

Neural Network を用いた所要時間提供情報の改善

～ETC2.0 データの活用～

阪神高速技術(株)技術部情報管理課 永廣 悠介
阪神高速道路(株)保全交通部交通企画課 西岡 悟史
(公財)日本道路交通情報センター副理事長 岡本 博

要 旨

阪神高速道路では、お客さまに所要時間情報を提供するため、車両検知器データより、高車交通量、低車交通量及び時間占有率を抽出し、走行速度、所要時間を算出している。一方で、ETC2.0 の普及により個別車両の実走行データが利用可能となっている。

本稿では ETC2.0 の実走行データを教師データとして、車両検知器から得られるデータを入力とする Neural Network を用いた所要時間推定モデルを構築した。構築したモデルでは一定の条件のもと、推定値が現行の所要時間推定方式よりも高精度となり、提供情報が改善できることが確認出来た。渋滞の多発している一部の区間を対象とした検証概要と今後の実装に向けた検討の方向性について報告する。

キーワード：所要時間、ETC2.0、AI、機械学習、Neural Network

はじめに

阪神高速道路では、車両検知器を用いて走行速度を算出し、道路情報板によりお客さまに所要時間情報を提供している。お客さまからは、実際にかかった時間との誤差が大きかった旨のご指摘をいただくこともあり、提供情報の精度向上が課題である。

誤差の要因としては大きく2つのものが考えられる。1つ目は車両検知器による速度の測定誤差である。2つ目は提供している所要時間情報が同時刻和（情報板に表示する時点での、行先までの構成区間ごとの所要時間を合計したもの）であることから、時間の経過とともに交通状況が変化することを反映できないことによる誤差である。

本稿では、ETC データを用いて、それらの誤差が実態的にどのようなものであるかを明らかにする。さらに、前者の測定誤差について、誤差を少なくするために人工知能（AI）の一分野である機械学習の適用を試みる。本稿では機械学習の一手法である Neural Network（以下 NN という）を用いた改善モデルを提案する。さらに、後者の誤差改善の方向性についても触れる。

1. 現状の所要時間情報の精度

所要時間情報としては、10km・20km 先までといった目的地までの途中の小区間での車両検知器による現行方式での所要時間の合計（その時点での各小区間の所要時間の総和、以下同時刻和という）を提供している。これに対応する実際の所要

時間として、従来型 ETC の入口アンテナから出口アンテナまでの所要時間を考える。ただし、検知器データは本線のみであるが、ETC データではランプ部分が付加されていることに留意する必要がある。区間設定のイメージを図-1 に示す。

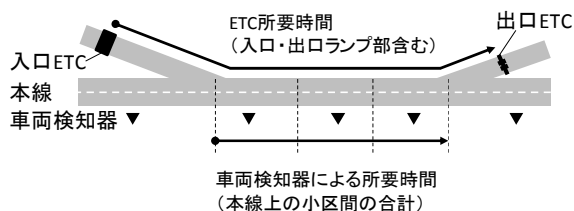


図-1 所要時間比較対象区間の設定

神戸線上り摩耶—西宮 IC 間 (12.8km) について、両者を対比したものが図-2 である。事故等の場合 (図中の×印) や所要時間が大きい場合に誤差が大きい事例が少なくない。事故等の場合では同時刻和約 10 分に対し ETC 所要時間の平均が約 15 分となる点もあり、実際の所要時間が大きくなる例が確認された。

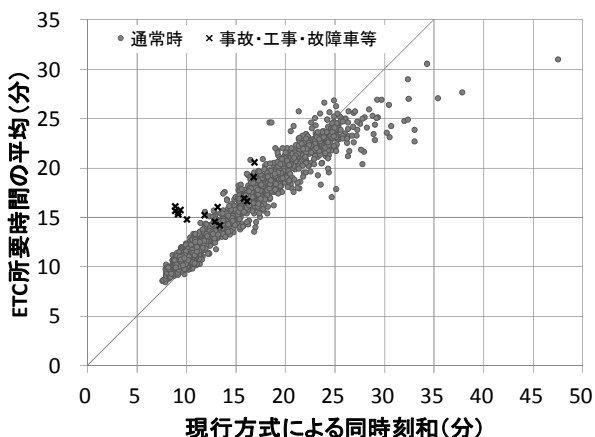


図-2 現行方式による同時刻和と ETC 所要時間の平均との対比

2. 検知器の現行方式での誤差

阪神高速道路での車両検知器による現行の所要時間算出のしくみについて説明する。車両検知器は超音波を路面に対して送出し、反射波の到達時間の違いで車両の有無、低車/高車 (車高 2.3m 以上) の違いを判定している (図-3)。ホールドタイム (検知状態を保持する時間) 以内に連続して車両検知した場合、同一車両として判定する。

