlación de Pearson 19

4. Metodología **21**
4.1. Fase 1: Entendimiento y preparación de datos 21
4.2. Fase 2: Modelamiento 22
4.2.1. Red *LSTM* 22
4.2.2. Modelo Mixto de Red Neuronal Bayesiana y Filtro de Partículas . . . 22
4.3. Fase 3: Resultados y evaluación 23

5. Desarrollo Metodológico **24**
5.1. Análisis descriptivo de los datos 24
5.2. Modelo *LSTM* 33

5.3. Modelo Mixto de Red Neuronal Bayesiana y Filtro de Partículas 43
5.4. Comparación entre modelo *LSTM* y modelo mixto de red neuronal Bayesiana con Filtro de Partículas 64

6. Recomendaciones sobre el prototipo **67**

7. Trabajos futuros **69**

8. Conclusiones Generales **70**

Bibliografía **72**

Anexo A. Análisis Exploratorio **74**

Anexo B. Red *LSTM* **78**
B.1. Gráficos de error 78
B.1.1. Recorrido 301, conjunto de prueba y bus seleccionado 79
B.1.2. Recorrido 315e, conjunto de prueba y bus seleccionado 81
B.1.3. Recorrido 506, conjunto de prueba y bus seleccionado 83
B.2. Gráfico de comparación valor real y valor predicho, bus de prueba 84
B.2.1. Bus 301 85
B.2.2. Bus 315e 88
B.2.3. Bus 506 91

Anexo C. Figuras **94**

Índice de Tablas

2.1. Tabla comparativa de puntuación y comentarios de aplicaciones. Fuente: Elaboración propia. 6
5.1. Tabla de valores únicos. Fuente: Elaboración propia. 26
5.2. Tabla de tiempo promedio de recorridos. Fuente: Elaboración propia. 33
5.3. Tabla resumen de resultados *LSTM*, umbral de 10 minutos. 40
5.4. Tabla resumen de resultados *LSTM*, umbral de 50 minutos. 40
5.5. Tabla resumen de resultados *LSTM*, umbral de 20 minutos. 41
5.6. Tabla resumen de resultados *LSTM*, umbral de 5 minutos. 41
5.7. Tiempo promedio de tramos, por recorrido y horario. Fuente: Elaboración Propia. 45
5.8. Tabla de *RMSE* para tramos de recorridos en horario valle y punta, para todo el conjunto de prueba. Fuente: Elaboración propia. 47
5.9. Tabla de *RMSE* para tramos de recorridos en horario valle y punta, para buses seleccionados de recorridos. Fuente: Elaboración propia. 54
5.10. Tabla de distribuciones iniciales para el Filtro de Partículas. Fuente: Elaboración propia. 57
5.11. Tabla de *RMSE* para tramos de recorridos en horario valle y punta, de los resultados del Filtro de Partículas. Fuente: Elaboración propia. 64
5.12. Comparación entre resultados de modelo LSTM y Filtro de Partículas. Fuente: Elaboración Propia. 65
A.1. Tabla de comparación de tiempos de viaje. Fuente: Elaboración propia. . . . 74
B.1. Tabla resumen de comparación métrica *RMSE* versus umbral. 78

Índice de Ilustraciones

1.1. Organigrama TranSapp. Fuente: Elaboración propia. 2
3.1. Estructura de una red neuronal artificial. Fuente:[11] 11
3.2. Celda *LSTM*. Fuente:[4] 12
3.3. Ejemplo de predicciones de red Bayesiana. Fuente: Elaboración propia. . . 14
5.1. Correlación datos *GPS*. Fuente: Elaboración propia. 27
5.2. Rutas de recorridos. Fuente: Elaboración propia. 28
5.3. Grupos de recorridos. Fuente: Elaboración propia. 29
5.4. Paraderos de recorridos seleccionados. Fuente: Elaboración propia. 30
5.5. Velocidad horario valle bus 301. Fuente: Elaboración propia. 31
5.6. Velocidad horario punta bus 301. Fuente: Elaboración propia. 31
5.7. Histograma de tiempo de recorridos bus 315e, horario valle. Fuente: Elaboración propia. 32
5.8. Histograma de tiempo de recorridos bus 315e, horario punta. Fuente: Elaboración propia. 32
5.9. Visualización de variables creadas para red *LSTM*. Fuente: Elaboración propia. 34
5.10. Correlación de variables para red *LSTM*. Fuente: Elaboración propia. . . . 34
5.11. Resultados red *LSTM* para todo el conjunto de prueba 301, umbral 10 minutos. Fuente: Elaboración propia. 37
5.12. Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbral 10 minutos. Fuente: Elaboración propia. 37
5.13. Resultados red *LSTM* para todo el conjunto de prueba 315e, umbral 10 minutos. Fuente: Elaboración propia. 38
5.14. Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbral 10 minutos. Fuente: Elaboración propia. 38
5.15. Resultados red *LSTM* para todo el conjunto de prueba 506, umbral 10 minutos. Fuente: Elaboración propia. 39
5.16. Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbral 10 minutos. Fuente: Elaboración propia. 39
5.17. Métrica *RMSE* para cada umbral de tiempo para todo el conjunto de prueba, horario Valle. Fuente: Elaboración propia. 42

5.18. Paraderos seleccionados para modelo mixto. Fuente: Elaboración propia. . . 44
5.19. Correlación de variables para modelos de redes neuronales Bayesianas. Fuente: Elaboración propia. 46
5.20. Histograma de predicciones de tiempo al recorrer tramos de la ruta, bus 301 horario valle. Fuente: Elaboración propia. 48
5.21. Histograma de predicciones de tiempo al recorrer tramos de la ruta, bus 301 horario valle. Fuente: Elaboración propia. 49
5.22. Histograma de predicciones de tiempo al recorrer tramos de la ruta, bus 315e horario valle. Fuente: Elaboración propia. 50
5.23. Histograma de predicciones de tiempo al recorrer tramos de la ruta, bus 315e horario punta. Fuente: Elaboración propia. 51
5.24. Histograma de predicciones de tiempo al recorrer tramos de la ruta, bus 506 horario valle. Fuente: Elaboración propia. 52
5.25. Histograma de predicciones de tiempo al recorrer tramos de la ruta, bus 506 horario punta. Fuente: Elaboración propia. 53
5.26. Modelo de Filtro de Partículas, con un ejemplo. Fuente: Elaboración propia. 56
5.27. Predicción modelo Filtro de partículas. Distribución a priori y a posteriori. Bus 301 horario valle. Fuente: Elaboración propia. 58
5.28. Predicción modelo Filtro de partículas. Distribución a priori y a posteriori. Bus 301 horario punta. Fuente: Elaboración propia. 59
5.29. Predicción modelo Filtro de partículas. Distribución a priori y a posteriori. Bus 315e horario valle. Fuente: Elaboración propia. 60
5.30. Predicción modelo Filtro de partículas. Distribución a priori y a posteriori. Bus 315e horario punta. Fuente: Elaboración propia. 61
5.31. Predicción modelo Filtro de partículas. Distribución a priori y a posteriori. Bus 506 horario valle. Fuente: Elaboración propia. 62
5.32. Predicción modelo Filtro de partículas. Distribución a priori y a posteriori. Bus 506 horario punta. Fuente: Elaboración propia. 63
6.1. Prototipo de Modelo Mixto. Fuente: Elaboración propia. 68
A.1. Velocidad horario valle bus 315e. Fuente: Elaboración propia. 74
A.2. Velocidad horario punta bus 315e. Fuente: Elaboración propia. 75
A.3. Velocidad horario valle bus 506. Fuente: Elaboración propia. 75
A.4. Velocidad horario punta bus 506. Fuente: Elaboración propia. 75
A.5. Histograma de tiempo de recorridos bus 301, horario valle. Fuente: Elaboración propia. 76
A.6. Histograma de tiempo de recorridos bus 301, horario punta. Fuente: Elaboración propia. 76
A.7. Histograma de tiempo de recorridos bus 506, horario valle. Fuente: Elaboración propia. 76

A.8. Histograma de tiempo de recorridos bus 506, horario punta. Fuente: Elaboración propia. 77
B.1. Resultados red *LSTM* para todo el conjunto de prueba 301, umbral 50 minutos. Fuente: Elaboración propia. 79
B.2. Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbral 50 minutos. Fuente: Elaboración propia. 79
B.3. Resultados red *LSTM* para todo el conjunto de prueba 301, umbral 20 minutos. Fuente: Elaboración propia. 79
B.4. Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbral 20 minutos. Fuente: Elaboración propia. 80
B.5. Resultados red *LSTM* para todo el conjunto de prueba 301, umbral 5 minutos. Fuente: Elaboración propia. 80
B.6. Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbral 5 minutos. Fuente: Elaboración propia. 80
B.7. Resultados red *LSTM* para todo el conjunto de prueba 315e, umbral 50 minutos. Fuente: Elaboración propia. 81
B.8. Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbral 50 minutos. Fuente: Elaboración propia. 81
B.9. Resultados red *LSTM* para todo el conjunto de prueba 315e, umbral 20 minutos. Fuente: Elaboración propia. 81
B.10. Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbral 20 minutos. Fuente: Elaboración propia. 82
B.11. Resultados red *LSTM* para todo el conjunto de prueba 315e, umbral 5 minutos. Fuente: Elaboración propia. 82
B.12. Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbral 5 minutos. Fuente: Elaboración propia. 82
B.13. Resultados red *LSTM* para todo el conjunto de prueba 506, umbral 50 minutos. Fuente: Elaboración propia. 83
B.14. Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbral 50 minutos. Fuente: Elaboración propia. 83
B.15. Resultados red *LSTM* para todo el conjunto de prueba 506, umbral 20 minutos. Fuente: Elaboración propia. 83
B.16. Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbral 20 minutos. Fuente: Elaboración propia. 84
B.17. Resultados red *LSTM* para todo el conjunto de prueba 506, umbral 5 minutos. Fuente: Elaboración propia. 84
B.18. Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbral 5 minutos. Fuente: Elaboración propia. 84

B.19. Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbrales de 50 y 20 minutos, horario valle. Fuente: Elaboración propia. 85
B.20. Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbrales de 10 y 5 minutos, horario valle. Fuente: Elaboración propia. 86
B.21. Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbrales de 50 y 20 minutos, horario punta. Fuente: Elaboración propia. 86
B.22. Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbrales de 10 y 5 minutos, horario punta. Fuente: Elaboración propia. 87
B.23. Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbrales de 50 y 20 minutos, horario valle. Fuente: Elaboración propia. 88
B.24. Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbrales de 10 y 5 minutos, horario valle. Fuente: Elaboración propia. 89
B.25. Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbrales de 50 y 20 minutos, horario punta. Fuente: Elaboración propia. 89
B.26. Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbrales de 10 y 5 minutos, horario punta. Fuente: Elaboración propia. 90
B.27. Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbrales de 50 y 20 minutos, horario valle. Fuente: Elaboración propia. 91
B.28. Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbrales de 10 y 5 minutos, horario valle. Fuente: Elaboración propia. 92
B.29. Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbrales de 50 y 20 minutos, horario punta. Fuente: Elaboración propia. 92
B.30. Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbrales de 10 y 5 minutos, horario punta. Fuente: Elaboración propia. 93
C.1. Metodología Crisp-DM. Fuente: Kenneth Jensen. 94

Capítulo 1

Antecedentes Generales

Hoy en día, el transporte público y privado alrededor del mundo es variado, contando con distintos servicios complementarios. Uno de estos servicios complementarios es la predicción de tiempos de llegada de los móviles a un punto de destino. Este servicio es implementado por la misma empresa de transporte, aunque a veces este servicio es entregado por empresas externas.

En el presente trabajo, se estudia el desarrollo de este servicio para una aplicación de dispositivos móviles que brinda servicios relacionados al transporte público de la región Metropolitana de Chile. Esta aplicación tiene distintos tipos de servicios, y entre ellos entrega la predicción de tiempos de espera o llegada (ya que para efectos de este trabajo tiene el mismo significado), basándose en un servicio externo que ocupa la aplicación, es decir, no es un servicio propio de predicción.

A continuación, se encuentra una descripción general de la empresa y su funcionamiento.

1.1. Empresa: Antecedentes Generales

TranSapp es una empresa que inició el año el año 2016, fundada por cuatro ingenieros de la Universidad de Chile. Surgió bajo la necesidad de poder responder a las inquietudes más comunes de los usuarios del transporte público de Santiago: el tiempo de espera en un paradero, la condición del sistema en general y la planificación de un viaje por la capital. Por esto fue que la empresa desarrolló una aplicación llamada TranSapp, que permite a los usuarios del transporte público de Santiago, conocer todo lo necesario para planificar su viaje.

El mercado en que se desenvuelve la empresa es el de aplicaciones móviles, que gran

parte de la población mundial ocupa. En este gran mercado, TranSapp se puede identificar en la categoría de servicios de Transporte, debido a los servicios que dispone a sus usuarios. Existe competencia directa, ya que aplicaciones como Google Maps o Moovit entregan servicios similares sobre tiempos de espera e indicaciones de viaje.

TranSapp no cuenta con áreas definidas, pero sí con cargos definidos, que se muestran en el organigrama de la Figura 1.1.

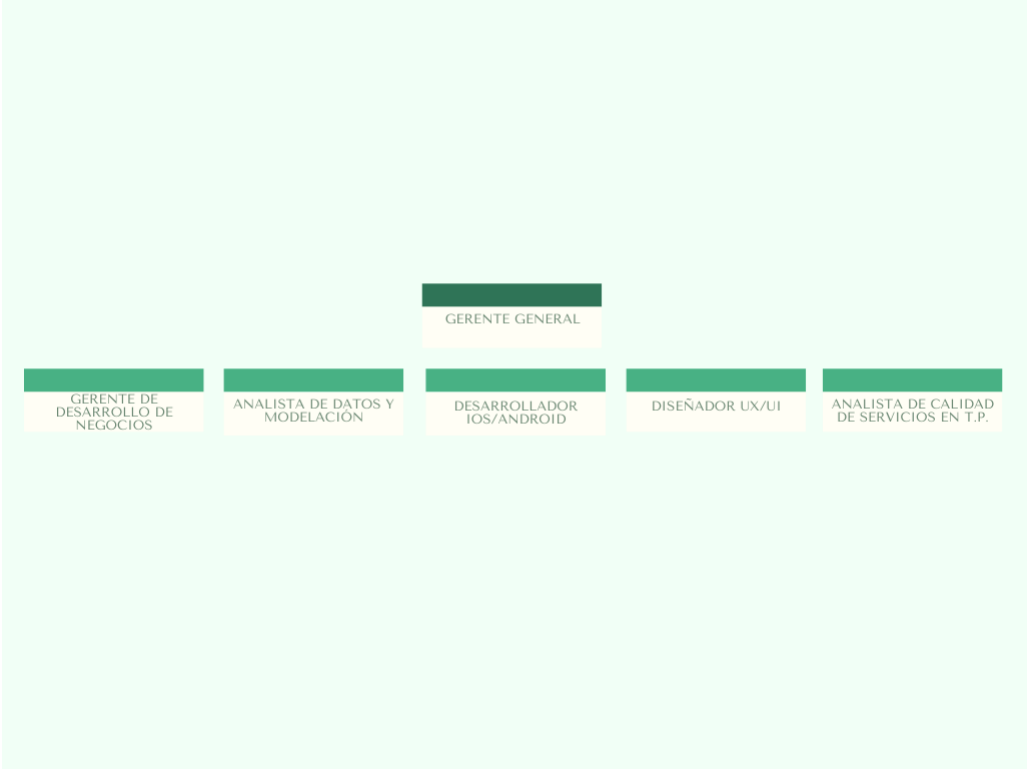

Figura 1.1: Organigrama TranSapp. Fuente: Elaboración propia.

Por otra parte, la misión y visión de TranSapp se presenta en su página web[1] como lo siguiente: *"¿Por qué lo hacemos? Porque merecemos la mejor experiencia de viaje, evidenciamos los problemas y te entregamos información del transporte público para tomar mejores decisiones. Utilizando tecnología de vanguardia construimos una comunidad de ciudadanos involucrados e informados con el sistema de transporte en la ciudad"*. Esto presenta la empresa como su tipo de misión y visión, y más que eso, la intención de su trabajo.

El producto que ofrece TranSapp es una aplicación para móviles con sistema operativo iOS o Android, y su tipo de producto es SaaS (Software como servicio), con una inversión hasta el año 2018 de $140.000 USD.

[1] www.transapp.cl

Los servicios que brinda la aplicación son:

- Tiempo de llegada del bus al paradero: Al seleccionar un paradero en el mapa de la aplicación, TranSapp indica los buses próximos a llegar, indicando un intervalo de tiempo.
- Reportes de los usuarios: Los usuarios pueden reportar problemas que identifiquen en los paraderos, por ejemplo, paradero en mal estado, sin iluminación, sin información de recorridos, entre otros.
- Planificación de viaje: La aplicación permite indicar el punto de inicio de un viaje y el punto final, y con esto muestra las rutas posibles y el tiempo estimado.

La ventaja que tiene la aplicación en comparación con las otras aplicaciones similares, son los reportes de los usuarios, y se muestran como alertas en los paraderos, con la hora en que fueron reportados. Esto permite que los usuarios del sistema público de transporte conozcan las condiciones a las que se enfrentarán si van a tomar un bus en particular, y si van a esperar en un paradero determinado.

Los usuarios de TranSapp son los usuarios de la aplicación móvil, que a la fecha de diciembre del año 2018, comprendían 100.000 usuarios, que generan más de 55.000 reportes mensuales. Según ranking de Apple Store, es la número 7 en la categoría de Navegación entre las aplicaciones más descargadas por chilenos, siendo sobrepasada por Moovit y Google Maps. La información que recauda TranSapp la obtiene de sus usuarios, quienes al instalar la aplicación pueden aceptar o denegar compartir datos. Estos datos son la ubicación del usuario al momento de realizar consultas, los viajes realizados y los reportes realizados, en base a su política de privacidad[2]. TranSapp obtiene también la estimación de tiempos de espera generado por SONDA, y la ocupa en su propia aplicación, así como obtiene datos del Directorio de Transporte Público Metropolitano.

Actualmente, el mercado donde se desenvuelve TranSapp, el mercado de las aplicaciones móviles para transporte y localización está dominado (en base a las descargas en las tiendas móviles) por actores de suma importancia, como Google Maps y Moovit. Google Maps ofrece los mismos servicios que TranSapp, pero es una aplicación que se ha implementado alrededor de todo el mundo, y según ranking de aplicaciones, es la número 1 en este aspecto[3]. Moovit también es una aplicación internacional, presente en más de 2.900 ciudades, con más de 550 millones de usuarios.

En base a lo anterior, TranSapp tiene la difícil tarea de posicionarse en el mercado nacional, y podría clasificarse como una empresa seguidora. El mercado de consultoría

[2] https://www.transapp.cl/politica-de-privacidad/
[3] https://play.google.com/store/apps/category/MAPS_AND_NAVIGATION?hl=es_CL

sobre calidad de servicio en Santiago genera ventas sobre los tres millones de dólares anuales, según investigación de TranSapp. Para el primer año de funcionamiento, se calculó como resultado una captura del 5 % del mercado.

Como modelo de negocio, el servicio para los pasajeros es gratuito. Los clientes son los reguladores y operadores, quienes pagan para conocer los problemas que reportan los usuarios en tiempo real, con el fin de evitar multas y sanciones[4]. Según el Gerente General de la empresa, se tiene como objetivo a largo plazo la expansión a regiones de Chile, brindando sus servicios basándose en el transporte de cada región, es decir, sobrepasar la barrera del transporte público de Santiago, Transantiago.

[4] http://www.uchile.cl/noticias/125223/transapp-la-aplicacion-que-busca-mejorar-los-viajes-en-transantiago

Capítulo 2

Descripción del proyecto

2.1. Descripción del proyecto y justificación

Como se expuso en el capítulo anterior, TranSapp cuenta con distintas áreas no bien definidas, pero que se comunican entre sí para lograr un correcto funcionamiento de la empresa. En este caso, el trabajo de memoria se desarrollará en el área de Análisis de datos y Modelamiento. Esta área la compone sólo un cargo, que es el Analista de datos y Modelación, que está encargado de generar el análisis de los datos obtenidos por TranSapp y los datos de fuentes externas. Con distintos softwares genera los algoritmos necesarios para que la aplicación funcione de la mejor forma, particularmente en el servicio de "cómo llegar", donde genera rutas posibles para el viaje del usuario entre un punto de la capital y su destino.

El servicio que entrega esta área influye y complementa los servicios de las otras áreas de la empresa, ya que con el análisis de datos que realiza y los modelos que genera, permite entregar avances, detectar problemas y entregar soluciones a los programadores de la aplicación.

La empresa, entre sus servicios, cuenta con un sistema predictor de tiempos de espera paraderos. Esto quiere decir que informa al usuario el tiempo estimado en que el bus llegará al paradero. Este servicio informa un rango de tiempo en que el bus llegará al paradero, generalmente en intervalos de 5 a 10 minutos.

El servicio de tiempos de espera de bus en el paradero se integró a la aplicación con un único proveedor, la empresa SONDA, contratada por Transantiago. Este último, al contratar los servicios de SONDA, también permite que aplicaciones independientes puedan hacer uso de estos servicios de forma gratuita, con la condición de haber postulado previamente y cumplir los requisitos de postulación. Al aceptar la postulación, los postulantes obtienen este sistema que puede ser incluido en la aplicación, pero sin poder realizar edi-

ciones a su funcionamiento, ya que no se conoce el código fuente que hace funcionar al sistema. La iniciativa de generar un modelo propio de predicción de tiempos de espera ha surgido previamente, pero por falta de integrantes en la empresa y por el trabajo en otros proyectos de mejora de la aplicación, no ha podido llevarse a cabo.

Como consecuencias a lo anterior, se tiene que:

- Al no tener un algoritmo de tiempos de espera propio, no se puede optimizar su funcionamiento.
- No se tiene control sobre el servicio que brindan a los usuarios (relacionado a tiempos de espera), por lo que cualquier error de este no es resoluble por parte de empresa.
- No se puede extender el uso de la aplicación a regiones de Chile, ya que se basa en el funcionamiento de Transantiago.

Como se expresó previamente, el servicio que entrega los tiempos de espera es uno independiente que ocupa la mayoría de las aplicaciones de este tipo en Santiago. Esto impone un estándar común en el sistema predictor de tiempos de espera de aquellas aplicaciones, lo que no permite progresar o destacarse en este mercado.

Por todo lo anterior, es que el proyecto consiste en desarrollar un sistema predictor de tiempos de espera propio, gestionable y editable. Para esto, el trabajo de título desarrolla un prototipo de este sistema predictor de tiempos de espera de buses, basándose en modelos de *Machine Learning*, y ocupando datos históricos de GPS (Global Positioning System).

No existe análisis que permita conocer la precisión del modelo actual de SONDA, pero la percepción de los usuarios puede dar indicios del problema. En la Tabla 2.1, se presenta una comparación entre TranSapp, Moovit y la aplicación oficial de Transantiago, sobre la puntuación y cantidad de comentarios, a julio de 2020.

Tabla 2.1: Tabla comparativa de puntuación y comentarios de aplicaciones.
Fuente: Elaboración propia.

	TranSapp	Moovit	Transantiago
Calificación (de 0 a 5 estrellas) Apple Store	2.8	4.5	2.5
Comentarios Apple Store	250	5000	160
Calificación (de 0 a 5 estrellas) Google Play Store	4.1	4.5	4.4
Comentarios Google Play Store	5592	798462	16059

Como se puede observar, TranSapp no es la aplicación que obtiene mejor valoración de los usuarios (en escala de 1 a 5 estrellas), y esto se observa en el detalle de los comentarios, en ambas tiendas de aplicaciones, Apple Store y Google Play Store, donde la mayoría son quejas relacionadas a los tiempos de espera de los buses del Transantiago. Específicamente, de las valoraciones de 1 estrella, en la tienda Google Play Store se registran 377 valoraciones de este tipo, siendo 92 de ellas debido a la imprecisión y al mal servicio del tiempo de espera de buses. Esto significa un 24.6 % de aquellas valoraciones. En la tienda App Store, en la misma categoría de valoración (1 estrella), comprenden 62 opiniones del total, y de estas, 27 de ellas son sobre el tiempo de espera de buses, alrededor de un 10.8 %.

Existen estudios donde se valora esta percepción del tiempo de espera de un bus en el paradero por parte de los usuarios, en base a encuestas. En [5] se expone que en ciudades de distintos países donde se implementa un modelo de tiempos de espera por aplicaciones o mensajes de texto, tiene como consecuencia que reduce la percepción del tiempo de espera real, e incrementa la satisfacción con el servicio de transporte. Específicamente, la percepción del tiempo real de espera disminuyó en un 13 % en Estocolmo (Suecia)[7] , un 26 % en Londres (Inglaterra)[13], y un 20 % en La Haya (Países Bajos)[2], desde que se inició este tipo de servicio. Sobre el aspecto de incremento de la percepción de seguridad del pasajero, en [15] se estudió la ansiedad que sufren los pasajeros de esperar el bus en la noche, y se concluyó que el 46 % de los pasajeros encuestados se sienten más seguros cuando saben que el bus está próximo a llegar.

Como conclusión, la oportunidad de elaborar un modelo propio que prediga tiempos de espera de buses representa una necesidad por parte de la empresa, ya que así no se dependería de un servicio del cual no se tiene control sobre su funcionamiento. Este tipo de servicio es valioso para los usuarios, desde distintos puntos de vista, y de ellos depende la valoración de la aplicación en el mercado, ya que ellos opinan.

2.2. Objetivos

2.2.1. Objetivo General

El objetivo general del trabajo de memoria es generar un prototipo de modelo de predicción de tiempos de espera que permita tomar decisiones estratégicas sobre la elección del modelo predictor implementado en la aplicación.

Para efectos de este trabajo, se define un prototipo de modelo como un modelo que cumple las siguientes condiciones:

- Que sea funcional.
- Que sea básico, ampliable y que pueda ser modificado.
- Que sirva como guía para modelos futuros.
- No es necesario que esté en código fuente de aplicación para móviles.

2.2.2. Objetivos Específicos

En los objetivos específicos, se tiene:

- Seleccionar recorridos a estudiar en el problema.
- Identificar y/o crear variables relevantes para el problema y para cada modelo.
- Generar y evaluar modelos, y elegir el de mejor desempeño.
- Concluir acerca de la precisión de los modelos, y la validez de implementación.
- Generar una recomendación de uso del mejor modelo.

2.3. Alcances

Debido a la gran cantidad de recorridos disponibles (que se expondrá en el análisis exploratorio), generar un modelo para cada uno de estos supone un extenso tiempo de análisis e implementación, así como costo computacional. Por esta razón, se seleccionarán grupos de recorridos con uno de ellos representando al grupo, y se trabajará con dicho recorrido.

No se tomará en cuenta los recorridos que son variaciones de recorridos establecidos, por ejemplo, un recorrido desviado por un evento en específico.

El desarrollo del problema es un prototipo, esto significa que no es un producto final que se pueda implementar de inmediato en la aplicación. Este prototipo debe ser adaptado por la empresa para su implementación.

No se comparará el desempeño del prototipo con datos en tiempo real, ni tampoco se experimentará con este tipo de datos, debido a la alta dificultad de este procedimiento, y a la contingencia nacional (Coronavirus).

El uso de softwares distintos para la realización de los modelos y/o prototipo no es una restricción, ya que, como se mencionó anteriormente, se debe adaptar el modelo a la aplicación.

2.4. Resultados esperados

Los resultados esperados del presente trabajo son:

- El desarrollo de un prototipo de sistema predictor de tiempos de espera para cada grupo de recorridos.
- Una recomendación de modelo de mejor desempeño para cada grupo de recorridos.
- Una comparación entre modelos desarrollados bajo métricas de desempeño.
- Conclusiones que permitan definir oportunidades de mejora o trabajos a futuro para mejorar el desempeño de los modelos.

Capítulo 3

Marco Teórico

Para el correcto entendimiento de los algoritmos, operaciones y métricas usadas en el presente trabajo, se expone en este marco teórico una descripción de estos algoritmos que se ocuparon en la realización del trabajo de título.

3.1. Redes Neuronales Artificiales

Los modelos de redes neuronales artificiales pertenecen a la gama de modelos de aprendizaje de máquinas e inteligencia artificial. Esto quiere decir que es un modelo configurable que tiene la capacidad de aprender por sí mismo la tarea que se le ordena, después de pasar por una etapa de entrenamiento.

Las redes neuronales artificiales están basadas en el funcionamiento de las redes neuronales biológicas. Es en estas últimas donde se transmiten impulsos nerviosos desde una neurona a otra contigua, transmitiendo señales con un objetivo concreto. Las redes neuronales artificiales emulan este proceso biológico, logrando que cada uno de sus componentes pueda desarrollar una tarea en específico.

Una red neuronal artificial está compuesta por unidades, llamadas neuronas, que están conectadas entre sí mediante enlaces. Las neuronas se agrupan en capas, y estas se califican en capas de entrada, ocultas y de salida.

La red recibe información en su capa de entrada. Esta capa procesa la información, y a cada neurona le asigna un valor, denominado ”peso”. La información de cada neurona se transmite a las neuronas enlazadas, de la misma capa o de las otras capas. A medida que se avanza en las capas de la red, las neuronas adquieren distintos pesos, llegando a la capa de salida el *output* o valor calculado por la red. La estructura de capas se observa en la Figura 3.1.

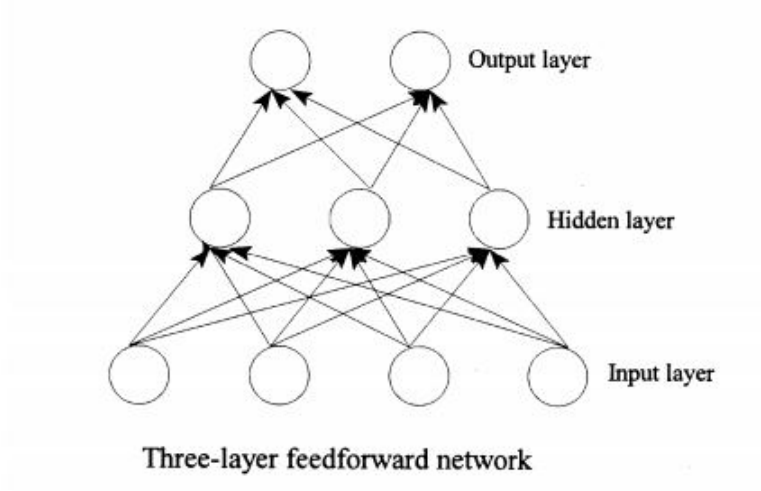

Figura 3.1: Estructura de una red neuronal artificial. Fuente:[11]

Las redes neuronales artificiales son ocupadas para numerosas tareas. Entre ellas, clasificación (reconocimiento de patrones, clasificación de imágenes), análisis de regresión, predicción de series temporales, procesamiento de datos, entre otros.