超音波パルスデータから検知データとして全車交通量、高車交通量、及び時間占有率を算定し、モデル式(1)により平均速度に変換しているが、速度が正しく推定出来ているかの検証は困難であり現在まで十分になされていなかった。

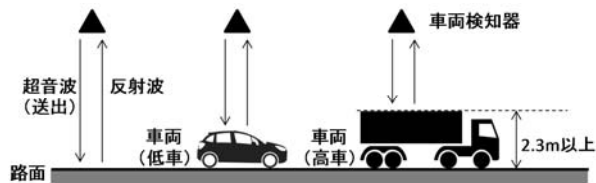


図-3 車両検知器による検出イメージ

平均速度 =

$$\frac{((4.97 \times \text{低車交通量}) + (8.20 \times \text{高車交通量})) / 1000}{((0.32 \times \text{占有時間カウント}) - (0.14 \times \text{全社交通量})) / 3600} \quad (1)$$

低車平均車長 (カタログ値等より設定) : 4.97m

高車平均車長 (カタログ値等より設定) : 8.20 m

占有時間カウントアップ周期 : 0.32 秒

ホールドタイム : 0.14 秒

一方で、ETC2.0 の普及により、個別車両の区間別の実走行速度データが利用可能となった。検知器ごとの推定所要時間(速度の逆数)と、ETC2.0 データによる当該箇所・時間帯での所要時間(同じく速度の逆数)を対比させた散布図が図-4 である。比較は渋滞の多発している表-1 に示す路線上の検知器のデータを対象とした。

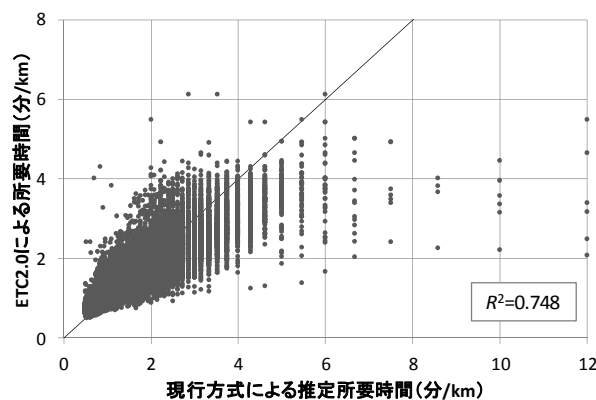


図-4 現行方式による推定所要時間と ETC2.0 による所要時間との対比 (2017/8/1 ~ 8/31, 後述のスクリーニング後のデータ)

表-1 車両検知器による推定所要時間と ETC2.0 による所要時間対比用データ (2017/8/1 ~ 8/31, 後述のスクリーニング後のデータ)

#	路線	検知器	検知器毎時間帯毎データ数
1	池田線 上り (豊中合併 ~ 福島出)	12箇所 S010U-2.5, S010U-3, S010U-3.5, S010U-4, S010U-4.5, S010U-5, S010U-5.5, S010U-6.2, S010U-6.5, S010U-7, S010U-7.5, S010U-8	27,553個
2	神戸線 上り (摩耶入 ~ 西宮IC出)	24箇所 S160U-15, S160U-15.5, S160U-15.9, S160U-16.4, S160U-17, S160U-17.4, S160U-18, S160U-18.6, S160U-19, S160U-19.5, S160U-19.9, S160U-20.4, S160U-21.1, S160U-21.5, S160U-22, S160U-22.3, S160U-23, S160U-23.5, S160U-24, S160U-24.5, S160U-25.1, S160U-25.5, S160U-26, S160U-26.4	64,312個
合計			91,865個

3. ETC2.0 データの概要

ETC2.0 対応の車載器は GPS により取得した緯度・経度の位置情報等をプローブデータとして時々刻々蓄積している。この蓄積データは路側機通過時に車載器からアップリンクされ、プローブ統合サーバで処理される。

プローブ統合サーバで処理され出力されるデータ様式には種々あるが、このうち「DRM リンク単位車両別旅行時間」(様式 2-1) と呼ばれるデータには、DRM (Digital Road Map, デジタル道路地図) でマップマッチングし DRM リンクへ紐付けされたプローブデータが、車両毎・DRM リンク単位の始点通過時刻、旅行時間、旅行速度として集計・格納されている。旅行時間データ格納までの流れを図-5 に示す。

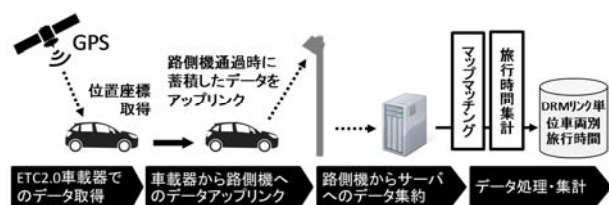


図-5 ETC2.0 による旅行時間集計フロー

本稿では車両検知器による推定所要時間精度の検証に、この DRM リンク単位車両別旅行時間データの利用を試みることにした。

データの利用にあたっては車両検知器に対し最

短距離にある DRM リンクを対応付け、対応付けされた DRM リンクの集計値を利用する。最短距離にある DRM リンクは、図-6 に示すとおり車両検知器から DRM リンクに下ろした垂線の長さが最短となるものを選択する。

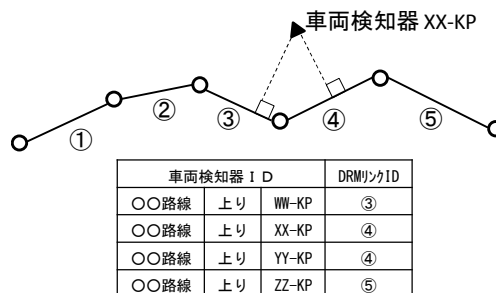


図-6 車両検知器と DRM リンクの対応付イメージ

車両検知器による所要時間は5分毎の統計値である。ETC2.0 による所要時間は、車両検知器に対応する DRM リンク ID と、対象5分間を条件に「DRM リンク単位車両別旅行時間」データから旅行速度を抽出し、平均を取った値を利用することとした。

4. NN モデルの概要

機械学習の一手法である NN (Neural Network) は、脳内のニューロンを模擬し、ある入力条件の場合にのみ発火する条件式をネットワーク状に配置したものである。用途としてはクラス分類 (2値, 多値)、回帰推定がある。本稿においては車両検知器計測値から算出できるデータを説明変数とし、ETC2.0 による所要時間を被説明変数とする回帰推定に NN モデルを使用した。

従来の回帰分析においては、説明変数と被説明変数との相関が回帰式により表された。一方、NN においては説明変数と被説明変数の組合せを学習データと教師データの組合せとして大量に与える事で、モデル内で重み係数を算出させる。学習時のモデル設定 (ハイパーパラメータ) や学習データ量にもよるが、繰り返し学習を進めることで当てはまりのよい回帰モデルが導出できる。NN モ

デルの詳細についてはどのような重み係数が設定されているのか理解が難しくブラックボックスのようである。

5. NN モデル実装環境

本稿では、様々な数理計算ツール、統計ツールが公開されており、機械学習の実装でよく利用されているスクリプト言語のひとつである Python 上で、機械学習、データマイニング用ツールキット Orange を使用し NN モデルの構築を行った。Orange はスクリプトを作成せず、マウスによる操作で NN による回帰モデルを構築することが出来る。NN モデルの構築環境は表-2 のとおりである。表中の Anaconda はインストーラにより簡易に Python 環境を導入できる。Orange の画面操作イメージを図-7 に示す。