Como se mencionó anteriormente, estos modelos necesitan un entrenamiento adecuado. Se le llama entrenamiento a entregarle datos al modelo, con la respuesta correcta del valor a predecir, con el fin de que el modelo pueda aprender de la configuración de estos datos, para cuando deba predecir el valor deseado con precisión, basándose en sus conocimientos de la etapa de entrenamiento. Esta predicción la realiza con el objetivo de disminuir el error de una métrica de desempeño, también llamada función de pérdida. A mejor desempeño, menor valor de la función de pérdida. Generalmente se entrenan los modelos con una fracción de los datos, para posteriormente probar su desempeño bajo a la métrica de error seleccionada.
Para más información, revisar [11].

3.1.1. *LSTM* (*Long Short Term Memory*)

La red *LSTM* (*Long Short Term Memory*) es un tipo de red neuronal recurrente, ocupada generalmente para el procesamiento de datos secuenciales.

Las redes neuronales recurrentes son un tipo de red neuronal, siendo su principal característica la de retener información útil de iteraciones pasadas, pudiendo ”recordar” información de estados pasados y configuraciones de iteraciones previas de la red, permitiendo configurar un nuevo estado en base a esta información.

Por esto es que este tipo de red, específicamente la red *LSTM*, permite manejar datos secuenciales o cronológicos, reteniendo información por mayor cantidad de iteraciones que otros tipos de red neuronales recurrentes. El entrenamiento de esta red, generalmente se realiza con *dropout* (activación y desactivación de neuronas en distintas capas), para evitar el overfitting. Generalmente el entrenamiento de estas redes es costoso en términos computacionales, ya que se debe entregar información por segmentos o lotes secuenciales.

Este tipo de red se compone por unidades (o celdas) que llevan el mismo nombre, *LSTM*. Estas unidades son las que crean la memoria temporal que permite retener información útil de estados o parámetros. Cada una de estas celdas se compone de tres compuertas que permiten el flujo de información:

- Compuerta de entrada: Esta compuerta controla el momento en que la información nueva puede ingresar a la memoria.
- Compuerta de olvido: Controla el momento y la cantidad de información útil que debe retenerse en la memoria, así como también elimina la información que no es considerada útil para el problema, lo que permite liberar espacio que estará disponible para nueva información.
- Compuerta de salida: Controla el momento en que se utiliza la información recopilada en la memoria de la red. La celda dispone de un mecanismo de optimización de las ponderaciones de los pesos de cada neurona, basado en el error de salida de la red.

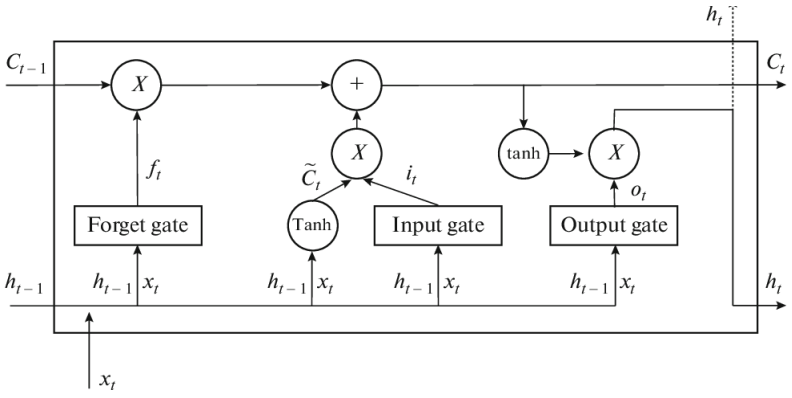

Figura 3.2: Celda *LSTM*. Fuente:[4]

En la Figura 3.2, se observa la estructura de una celda *LSTM*. Al tiempo t, el dato de entrada es x_t, la compuerta de entrada es i_t, la compuerta de olvido es f_t, y la compuerta de salida es o_t, el estado de entrada de la celda es $\hat{C}_t$, el estado de salida de la celda es C_t, y la capa de salida del estado es h_t.

El estado de salida h_t se calcula como:

$$\begin{gathered} i_t = \delta(W_{ix}X_t + W_{ih}h_{t-1} + b_i), \quad f_t = \delta(W_{fx}X_t + W_{fh}h_{t-1} + b_f) \\ o_t = \delta(W_{ox}X_t + W_{oh}h_{t-1} + b_o), \quad \hat{C}_t = tanh(W_{Cx}X_t + W_{Ch}h_{t-1} + b_C) \\ C_t = i_t * \hat{C}_t + f_t * C_{t-1}, \quad h_t = o_t * tanh(C_t) \end{gathered} \tag{3.1}$$

Donde W_{ix},W_{fx},W_{ox},W_{cx} son los coeficientes de pesos que conectan a x_t con las tres compuertas y a $\hat{C}_t$;W_{ih},W_{fh},W_{oh},W_{Ch} son coeficientes de peso que conectan a h_{t-1} a las tres compuertas y a $\hat{C}_t$; y $\delta(x) = \frac{1}{1+exp(-x)}$; $tanh(x) = \frac{exp(x)-exp(-x)}{exp(x)-exp(-x)}$.

Los estados de salida C_t y h_t son utilizados como los datos de entrada de la próxima celda *LSTM*.

Como se puede observar, la explicación del funcionamiento de este tipo de red es complejo, y se basa en las ecuaciones expuestas previamente, obtenidas de [4]. En dicho trabajo, el autor realiza la comparación de tres modelos que predicen el tiempo de llegada de buses a un punto. Estos tres modelos son una red neuronal perceptrón multicapa, una red *LSTM* y un modelo de regresiones. Al compararlos bajo la métrica MAE, la red *LSTM* obtiene el mejor desempeño.

3.1.2. Red Neuronal Bayesiana

Las redes neuronales artificiales Bayesianas son redes que ofrecen una interpretación probabilística de sus resultados, inferida de la distribución de los pesos de sus neuronas. Este tipo de red puede ser del tipo recurrente o convolucional.

En [10], se propone un tipo de red neuronal Bayesiana, que tiene las siguientes ventajas:

- Se adapta bien a gran cantidad de datos
- Se adapta a modelos complejos (permite combinar modelos de distintos tipos de red)
- No es difícil de implementar, y no es necesario poseer gran conocimiento de aprendizaje de máquinas.

Una red neuronal Bayesiana es, en resumen, un algoritmo que genera resultados de los que es posible identificar una tendencia o distribución de estos.

El método para elaborar estas redes, en el presente trabajo, es ocupar *dropout* aleatorio en las neuronas, y realizar ”T” pasos estocásticos por la red, recolectando sus resulta-

dos. La varianza de estos resultados se interpreta como la incertidumbre de la predicción.

La técnica de *dropout*, como se mencionó anteriormente, significa activar y desactivar neuronas aleatoriamente en la red, para evitar overfitting. En este caso, se ocupa una técnica llamada MC (Monte Carlo) dropout, ya que se realizan los ”T” pasos por la red mencionados, se registran los resultados, y se calcula el promedio y la varianza de estos.[10]

Bajo este método, si se desea predecir un valor en específico con una red neuronal Bayesiana, por ejemplo, el tiempo de un trayecto, la red realizará ”T” predicciones de dicho valor. El promedio de estos valores será el valor de la predicción.

Aparte de poder encontrar un valor predicho, lo importante de este tipo de red neuronal, es que el vector de predicciones de largo ”T” contiene de forma intrínseca el error de predicción para aquel dato. Es posible ingresar todo este valor de predicciones en distintos modelos, con el fin de captar dicho error de predicción. En la Figura 3.3, se muestra el ejemplo de un histograma del vector de predicciones de un registro calculado con una red neuronal Bayesiana.

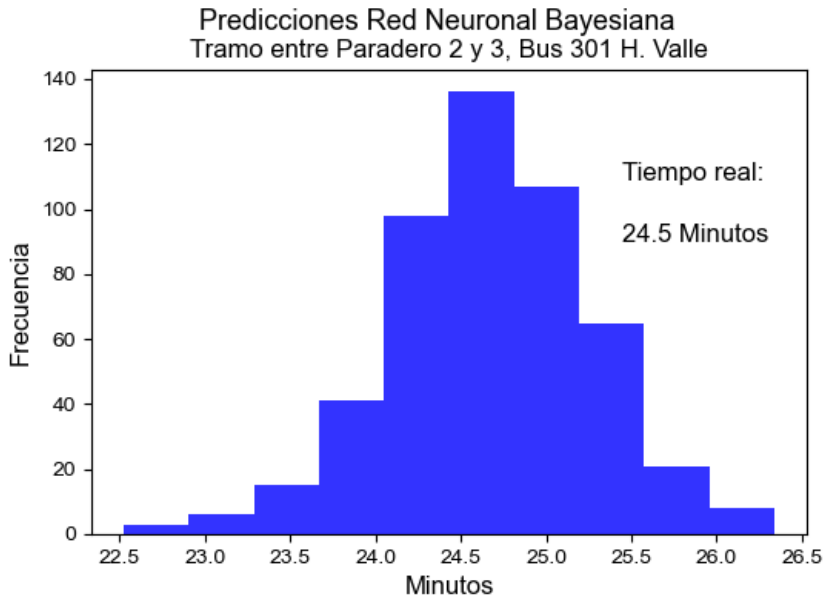

Figura 3.3: Ejemplo de predicciones de red Bayesiana. Fuente: Elaboración propia.

3.2. Métodos de Monte Carlo

Muchos problemas se pueden describir como un estado que cambia a través del tiempo, siendo medido con instrumentos que conllevan de forma intrínseca un error de medición. Esto se asemeja a medir el tiempo de llegada de un bus a un punto determinado, bajo un instrumento *GPS* que genera registros de la posición del bus.

Este tipo de problemas puede ser expresado como un sistema que evoluciona de forma secuencial y recursiva, representado por una variables de estado, que evidencian el estado del sistema en el instante ”n”, y variables de observación o medición, que relacionan las variables de estado con algún error asociado a la medición.[6][3]

Bajo esta formulación del problema, puede ser visto desde el punto de vista Bayesiano, lo que significa calcular de forma recursiva la función de distribución de probabilidad de las variables de estado, de forma previa y posterior, sujeto a las mediciones entregadas para cada instancia y su error relativo.

Los métodos secuenciales de Monte Carlo, entregan una solución a este problema, ya que permiten obtener una distribución de probabilidad posterior del estado de manera completa, permitiendo conocer fácilmente la moda, varianza y media de manera sencilla.[2]

Para definir el problema asociado a este tipo de métodos, se considera un modelo de procesos en tiempo discreto, donde existe un vector de estados descrito por la ecuación (3.2):

$$x_k = f_k(x_{k-1}, \omega_{k-1}) \tag{3.2}$$

Donde k $\in \mathbb{N}$ son los instantes de tiempo, y donde f es una función probablemente no lineal dependiente del vector de estado x_{k-1} y de un vector de ruido de proceso $\omega \in \mathbb{R}^{n_\omega}$

También se considera un vector de observación (o mediciones) $z \in \mathbb{R}^{n_z}$, que se relaciona con el vector de estados x, y un vector de errores de medición $v \in \mathbb{R}^{n_v}$, por medio de una función probablemente no lineal h, como se detalla en la ecuación (3.3):

$$z_k = h_k(x_k, \nu_{k-1}) \tag{3.3}$$

Estos problemas tienen el objetivo obtener información sobre el estado x_k basado en el conjunto de mediciones z, teniendo en cuenta el error de estas mediciones, como se detalla en la ecuación 2. En base a esto, se desea construir la función de distribución de probabilidad (PDF en inglés, Probability Distribution Function) del vector de estados.

Al principio del procedimiento, se asume que la PDF a *priori* se encuentra disponible, y con esto, la distribución final se puede obtener de manera recursiva, en dos etapas: predicción y actualización.

Suponiendo que la PDF de x_{-1} está disponible, se ocupa la ecuación de Chapman-Kolmogorov para obtener la predicción de la PDF a *priori* del estado en el tiempo k, como se detalla en la ecuación (3.4):

$$p(x_k|z_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|z_{1:k-1})\mathrm{d}x_{k-1} \tag{3.4}$$

Para la etapa de actualización, se utiliza la observación z_k para actualizar la pdf a priori calculada previamente, ocupando la regla de Bayes:

$$p(x_k|z_{1:k-1}) = \frac{p(z_k|x_k)p(x_k|z_{1:k-1})}{p(x_k|z_{1:k-1})} \tag{3.5}$$

La etapa de actualización ocupa la observación z_k para modificar la PDF a *priori*, con el objetivo de obtener la PDF a *posteriori* del vector de estados en tiempo actual.

Las ecuaciones 3 y 4 muestran la relación de recursividad de la solución óptima Bayesiana. Esta recursividad de la PDF a *posteriori* es solo conceptual, y no puede ser determinada de manera analítica.

$$p(z_k|z_{1:k-1}) = \int p(z_k|x_k)p(x_k|z_{1:k-1})\mathrm{d}x_k \tag{3.6}$$

Entre estos métodos, se encuentra el denominado *Bootstrap Filter*, o Filtro de Partículas.

3.2.1. Filtro de Partículas

Algunos autores han resuelto problemas asociados a tiempo de espera de buses con un algoritmo de Filtro de Kalman , que es similar a los métodos secuenciales de Monte

Carlo, ya que se describen variables de estado y observación, pero tienen como condición que estas variables distribuyan de forma Gaussiana, y unimodales. Esto representa una desventaja, debido a que no necesariamente estas variables distribuyen de esa forma en la mayoría de estos problemas.[16]

En [8], los autores implementan tres modelos distintos para predecir el tiempo de llegada de un bus, y finalmente comparan los resultados. En esta instancia, el Filtro de Kalman obtuvo mejores resultados para la métrica de comparación, que en este caso fue el MAPE, en contraste con una red neuronal y un modelo de promedios históricos.

En cambio, el Filtro de Partículas no impone esta condición, de hecho, permite también que se tengan distribuciones bimodales. Esta metodología se ocupa para aproximar la solución óptima Bayesiana cuando la solución analítica es difícil de encontrar.

El Filtro de Partículas es una técnica que implementa un Filtro Bayesiano a través de simulaciones de Monte Carlo. El objetivo es representar la PDF a *posteriori* del vector de estados, por medio de un conjunto de muestras aleatorias con pesos asociados, llamadas partículas, y de este modo, calcular estimaciones basadas en aquellas partículas.

Sea $x_k^i, i = \{1, ...N_s\}$ las partículas con sus pesos correspondientes $w_k^i, i = \{1, ...N_s\}$, donde N_s es el número de partículas. Los pesos se normalizan, esto quiere decir que la suma de ellos es igual a 1. Con esto, la PDF a *posteriori* en el tiempo k puede ser aproximada por la ecuación (3.7), en donde $\delta(\cdot)$ es la función delta de Dirac.

$$p(x_k|z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^{i=1} \delta(x_k - x_k^i) \tag{3.7}$$

La actualización de pesos se detalla en la ecuación (3.8):

$$w_k^i = w_{k-1}^i \frac{p(z_k|x_k^i)p(x_k^i|x_{k-1}^i)}{q(x_k^i|x_{0:k-1}^i, z_{1:k})} \tag{3.8}$$

Se puede notar que cuando el número de partículas tiende a infinito (N_s), el cálculo de la ecuación (3.8) converge a la PDF a posteriori real $p(x_k|z_{1:k})$.[3][14]

3.3. Métricas para evaluar desempeño de modelos

3.3.1. *MSE* (Mean Squared Error)

El *MSE* (Mean Squared Error) o error cuadrático medio, es una métrica ocupada en problemas de regresión, que mide el promedio de los errores al cuadrado. Esto es, la diferencia del valor real y el valor predicho por el algoritmo.

Su cálculo se muestra en la ecuación (3.9):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2 \tag{3.9}$$

Donde:

- n = Cantidad de datos
- $\hat{Y}_i$ = Valor predicho para dato i
- Y_i = Valor real dato i

Esta métrica de error representa la varianza del estimador, y por lo tanto, tiene la misma unidad de medida de la cantidad que se estima, pero al cuadrado. Al calcular el cuadrado de los errores, no se genera ningún valor negativo para esta métrica.

Otra característica importante de esta métrica es que amplifica los errores mayores, ya que calcula el cuadrado del error. Para más información, revisar [9].

3.3.2. *RMSE* (Root Mean Squared Error)

El *RMSE* (Root Mean Squared Error) o la raíz del error cuadrático medio, es una métrica ocupada en problemas de regresión, que mide la raíz del promedio de los errores al cuadrado. El valor resultante tiene la misma unidad que los datos ocupados. Su cálculo se basa en la raíz cuadrada del *MSE*.

Su cálculo se muestra en la ecuación (3.10):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2} \tag{3.10}$$

Generalmente se ocupa esta métrica para facilitar la interpretación del resultado del error, ya que al encontrarse en las mismas unidades de medida, se obtiene una impresión mejor de cuán preciso es el modelo.

3.3.3. Correlación de Pearson

El coeficiente de correlación de Pearson mide la dependencia lineal entre dos variables cuantitativas, independiente de la escala de medida de las variables.

Este indicador es útil para distinguir entre qué variables están relacionadas entre sí, y qué tipo de relación también (correlación positiva, negativa, no existente).

Se calcula como se muestra en la ecuación (3.11):

$$\rho_{(X,Y)} = \frac{\sigma_{XY}}{\sigma_X \sigma_Y} \tag{3.11}$$

Donde:

- σ_{XY} = Covarianza de (X,Y)
- σ_X = Desviación estándar de la variable X
- σ_Y = Desviación estándar de la variable Y

El valor de este indicador muestra la relación de las variables, y puede tomar distintos valores:

- Si $\rho_{(X,Y)}$= 1, la correlación es positiva perfecta. Esto significa que si una variable aumenta, la otra también lo hace, en la misma proporción. En el caso que una disminuya, el comportamiento es análogo.
- Si 0 < $\rho_{(X,Y)}$ < 1, la correlación es positiva. Es igual que el caso de correlación perfecta positiva, pero no aumentan o disminuyen en la misma proporción.
- Si $\rho_{(X,Y)}$= 0, no existe correlación. Las variables son independientes.
- Si $\rho_{(X,Y)}$= -1, la correlación es negativa perfecta. Esto significa que si una variable aumenta, la otra disminuye, en la misma proporción. En el caso que una disminuya, el comportamiento es análogo.
- Si -1 < $\rho_{(X,Y)}$ < 0, la correlación es negativa. Es igual que el caso de correlación perfecta negativa, pero no aumentan o disminuyen en la misma proporción.

Este indicador es importante para seleccionar un conjunto variables relevantes de una gama de variables, siempre y cuando cuenten con una correlación positiva. [12]

Capítulo 4

Metodología

Según los objetivos del trabajo de memoria y el proceso que se debe realizar para cumplirlos, es pertinente una metodología de trabajo que incluya el procesamiento y entendimiento de datos, y que esto permita desarrollar modelos que deben ser comparados en base a su desempeño. Tomando como apoyo la bibliografía, específicamente [1], la metodología más indicada es CRISP-DM, ya que reúne las condiciones anteriormente mencionadas.

Bajo este esquema, para realizar el trabajo de memoria, se identifican tres fases necesarias, que se detallan a continuación.

4.1. Fase 1: Entendimiento y preparación de datos

El punto central de esta fase es la exploración de los datos. Existen variables que se encuentran en las distintas bases de datos y están relacionadas entre sí, lo que permite realizar operaciones entre ellas para crear nuevas variables. Como se cuenta con distintas bases de datos, es necesario poder cohesionarlas de modo lógico, para facilitar la implementación de los modelos que se desarrollarán. Para poder comprender los datos, se desarrollará un análisis exploratorio que permita visualizar correlaciones, los valores máximos y mínimos en que se encuentran los datos, y los posibles errores con los que cuentan, para poder solucionarlos lo antes posible.

Generar variables relevantes para el problema, así como elegir las variables importantes entre las que ya se cuenta, supone una tarea imprescindible. Esta tarea se realizará de forma independiente para el modelo que se ejecutará, ya que se necesitan distintas estructuras de datos como entrada.

Otro punto importante es la elección de los datos a estudiar, ya que existen múltiples

criterios para dividir los datos, entre ellos, recorridos, horarios, días. Se ocupará un horizonte de tiempo de datos que permita el funcionamiento de los modelos, y que este sea comparable. Esta fase, bajo la metodología CRISP-DM, es posible avanzar a la siguiente etapa y volver a etapas anteriores, dependiendo de las necesidades del modelo.

4.2. Fase 2: Modelamiento

La segunda fase contempla la etapa de modelamiento. Se destina una fase completa a esta tarea, ya que es la parte más importante del trabajo de memoria: diseñar un modelo que obtenga resultados de sentido lógico, acorde al análisis exploratorio realizado previamente.

4.2.1. Red *LSTM*

La red *LSTM*, como se mencionó previamente, es un tipo especial de red neuronal artificial que tiene la capacidad de almacenar información útil en su memoria artificial a medida que realiza iteraciones. Esta red necesita una preparación especial de los datos, ya que deben ser ingresados como series de tiempo. Por esto, se deben dar formato a los datos de entrada, para poder ejecutar de la mejor forma este algoritmo.

Esta red es configurable en sus distintos hiper parámetros, y no existe una configuración óptima de antemano. La mejor configuración debe ser encontrada de forma empírica, cambiando manualmente los valores de dichos parámetros.

Finalmente, se ejecutará este algoritmo, obteniéndose resultados comparables entre sí, y comparables con el modelo mixto.

4.2.2. Modelo Mixto de Red Neuronal Bayesiana y Filtro de Partículas

La red neuronal Bayesiana, como se explicó previamente, es una red que utiliza la técnica de *dropout* para realizar múltiples predicciones para un solo registro. En este caso, se expondrá la estructura de este modelo y la forma en que ocupa los datos como entrada. Este vector de predicciones, más las características de la distribución de las predicciones, son los datos de entrada del algoritmo de Filtro de Partículas.

El Filtro de Partículas toma estos datos como entrada, y de forma recursiva, predice

el tiempo de llegada de los buses de forma probabilística distribuyendo un conjunto de partículas, formando una función de densidad de probabilidad. Al ordenar las partículas en base a su probabilidad, el Filtro calcula el valor predicho en base al peso de estas partículas. Vale mencionar que este modelo necesita ecuaciones que caractericen el movimiento de un bus, en términos de tiempo. Por esto, en esta etapa se debe enunciar dichas ecuaciones y explicar su funcionamiento.

4.3. Fase 3: Resultados y evaluación

Al terminar la etapa anterior, la siguiente fase es la evaluación del desempeño de cada modelo. Como se generarán divisiones en los datos por horarios y grupos de recorridos, se comparará el desempeño de cada modelo entre las divisiones de datos, para finalmente comparar entre los modelos, y generar una recomendación. Es importante poder comparar entre los modelos, y debido a esta razón se elegirá un bus al azar de un recorrido de cada grupo, y con estos buses se ejecutarán ambos modelos de redes neuronales. Aún así, en los modelos que sean posibles de implementar con todos los buses de los recorridos seleccionados, se realizará esta tarea. El modelo de Filtro de Partículas se debe implementar para un bus a la vez, y esta es la razón principal de elegir un bus de prueba. Como aclaración, en el presente documento un recorrido significa una línea de buses, como por ejemplo, el recorrido 506. Estos recorridos son llevados a cabo por móviles, que son los buses, y cada uno tiene una patente distinta. Es por esto, que un bus de prueba es un bus aleatorio seleccionado del conjunto de buses de un recorrido, con una patente única. Para facilitar la comprensión aún más, se expone lo siguiente: Un bus de prueba del recorrido 506 podría ser el bus de patente ABCD98, que realiza el recorrido 506 de ida y vuelta en un día en específico.

La métrica a ocupar es el error medio cuadrado (*MSE*), ya que, según lo expuesto en el marco teórico, es una métrica de error que reúne las condiciones necesarias para poder medir el desempeño de dichos modelos. El *MSE* tiene las mismas unidades de medida, pero al cuadrado, que en este caso, serían minutos al cuadrado. Por esta razón se analizarán los resultados con el *RMSE*, que tiene las mismas unidades de medida, en este caso, minutos, que es más intuitivo para su análisis.

La metodología *Crisp-DM* se puede observar en la Figura C.1 en anexos. La metodología que se ocupa en el presente trabajo, como se expresó previamente, es una variación de *CRISP-DM*, ya que no se contempla una etapa de despliegue.

Capítulo 5

Desarrollo Metodológico

5.1. Análisis descriptivo de los datos

Se cuenta con datos GPS de la flota de buses de Transantiago, con una base de datos de los puntos geográficos de las rutas, y una base de datos de paraderos. Para realizar modelos comparables en el mismo horizonte de tiempo, y con el fin de acotar el problema, se trabajará con datos del mes de mayo del año 2019, ya que cuenta con mayor cantidad de datos *GPS* que otros meses. Esta base de datos registra la posición de todos los buses de la flota de Transantiago, de todos los días del mes, generando registros cada 30 segundos. La elección del horizonte de tiempo de un mes se debe a que los recorridos del Transporte público de la región Metropolitana cambian constantemente debido a desvíos, construcciones en las calles, o eventos de índole social, y con esto también cambian los paraderos y los tiempos de viaje. En el mes mayo no se generaron grandes cambios para el conjunto de recorridos, y sólo hubo que realizar pequeñas modificaciones en las bases de datos de paraderos y rutas.

Como se mencionó anteriormente, también se cuenta con una base de datos de paraderos, y otra de rutas. Estas bases se complementan entre sí, y permiten visualizar la ruta de un bus. A continuación, se exponen las variables de cada base de datos.

Bases de datos

Datos *GPS*

- Patente del Bus: Patente de la máquina.
- Servicio de SONDA: Servicio identificado por SONDA.
- Servicio de usuario: Servicio que puede ver el usuario en el cartel del bus. Por ejemplo, un usuario en un paradero ve que pasa un bus con el servicio 315, cuando

realmente puede ser un recorrido 315 con alguna desviación por algún evento como una feria o arreglos en las calles.

- Expedición: Identificador único para cada uno de los viajes. Este valor es único para cada viaje.
- Día y hora: Día y hora del registro.
- Latitud y Longitud: Coordenadas de latitud y longitud.
- Coordenada X e Y en UTM: Coordenadas en UTM (Universal Transverse Mercator), que facilita la proyección en un plano. Es otro tipo de coordenadas, que pueden ser transformadas a Longitud y Latitud.
- Distancia en ruta: Distancia del bus desde el comienzo de la ruta o terminal de buses, en unidad de metros.
- Distancia a la ruta: Distancia que indica en cuánto se ha desviado el bus de su ruta, en metros.
- Velocidad instantánea: Velocidad del bus al momento del registro, registrada en Km/H.
- Operador: Empresa operadora del servicio.
- Identificador de sentido: Indica si la expedición es de ida o vuelta.

Datos de información de ruta

- Código de ruta: Código asignado a cada ruta. Una ruta es el conjunto de todos los puntos que forman el viaje de un recorrido. Por ejemplo, la ruta 506 indica todos los puntos por los que debe pasar un bus del recorrido 506.
- Latitud: Latitud del punto registrado.
- Longitud: Longitud del punto registrado.
- X UTM: Coordenada x del punto, en UTM (Universal Transverse Mercator).
- Y UTM: Coordenada y del punto, en UTM (Universal Transverse Mercator).

Datos de paraderos

- Id de paradero: Se cuenta con un código único para cada paradero.
- Nombre: Se identifica con un nombre a cada paradero, como ”Parada 6 Quinta Normal”.

- Código TS: Código del bus que pasa por el paradero. Como varios buses pueden tener el paradero en sus rutas, este valor no es único.
- Código usuario: Código que ven los usuarios del bus que pasa por ese paradero.
- Sentido Servicio: Indica si el bus que pasa por ese paradero lo hace en su recorrido de ida o de vuelta.
- Comuna: Indica en qué comuna se encuentra el paradero.
- Eje: Indica la calle principal en que se encuentra cercano el paradero.
- Desde (Cruce 1): Indica el inicio tramo del paradero.
- Hacia (Cruce 2): Indica el fin tramo del paradero.
- X UTM: Coordenada x del paradero en UTM.
- Y UTM: Coordenada y del paradero en UTM.
- Latitud: Coordenadas de la latitud del paradero.
- Longitud: Coordenadas de la longitud del paradero.

Como se puede apreciar, se tienen distintas bases de datos que cuentan con variables que pueden combinarse. En la Tabla 5.1 se pueden observar la cantidad de valores únicos existentes.

Tabla 5.1: Tabla de valores únicos. Fuente: Elaboración propia.

Variable	**Cantidad**
Patentes	7591
Expediciones	1229
Recorridos	755
Rutas	1442
Paraderos	11256

Para poder observar relaciones entre las variables cuantitativas de la base de datos de *GPS*, se generó un gráfico de correlaciones basado en el cálculo de Pearson, que se observa en la Figura 5.1. Según el gráfico, es difícil notar relaciones lógicas, aunque, por ejemplo, la correlación entre la distancia a la ruta y la velocidad es negativa. Esto podría significar que a mayor desvío de la ruta, es menor la velocidad, lo que podría indicar que existe congestión vehicular o algún embotellamiento.

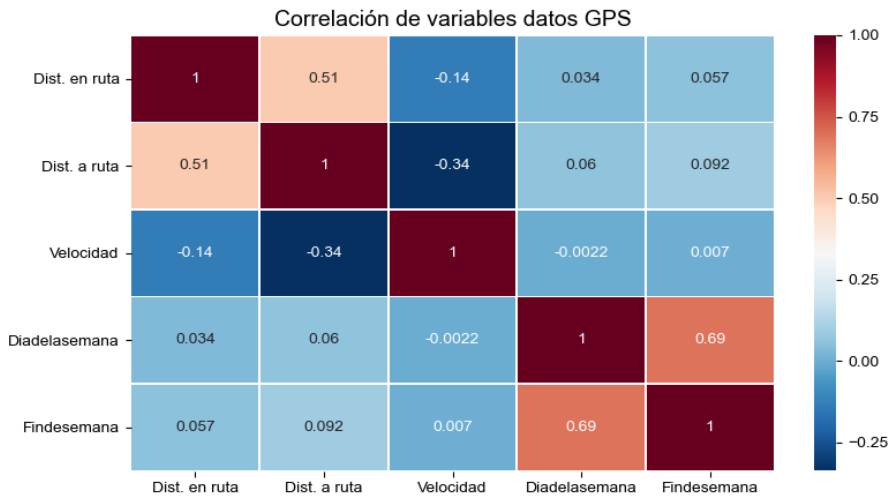

Figura 5.1: Correlación datos *GPS*. Fuente: Elaboración propia.