表-2 実行環境

#	名称	備考
1	Windows	基本ソフトウェア。
2	Python	Windows、Linux等幅広いOS上で動作するプログラム言語。統計・解析等幅広い領域のツールが公開されている。
3	Anaconda	インストーラにより簡易にPythonの開発・実行環境を導入できるパッケージ。
4	Orange	データ視覚化、機械学習、データマイニング用ツールキット。マウスによる視覚的な操作が可能。

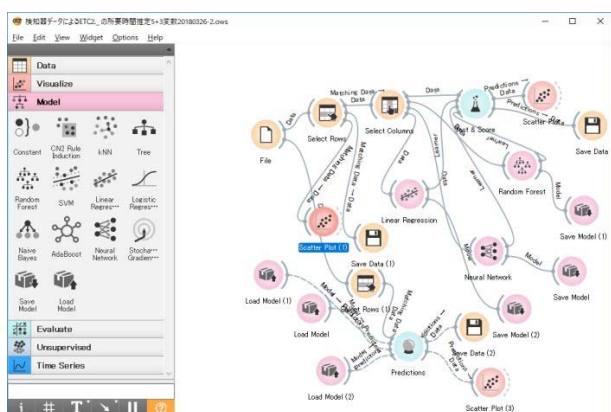


図-7 Orange の操作画面例

6. 所要時間推定精度の評価

所要時間を推定した結果がどの程度実測値に近

い値となっているかについて評価する指標として、本稿では決定係数 R^2 を用いた。 R^2 は以下の式により求められる値で、

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_{true,i} - y_{pred,i})^2}{\sum_{i=1}^n (y_{true,i} - \overline{y_{true}})^2} \quad (2)$$

y_{true} は実測値、 y_{pred} は推定値、 $\overline{y_{true}}$ は y_{true} の平均値、 n はデータ件数である。

実測値のばらつきのうち、予測値により推定できている割合を示すもので、 R^2 の値が 1 に近いほど推定精度が高い。

図-4 の現行方式による推定所要時間を y_{pred} 、ETC2.0 による所要時間を y_{true} とした場合、 R^2 の値は 0.748 となる。

7. NN モデル学習の設定

NN による推定を進めるにあたり、モデルに与える説明変数の選定、NN の学習設定、及び学習データのスクリーニングについて検討した。

NN の学習設定は人間により設定する必要があり、試行錯誤により適切な値を求めていく。これらの設定のことをハイパーパラメータという。Orange 上の NN で学習を行う際の設定項目には表-3 の項目があり、それぞれ表-3 に示す設定値で良好な推定ができた。

表-3 ハイパーパラメータの設定

#	項目	概要	設定値
1	Hidden layers	隠れ層のニューロン数	20, 20, 20
2	Activation	活性化関数	Logistic
3	Solver	重みベクトルの最適化アルゴリズム	Adam
4	Alpha	L2正則化パラメータ	0.0001
5	Max iterations	最大繰り返し回数	300

被説明変数及び説明変数は表-4 の組み合わせとし、学習データのスクリーニング設定については表-5 のとおり試行錯誤により決定した。これは、着目区間・着目時間帯 (5 分間) に対し、下

流区間・上流区間および前時間帯の影響があることを想定して作成した100種以上の説明変数から選定したものである。

ETC2.0 データには異常値が含まれている可能性がある。本検討では異常値による教師データの確かさへの影響を排するため、5分間のデータ件数が少ないデータを除外する条件として(表-5)の1を、バラツキの大きいデータを除外する条件として表-5の2を設定した。表-5の3~6については、事故・工事・故障車等がある場合は通常と状態が著しく異なり、データ数も少ないことからモデルの学習に寄与しないため除外することとした。表-5の7~9については、車両検知器データを所要時間推定の説明変数としていることから、検知器に故障がある場合は無効データとなるため除外している。

表-4 被説明変数, 説明変数の選択

#	被説明変数	備考
1	ETC2.0のkmあたり所要時間	分
#	説明変数	備考
1	検知器名称	
2	日種	平日, 土日, 祝日
3	時刻	5分単位
4	第二走行車線 交通量	台/5分
5	第二走行車線 高車交通量/交通量	
6	第二走行車線 時間占有率/交通量	1/(台/5分)
7	下流側 第二走行車線 時間占有率/交通量	1/(台/5分)
8	5分前 第二走行車線 時間占有率/交通量	1/(台/5分)
9	上流側 第二走行車線 時間占有率/交通量	1/(台/5分)