En primera instancia, se esperaba estudiar 16 recorridos distintos, por solicitud de la empresa. Esto significa una tarea fuera del alcance de este trabajo de título. Por esto es que se agruparon estos 16 recorridos en tres conjuntos. Estos conjuntos de recorridos tienen inicios y fines de rutas en similares coordenadas de latitud y longitud, y cantidad de paraderos también similares . En las Figuras 5.2 y 5.3 se pueden observar las rutas y los grupos de recorridos, respectivamente. De dichos grupos, se seleccionó un recorrido de cada uno de forma aleatoria, resultando electos el recorrido 301, 315e y 506.

El grupo 301 tiene características de ruta norte-sur, bajo similares coordenadas, tal como el grupo 506 de rutas este-oeste. El grupo 315e comprende recorridos cortos, y algunos que son variaciones de recorridos normal, como el mismo 315e, ya que es el recorrido expreso de 315.

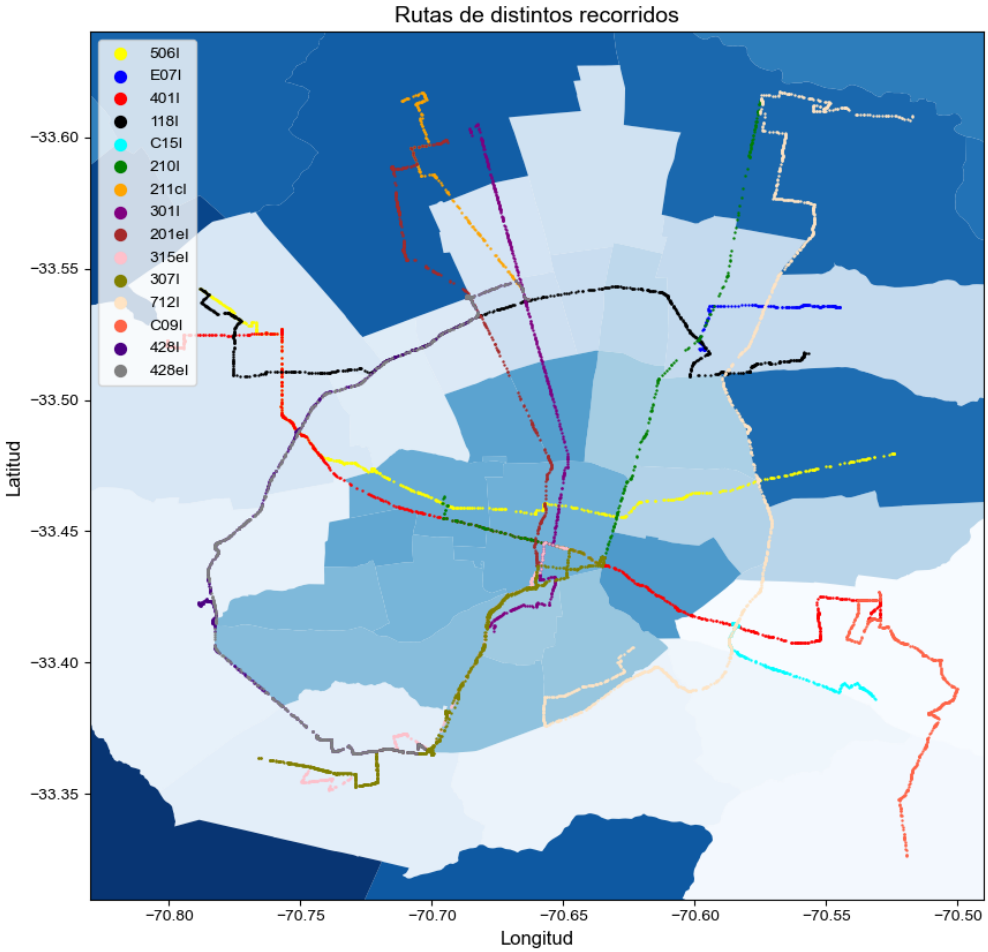

Figura 5.2: Rutas de recorridos. Fuente: Elaboración propia.

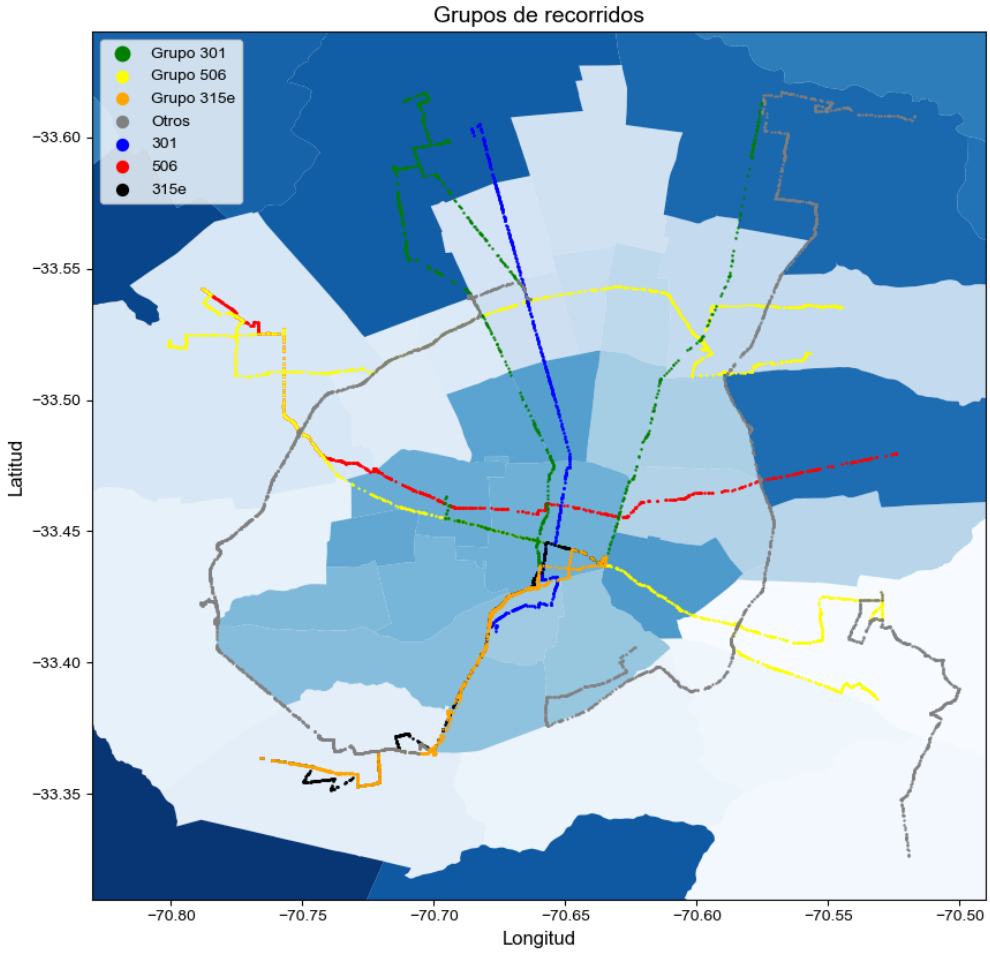

Figura 5.3: Grupos de recorridos. Fuente: Elaboración propia.

Con el fin de seguir acotando, no se dividieron los datos entre días de semana y fin de semana. En vez de eso, sí se dividió la base de datos de *GPS* en horarios, descritos oficialmente por Transantiago: horario valle (09:00 a 17:59 hrs.) y horario punta (18:00 a 19:59 hrs.). Transantiago define estos horarios y distintos cobros para ellos, teniendo costos mayores de pasaje en horario punta, y mayor congestión vehicular y de metro. Solo se trabajó con el sentido de los recorridos de ida, no de vuelta, ya que las rutas son similares geográficamente y en tiempos de viaje, como se puede apreciar en la Tabla A.1 en Anexo A. Aun así, este aspecto se puede cambiar fácilmente en los parámetros de los modelos. Con esto, se evaluarán los modelos para los tres recorridos seleccionados, que son 301, 315e y 506, para dos horarios, que son horario valle y horario punta. Esto genera 6 conjuntos de datos distintos, que implica a tener 6 modelos distintos.

En la Figura 5.4, se muestra una gráfica de los paraderos de los recorridos seleccionados. El recorrido 301 tiene 76 paraderos, el recorrido 315e tiene 39 paraderos, y el recorrido 506 tiene 90 paraderos.

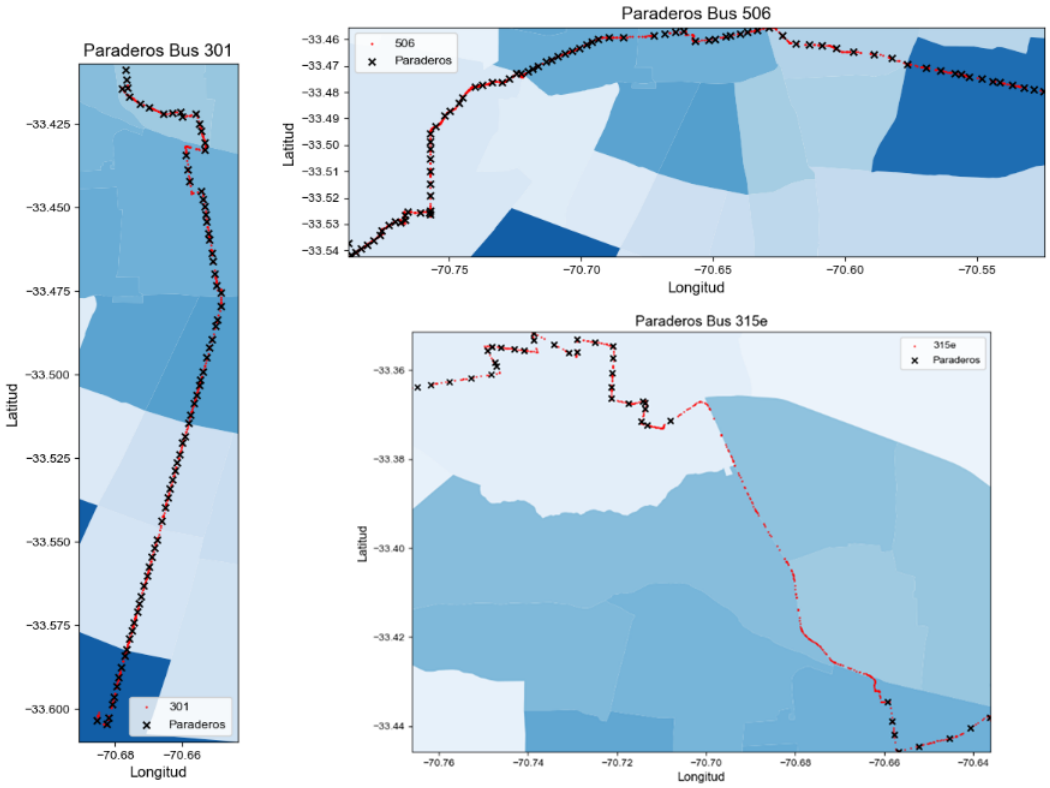

Figura 5.4: Paraderos de recorridos seleccionados. Fuente: Elaboración propia.

Seleccionando buses aleatorios de los recorridos seleccionados, se generaron gráficos de velocidad en horario punta y horario valle. Por ejemplo, se observa en la Figura 5.5 y 5.6 el gráfico de velocidad del recorrido 301 en horario valle y horario punta, respectivamente. se observa que los gráficos no difieren en alto grado, y las velocidades promedio son distintas de forma mínima. Los gráficos de los buses 315e y 506, se encuentran en el anexo A, y sucede lo mismo en términos de la diferencia de velocidad.

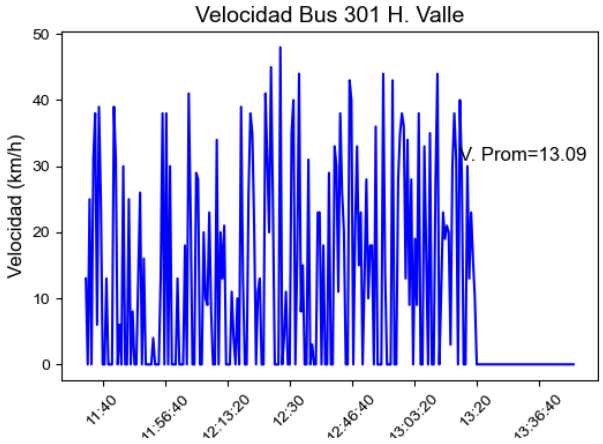

Figura 5.5: Velocidad horario valle bus 301. Fuente: Elaboración propia.

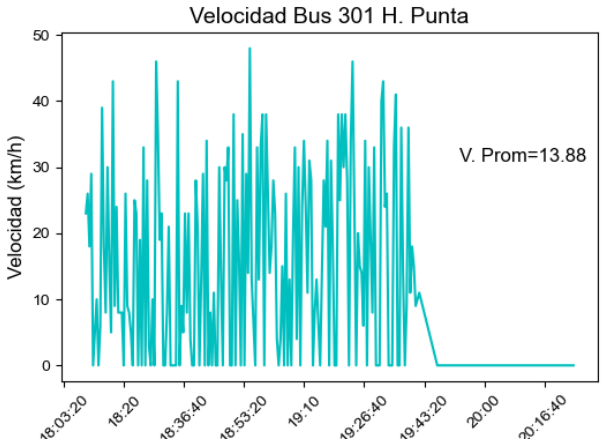

Figura 5.6: Velocidad horario punta bus 301. Fuente: Elaboración propia.

También, se generaron histogramas de los tiempos de viaje de los recorridos seleccionados para ambos horarios, para visualizar la distribución de los datos, como se observa en la Figura 5.7, tomando los buses 315e. En estos gráficos se puede ver que se tiene una tendencia bien definida, y no existe mucha dispersión en los datos. Esto sucede también para el recorrido 506 y 301. Los gráficos restantes se encuentran en el anexo A.

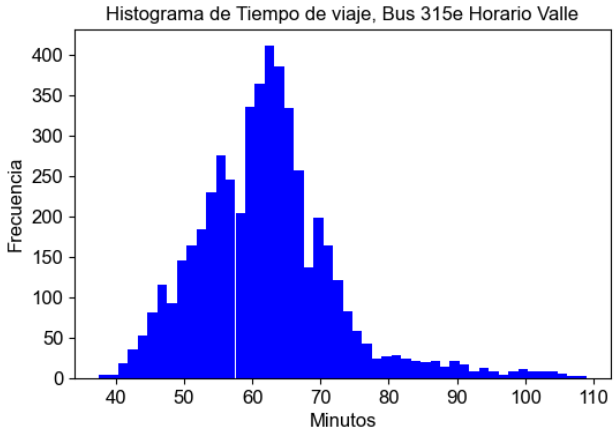

Figura 5.7: Histograma de tiempo de recorridos bus 315e, horario valle. Fuente: Elaboración propia.

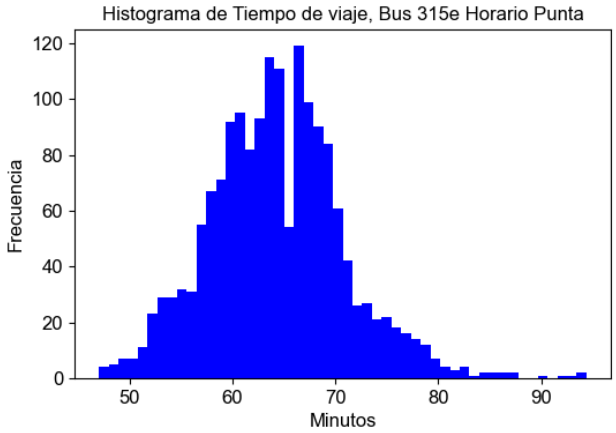

Figura 5.8: Histograma de tiempo de recorridos bus 315e, horario punta. Fuente: Elaboración propia.

En la Tabla 5.2 se muestra el promedio de tiempo de viajes, en minutos. La mayor diferencia se encuentra en el recorrido 301, donde la diferencia es de alrededor de 11 minutos. En el recorrido 315e y 506, no existe gran diferencia, pero el tiempo de horario punta es mayor de forma mínima. Donde sí radica una diferencia mas notoria, es en la dispersión de estos tiempos de viaje. En el horario Punta es de menor valor, lo que podría suceder dado que en este horario es más probable encontrar congestión a la misma hora, por lo que el tiempo de viaje no varía mucho.

Tabla 5.2: Tabla de tiempo promedio de recorridos. Fuente: Elaboración propia.

	Tiempo de viaje (μ, σ)	
Bus	Horario Valle (minutos)	Horario Punta (minutos)
301	85.86, 14.01	95.70, 10.85
315e	62.06, 10.27	64.30, 6.61
506	98.54, 14.98	98.94, 9.10

Finalmente, para cada registro se creó una variable que toma el valor del número del día de semana, siendo día lunes el valor 1 y domingo el valor 7, y una variable binaria que toma valor 1 si es día de semana, y valor 0 si es día de fin de semana. Este análisis exploratorio es general, con el objetivo de mostrar distintos aspectos de los datos. Para los dos modelos que se ejecutarán, se deben crear variables necesarias para adaptar los datos acorde al funcionamiento de cada uno. Es por esto, que en la sección de desarrollo de cada modelo, se desarrollará un pequeño análisis de datos y creación de variables relevantes.

5.2. Modelo *LSTM*

El objetivo general del presente trabajo es elaborar un prototipo de sistema predictor de tiempos de espera basado en modelos de Machine Learning. Para esto, se desarrolló un modelo de red neuronal artificial *LSTM*. Como se mencionó anteriormente, este modelo tiene la capacidad de almacenar información útil en su memoria interna, y por esto se tiene como hipótesis que podría tener un buen funcionamiento en la resolución de este problema.

Preparación de datos

Las variables de la base de datos *GPS* por sí solas no son un una base de datos válida como entrada de este modelo. Por esto, se crearon dos variables distintas en base a cálculos con la base de datos. El procedimiento para esto fue el siguiente: Se seleccionó cada bus de un recorrido, y para cada registro, se calculó el tiempo y la distancia a cada paradero. El tiempo a cada paradero se adjuntó como variable a la base de datos, como ”tiempo al objetivo”, y la distancia a cada paradero, como ”distancia al objetivo”, como se observa en el ejemplo de la Figura 5.9. Calcular el momento en que un bus llega al paradero de forma precisa es prácticamente improbable, debido al error de los sensores de *GPS*. Por esto fue que se identificó que un bus llega a un paradero cuando entra en un rango de metros a la redonda, con un valor de diferencia. Este valor se calculó como

la diferencia entre la posición del bus y la posición del paradero. Si este valor, en coordenadas UTM, es menor o igual a 0.01, se acepta que el bus se encuentra en el paradero. El valor 0.01 en coordenadas UTM tiene una equivalencia a 10 metros. También se creó una variable que indica el número de la hora del registro (por ejemplo, si el registro es de las 16:15:40, esta variable toma el valor 16).

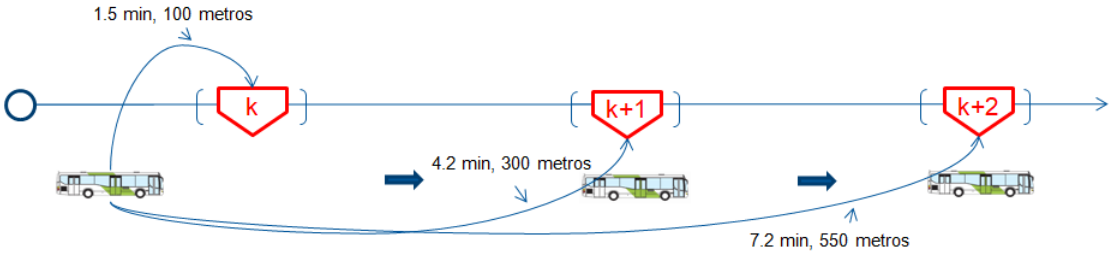

Figura 5.9: Visualización de variables creadas para red *LSTM*. Fuente: Elaboración propia.

En la figura anterior, se expone el ejemplo del registro GPS de un bus. Se calculó que para llegar al paradero ”k”, el bus se demora 1.5 minutos y está a una distancia de 100 metros, y que para llegar al paradero ”k+1”, el mismo bus se demora 4.2 minutos, y se encuentra a 300 metros. Este cálculo se realizó para todos los registros de los buses de los recorridos seleccionados. Finalmente, la variable que se intenta predecir es ”tiempo al objetivo”, ya que calcularía cuánto demora a cada paradero de su ruta. En la Figura 5.10, se observa el cálculo de correlación de Pearson incluyendo las nuevas variables.

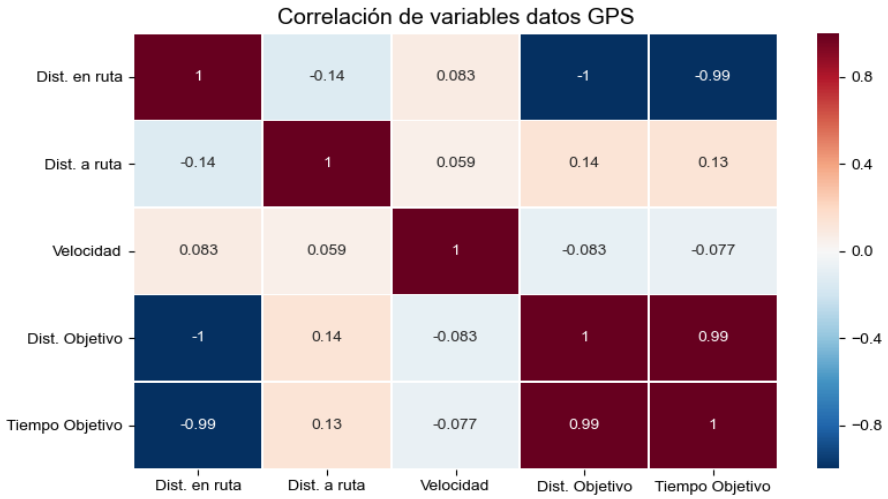

Figura 5.10: Correlación de variables para red *LSTM*. Fuente: Elaboración propia.

En la Figura 5.10, se puede notar que los coeficientes de correlación son los esperados de forma lógica, ya que al aumentar la distancia en ruta, disminuye el tiempo en que el bus llega a un paradero, y al disminuir la distancia al objetivo, disminuye también el tiempo al objetivo. Esto muestra congruencia entre las variables antes mencionadas.

También es necesario cambiar el formato de las variables de entrada, y convertirlas en series de tiempo para que el modelo pueda procesarlas. Se definieron series de tiempo de 10 datos, esto quiere decir que cada serie cuenta con 10 registros hace atrás, siendo cada registro entre 30 segundos, se tiene un total de 5 minutos captada en cada serie de tiempo. Este número depende del modelo y el efecto que se quiera captar. Por ejemplo, para un modelo anual o semestral, es necesario al menos una serie que contenga 50 registros. En este caso, como el suceso que se quiere predecir es la llegada de un bus al paradero, es necesario captar el tiempo, la posición y la velocidad del bus momentos antes, ya que se puede esperar de que sean factores importantes.

En base a lo anterior, las variables de entrada del modelo son: coordenadas del bus en UTM, velocidad del bus, hora, distancia a la ruta y distancia al punto. La variable a predecir es el tiempo objetivo. La forma de representar estas variables en la base de datos de entrada, se compone de conjuntos de datos. Cada conjunto se compone de un recorrido, y sus subconjuntos son los buses del recorrido. Cada subconjunto contiene filas, que incluyen el tiempo objetivo al paradero, la distancia objetivo, la velocidad del bus y las coordenadas del bus.

Configuración de la red

El conjunto total de datos se dividió en tres subconjuntos: entrenamiento (40 % de los datos), validación y prueba, estos últimos contemplando un 50 % cada uno de los datos restantes. El conjunto de datos totales comprende todo el mes antes mencionado, por lo que se divide en días de forma aleatoria.

La red *LSTM* cuenta con distintos parámetros configurables. El primero, llamado *epoch*, indica la cantidad de veces que la red recorre los datos de entrenamiento y actualiza los pesos de las neuronas. El otro es el *learning rate*, que determina el tamaño de cada ciclo, buscando minimizar la función de pérdida de la red. Si este parámetros es de gran magnitud, la red puede caer en el error de identificar sólo un mínimo global. Estos parámetros varían para cada red, y no existe una regla general para elegir los mejores valores, siendo la mejor opción una estrategia de prueba y error, de forma heurística. En base a esto, después de buscar los mejores valores en base a prueba y error, se configuró cada modelo con 100 *epochs*, y un *learning rate* de 0.0001. El *MSE* fue seleccionado como

función de pérdida.

Otra configuración importante de la implementación de las redes, es que al calcular las variables ”tiempo al objetivo” y ”distancia al objetivo”, se calcula con cada registro *GPS*. Con estos datos , la red realiza predicciones desde distancias mayores y tiempos mayores al que probablemente esté una persona. Por ejemplo, que alguien espere una micro que tiene estimado 50 minutos de llegada al paradero, es distinto a que esté a 10 minutos del paradero. Por esto, se generaron umbrales de tiempo de 50, 20, 10 y 5 minutos, para calcular el rendimiento de la predicción de la red. En el presente documento se define como umbral de tiempo la máxima distancia del bus al paradero traducida en minutos. Por ejemplo, el umbral de tiempo de 10 minutos significa que el bus está a lo más a una distancia del paradero tal que su tiempo de llegada calculado es de 10 minutos. En los resultados, se mostrará el valor del umbral de 10 minutos, y en el anexo B se pueden encontrar los gráficos de los otros umbrales.

Implementación de las redes y resultados

Como se mencionó en la metodología, los modelos deben ser comparables. Con esto, se entrenaron los modelos, y se probaron con el porcentaje de los datos antes descrito. Pero también se seleccionó un bus al azar de cada conjunto de datos, con el objetivo de poder comparar la predicción de las redes *LSTM* con el Filtro de Partículas. Como el horario Punta comprende menos horizonte de tiempo que el horario Valle, implica que se generen menos datos también. Como se podrá observar a continuación, en distintos casos la cantidad de datos tiene una diferencia de gran magnitud, por lo que se generaron gráficos con distintas escalas, con el fin de poder observar de buena forma el error del modelo. A continuación, se presentan los resultados de los buses de los recorridos seleccionados.

Resultados recorrido 301

Al ejecutar la red para el recorrido 301, los valores obtenidos para todo el conjunto de prueba y para el bus de prueba, son similares, ya que el valor de la métrica *RMSE* es menor para el horario valle, en ambos casos, como se puede apreciar en las Figuras 5.11 y 5.12. Este resultado podría esperarse, ya que, como se mencionó anteriormente, el horario punta se caracteriza por poseer mayor congestión vehicular, y mayor cantidad de personas ocupando el servicio de transporte. Esto podría ser la causa de que la predicción para este horario sea más errada.

En la Figura 5.11, el error de predicción se centra en 0, pero se observa que existe mayor error positivo que negativo. Esto significa que la red predice que el bus llegará antes del tiempo real.

En la Figura 5.12, la situación es similar a la de todo el conjunto de prueba. El error se centra en 0, pero se observa que la predicción es errónea en casos donde el tiempo de llegada real es mayor. Lo interesante de este gráfico es que existen datos multimodales, que podrían generarse por distintas circunstancias. En este caso, es probable debido a la representación en el gráfico, ya que puede tratar de aproximar el valor del error a números enteros, y no existiría una concentración empírica del error.

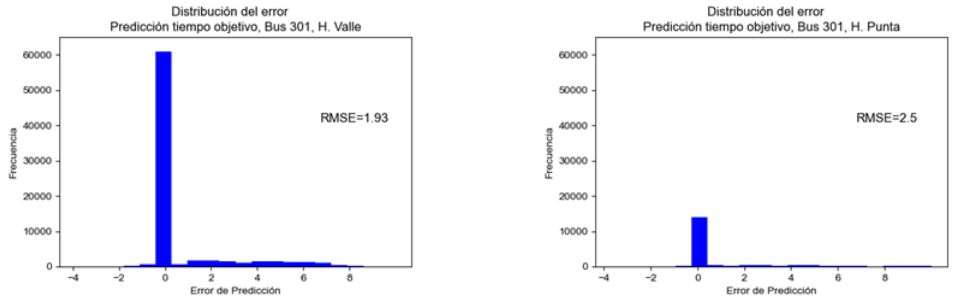

Figura 5.11: Resultados red *LSTM* para todo el conjunto de prueba 301, umbral 10 minutos. Fuente: Elaboración propia.

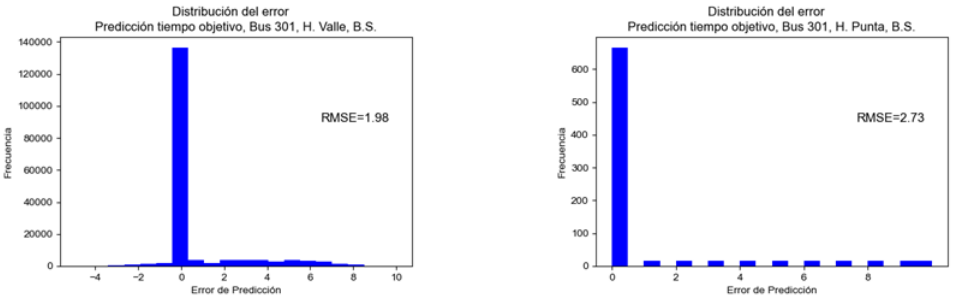

Figura 5.12: Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbral 10 minutos. Fuente: Elaboración propia.

Resultados recorrido 315e

Los resultados del recorrido 315e son similares al del recorrido 301, ya que las redes obtienen peor desempeño en la predicción del horario punta, como se observa en las Figuras 5.13 y 5.14. Lo interesante de estos resultados es la similitud de desempeño entre el conjunto de prueba y el bus seleccionado. Esto indica que el bus de prueba puede ser una muestra representativa del conjunto en su totalidad.

Al igual que en el recorrido 301, el error de la Figura 5.13 se centra en el valor 0 para el horario valle, no así en el horario punta, donde parece existir un mayor error de predicción negativo, lo que implica que el valor real del tiempo de llegada de los buses es menor al que predijo la red. Esta es la razón de que el *RMSE* sea de alrededor de 8.59 minutos.

En la Figura 5.14, se observan resultados similares que en todo el conjunto de prueba 315e. En la figura de horario punta, se observa que la predicción de la red tiende totalmente a ser un tiempo mayor que el tiempo real.

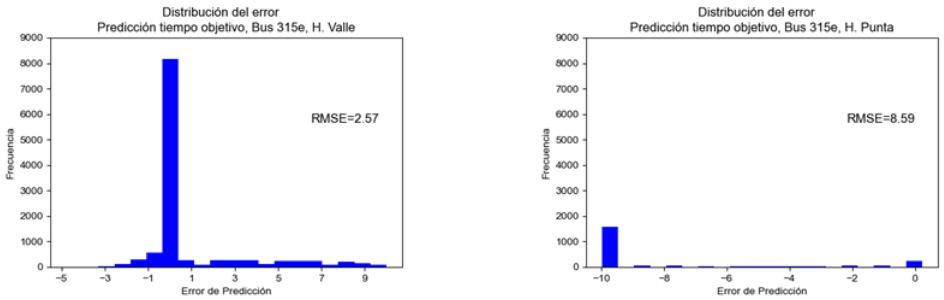

Figura 5.13: Resultados red *LSTM* para todo el conjunto de prueba 315e, umbral 10 minutos. Fuente: Elaboración propia.

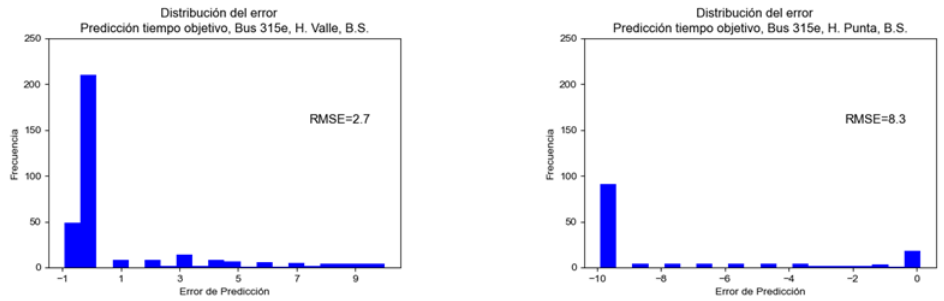

Figura 5.14: Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbral 10 minutos. Fuente: Elaboración propia.

Resultados recorrido 506

Por último, los resultados del recorrido 506, para todo el conjunto de prueba y un bus de prueba, no difieren en mayor medida, y tampoco entre horarios. La precisión del modelo, bajo la métrica *RMSE*, es alta, lo que significa que el modelo predice bien para ambos conjuntos. También es posible inferir que el bus de prueba elegido es un buen representante del conjunto, en base a las Figuras 5.15 y 5.16.

Al igual que en el recorrido 301, parece ser que la diferencia de horarios es mínima, y la predicción se centra en el valor 0 para el conjunto de prueba y el bus seleccionado.

Aún así, se observa que en los valores distintos de 0, la red tiende a predecir un tiempo de llegada mayor al tiempo real.

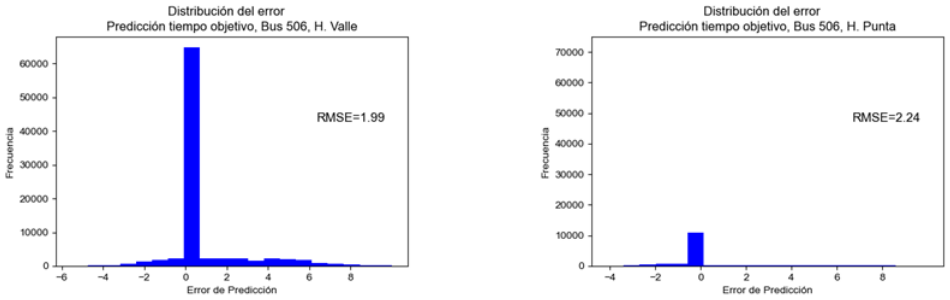

Figura 5.15: Resultados red *LSTM* para todo el conjunto de prueba 506, umbral 10 minutos. Fuente: Elaboración propia.

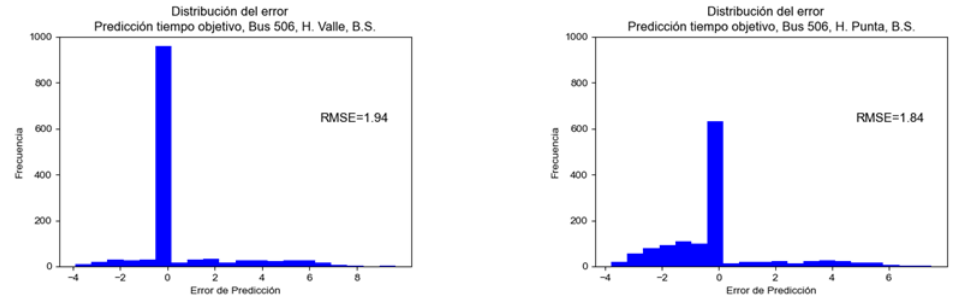

Figura 5.16: Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbral 10 minutos. Fuente: Elaboración propia.

Análisis de resultados y conclusiones

Habiendo comparado entre resultados del mismo recorrido en diferentes horarios, con el umbral de 10 minutos, se generó la Tabla resumen 5.3. En esta tabla se observa que se obtienen desempeños similares en diferencia de horarios en los tres recorridos, 301, 315 y 506. El peor desempeño se obtiene en el recorrido 315e. No existe una razón concreta de por qué esto sucede, pero se piensa que se debe al tipo de recorrido. El recorrido 315e es un recorrido expreso, lo que quiere decir que no se detiene en todos los paraderos de la ruta normal del recorrido 315. Como se puede observar en 5.4, existe un tramo del bus 315e donde no existen paraderos, y es casi un tercio de la ruta. Esto generalmente sucede en los recorridos expreso, y los recorridos del conjunto 315e. Esto podría ser una de las causas del mal desempeño de la red en este tipo de recorridos, probablemente por la imprecisión del tiempo que toma recorrer este tramo, o por la cantidad de datos *GPS* registrados.

El menor error se encuentra en el recorrido 506, por lo que se puede concluir que el modelo se adapta mejor a la predicción de este tipo de recorridos. En el caso del recorrido 315e, el modelo no predice bien el tiempo de llegada de los buses a los paraderos en horario punta.

Lo más importante de estos resultados es que tienen en común que en el horario punta se obtiene mayor error de predicción al calcular que un bus llega un paradero, probablemente por las condiciones antes mencionadas, mayor congestión vehicular y de usuarios del sistema de transporte. Esto se observa en el *RMSE*, ya que el error en minutos es mayor, teniendo, por ejemplo, un error de casi 9 minutos en el recorrido 315e en el horario punta.

Tabla 5.3: Tabla resumen de resultados *LSTM*, umbral de 10 minutos.

Bus	Datos	Horario Valle *RMSE* (min)	Horario Punta *RMSE* (min)
301	Todo el conjunto de prueba	1.93	2.50
	Bus selec.	1.98	2.73
315e	Todo el conjunto de prueba	2.57	8.59
	Bus selec.	2.70	8.30
506	Todo el conjunto de prueba	1.99	2.24
	Bus selec.	1.94	1.84

Como se mencionó anteriormente, se generaron umbrales de tiempo para evaluar los resultados de los modelos. Previamente se expusieron los resultados del umbral de 10 minutos.

Se generaron tablas resumen del desempeño del modelo para un umbral de tiempo de 50 minutos, 20 minutos y para 5 minutos, las Tablas 5.4, 5.5 y 5.6 presentan estos resultados, respectivamente.

Tabla 5.4: Tabla resumen de resultados *LSTM*, umbral de 50 minutos.

Bus	Datos	Horario Valle *RMSE* (min)	Horario Punta *RMSE* (min)
301	Todo el conjunto de prueba	7.31	9.61
	Bus selec.	7.49	10.50
315e	Todo el conjunto de prueba	12.97	23.03
	Bus selec.	13.97	17.51
506	Todo el conjunto de prueba	6.38	11.27
	Bus selec.	6.4	13.28

Tabla 5.5: Tabla resumen de resultados *LSTM*, umbral de 20 minutos.

Bus	Datos	Horario Valle *RMSE* (min)	Horario Punta *RMSE* (min)
301	Todo el conjunto de prueba	2.43	4.17
	Bus selec.	2.66	4.99
315e	Todo el conjunto de prueba	5.95	15.44
	Bus selec.	6.44	14.35
506	Todo el conjunto de prueba	2.96	5.39
	Bus selec.	3.11	6.67

Tabla 5.6: Tabla resumen de resultados *LSTM*, umbral de 5 minutos.

Bus	Datos	Horario Valle *RMSE* (min)	Horario Punta *RMSE* (min)
301	Todo el conjunto de prueba	1.03	1.10
	Bus selec.	1.05	1.03
315e	Todo el conjunto de prueba	1.12	4.46
	Bus selec.	1.14	4.36
506	Todo el conjunto de prueba	1.12	1.14
	Bus selec.	1.11	1.13

De las tablas anteriores, se observa que en el umbral de 50 minutos los resultados del *RMSE* son mayores que en los otros umbrales de tiempo, teniendo un error de 10 minutos o más. Esto significa que las redes predicen de peor forma la llegada de un bus a un paradero si está a 50 minutos de lejanía. Esto sucede porque, probablemente, predecir el tiempo de llegada de un bus tan lejano es más difícil que uno cercano, ya que múltiples eventos pueden pasar en la ruta restante. Al igual que en los otros umbrales de tiempo, la peor predicción se obtiene en el recorrido 315e, lo que da aún mas luces de que este tipo de recorridos no se adaptan de buena forma al funcionamiento de las redes *LSTM*.

Al variar los umbrales de tiempo, la predicción se comporta de la misma manera que en el umbral de 10 minutos, donde el recorrido 301 y 506 predicen un tiempo mayor de llegada, generando errores positivos, no así el recorrido 315e, donde sucede lo contrario.

A medida que disminuye el tiempo del umbral, mejora la predicción, como se observa en la Figura 5.17, donde se graficó el *RMSE* por umbral de tiempo para los datos del conjunto de prueba de horario Valle . Comparando entre distintas tablas, al pasar del umbral de 50 minutos al de 20 minutos, mejora la predicción, y el horario punta nuevamente posee peores resultados que el horario valle. Al observar el umbral de 5 minutos de lejanía de un bus al paradero, la predicción mejora notablemente en comparación a los otros umbrales de tiempo. Se observa, en base al *RMSE*, que el error de predicción no supera los 4.46 minutos, que es el valor más alto, en todo el conjunto de prueba del recorrido 315e. Esto sucede debido a que predecir la llegada de un bus a pocos minutos de distancia parece más fácil, ya que depende directamente de los datos previos, y proba-

blemente de la velocidad del bus. En Anexo B se encuentra la Tabla B.1, que resume las tablas anteriores, evidenciando la disminución del *RMSE* al disminuir el umbral de tiempo.

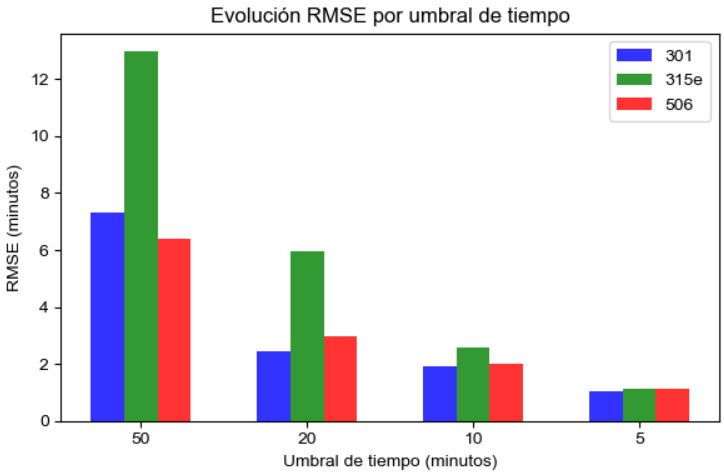

Figura 5.17: Métrica *RMSE* para cada umbral de tiempo para todo el conjunto de prueba, horario Valle. Fuente: Elaboración propia.

Como conclusión, las redes *LSTM* implementadas para los distintos conjuntos de datos, tienen un desempeño distinto en horarios distintos, siendo el horario valle el de mejor valor del *RMSE*.

Al analizar los umbrales de tiempo, las redes predicen mejor el tiempo de llegada de buses de los tres recorridos en el umbral de 5 minutos. A mayor cantidad de minutos de lejanía del bus, peor es la predicción.

En el anexo B se encuentran las figuras del valor real de tiempo al objetivo y el valor predicho por las redes *LSTM* para el bus de prueba. Los resultados para todo el conjunto de prueba son similares, y se comportan de la misma forma. Como se generaron series de tiempo de 7 registros, los resultados muestran cuando el bus llega a un paradero, donde el tiempo va decreciendo hasta llegar a 0. Lo importante y notorio de estas figuras es que se observa que la predicción mejora al disminuir el tiempo, como se explicó antes, pero en todas las ocasiones, las redes subestiman el tiempo, indicando que el bus llega antes de lo esperado.

El modelo generado en esta sección mide el desempeño general de predecir el momento en que un bus llega al paradero, y no realiza distinción en tramos de ruta, como se implementará en el modelo de Filtro de Partículas.

5.3. Modelo Mixto de Red Neuronal Bayesiana y Filtro de Partículas

El modelo de red neuronal *LSTM* genera predicciones del tiempo de llegada de un bus a los paraderos, tomando como datos la ruta completa. En esta ocasión, se tratará el problema de forma diferente.

El Filtro de Partículas genera predicciones basadas en una distribución de probabilidad, siguiendo el camino de un móvil. Por esto es que no funcionaría generando predicciones para la ruta completa. La forma de generar el problema, para este modelo, se basa en dividir la ruta en tramos. Al dividir la ruta en tramos, se puede generar una predicción de tiempo de llegada del bus a cada fin del tramo, que es un paradero. En el mejor de los casos, cada tramo es cada uno de los caminos entre paraderos consecutivos. Como el presente trabajo se trata de desarrollar un prototipo basado en modelos, se generó una división de la ruta de los buses en tres tramos, tomando como criterio de división los paraderos que visualmente dividen la ruta, y que tienen mayor cantidad de registros de buses que pasan por ellos. Estos registros de cuando un bus se encuentra en un paradero se calcularon de la misma forma que en las redes *LSTM*.

En la Figura 5.18, se muestran los paraderos escogidos bajo el criterio antes expuesto. El primer paradero seleccionado, es el terminal de buses, que sirve para registrar la hora en que el bus inició su recorrido. Se espera que el tiempo entre el terminal y el primer paradero seleccionado sea mínimo, por lo que no se toma en cuenta como un tramo en sí. Los paraderos seleccionados con la ”S” de color amarillo definen los tramos, que para los recorridos seleccionados, 301, 315e y 506, dividen la ruta en tres tramos.

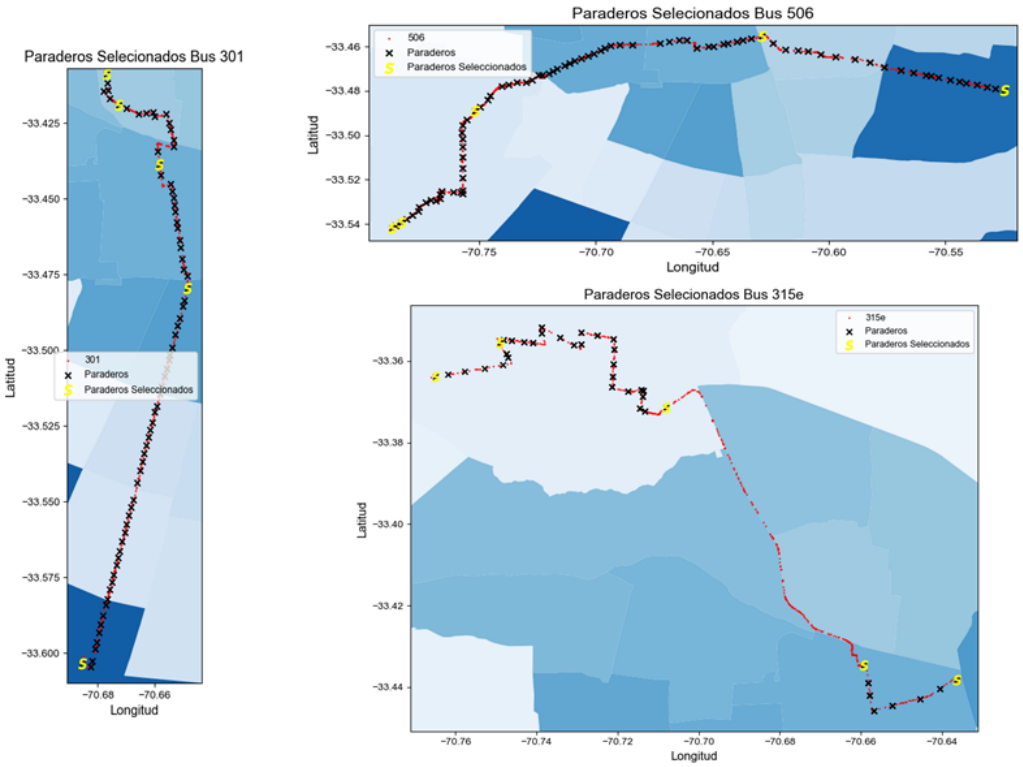

Figura 5.18: Paraderos seleccionados para modelo mixto. Fuente: Elaboración propia.

La idea de este modelo es que para cada tramo se realice una predicción de tiempo de llegada de forma probabilística, es decir, que se pueda identificar una función de densidad de probabilidad del tiempo de llegada, siendo el valor estimado el que tenga la probabilidad mayor. Para comparar con el modelo *LSTM*, se calculará el error de la ruta completa, bajo la métrica *RMSE*. El primer paso para obtener los resultados esperados, es desarrollar redes neuronales Bayesianas para cada una de las rutas, en los dos horarios distintos: valle y punta.

En la Tabla 5.7, se observa el tiempo promedio por tramo de los recorridos seleccionados. Se puede notar que, aunque sean horarios distintos, la diferencia de tiempo que le toma a un bus recorrer el tramo seleccionado, no es de gran magnitud, y parece ser que esta distinción entre horarios no afecta tanto en los tiempos de viaje.

Tabla 5.7: Tiempo promedio de tramos, por recorrido y horario. Fuente: Elaboración Propia.

Recorrido	Horario	Tiempo promedio primer tramo (min.)	Tiempo promedio segundo tramo (min.)	Tiempo promedio tercer tramo (min.)
301	Valle	20.49	21.39	43.59
	Punta	20.10	24.29	47.48
315e	Valle	25.36	15.08	15.60
	Punta	26.27	14.79	17.44
506	Valle	25.68	42.91	28.94
	Punta	24.16	41.16	29.25

Redes neuronales Bayesianas

Preparación de datos

Como se mencionó anteriormente, el funcionamiento de una red neuronal Bayesiana radica en realizar ”T” predicciones para cada registro, obteniéndose así un conjunto de valores para cada uno de estos registros de forma independiente.

Lo que se desea predecir es el tiempo que le toma a un bus para cada uno de los tramos antes expuestos. Para esto, después de dividir los datos por horario y por bus, se calculó el tiempo de que le toma a cada bus de la base de datos recorrer cada tramo. Para esta gama de modelos, que es uno para cada bus, horario y tramo, resultan 18 redes neuronales Bayesianas. Se generó una base de datos que contiene las siguientes variables: distancia en ruta del bus al inicio del tramo, la velocidad del bus en ese instante, la distancia del paradero del fin del tramo, la variable día de la semana, la variable de fin de semana, la hora en que el bus inició el recorrido, y la hora actual. La variable a predecir es el tiempo que le toma al bus recorrer el tramo.

En la Figura 5.19, se observa la correlación de las variables para el modelo de redes neuronales Bayesianas. De la figura, se puede apreciar que ninguna correlación tiene alto valor, por lo que realizar inferencias sobre estos valores no es adecuado. La correlación de la variable de día de semana y fin de semana es alta, de forma lógica, ya que siempre el día sábado y domingo, que en la variable de día de semana toman el valor 6 y 7, en la variable de fin de semana toman el valor 1.

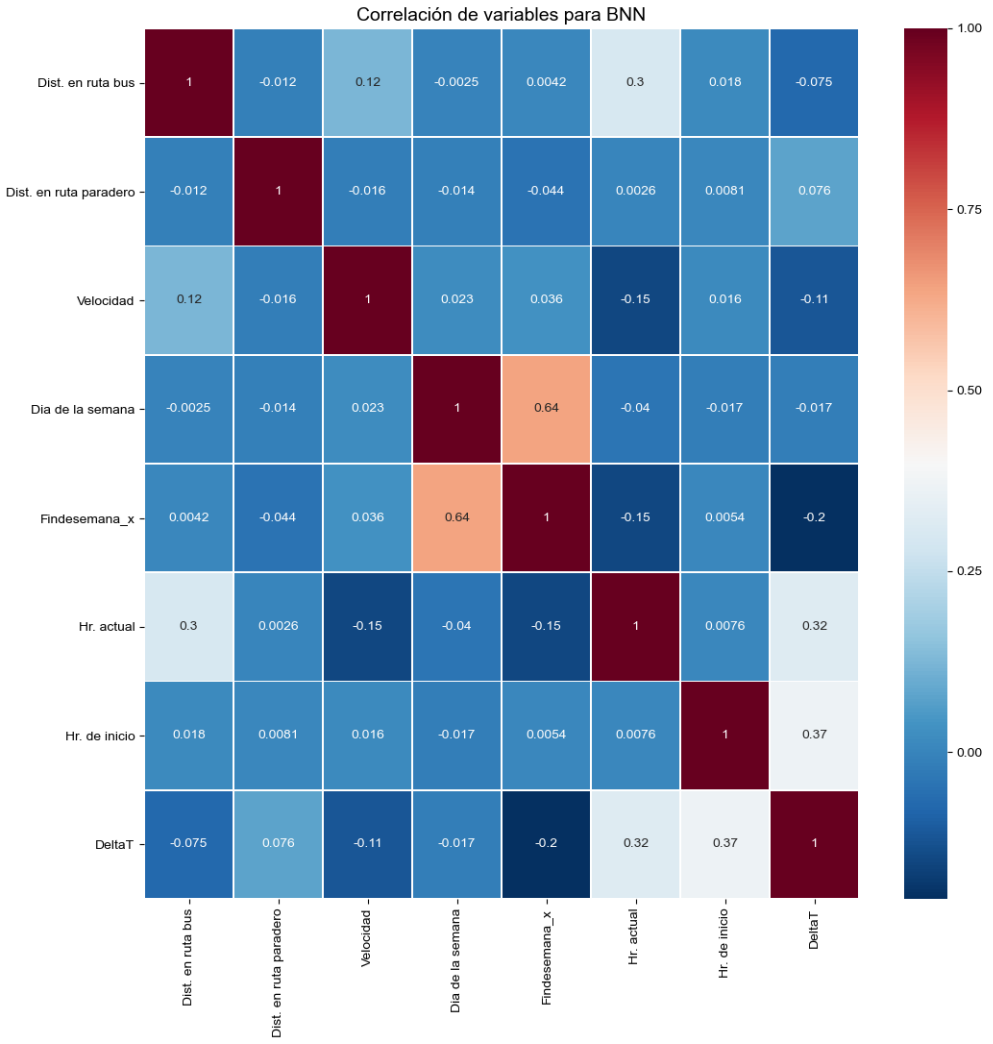

Figura 5.19: Correlación de variables para modelos de redes neuronales Bayesianas. Fuente: Elaboración propia.

Para efectos del cálculo del tiempo predicho para cada tramo, ya que se realizan ”T” cálculos de este valor, se calcula la media de estos valores y se almacena como el valor predicho. Esto se realiza para el conjunto de prueba. Para los buses de prueba de los recorridos seleccionados, este procedimiento no se realiza, ya que la cantidad ”T” de predicciones se procesa en el algoritmo de Filtro de Partículas, sin haber realizado cálculos de la media o la varianza de este valor.

Configuración de las redes neuronales Bayesianas

Las redes Bayesianas, igual que en el caso de las redes *LSTM*, tienen distintos parámetros configurables. Entre ellos, la cantidad de *epochs*, la cantidad de capas y la tasa

de *dropout*, que indica el porcentaje de las neuronas que se desactivarán por cada paso de la red. Después de realizar pruebas variando los parámetros, se eligió la configuración recomendada por [10], configurando las 18 redes con 3 capas, cada una con 200 neuronas, con 100 *epochs*, y una tasa de *dropout* de 0.05. Al igual que en el modelo *LSTM*, se dividieron los datos de cada conjunto en 60 % de entrenamiento y 40 % de prueba.

Análisis de resultados y conclusiones

Se implementaron las redes para los conjuntos de horarios y recorridos, probando su desempeño con todo el conjunto de prueba y con los buses de prueba seleccionados de los recorridos.

En la Tabla 5.8 se observan los resultados del desempeño de las redes para el todo el conjunto de prueba. En él se observa que para los tres buses no existe ninguna tendencia marcada de que en los tres tramos la predicción sea mejor para un horario u otro. Aún así, generalmente los valores más altos de *RMSE* se obtienen para el horario punta. El valor del *RMSE*, en minutos, indica que el error de todos los conjuntos no sobrepasa los 5 minutos, y tiene un promedio alrededor de 4 minutos. Esto muestra que el desempeño de las redes no es deficiente si se piensa que todas tienen la misma configuración.

Si bien los valores del *RMSE* son altos, esto puede deberse a que el valor predicho es la media de los ”T” valores predichos para cada observación, que pueden variar en gran magnitud debido a la tasa de *dropout*.

Tabla 5.8: Tabla de *RMSE* para tramos de recorridos en horario valle y punta, para todo el conjunto de prueba. Fuente: Elaboración propia.

Recorrido	Horario	P12 (*RMSE*) min.	P23 (*RMSE*) min.	P34 (*RMSE*) min-
301	H. Valle	3.93	3.12	4.73
	H. Punta	2.84	4.44	4.73
315e	H. Valle	2.95	3.52	4.04
	H. Punta	3.59	3.77	4.45
506	H. Valle	3.43	5.43	3.09
	H. Punta	2.77	4.36	2.93

Para los buses de prueba de los recorridos seleccionados, se realizaron predicciones del tiempo que toma recorrer cada tramo. En las siguientes figuras, se pueden observar histogramas de las predicciones para cada tramo, cada bus y horario, y el valor real del tiempo que le tomó al bus recorrer ese tramo. Se configuró ”T” igual a 500, debido a que estos vectores de predicciones deben ingresar al Filtro de Partículas como valores iniciales de las partículas. La cantidad de estas últimas es un parámetro que se debe elegir

en base a prueba y error, y por lo tanto, es bueno tener una cantidad que permita tener cierta holgura, es decir, que no sea una cantidad menor a la que podría resultar óptima.

Predicciones de tiempo de recorrido entre tramos, bus de prueba 301

En las Figuras 5.20 y 5.21 se muestran los resultados de las predicciones de las redes neuronales Bayesianas para el bus 301.

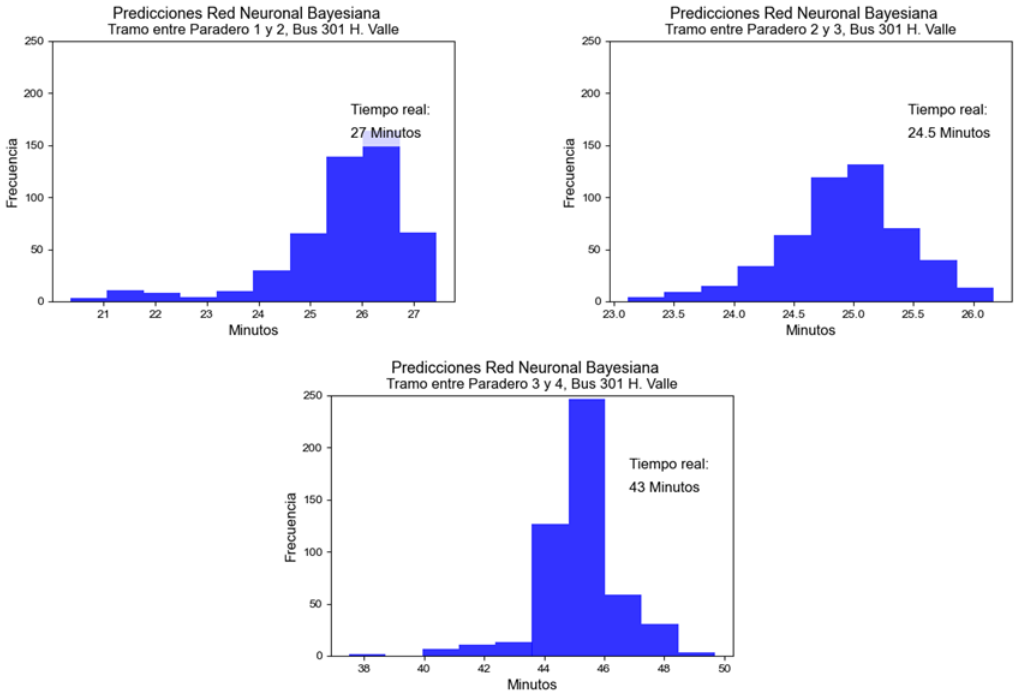

Figura 5.20: Histograma de predicciones de tiempo al recorrer tramos de la ruta, bus 301 horario valle. Fuente: Elaboración propia.

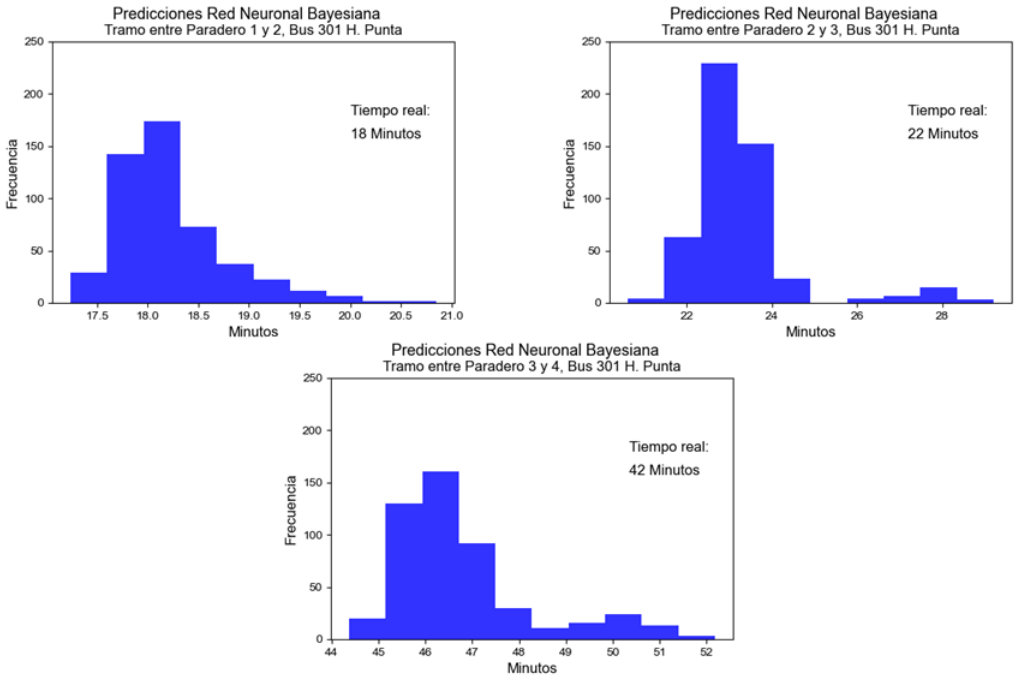

Figura 5.21: Histograma de predicciones de tiempo al recorrer tramos de la ruta, bus 301 horario valle. Fuente: Elaboración propia.