表-5 学習データのスクリーニング条件

#	条件
1	3 < ETC2.0のデータ件数
2	0.0 ≤ ETC2.0のkmあたり所要時間(分)の標準偏差 ≤ 0.2
3	検知器のkmあたり所要時間(分) < 15
4	着目区間 渋滞状況: 非渋滞、自然渋滞(事故・工事・故障車等による渋滞を除く)
5	着目区間の5分前 渋滞状況: 非渋滞、自然渋滞(事故・工事・故障車等による渋滞を除く)
6	下流区間 渋滞状況: 非渋滞、自然渋滞(事故・工事・故障車等による渋滞を除く)
7	着目区間 第一走行車線の検知器:正常 または 未設置 第二走行車線の検知器:正常
8	着目区間の5分前 第一走行車線の検知器:正常 または 未設置 第二走行車線の検知器:正常
9	下流区間 第一走行車線の検知器:正常 または 未設置 第二走行車線の検知器:正常

8. NN モデルの学習

7. の設定で NN モデルの学習を行った。学習には表-1 のデータを表-5 の条件でスクリーニングし使用した。また、1ヶ月分のデータのうち、平日、週末、及び祝日それぞれ1日ずつ計3日分を検証データ、残りの28日分を学習データとして使用した。データの概要を表-6に示す。

表-6 学習・検証データの概要(スクリーニング後)

年月日	日種	学習/検証	検知器毎時間帯毎データ数		
			池田線	神戸線	計
2017/8/1	平日	学習	125	324	449
2017/8/2	平日	学習	171	82	253
2017/8/3	平日	学習	314	547	861
2017/8/4	平日	学習	555	1,707	2,262
2017/8/5	週末	学習	796	2,620	3,416
2017/8/6	祝日	学習	1,395	2,690	4,085
2017/8/7	平日	学習	522	1,353	1,875
2017/8/8	平日	学習	492	1,744	2,236
2017/8/9	平日	学習	594	1,838	2,432
2017/8/10	平日	学習	607	1,905	2,512
2017/8/11	祝日	学習	1,305	2,765	4,070
2017/8/12	週末	学習	1,408	2,582	3,990
2017/8/13	祝日	学習	1,449	2,657	4,106
2017/8/14	祝日	学習	1,444	3,113	4,557
2017/8/15	祝日	学習	1,283	2,729	4,012
2017/8/16	平日	学習	1,349	2,971	4,320
2017/8/17	平日	学習	665	2,231	2,896
2017/8/18	平日	学習	712	2,036	2,748
2017/8/19	週末	学習	1,341	2,624	3,965
2017/8/20	祝日	学習	1,438	2,958	4,396
2017/8/21	平日	学習	779	2,018	2,797
2017/8/22	平日	学習	783	1,973	2,756
2017/8/23	平日	学習	664	1,921	2,585
2017/8/24	平日	学習	760	1,908	2,668
2017/8/25	平日	学習	738	1,985	2,723
2017/8/26	週末	検証	1,337	2,803	4,140
2017/8/27	祝日	検証	1,474	2,689	4,163
2017/8/28	平日	検証	851	1,944	2,795
2017/8/29	平日	学習	564	1,935	2,499
2017/8/30	平日	学習	842	2,007	2,849
2017/8/31	平日	学習	796	1,653	2,449
合計	計	学習	23,891	56,876	80,767
		検証	3,662	7,436	11,098
			27,553	64,312	91,865

学習モデルは①池田線、②神戸線、③池田線+神戸線の3パターンで学習と検証を実施した。

学習及び検証結果の決定係数の値を表-7に示す。なお、「学習結果」の決定係数 R^2 は学習データに対する10-分割交差検証による値、「検証結果」の決定係数 R^2 は表-6で分離しておいた検証用データに対する推定結果である。