De las Figuras 5.20 y 5.21, se observa que la predicción del tramo entre paradero 2 y 3 tiene una tendencia al valor real, pero para los tramos restantes, la predicción no es tan precisa como para los otros tramos.

Para el tramo entre paradero 1 y 2 en horario valle, la predicción tiende a ser un tiempo menor que el real. Esto no sucede en el horario punta, donde es más acertada en comparación al horario valle.

En el tramo entre paradero 3 y 4 en horario valle, las predicciones tienden a estimar un tiempo mayor que el valor real de 43 minutos. Esto sucede de la misma forma en el horario punta.

Predicciones de tiempo de recorrido entre tramos, bus de prueba 315e

En las Figuras 5.22 y 5.23 se muestran los resultados de las predicciones de las redes neuronales Bayesianas para el bus 315e.

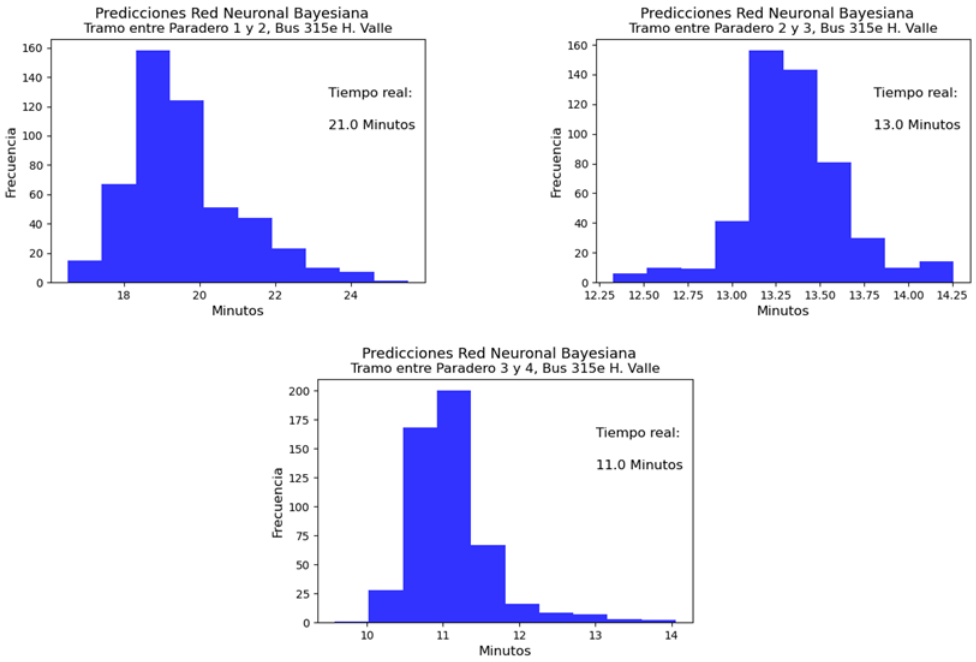

Figura 5.22: Histograma de predicciones de tiempo al recorrer tramos de la ruta, bus 315e horario valle. Fuente: Elaboración propia.

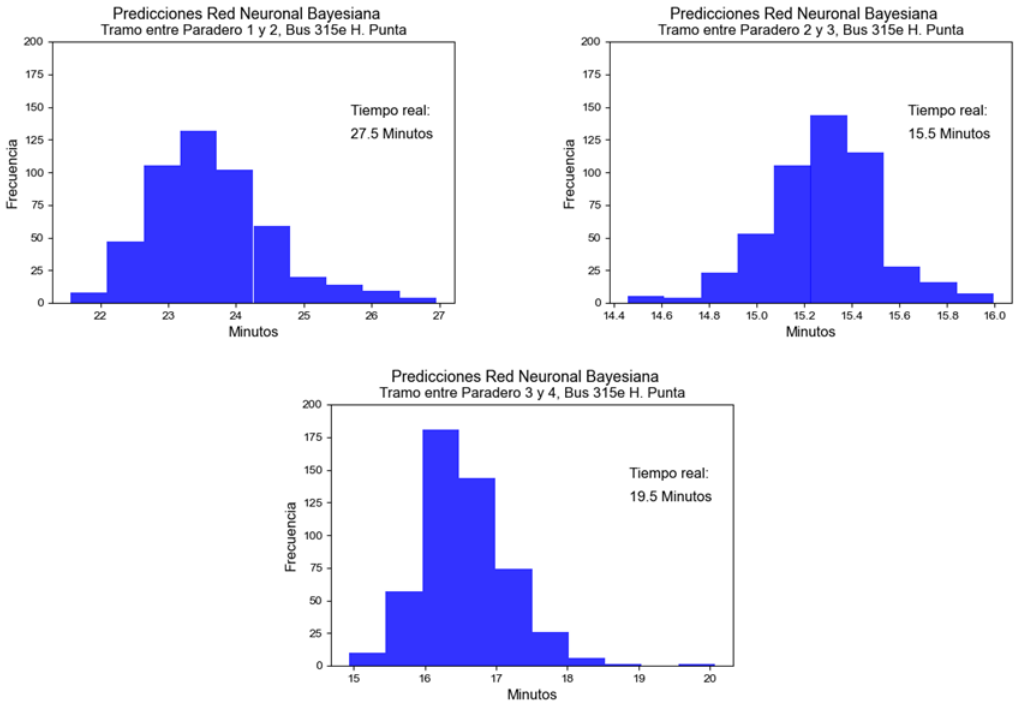

Figura 5.23: Histograma de predicciones de tiempo al recorrer tramos de la ruta, bus 315e horario punta. Fuente: Elaboración propia.

En comparación al bus 301, las Figuras 5.22 y 5.22 muestran que las predicciones tienen una tendencia mayor al tiempo real, por lo que la predicción parece más acertada.

En el tramo entre paradero 1 y 2, las predicciones tienden de mejor forma al valor real de 21 minutos para el horario valle, pero no sucede así para el horario punta, donde tienden a predecir un tiempo menor que el real de 27.5 minutos.

En el tramo entre paradero 2 y 3, las predicciones tienden al valor real en ambos horarios. Finalmente, las predicciones para el tramo entre paradero 3 y 4, para el horario valle son acertadas, tendiendo al valor real de 11 minutos, pero en el horario punta tienden a un valor menor que el real de 19 minutos.

Predicciones de tiempo de recorrido entre tramos, bus de prueba 506

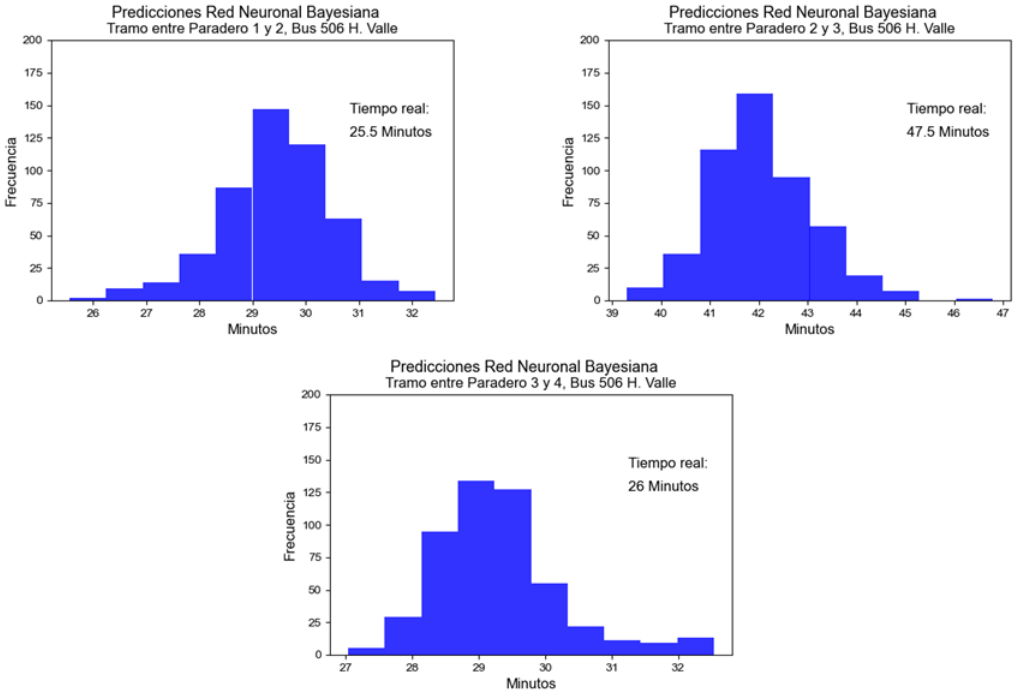

Figura 5.24: Histograma de predicciones de tiempo al recorrer tramos de la ruta, bus 506 horario valle. Fuente: Elaboración propia.

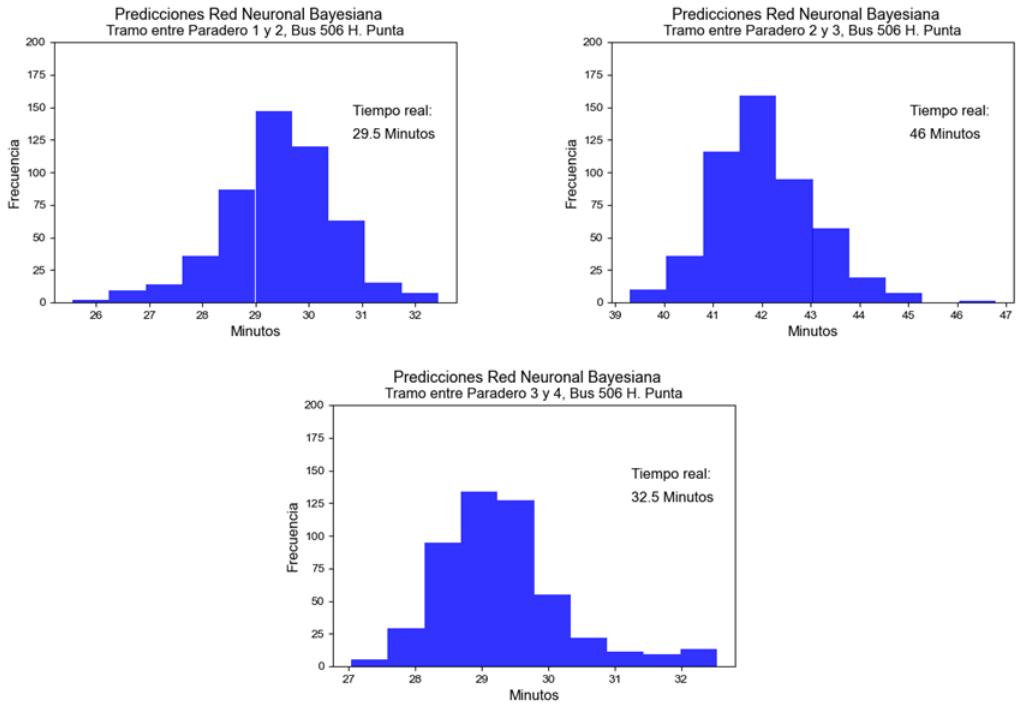

Figura 5.25: Histograma de predicciones de tiempo al recorrer tramos de la ruta, bus 506 horario punta. Fuente: Elaboración propia.

De las Figuras 5.24 y 5.25, se observa que no todas las predicciones tienden al valor real, y algunas se desvían en gran cantidad, como por ejemplo el tramo entre paradero 1 y 2 en ambos horarios. En el horario valle, las predicciones tienden a un valor más alto que el valor real de 25.5 minutos, mientras que en el horario punta, estas tienden a predecir un valor menor que el real de 29.5 minutos.

Para el tramo entre paradero 2 y 3, y el tramo entre paradero 3 y 4, las predicciones tienden a ser más acertadas en el horario punta que en el horario valle, como se puede observar en las figuras anteriores.

De las figuras anteriores, se observa que las predicciones no siempre tienden al valor real, y a veces tienen un gran desvío. Esto puede suceder por la configuración o el entrenamiento de la red, pero es difícil identificar el aspecto que provoca que la red no pueda predecir bien el valor. Esto es el resultado esperado de una red neuronal Bayesiana, que el resultado de una predicción no sea un valor concreto, sino un vector de valores que puedan captar el error asociado a cada predicción.

La Tabla 5.9 resume los resultados del *RMSE* para los buses de prueba de los reco-

rridos seleccionados. En esta tabla se puede visualizar que para los buses de prueba el valor del *RMSE* es más bajo, pero sigue la tendencia de que en distintos tramos es mayor, como también para distintos horarios. Por ejemplo, el bus 301 en todo el conjunto de prueba, el valor de *RMSE* en promedio de los tres tramos es mayor en el horario punta que en el horario valle. Esto también sucede para el bus 315e, no así para el recorrido 506. En general, si se desea identificar alguna tendencia, en este modelo y en el modelo *LSTM* parece ser que el recorrido en horario punta es más difícil de predecir acertadamente. Basándose en el valor del *RMSE*, el error en minutos es más bajo en promedio que el calculado en el rendimiento del conjunto de prueba. En este caso, se tiene un promedio de alrededor de 2 minutos.

Tabla 5.9: Tabla de *RMSE* para tramos de recorridos en horario valle y punta, para buses seleccionados de recorridos. Fuente: Elaboración propia.

Recorrido	Horario	P12 (*RMSE*) min.	P23 (*RMSE*) min.	P34 (*RMSE*) min-
301	H. Valle	1.83	0.64	2.56
	H. Punta	0.58	1.7	4.96
315e	H. Valle	1.77	0.42	0.61
	H. Punta	4.02	0.36	3.06
506	H. Valle	4.02	5.61	3.37
	H. Punta	1.90	1.65	0.89

Finalmente, los vectores que predicen el tiempo que recorre el bus de prueba en cada tramo, son uno de los datos de entrada del Filtro de Partículas, que ocupa estos datos para captar la predicción del tiempo y el ruido asociado.

Filtro de Partículas

El modelo de Filtro de Partículas, como se explicó en el marco teórico, realiza predicciones del estado de un móvil, en dos etapas: predicción y actualización. Este modelo genera una función de densidad de probabilidad con los valores de predicción. Para generar estos cálculos, distribuye partículas, cada una con un peso asociado, y que juntas generan esta función de densidad de forma recursiva.

Se desarrolló un modelo acorde al explicado anteriormente en las redes neuronales Bayesianas, que es predecir tramo por tramo el tiempo que se demora en recorrer un bus dichos tramos. Para poder ejecutar este modelo, es necesario poder caracterizar el movimiento del bus entre estos tramos bajo un sistema de ecuaciones, que contempla ecuaciones de estados y ecuaciones de observación. Este modelo no se puede ejecutar para todo el recorrido de prueba, ya que la predicción del Filtro está diseñada para cada

móvil en específico. Por esto es que en modelos anteriores se escogió un bus de prueba, para poder realizar comparaciones entre modelos.

Las variables que se crearon fueron:

- $X1$, que representa la hora de llegada de un bus al paradero del inicio del tramo.
- $X2$, que representa la hora de inicio del recorrido, es decir, el momento en que sale del terminal de buses. Este valor es constante para las mediciones de un bus en particular, ya que para completar una ruta, el bus lógicamente sólo deja el terminal una vez.
- RN, que es el vector de "T" predicciones del tiempo que demora un bus en recorrer el tramo, que proviene del cálculo de la red neuronal Bayesiana. Este vector contiene el error de predicción del cálculo de tiempo de forma intrínseca, que es captado por la diferencia entre los valores predichos.
- Y, que representa la hora de llegada del bus al paradero del fin del tramo.
- $V_{GPS(k)}$, que representa el error del aparato *GPS* de los buses, en minutos.

Las variables $X1$, $X2$ e Y, que representan horas, no pueden ser ingresadas con el formato de horas como tal y por eso deben convertirse a minutos. Esto se realiza transformando la hora del registro en minutos desde las 00:00 horas. Por ejemplo, si el registro fuese de las 15:30 horas, sería igual a 930 minutos. Con las variables expuestas, se generaron las siguientes ecuaciones para relacionarlas:

$$X1_{(k+1)} = X1_{(k)} + RN_{(k)} \tag{5.1}$$

Ecuación de estado número 1.

$$X2_{(k+1)} = X2_{(k)} \tag{5.2}$$

Ecuación de estado número 2.

$$Y_k = X1_{(k)} + x2_{(k)} + V_{GPS(k)} \tag{5.3}$$

Ecuación de observación.

Estas ecuaciones son las que se mencionaban en el marco teórico, siendo las que describen el movimiento de un bus en efectos de este problema. Para mejor entendimiento del problema, observar la Figura 5.26, que además contiene un ejemplo.

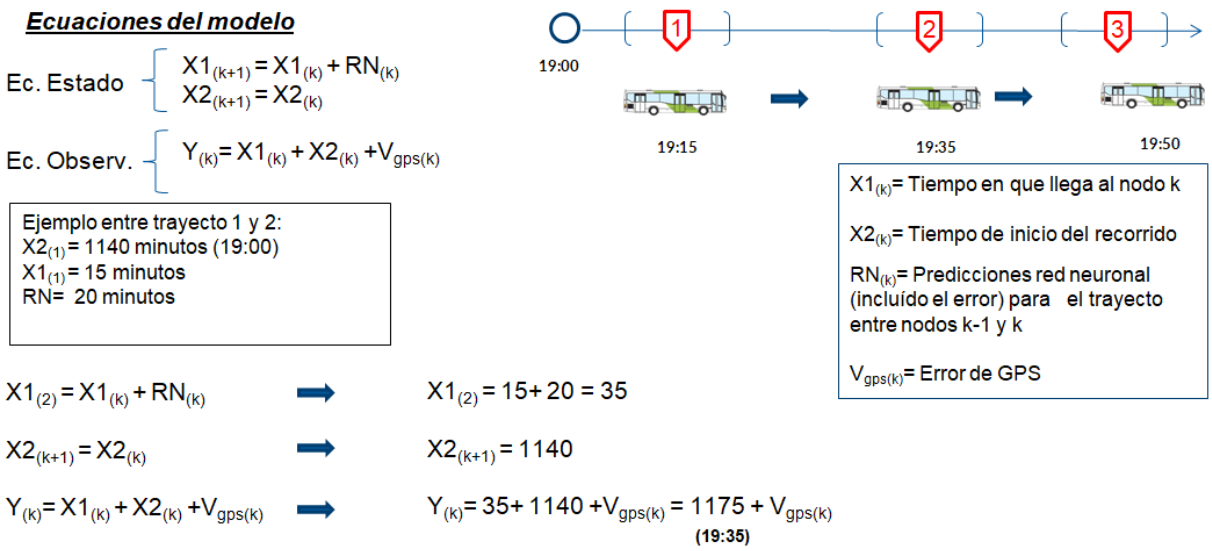

Figura 5.26: Modelo de Filtro de Partículas, con un ejemplo. Fuente: Elaboración propia.

En la Figura 5.26, se ejemplifica la predicción de llegada de un bus entre el paradero 1 y 2, teniendo como hora de salida del terminal a las 19:00 horas. Se puede observar que las ecuaciones del sistema no son de mayor complejidad, pero definen bien el movimiento de un bus entre un tramo de la ruta.

Configuración del modelo de Filtro de Partículas

Los parámetros a configurar del Filtro de Partículas son los valores de error, el número de partículas y la distribución inicial de valores.

El valor del error de predicción de los datos es captado al ingresar el vector de predicción generado por la red neuronal Bayesiana. En este caso, el vector es de 500 valores, pero en el algoritmo se configura que el valor a ocupar sea igual al número de partículas. En base a prueba y error, se seleccionó una cantidad de 200 partículas, y con esto, se toman como valores los 200 primeros valores del vector de predicción.

El error del aparato *GPS* del bus depende intrínsecamente del aparato, ya que algunos miden con peor precisión que otros. Como el modelo calcula todo en minutos, este parámetro también debe estar en minutos, y para efectos del Filtro de Partículas, se debe ingresar una distribución con una media y una varianza dadas, ya que de esta forma el modelo toma valores de dicha distribución de forma aleatoria, generando un ruido de proceso que no es constante. El valor de la media y la varianza de este parámetro es difícil de seleccionar, pero con el objetivo de contar con holgura y pensando que el aparato crea registros cada 30 segundos, se ingresó una distribución gausiana con media igual a 0 y

varianza igual a 1.5 minutos. Si se pudiese saber de forma concreta y correcta el error del aparato *GPS*, y si fuese el mismo tipo de aparato para todos los buses, este valor puede ser ingresado como un valor discreto.

Por último, para efectos del modelo, el tiempo de llegada del bus desde el terminal o paradero cero al paradero de inicio del primer tramo, también se modela como un valor aleatorio de la distribución asociada al cálculo de este valor con la base de datos. Esto quiere decir, que se calculó el tiempo de llegada de los buses desde el terminal al paradero de inicio del primer tramo, y se ajustó una distribución a estos datos. La distribución resultante es la que se ingresa al Filtro de Partículas.

Análisis de resultados y conclusiones

Con la configuración anterior, se ejecutó el Filtro de Partículas para los buses de prueba de los recorridos seleccionados. Para cada bus de prueba y su horario, como se explicó anteriormente, se ajustó una distribución gaussiana a la distribución de los tiempos de llegada del terminal al primer paradero a estudiar, que es el inicio del primer tramo. La media y la desviación de las distribuciones se encuentran en la Tabla 5.10. En esta Tabla se observa que los mayores valores de media y desviación estándar se registran para el recorrido 315e, y distan hasta 6 unidades en media del valor mínimo de la tabla.

Tabla 5.10: Tabla de distribuciones iniciales para el Filtro de Partículas.
Fuente: Elaboración propia.

Recorrido	Horario	Distribución Gaussiana (media μ, desviación est. σ)
301	Valle	3.03 , 0.70
	Punta	3.26 , 1.51
315e	Valle	7.25 , 3.16
	Punta	7.40 , 1.35
506	Valle	2.08 , 0.61
	Punta	1.91 , 0.58

En las secciones siguientes, se mostrarán los resultados de las predicciones del Filtro de Partículas. Se observará que los resultados están en minutos, del orden de 700 o 1000, pero se debe recordar que este número se debe transformar a horas, tomando como inicio las 00:00 horas para obtener el valor como la hora de llegada.

Resultados predicción de tiempo por tramos, bus de prueba 301

La predicción de tiempos de llegada de tramos para el bus de prueba del recorrido 301 se muestra en las Figuras 5.27 y 5.28, para horario valle y horario punta, respectivamente.

De las figuras de horario valle se observa que el error de predicción es de menos de un minuto. La función de densidad de probabilidad previa muestra la organización de las partículas en la etapa a priori. De ellas se observa que en los tres tramos, las partículas se distribuyen de forma que tienen la misma probabilidad, pero en el cálculo de la predicción final (distribución a posteriori), las partículas tienden al valor real, que tiene como etiqueta ”Observación”.

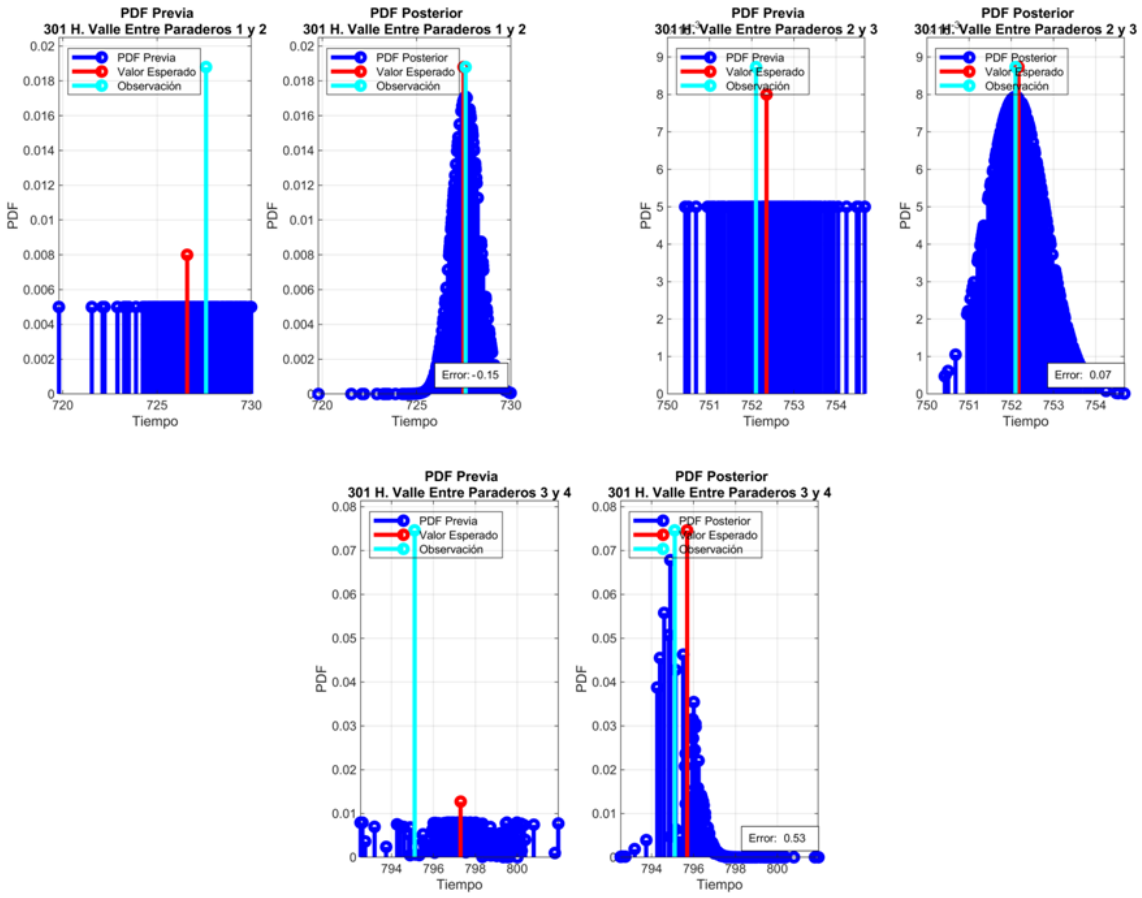

Figura 5.27: Predicción modelo Filtro de partículas. Distribución a priori y a posteriori. Bus 301 horario valle. Fuente: Elaboración propia.

En la Figura 5.28, se observa que los resultados son similares al horario valle, excepto por el tramo entre paraderos 3 y 4, donde el error es de más de dos minutos. En comparación al horario valle, en ambos horarios el último tramo es en el que se genera más error.

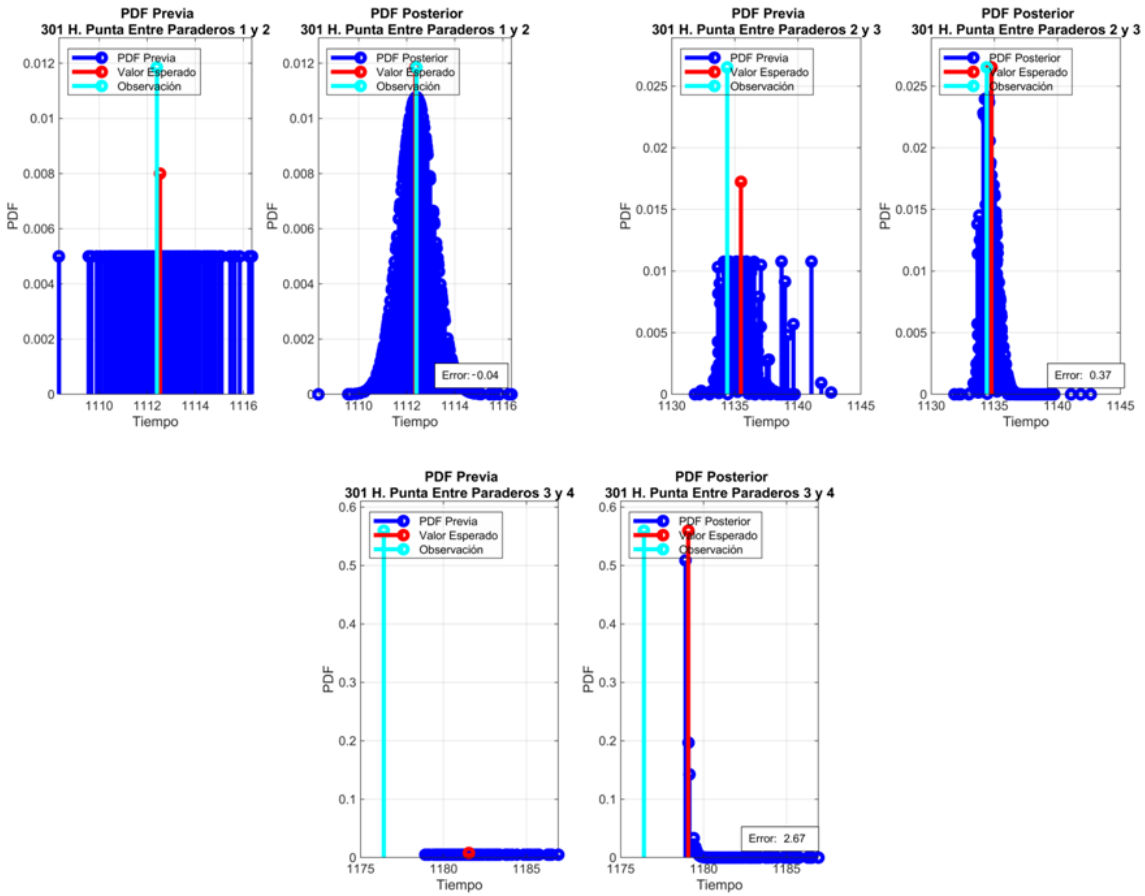

Figura 5.28: Predicción modelo Filtro de partículas. Distribución a priori y a posteriori. Bus 301 horario punta. Fuente: Elaboración propia.

Lo que tienen en común estas predicciones es que para el tramo entre paradero 2 y 3, y tramo entre paradero 3 y 4, el error es positivo, lo que indica que el valor predicho es menor al real.

Resultados predicción de tiempo por tramos, bus de prueba 315e

La predicción del Filtro de partículas para el bus de recorrido 315e es similar al bus 301. En este caso, el error de predicción no supera el minuto. En el horario punta, el error es de menor magnitud, y se observa que las partículas en la función de densidad de probabilidad posterior para el tramo 1 y el tramo 3, no se agrupan cerca del valor real, y debido a esto el cálculo del valor predicho posee mayor error que en el horario valle. Esto sucede debido a que la predicción de la red neuronal Bayesiana es peor en el horario punta que en el horario valle.