表-7 モデル毎の決定係数 R^2

	学習結果		⇒	検証結果	
	データ数	R^2		データ数	R^2
①池田線 学習モデル	23,891	0.959	⇒	3,662	0.948
②神戸線 学習モデル	56,876	0.956	⇒	7,436	0.952
③池田線+神戸線 学習モデル	80,767	0.959	⇒	11,098	0.953

①～③のモデルそれぞれについて学習モデルによる推定所要時間と ETC2.0 による所要時間との対比を図-8～図-10 に示す。

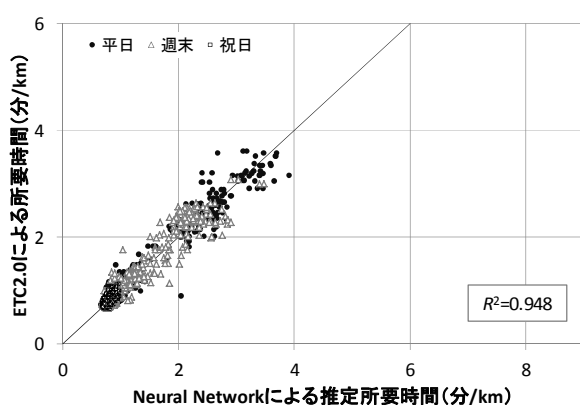


図-8 ①池田線 学習モデルによる推定所要時間と ETC2.0 による所要時間との対比 (検証データ)

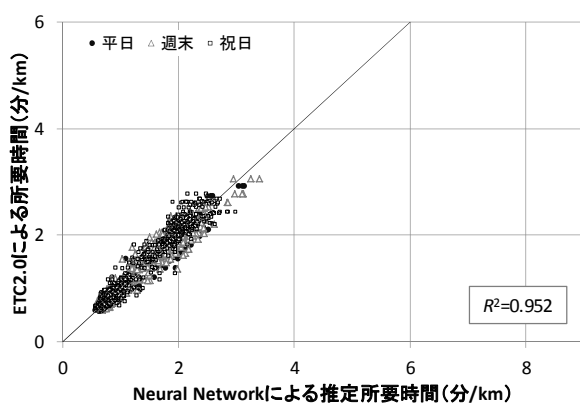


図-9 ②神戸線 学習モデルによる推定所要時間と ETC2.0 による所要時間との対比 (検証データ)

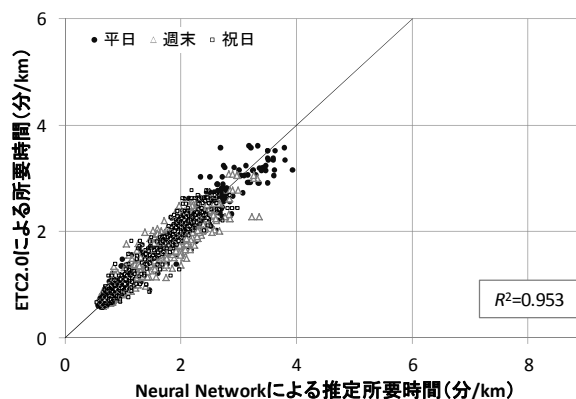


図-10 ③池田線+神戸線 学習モデルによる推定所要時間と ETC2.0 による所要時間との対比 (検証データ)

3 モデルとも決定係数 R^2 の値は 1 に近い値となった。また、図-8～図-10 においても推定値と ETC2.0 による所要時間を対比した点が 45 度線上に分布しており、本稿で構築した NN モデルにより ETC2.0 による所要時間を精度よく推定出来ることが確認できた。

次に各モデル間の検証結果の比較について述べる。①池田線学習モデルと②神戸線学習モデルを比較すると、決定係数 R^2 の値は 0.948 と 0.952 と似通った値となった。図-8 と図-9 を見比べると図-8 では右上 (所要時間が長い方向) に分布する点が図-9 よりも多い。これは平日 (8 月 28 日) において、池田線の方が神戸線よりも所要時間が長いデータが多いことによる。路線毎の交通状況に差があるが、それぞれの路線とも精度よく推定出来ていることが分かる。