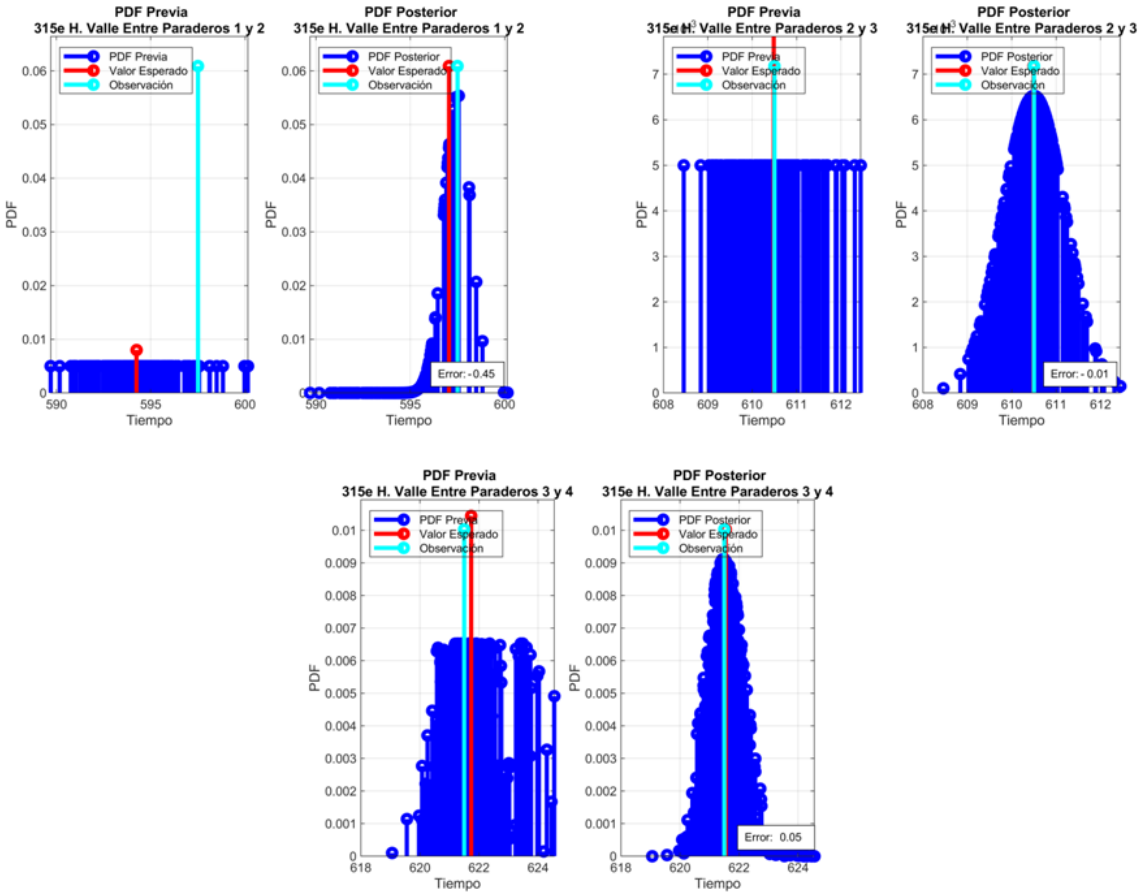

Figura 5.29: Predicción modelo Filtro de partículas. Distribución a priori y a posteriori. Bus 315e horario valle. Fuente: Elaboración propia.

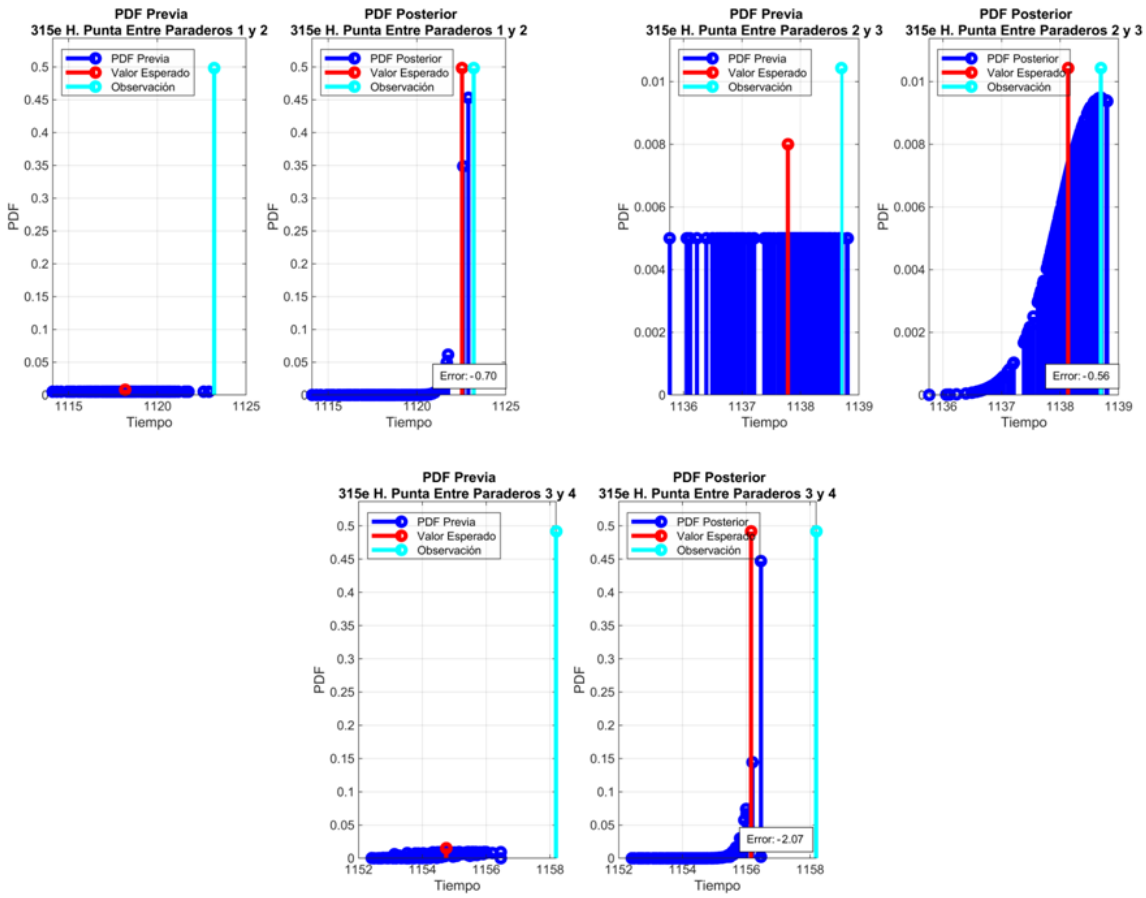

Figura 5.30: Predicción modelo Filtro de partículas. Distribución a priori y a posteriori. Bus 315e horario punta. Fuente: Elaboración propia.

En ambos horarios existen valores de error negativo, como en los primeros dos tramos del horario valle. Esto indica que el bus llegó después de lo predicho. Para el horario punta, se comporta de esta manera en los tres tramos.

Resultados predicción de tiempo por tramos, bus de prueba 506

Para el bus de prueba del recorrido 506, sucede lo contrario que en el bus de prueba del recorrido 315. Se observa que las predicciones del horario punta poseen menor error que en el horario valle, siendo en este último horario donde las partículas se ordenan de forma distante al valor real. La principal causa de esto es que este cálculo depende directamente de las predicciones de la red neuronal Bayesiana correspondiente, donde al observar la Tabla 5.9 que resume los valores de error con métricas de cada bus y horario, se puede dichos valores son mayores para el horario valle, indicando que la predicción de tiempo no es tan precisa como en el otro horario.

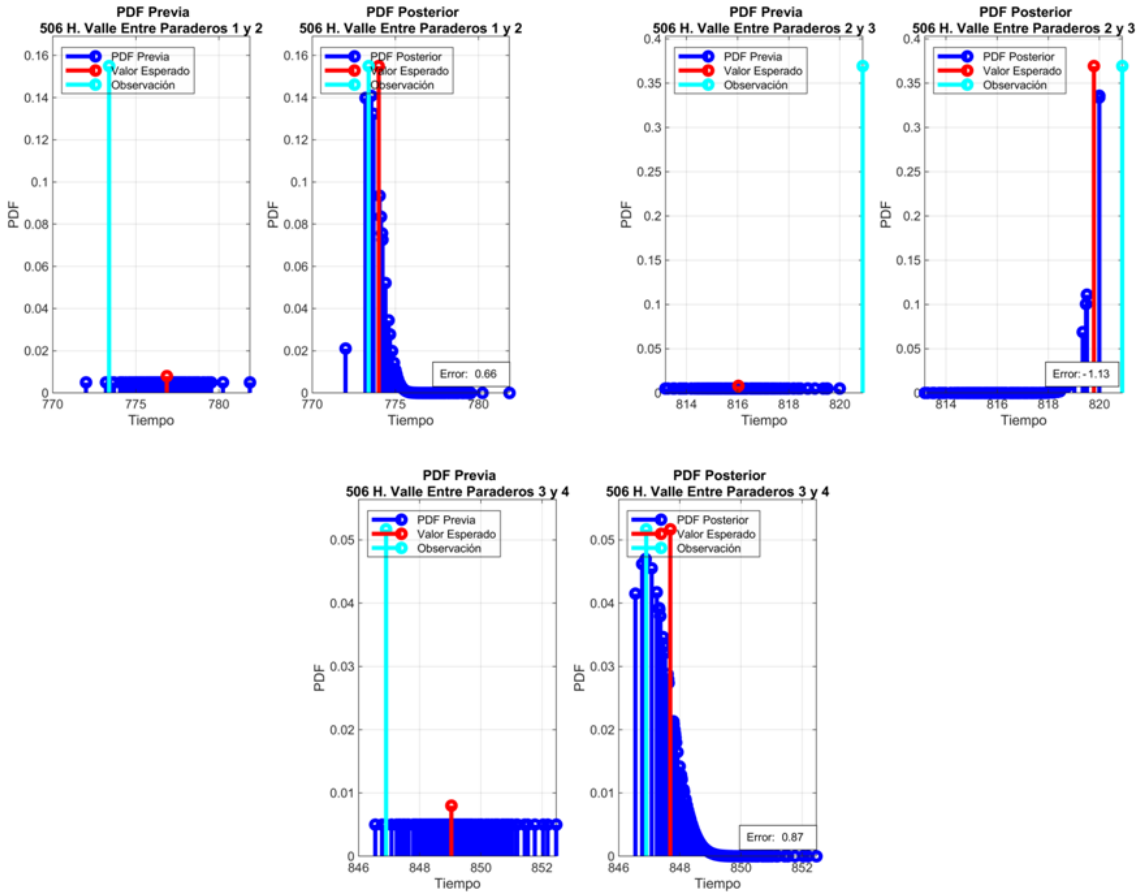

Figura 5.31: Predicción modelo Filtro de partículas. Distribución a priori y a posteriori. Bus 506 horario valle. Fuente: Elaboración propia.

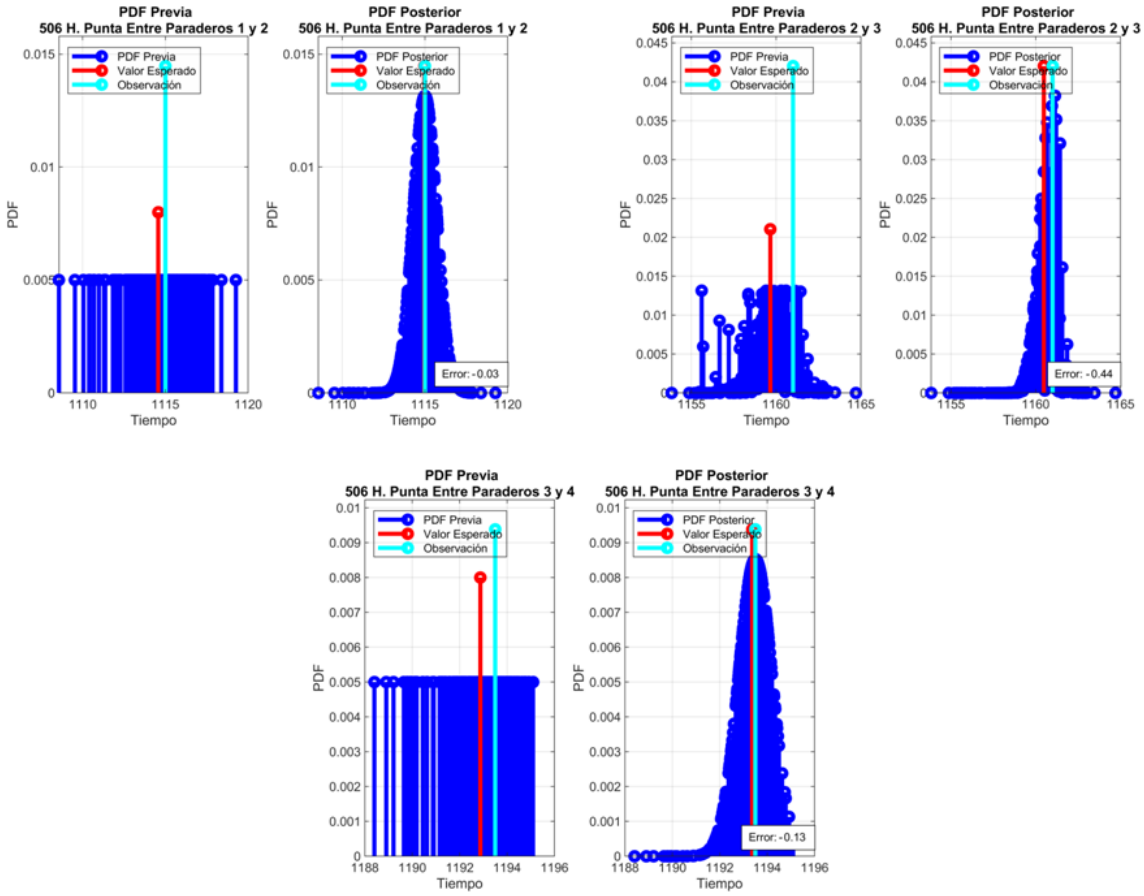

Figura 5.32: Predicción modelo Filtro de partículas. Distribución a priori y a posteriori. Bus 506 horario punta. Fuente: Elaboración propia.

En ambos horarios existen valores de error negativo, como en el segundo tramo del horario valle. Para el horario punta, el error es negativo en los tres tramos, lo que significa que el valor de la predicción es menor al valor real.

Conclusiones generales

Finalmente, se presenta un resumen de los resultados en la Tabla 5.11. Se observa que para el bus de prueba 301 y 315e, el error es de menor magnitud en el horario punta que en el horario valle, no así para el bus 506, donde sucede lo contrario. Esto puede suceder debido a las predicciones de las redes neuronales Bayesianas en mayor medida que en el funcionamiento del Filtro de Partículas.

Otro aspecto importante que se observa en la Tabla 5.11, es que no existe una tendencia a predecir tiempos de llegada menores o mayores a los reales, ya que para distintos buses en distintos horarios, se observa errores de predicciones positivos (valor predicho mayor al real) y negativos (valor predicho menor al real).

Este modelo mixto, que realiza predicciones por tramos, no genera gran error de predicción basándose en los valores del *RMSE*, donde el mayor error de predicción es de menos de dos minutos para la ruta total de los buses.

Es importante notar que los tramos generados para probar el funcionamiento de este modelo incluyen una gran cantidad de paraderos de distancia. Si esto se cambiase especificando que un tramo es la distancia entre paraderos consecutivos, se presume que el resultado podría mejorar.

Tabla 5.11: Tabla de *RMSE* para tramos de recorridos en horario valle y punta, de los resultados del Filtro de Partículas. Fuente: Elaboración propia.

Recorrido	Horario	Error primer tramo	Error segundo tramo	Error tercer tramo	*RMSE* ruta total minutos
301	Valle	-0.15	0.07	0.53	0.32
	Punta	-0.04	0.37	2.67	1.56
315e	Valle	-0.45	-0.01	0.05	0.26
	Punta	-0.70	-0.56	-2.07	1.30
506	Valle	0.66	-1.13	0.87	0.91
	Punta	-0.03	-0.44	-0.13	0.27

5.4. Comparación entre modelo *LSTM* y modelo mixto de red neuronal Bayesiana con Filtro de Partículas

Es difícil realizar una comparación directa entre los dos modelos presentados anteriormente, ya que funcionan de forma distinta debido a las necesidades de cada uno. El modelo *LSTM* necesita datos de entrada como series de tiempo, y el modelo mixto trata el problema con la necesidad de ser definido por tramos, y no como ruta completa, como el modelo *LSTM*.

Debido a lo anterior, y que el Filtro de Partículas no ocupa todo el conjunto de prueba para realizar predicciones, y sólo puede ser ejecutado con un bus a la vez, se seleccionó un bus de prueba, y se ejecutaron los modelos para los datos de dichos buses. Con esto, se obtuvieron resultados que de una forma podrían ser comparables en base a las métricas de error *MSE* y *RMSE*, ya que es posible comparar el desempeño de los modelos para la ruta completa.

En el caso del modelo *LSTM*, no es necesario realizar cálculos con los resultados, ya que este modelo soluciona de forma general el problema de tiempos de espera para toda la ruta. Sin embargo, como el modelo mixto soluciona el problema por tramos, es necesario llevar estas soluciones a la ruta completa, y un método es calcularlo en base a los

errores individuales de cada tramo. Este cálculo se muestra en la Tabla 5.11, donde los valores de *RMSE* se calculan de esta forma.

Con estos cálculos realizados, es necesario comparar bajo dichas métricas el desempeño de los modelos de la mejor forma, y no puede ser con todo el conjunto de prueba, ya que el Filtro de partículas no ocupa todo este conjunto para las predicciones. Basándose en la Tabla 5.7, los tramos seleccionados varían entre 20 y 40 minutos, por lo que para comparar el rendimiento de este modelo con el rendimiento del modelo *LSTM*, se debe elegir el umbral de este último que sea comparable. Con esta condición, se eligió el umbral de 20 minutos, que resulta ser casi el valor intermedio de los umbrales extremos (50 y 5 minutos). Teniendo esto como condición, se generó la Tabla 5.12, que compara las métricas de error de ambos modelos.

Tabla 5.12: Comparación entre resultados de modelo LSTM y Filtro de Partículas. Fuente: Elaboración Propia.

Recorrido	Horario	*RMSE* LSTM (minutos)	*RMSE* FP (minutos)
301	Valle	2.66	0.32
	Punta	4.99	1.56
315e	Valle	6.44	0.26
	Punta	14.35	1.30
506	Valle	3.11	0.91
	Punta	6.67	0.26

De la Tabla 5.12, se observa que ambos modelos, para los buses 301 y 315e, el error es menor para el horario valle y mayor para el horario punta, pero para el bus 506 sucede lo contrario. Lo importante de esta comparación, es que el modelo *LSTM* obtiene peores resultados en comparación al modelo mixto, de forma general, ya que todos los valores de las métricas de error son más altos. Si bien el error del modelo *LSTM* no supera los cuatro minutos en algunos tramos y horarios, sí existen valores más altos, como 6.44 minutos y 14.35 minutos de error en el bus 315 para el horario valle y punta, respectivamente. Este tipo de error puede ser de gran magnitud dependiendo de la percepción del usuario del transporte público.

El error del modelo mixto no sobrepasa los dos minutos en los buses seleccionados basándose en el *RMSE*, y en comparación al modelo *LSTM* el error es notoriamente más bajo, lo que indica un mejor desempeño.

El desempeño de cada modelo depende directamente de la forma de tratar los datos. En el modelo *LSTM*, es un solo algoritmo que genera predicciones de tiempos de llegada, y sólo necesita una serie de datos de 7 observaciones pasadas de un bus (aunque es un parámetro configurable), y el error depende del umbral de tiempo. Según el análisis de

resultados de este modelo, el error disminuye al disminuir el umbral de tiempo, indicando que es un algoritmo que funciona mejor en tramos cortos o cuando el bus está cercano al paradero. Esa es la ventaja de dicho modelo, que funciona de forma independiente, y aunque el costo computacional es alto, puede llevarse a cabo con la suficiente potencia de software.

El desempeño del modelo mixto depende directamente de las predicciones de la red neuronal Bayesiana. Este modelo mixto es más costoso en recursos computacionales que el modelo *LSTM*, ya que se deben ejecutar dos algoritmos que funcionan juntos, que son la red neuronal Bayesiana y el Filtro de Partículas. En este modelo, si el vector de predicciones de la red no son precisas, la predicciones del Filtro de Partículas tampoco lo serán, lo que indica una condicionante al modelo. La ventaja es que el algoritmo de Filtro de Partículas es recursivo, y genera predicciones para los tramos de la ruta. Aún así, es posible ejecutarlo una vez para predecir el tiempo de llegada a cada tramo, e ir actualizándolo cada vez que el bus recorra un tramo de ruta. Como se mencionó anteriormente, si se divide la ruta en tramos de paraderos consecutivos, se necesitarían tantas redes neuronales Bayesianas como tramos, e implementar el Filtro de Partículas por cada tramo, lo que necesitaría grandes recursos computacionales. La mayor ventaja de este modelo, es que aparte de predecir el tiempo de llegada de un bus a un paradero, también calcula la probabilidad de que el bus llegue en otro momento, siendo este un dato adicional a la predicción en sí, y que agrega valor a la predicción en sí.

La comparación anterior se realiza en base a los aspectos comparables de cada modelo, ya que es difícil comparar soluciones cuando se resuelve el problema de forma distinta y bajo distintos puntos de vista. En base a las ventajas de y desventajas de cada modelo, el modelo mixto de red neuronal Bayesiana parece presentar más ventajas en general que el modelo *LSTM*.

Capítulo 6

Recomendaciones sobre el prototipo

Basándose en los resultados de las métricas de error, el modelo mixto de red neuronal Bayesiana con el Filtro de Partículas posee mejor desempeño para los tres conjuntos de recorridos. Por esto es que la recomendación general a la empresa es seleccionar este prototipo de modelo, modificarlo para que sea viable en la aplicación, e intentar una primera implementación que de luces de la precisión con datos en tiempo real.

Si bien el modelo necesita más recursos computacionales que el modelo *LSTM*, las ventajas en la precisión de la predicción parecen compensar esta desventaja, y más aún cuando parece funcionar para tramos de longitud variable. La mayor desventaja de la implementación de este modelo, es que cada tramo debe configurarse para que comprenda la distancia entre paraderos consecutivos.

En base a todo lo anterior, la recomendación de prototipo es la que se muestra en la Figura 6.1. donde el orden es el siguiente: El bus sale del terminal, y se registra su hora de salida. El bus llega al primer paradero, y se generan las ”T” predicciones de la red neuronal Bayesiana, estas predicciones se envían al Filtro de Partículas, y el Filtro genera la función de densidad de probabilidad del tiempo de llegada, con la predicción de la hora de llegada. Este valor tiene dos aspectos importantes, la predicción de la hora de llegada y la probabilidad asignada a esta predicción. El segundo dato, que es la probabilidad de ocurrencia, podría ser útil para la empresa como una diferenciación en el mercado de las aplicaciones similares, ya que, aparte de entregar un tiempo de espera o llegada estimado, adicionalmente podría entregar la probabilidad estimada con que el bus llegará a esa hora, dato que actualmente no poseen las aplicaciones de transportes. Los parámetros de este modelo pueden configurarse bajo una etapa de experimentación, donde podrían encontrarse los mejores parámetros para su funcionamiento.

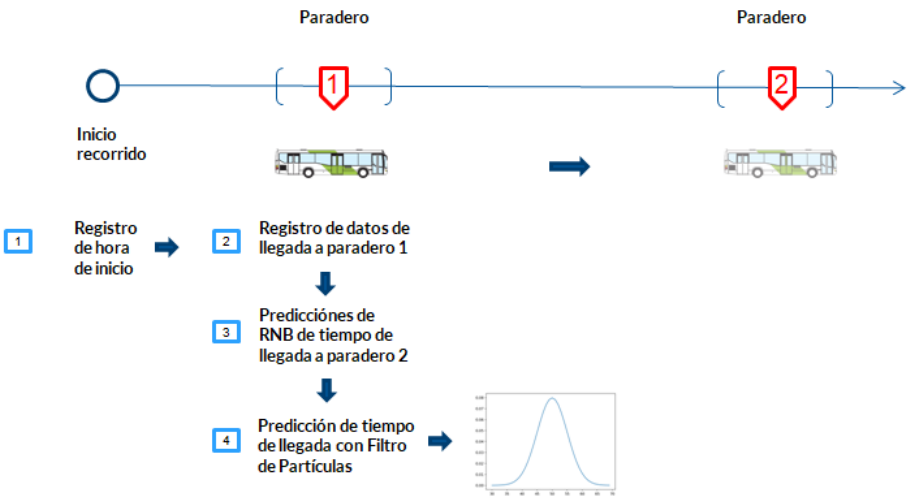

Figura 6.1: Prototipo de Modelo Mixto. Fuente: Elaboración propia.

Capítulo 7

Trabajos futuros

Como extensiones a este trabajo, o trabajos futuros que complementarían el presente trabajo, se tiene:

- El estudio de otros recorridos distintos, o la elección de distintos recorridos representantes de los conjuntos generados.
- Ejecutar los modelos con datos de otros meses, y con algunos meses atípicos o que resulten interesantes de observar por los hechos ocurridos en ellos.
- Estudiar el desempeño del modelo mixto cambiando parámetros, como el número de neuronas, el número de partículas, entre otros, y comparar su rendimiento.
- Estudiar el rendimiento de los modelos con datos en tiempo real.

Capítulo 8

Conclusiones Generales

El presente trabajo de memoria brindó dos soluciones distintas al mismo problema: Predecir el tiempo de llegada de un bus a un paradero. El objetivo principal, desarrollar un prototipo de modelo con la capacidad de predecir este tiempo, se cumplió al desarrollar dos propuestas de modelos, el modelo *LSTM* y un modelo mixto de redes neuronales Bayesianas y Filtro de Partículas.

Al ser un amplio conjunto de estudio, se generaron conjuntos de datos agrupando recorridos en base a las coordenadas de sus rutas, y eligiendo un representante de dichos grupos de forma aleatoria. Los modelos predicen de forma similar el tiempo de llegada de los buses para los conjuntos de recorridos de 301 y 506, pero para el conjunto de recorridos de 315e genera más error en la predicción.

Al dividir los datos en horarios oficiales de Transantiago, los resultados son casi similares en ambos horarios punta y valle, y sólo en algunas ocasiones los resultados tienen gran diferencia de error entre horarios.

Para poder realizar comparaciones entre modelos, fue necesario elegir un bus representante de los recorridos 310, 315e y 506, ya que el algoritmo de Filtro de Partículas genera predicciones puntuales para cada bus. Los resultados de comparación de estos modelos se basan en dicho bus, llamado bus de prueba de los recorridos elegidos.

Como conclusiones y aspectos importantes sobre el modelo *LSTM*, dicho modelo necesita que los datos sean ingresados como series de tiempo. Se configuró este algoritmo con el fin de crear un modelo que tenga la capacidad de predecir en toda la ruta, y se obtuvo errores altos en comparación al modelo mixto. Estos errores varían dependiendo del horizonte de tiempo desde donde se encuentra el bus, esto quiere decir que a mayor lejanía del bus, peor es la predicción. Con esto, el modelo *LSTM* tiene la capacidad de predecir, con un error promedio de alrededor de 3 minutos, buses que se encuentran a

una lejanía de 5 minutos o menos. Ahora, si se quiere predecir un bus que se encuentra a una lejanía mayor, el error promedio puede llegar a ser de hasta 14 minutos para un bus que esté a una lejanía de 20 minutos. Por esto es que se concluye que el modelo podría tener un buen desempeño sólo en predicciones de tramos cortos. También es importante notar que los resultados con todo el conjunto de prueba son similares a los resultados obtenidos con el bus de prueba seleccionado.

Sobre el modelo mixto, las predicciones presentan un error mucho más bajo en comparación al modelo *LSTM*. Este modelo tiene como necesidad ser implementado por tramos. Los tramos estudiados en este trabajo varían entre 15 a 45 minutos, y el modelo no presenta un error mayor a 2 minutos, lo que indica que para tramos de menor y mayor genera predicciones más acertadas que el modelo anteriormente mencionado. Se concluye que este modelo puede ser implementado por tramos de diferente longitud, y el error debería ser similar. Las limitaciones de este modelo residen en que la precisión depende directamente de las predicciones de la red neuronal Bayesiana, porque si son erradas, también lo será la predicción final.

Como conclusión final, los dos modelos ejecutados en el presente trabajo poseen desempeños distintos, pero también en circunstancias distintas. Este desempeño varía dependiendo las condiciones de predicción, de los datos y la distancia del bus al paradero en que se requiere predecir su llegada. Aún así, realizando una comparación general de su desempeño, el modelo mixto genera una predicción de tiempo con mucho menos error que el modelo *LSTM*, generando como recomendación a la empresa que la mejor opción de implementación, siendo este el de mayor potencial y precisión.

Bibliografía

[1] Ana Azevedo and M.F. Santos. KDD, SEMMA and CRISP-DM: A parallel overview. *IADIS European Conference Data Mining*, pages 182–185, 2008.

[2] K. Dziekan and K. Kottenhoff. Dynamic at-stop real-time information displays for public transport: effects on customers. *Transportation. Research. Part A: Policy Practice.*, 41(6):489–501, 2007.

[3] Aditya Tulsyan et al. Particle filtering without tears: A primer for beginners. *Computers and Chemical Engineering*, 95:139–144, 2016.

[4] Anton Agafonov et al. Bus arrival time prediction with lstm neural network. *Lecture Notes in Computer Science*, 11554:11–18, 2019. doi: http://doi-org-443.webvpn.fjmu.edu.cn/10.1007/978-3-030-22796-8_2.

[5] Hui Lu et al. The impact of real-time information on passengers' value of bus waiting time. *Transportation Research Procedia.*, 31:18–34, 2018.

[6] M. Arulampalam et al. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):175–177, 2002.

[7] P. Kronborg et al. Fungera transportinformatik i prakitken? 4 fallstudier i syfte att undvika misstaf i framtiden. 18(2002), 2002.

[8] Wei Fan et al. Dynamic travel time prediction models for buses using only gps data. *International Journal of Transportation Science and Technology*, 4(4):353–366, 2015.

[9] Zhou Wang et al. Mean squared error: Love it or leave it? *IEEE SIGNAL PROCESSING MAGAZINE*, 26:98–117, 2009. doi: 10.1109/MSP.2008.930649.

[10] Yarin Gal. Uncertainty in deep learning. 2016. URL http://mlg.eng.cam.ac.uk/yarin/thesis/thesis.pdf.

[11] M. Hajmeer I.A. Basheer. Artificial neural networks: fundamentals, computing, design, and application. *Journal of Microbiological Methods*, 43:3–31, 2000. doi: 10.1016/s0167-7012(00)00201-3.