③池田線+神戸線学習モデルについては他の 2 モデルと決定係数 R^2 は同程度の値となった。図-10 の分布についても図-8, 図-9 と大きな差異は無く、複数路線で学習したモデルでの推定結果と路線個別に学習したモデルでの推定結果とでそれほど差異は無い結果となった。推定対象路線を拡大する場合に、路線個別の推定モデルと複数路線を同時に推定するモデルのどちらが望ましいかについては今後の更なる検証が必要である。

9. 同時刻和所要時間と追跡和所要時間

はじめに で述べた所要時間の同時刻和の誤差改善について検討する。

実際の走行での所要時間を求めるためには、道路情報板から目的地までについて途中の小区間ごとの所要時間（時間経過とともに交通状況に応じて変化する）を追跡し、合計することが必要となる。これを追跡和所要時間と呼ぶこととする。ただし、これは事後的にのみ算出できるものである。

1. で示した神戸線上り摩耶—西宮 IC 間（12.8km）について、NN モデルでの所要時間推定値を用いた追跡和と ETC の所要時間とを対比したものが図-11 である。現行方式での同時刻和である図-2 に比べ、所要時間の大きい範囲での過大推定が改善されていることがわかる。ただし ETC 所要時間よりやや小さい値となる傾向については更なる検証を要する。

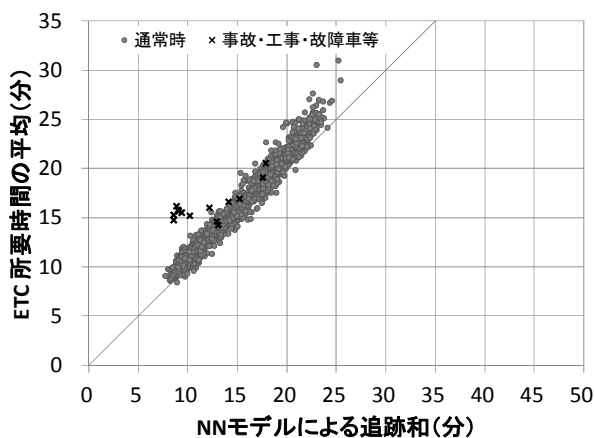


図-11 NN モデルによる追跡和と ETC 所要時間の平均との対比

NN モデルによる推定値で、情報板通過時点で利用可能な同時刻和と事後的に算定される追跡和とを対比したものが図-12 である。両者は非常に高い相関を持っている。言い換えれば、所要時間に追跡和でなく同時刻和を用いることの誤差は小さいと言える。

NN 追跡和の代わりに NN 同時刻和を用いて ETC 所要時間との対比をしたものが図-13 である。

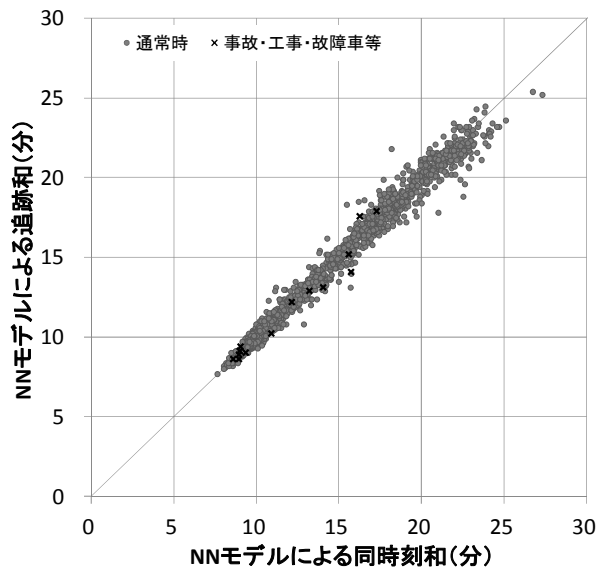


図-12 NN モデルによる同時刻和と追跡和の対比

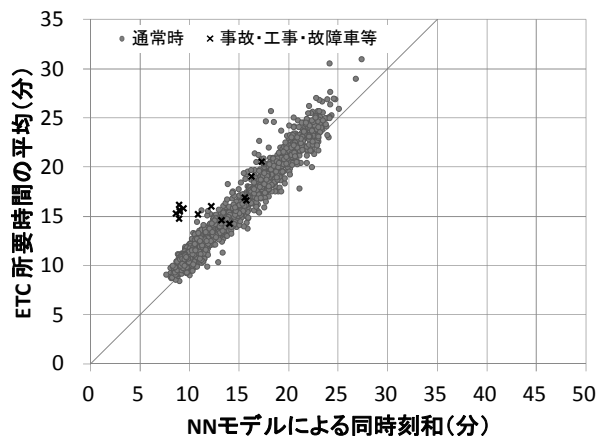


図-13 NN モデルによる同時刻和と ETC 所要時間の平均との対比

これでも、図-2 の現行方式に比べ、所要時間が大きい範囲での過大推定が改善されている。情報板での所要時間提供が 5 分単位の表示であることを考えれば、十分実用可能であると思われる。ただし、事故等特異事象の場合については、依然として誤差が大きくなっている。これは、特異事象のデータ数が相対的に少なかったため学習データから除しており、NN モデルに反映できていないことが要因と考えられる。

所要時間提供時に、その時点では未来の値である追跡和所要時間を適切に予測する方法、特に、事故・工事・故障車等の特異事象での予測方法と

しては、別途の推定方法が必要と思料される。

10. まとめ

本稿では、所要時間推定の新たな試みとして Neural Network を用いた推定モデルを構築し、ETC2.0 の旅行時間データを推定モデルの学習に活用することで検知器データによる所要時間推定精度を改善できることを明らかにした。

同時刻和の誤差改善については、事故等が無い条件下においては NN モデルによる同時刻和が現行方式における誤差を改善できることを確認した。

今後の実装に向けては、本稿で構築した NN モデルでは推定対象区間の拡大に対しての汎化性能が課題として確認された。汎化性能が向上する NN モデルの改良手法が存在するか、または区間毎に個別の推定モデルを構築し並列計算させる方が高精度となるのかについて今後検証していく必要がある。

また、本モデルによっても誤差が大きく残るものとして、事故等特異事象による所要時間の急な増加や、交通規制解除後の所要時間の急な減少など、情報提供時点後の交通状況の変化による誤差がある。これは、数時間の範囲での将来予測を必要とするもので、通常時を対象とする本モデルとは別の手法が必要となる。お客さまへの所要時間提供情報の改善の面では今後の重要な課題である。

謝辞：本稿を執筆するにあたり名古屋大学大学院情報科学研究科川西康友助教に助言をいただいた。ここに感謝の意を表す。

参考文献

- 1) 関静夫, 堀江康之, 田廣恭幸: 交通管制データウェアハウスの改修設計, 阪神高速道路第43回技術研究発表会論文集, pp. 303-308, 2011.
- 2) 巢籠悠輔: 詳解ディープラーニング TensorFlow・Keras による時系列データ処理, 2017.
- 3) 荒屋真二: 人工知能概論 (第2版), 2004.

IMPROVING THE TRAVEL TIME INFORMATION BY USING A NEURAL NETWORK: EFFECTIVE USE OF THE ETC 2.0 DATA

Yusuke NAGAIRO, Satoshi NISHIOKA and Hiroshi OKAMOTO

In order to provide travel time information to the users of the Hanshin Expressway, data collected through the on-road vehicle detectors is used to calculate travel speeds and times from the traffic volumes of standard-sized cars and tall vehicles and the time occupancy ratio. Meanwhile, with an increasing number of vehicles equipped with the ETC 2.0 devices, actual travel data of vehicles is becoming available. In this study a travel time estimation model was developed to process the data from the vehicle detectors by using a neural network trained on the actual ETC 2.0 travel data. The model was found to be capable of more accurately estimating travel times compared to the current estimation method under certain conditions, indicating the possibility of improving the travel time information to be provided to the road users. This report summarizes verification on some frequently congested sections of the Hanshin Expressway and presents a direction for future investigation toward practical implementation.

永廣 悠介



阪神高速技術株式会社
技術部 情報管理課
Yusuke Nagahiro

西岡 悟史



阪神高速道路株式会社
保全交通部 交通企画課
Satoshi Nishioka

岡本 博



(公財)日本道路交通情報センター
副理事長
Hiroshi Okamoto