[12] Joseph Magiya. Pearson coefficient of correlation explained, 2019. URL https://towardsdatascience.com/pearson-coefficient-of-correlation-explained-

369991d93404.

[13] C. L. Schweiger. Customer and media reactions to real-time bus arrival information systems. *Transportation Research Board, Report No. 48.*, 2003.

[14] Maarten Speekenbrink. A tutorial on particle filters. *Journal of Mathematical Psychology*, 73:140–152, 2016.

[15] M. Wardman. Measured impacts of real-time control and information systems for bus services. *Transport Direct*, 2003.

[16] Álvaro Solera Ramírez. El filtro de kalman, 2003. URL https://activos.bccr.fi.cr/sitios/bccr/investigacioneseconomicas/DocMetodosCuantitativos/Filtro_de_Kalman.pdf.

Anexo A

Análisis Exploratorio

Tabla A.1: Tabla de comparación de tiempos de viaje. Fuente: Elaboración propia.

		Tiempo promedio de recorrido (μ, σ)	
Recorrido	Horario	Trayecto Ida	Trayecto Vuelta
301	H. Valle	85.86, 14.01	80.49, 13.56
	H. Punta	95.70, 10.85	96.02, 12.23
315e	H. Valle	62.06, 10.27	58.55, 6.04
	H. Punta	64.30, 6.61	60.45, 7.54
506	H. Valle	98.54, 14.98	99.51, 16.78
	H. Punta	98.94, 9.10	102.45, 12.03

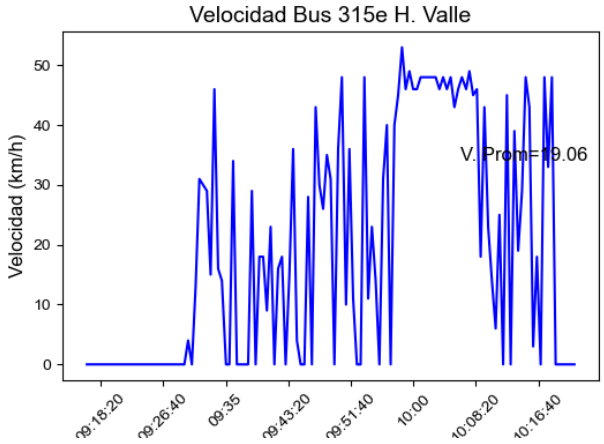

Figura A.1: Velocidad horario valle bus 315e. Fuente: Elaboración propia.

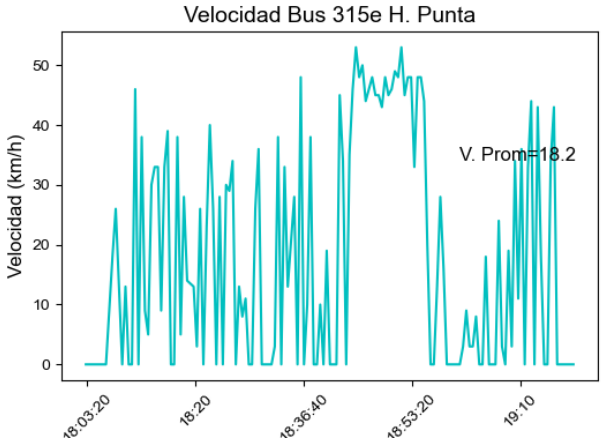

Figura A.2: Velocidad horario punta bus 315e. Fuente: Elaboración propia.

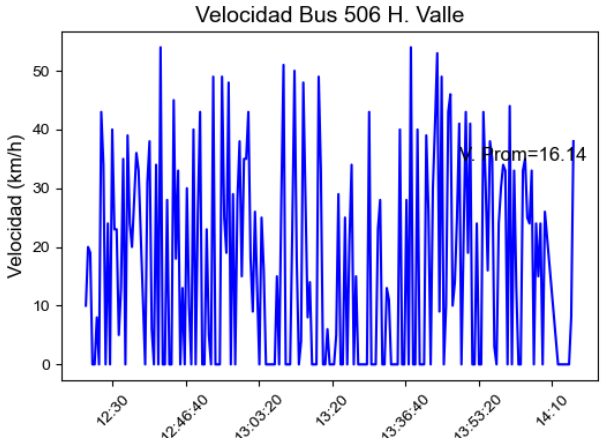

Figura A.3: Velocidad horario valle bus 506. Fuente: Elaboración propia.

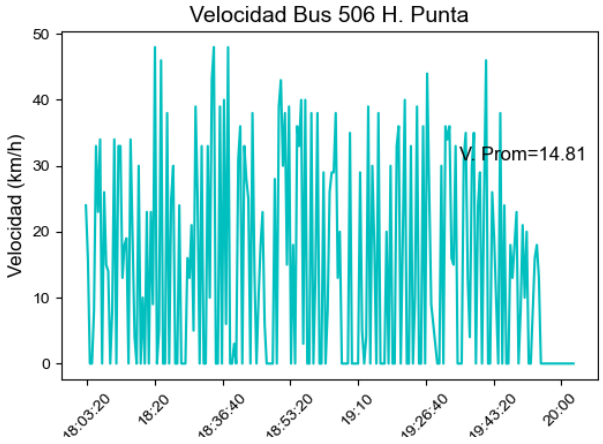

Figura A.4: Velocidad horario punta bus 506. Fuente: Elaboración propia.

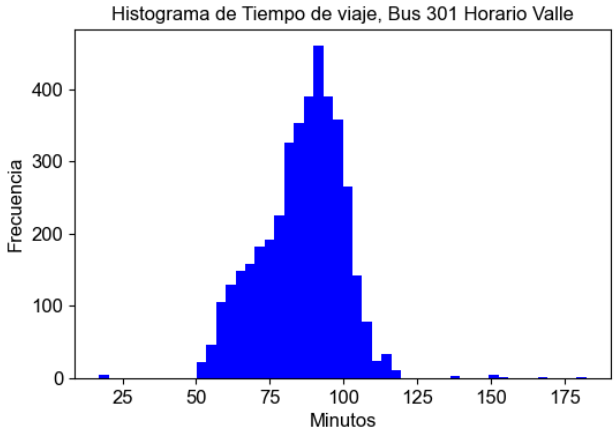

Figura A.5: Histograma de tiempo de recorridos bus 301, horario valle. Fuente: Elaboración propia.

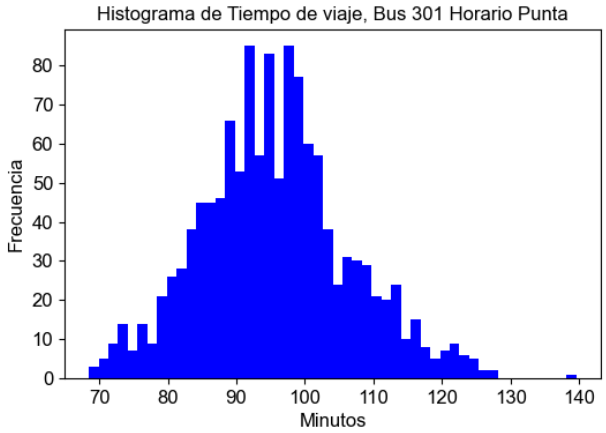

Figura A.6: Histograma de tiempo de recorridos bus 301, horario punta. Fuente: Elaboración propia.

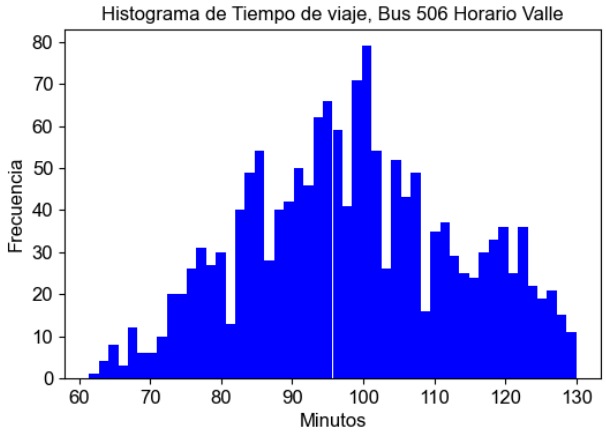

Figura A.7: Histograma de tiempo de recorridos bus 506, horario valle. Fuente: Elaboración propia.

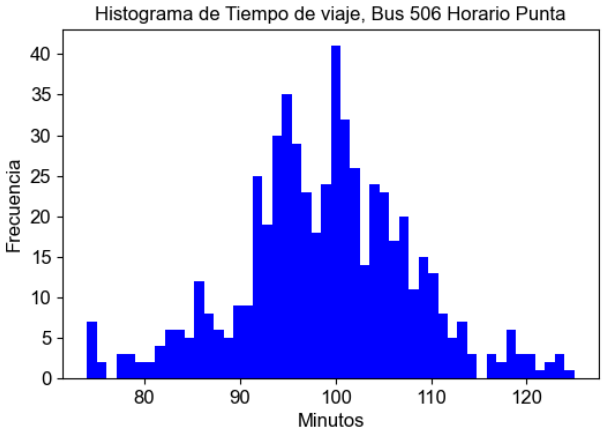

Figura A.8: Histograma de tiempo de recorridos bus 506, horario punta. Fuente: Elaboración propia.

Anexo B

Red *LSTM*

Tabla B.1: Tabla resumen de comparación métrica *RMSE* versus umbral.

		RMSE				
Bus	Datos	Horario	Umbral 50 min	Umbral 20 min	Umbral 10 min	Umbral 5 min
301	Todo conjunto de prueba	H. Valle	7.31	2.43	1.93	1.03
		H. Punta	9.61	4.17	2.50	1.10
	Bus seleccionado	H. Valle	7.49	2.66	1.98	1.05
		H. Punta	10.50	4.99	2.73	1.03
315e	Todo conjunto de prueba	H. Valle	12.97	5.95	2.57	1.12
		H. Punta	23.03	15.44	8.59	4.46
	Bus seleccionado	H. Valle	13.97	6.44	2.70	1.14
		H. Punta	17.51	14.35	9.30	4.36
506	Todo conjunto de prueba	H. Valle	6.38	2.96	1.99	1.12
		H. Punta	11.27	5.39	2.24	1.14
	Bus seleccionado	H. Valle	6.40	3.11	1.94	1.11
		H. Punta	13.28	6.67	1.84	1.13

B.1. Gráficos de error

B.1.1. Recorrido 301, conjunto de prueba y bus seleccionado

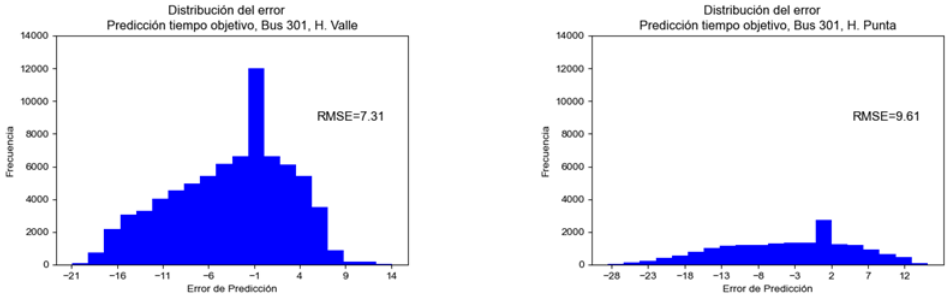

Figura B.1: Resultados red *LSTM* para todo el conjunto de prueba 301, umbral 50 minutos. Fuente: Elaboración propia.

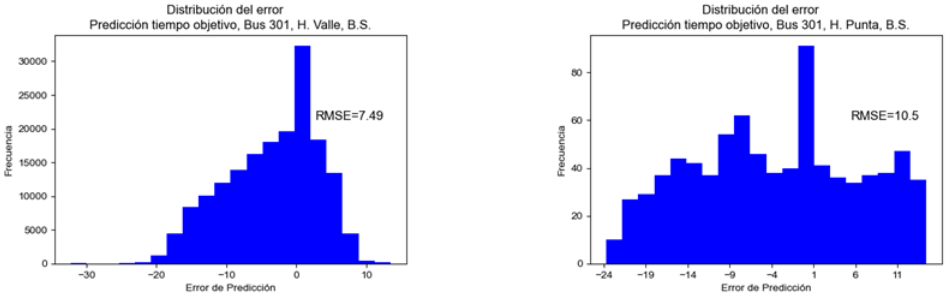

Figura B.2: Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbral 50 minutos. Fuente: Elaboración propia.

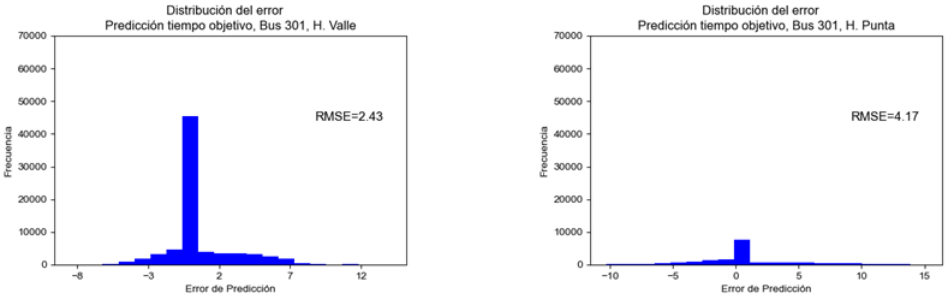

Figura B.3: Resultados red *LSTM* para todo el conjunto de prueba 301, umbral 20 minutos. Fuente: Elaboración propia.

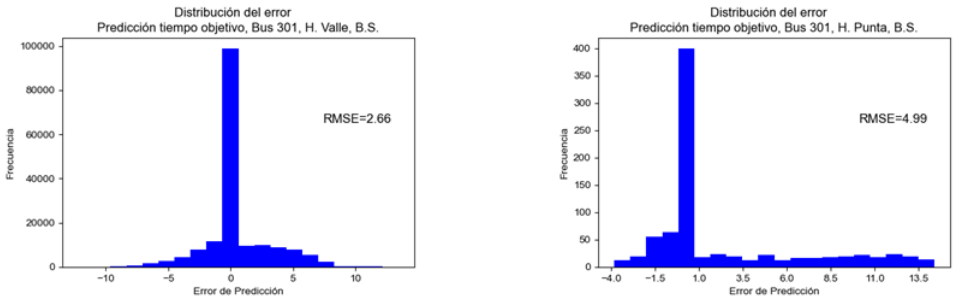

Figura B.4: Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbral 20 minutos. Fuente: Elaboración propia.

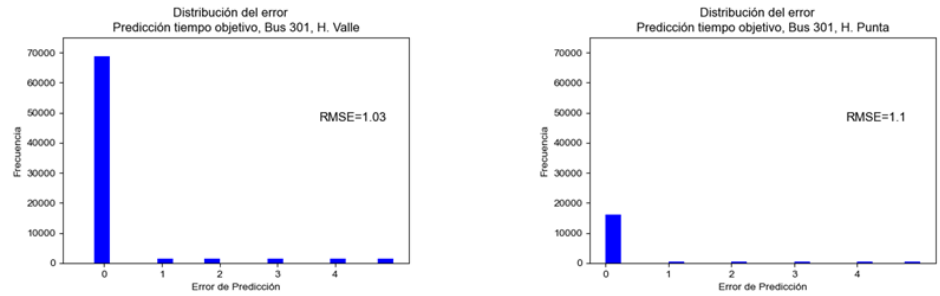

Figura B.5: Resultados red *LSTM* para todo el conjunto de prueba 301, umbral 5 minutos. Fuente: Elaboración propia.

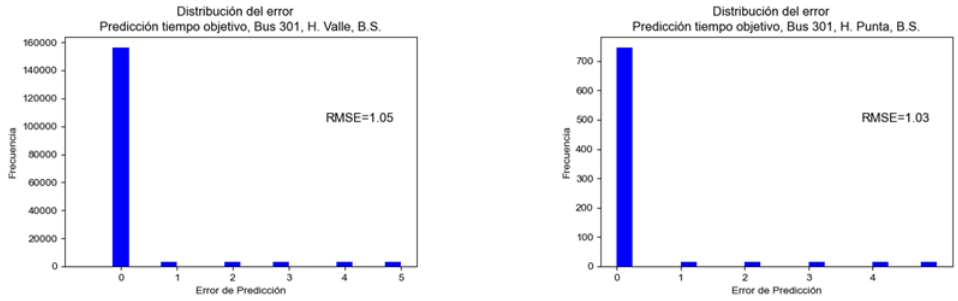

Figura B.6: Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbral 5 minutos. Fuente: Elaboración propia.

B.1.2. Recorrido 315e, conjunto de prueba y bus seleccionado

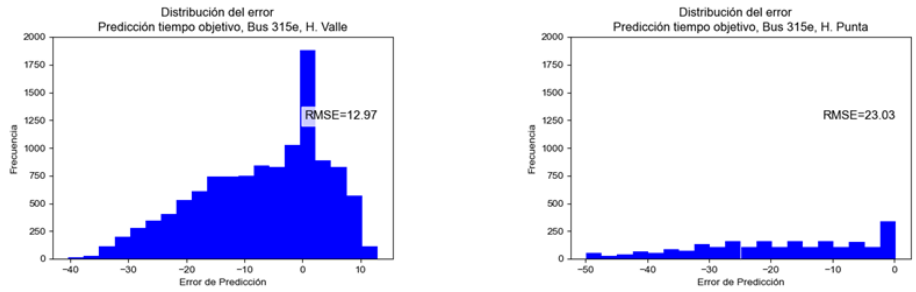

Figura B.7: Resultados red *LSTM* para todo el conjunto de prueba 315e, umbral 50 minutos. Fuente: Elaboración propia.

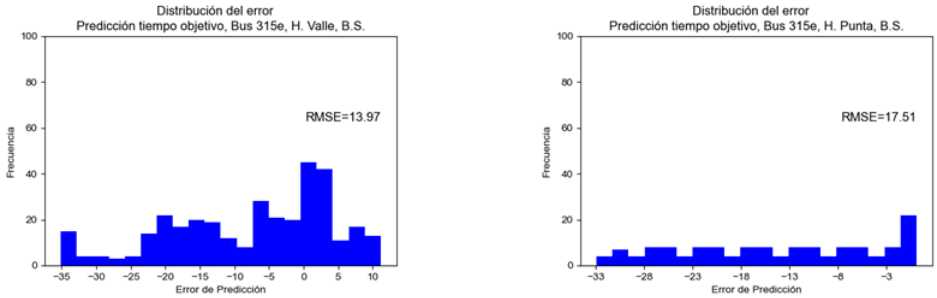

Figura B.8: Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbral 50 minutos. Fuente: Elaboración propia.

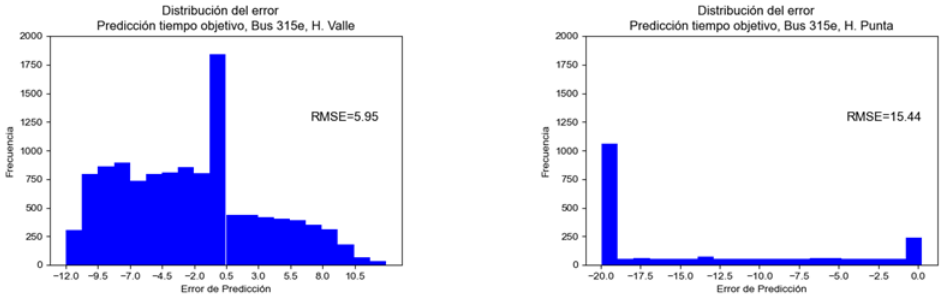

Figura B.9: Resultados red *LSTM* para todo el conjunto de prueba 315e, umbral 20 minutos. Fuente: Elaboración propia.

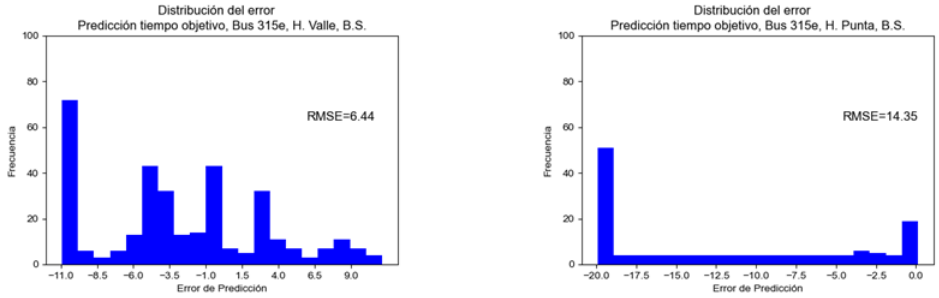

Figura B.10: Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbral 20 minutos. Fuente: Elaboración propia.

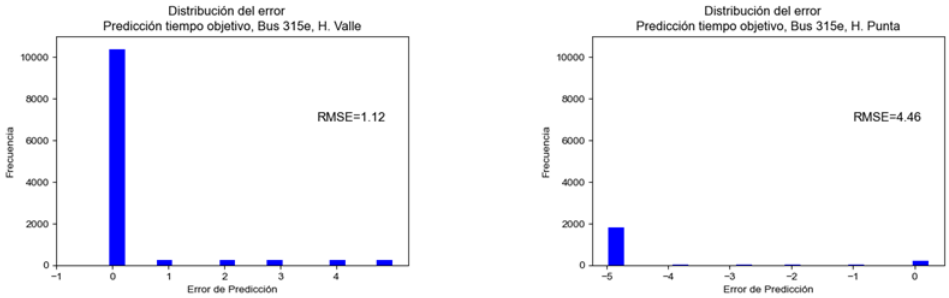

Figura B.11: Resultados red *LSTM* para todo el conjunto de prueba 315e, umbral 5 minutos. Fuente: Elaboración propia.

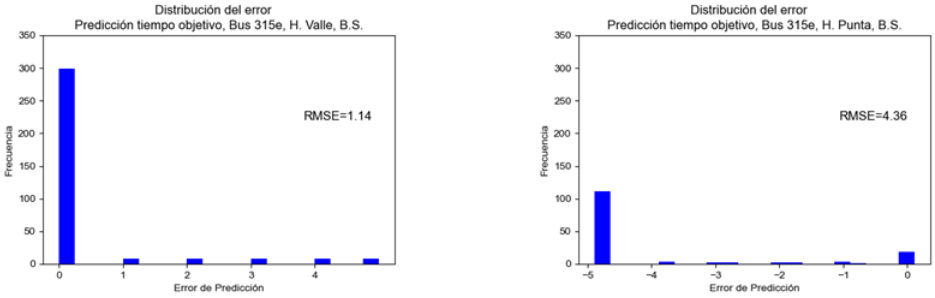

Figura B.12: Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbral 5 minutos. Fuente: Elaboración propia.

B.1.3. Recorrido 506, conjunto de prueba y bus seleccionado

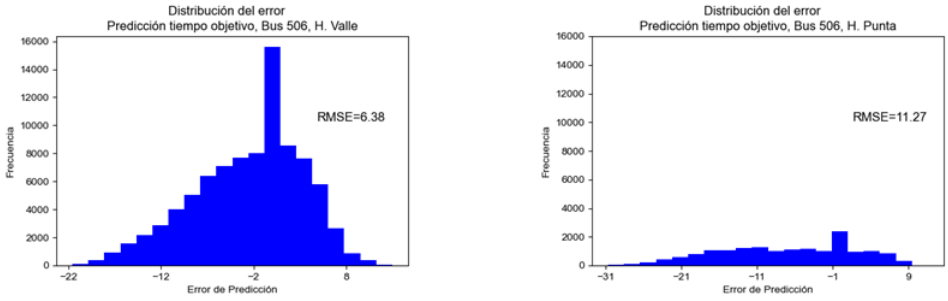

Figura B.13: Resultados red *LSTM* para todo el conjunto de prueba 506, umbral 50 minutos. Fuente: Elaboración propia.

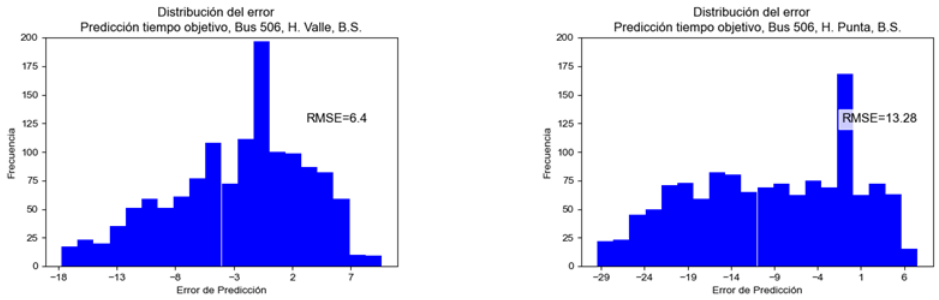

Figura B.14: Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbral 50 minutos. Fuente: Elaboración propia.

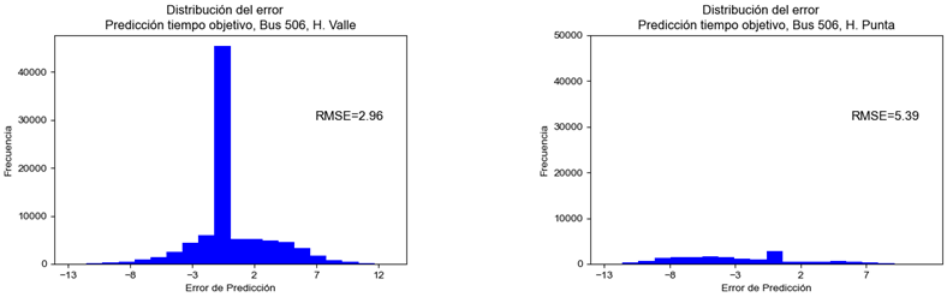

Figura B.15: Resultados red *LSTM* para todo el conjunto de prueba 506, umbral 20 minutos. Fuente: Elaboración propia.

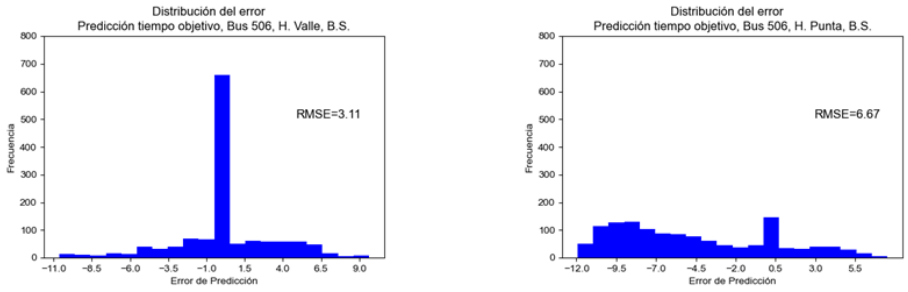

Figura B.16: Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbral 20 minutos. Fuente: Elaboración propia.

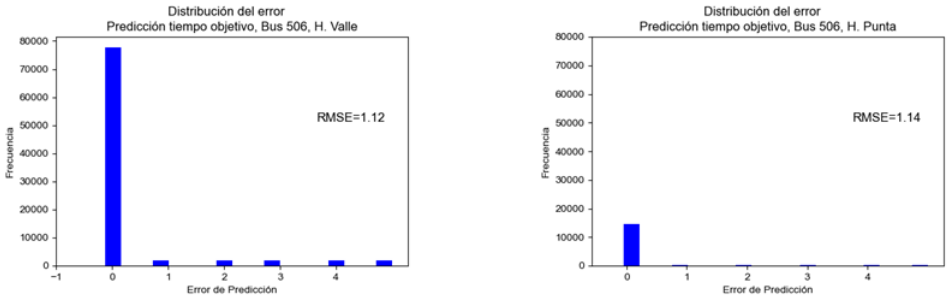

Figura B.17: Resultados red *LSTM* para todo el conjunto de prueba 506, umbral 5 minutos. Fuente: Elaboración propia.

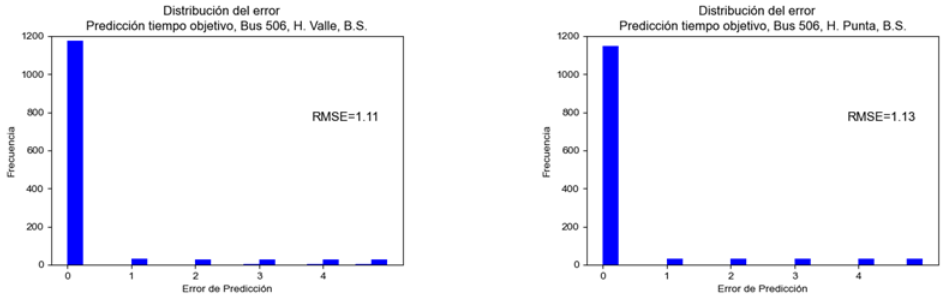

Figura B.18: Resultados red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbral 5 minutos. Fuente: Elaboración propia.

B.2. Gráfico de comparación valor real y valor predicho, bus de prueba

B.2.1. Bus 301

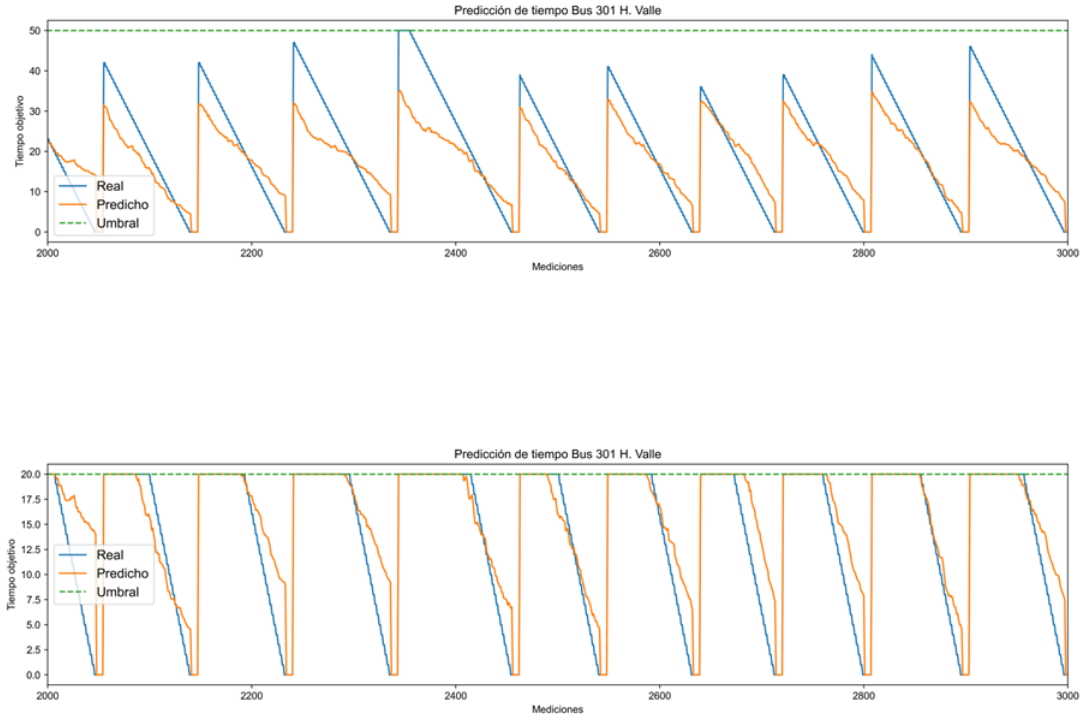

Figura B.19: Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbrales de 50 y 20 minutos, horario valle. Fuente: Elaboración propia.

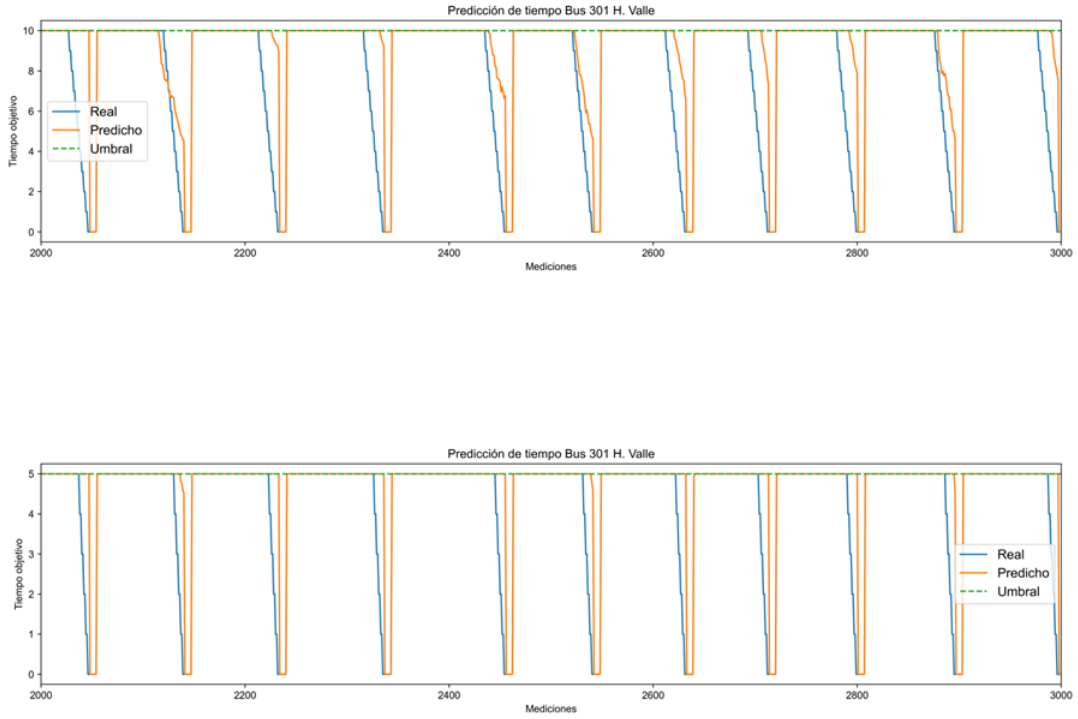

Figura B.20: Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbrales de 10 y 5 minutos, horario valle. Fuente: Elaboración propia.

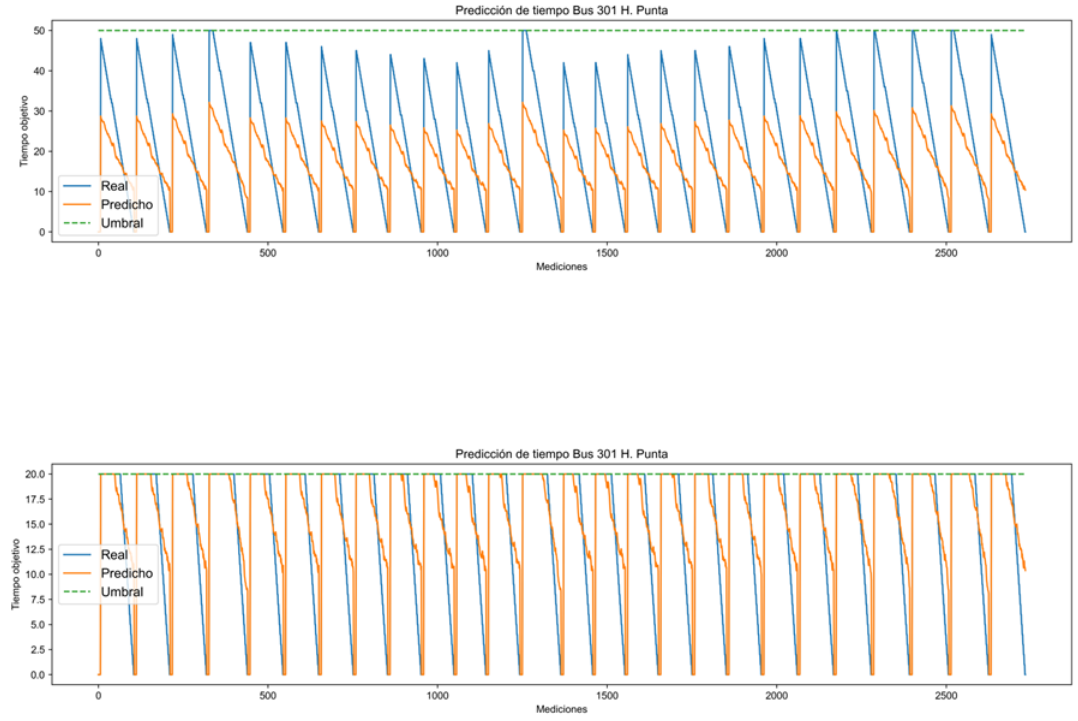

Figura B.21: Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbrales de 50 y 20 minutos, horario punta. Fuente: Elaboración propia.

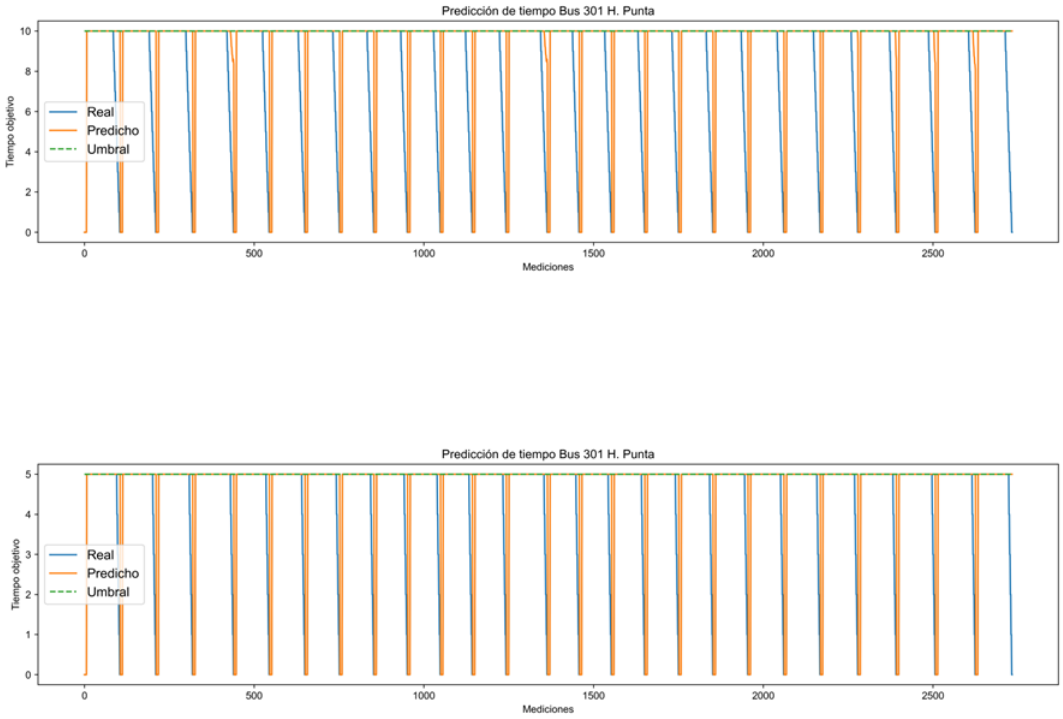

Figura B.22: Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 301, umbrales de 10 y 5 minutos, horario punta. Fuente: Elaboración propia.

B.2.2. Bus 315e

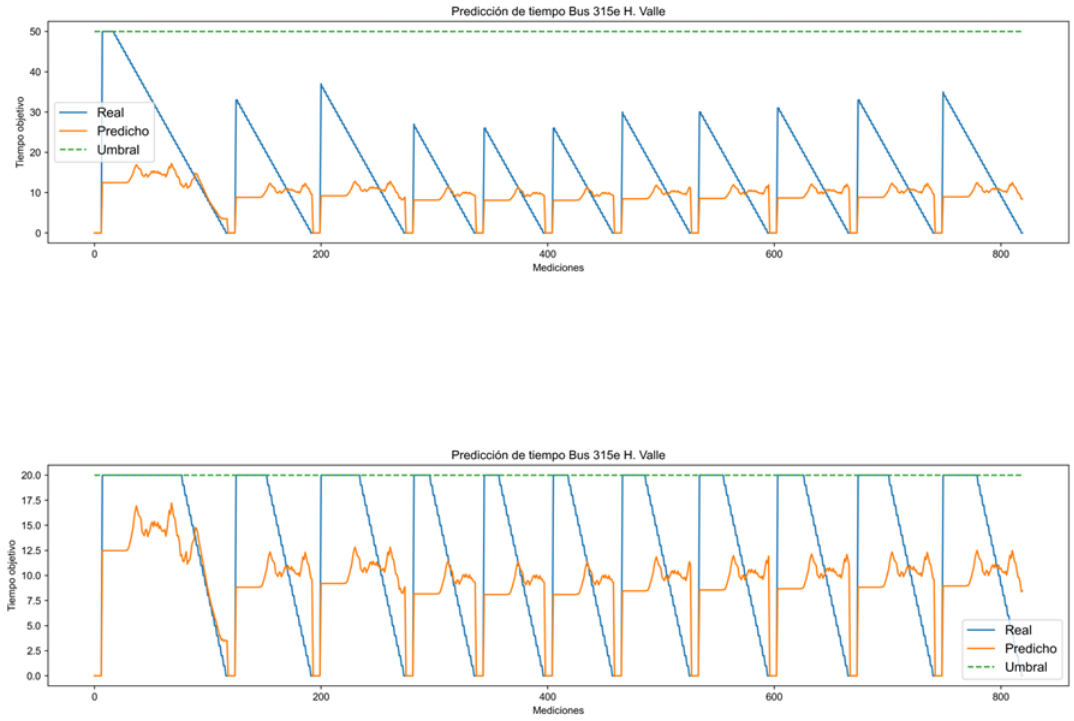

Figura B.23: Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbrales de 50 y 20 minutos, horario valle. Fuente: Elaboración propia.

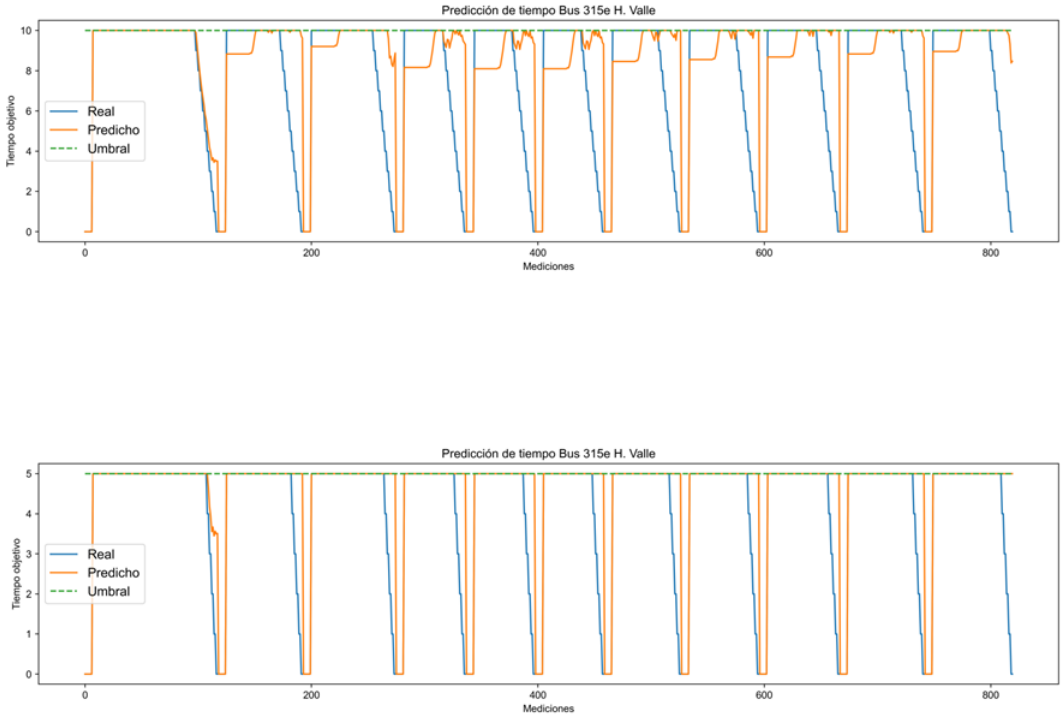

Figura B.24: Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbrales de 10 y 5 minutos, horario valle. Fuente: Elaboración propia.

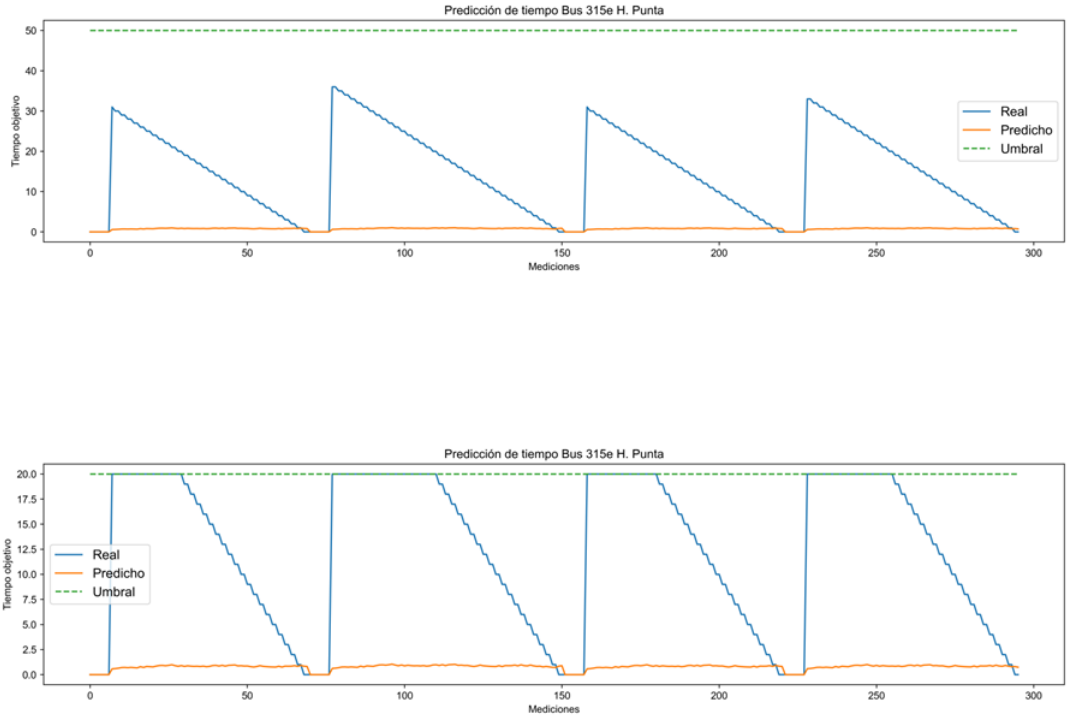

Figura B.25: Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbrales de 50 y 20 minutos, horario punta. Fuente: Elaboración propia.

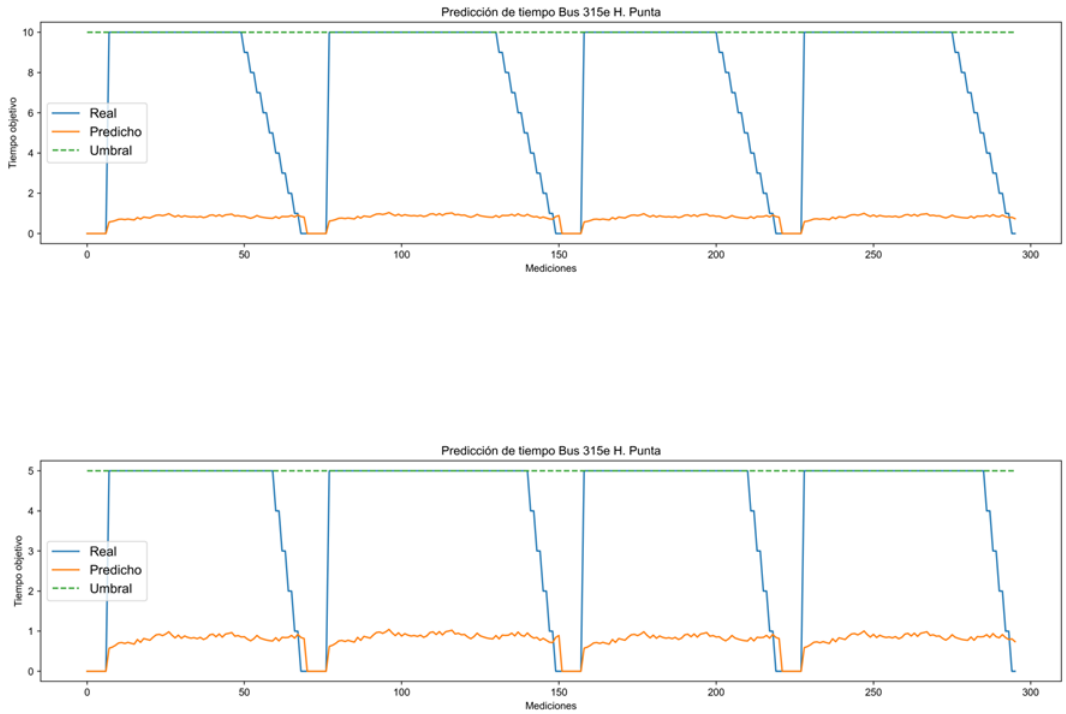

Figura B.26: Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 315e, umbrales de 10 y 5 minutos, horario punta. Fuente: Elaboración propia.

B.2.3. Bus 506

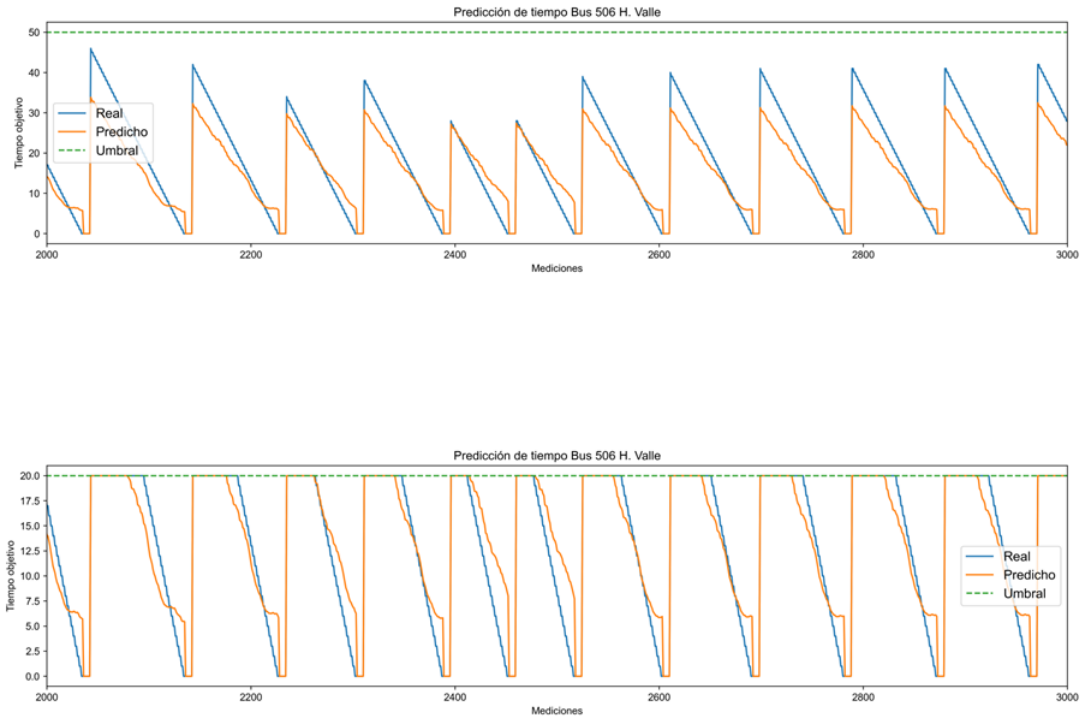

Figura B.27: Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbrales de 50 y 20 minutos, horario valle. Fuente: Elaboración propia.

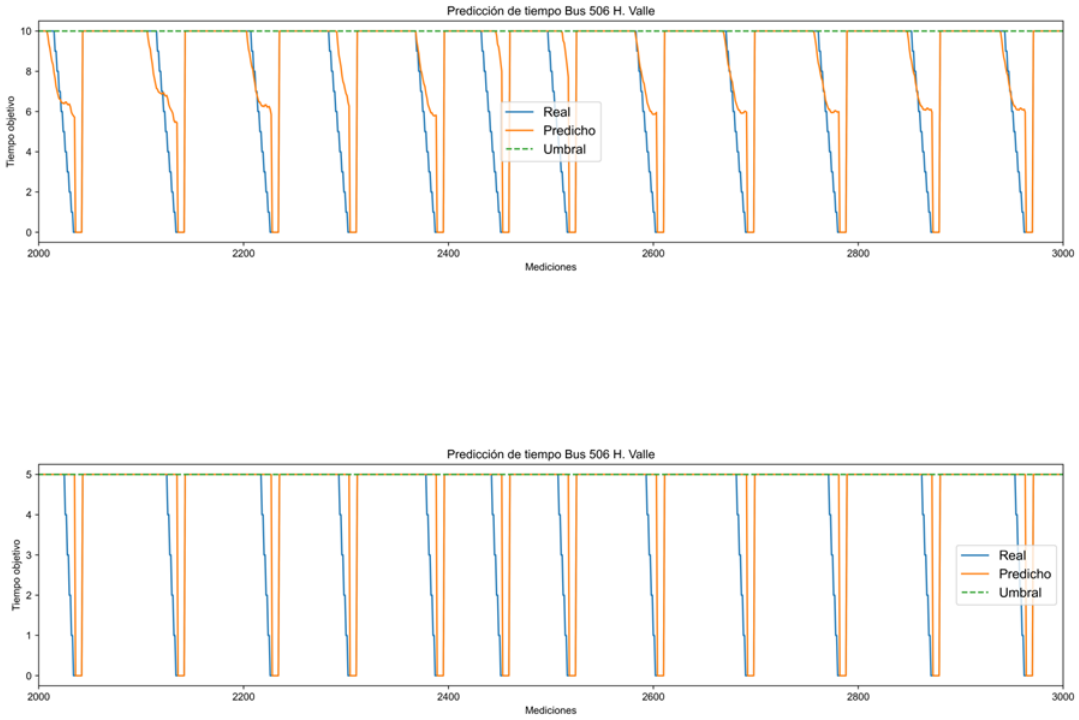

Figura B.28: Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbrales de 10 y 5 minutos, horario valle. Fuente: Elaboración propia.

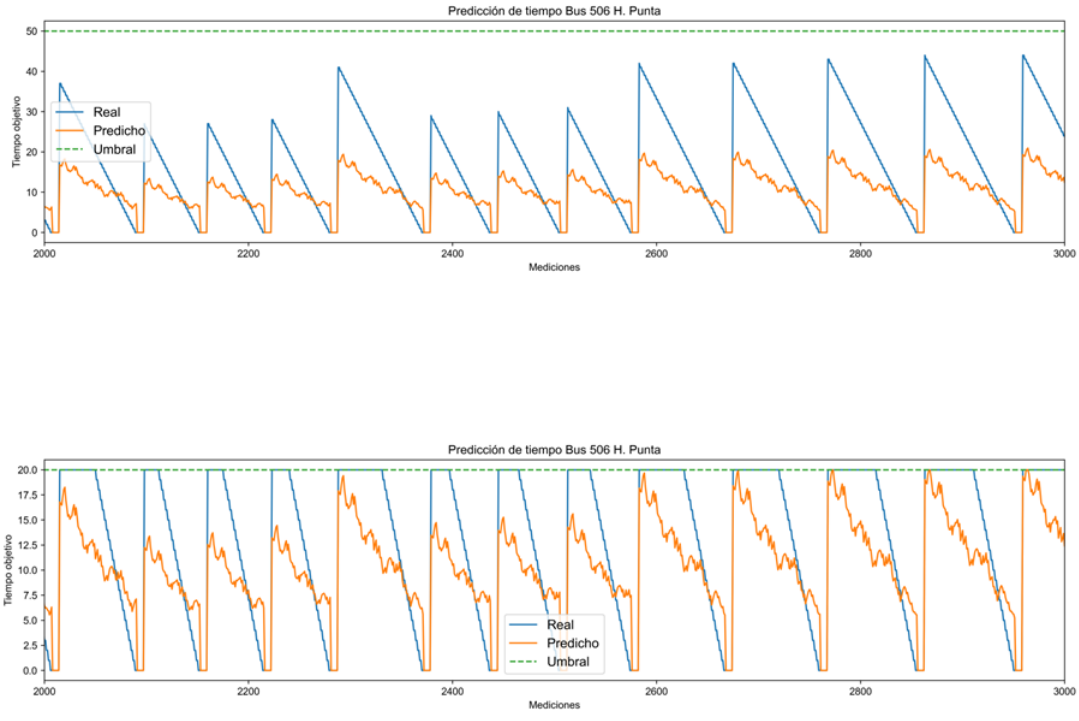

Figura B.29: Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbrales de 50 y 20 minutos, horario punta. Fuente: Elaboración propia.

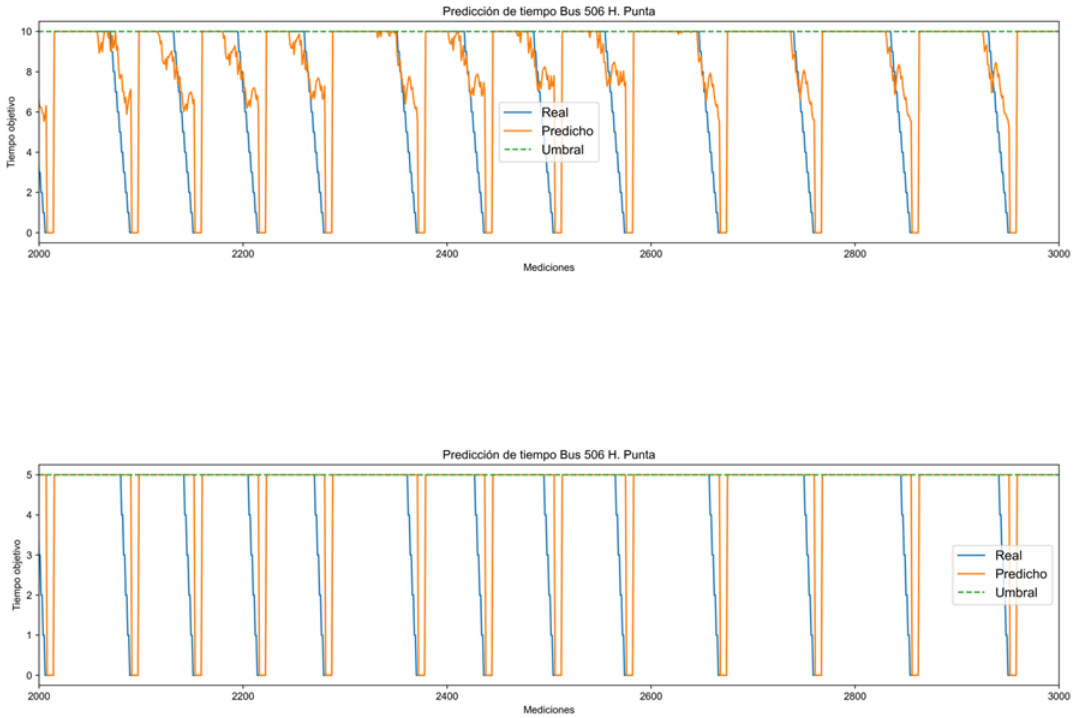

Figura B.30: Comparación entre valor real y valor predicho por red *LSTM* para el bus de prueba seleccionado de recorrido 506, umbrales de 10 y 5 minutos, horario punta. Fuente: Elaboración propia.

Anexo C

Figuras

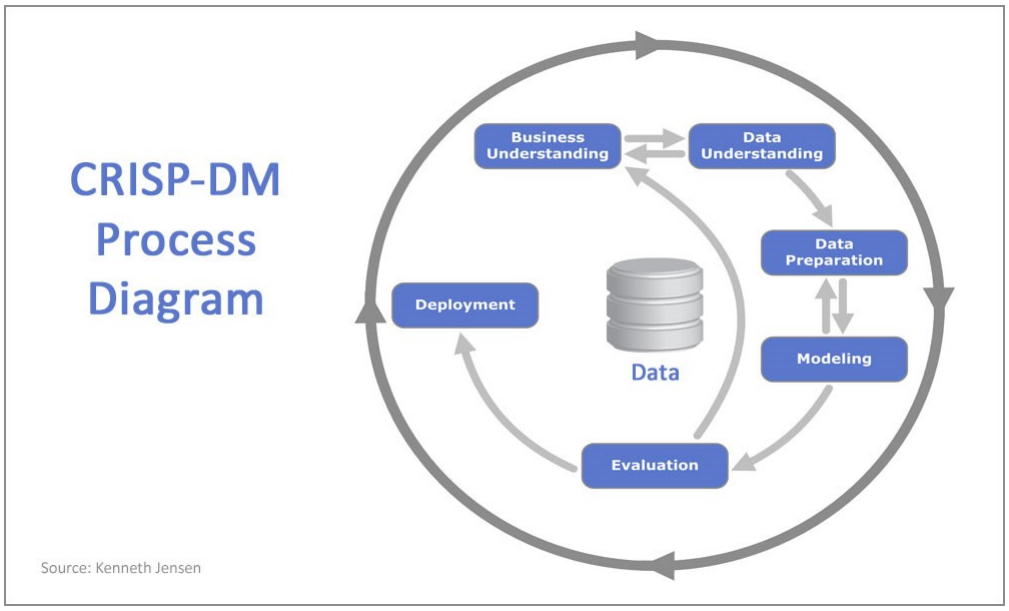

Figura C.1: Metodología Crisp-DM. Fuente: Kenneth Jensen.