

적응성 있는 단기간 속도 예측모형 개발에 관한 연구

Adaptive Short-Term Vehicle Speed Prediction Models

조범철

(서울시립대학교 도시공학과 대학원)

이승재

(서울시립대학교 도시공학과 조교수)

목 차

- | | |
|------------------|----------------|
| I. 서론 | 2. 입력변수설정 |
| II. 기존모형의 이론적 고찰 | 3. 모형결정 |
| 1. 다중회귀분석 | IV. 적용 및 결과분석 |
| 2. 시계열분석 | 1. 정확성 분석 |
| 3. 인공신경망 | 2. 적응성 분석 |
| 4. 칼만필터링 | V. 결론 및 향후연구과제 |
| III. 모형구축 | 참고문헌 |
| 1. 자료수집 | |

요 약

본 논문은 도로를 주행하는 차량의 지점속도에 대하여 단기간(short-term)으로 예측하는 네 가지의 모형들에 대한 개발 및 결과의 비교하고 평가했다. 사용된 기법들로는 다중회귀분석, 시계열분석 (ARIMA), 인공 신경망, 칼만필터링 등이며, 모형의 구축을 위하여 다수의 독립변수 및 입력변수가 요구되는 다중회귀분석과 인공 신경망에서는 연속방정식에서 고려되는 변수들간의 단순상관계수 및 편상관계수의 계산을 통해서 입력변수가 설정이 되었으며, 시계열분석(ARIMA)과 칼만필터링 등 단일 입력 변수만을 요하는 모형에서는 바로 전 시간대와 현재시간대의 간격동안 속도의 변화량을 입력변수로 설정하였다. 속도를 비롯해서 교통 데이터는 현장자료를 사용하였는데, 이는 서울의 한강옆에 위치한 올림픽대로 중 한강대로에 위치한 검지기 3개를 통해서 천호동 방향으로 이동하는 교통류에 대해서 17시간 (00시~17시)동안 수집했다. 17시간 수집했는데 그중에 검지된 속도는 14km/h에서 98km/h까지 변하는 등, 수집된 자료에는 다양한 교통상태가 포함되어 있는데 이는 각 모형들의 정확한 예측력과 적응성을 평가하기 위함이었다. 각 모형은 예측하고자 하는 시점으로부터 1, 5, 10, 15분 후의 속도를 예측하는 것으로 총 4가지의 예측시간간격으로 각각 실험되었다. 결과는 전반적으로 신뢰성 있게 나왔으나 그 중에서도 정확성면에서는 인공신경망과 칼만필터링이 우수했고 적응성면에서는 칼만필터링이 탁월했다. 또한 1분 후의 속도를 예측하는 결과들은 모형들간에 거의 비슷한 정확도를 보여주었는데 이는 입력변수의 설정이 중요한 것임을 보여주는 것이라 판단된다.

I. 서론

도시교통체계 안에서 교통량의 증가와 각종 도시활동의 증가로 인해 교통혼잡이 빈번히 일어나고 있으며 이로 인해 국가 전체적으로도 경제적, 사회적, 환경적인 손실이 상당하다. 이로 인해서 1996년 건설교통부와 경찰청의 주관 하에서 지능형 교통시스템(Intelligent Transport System) 기본계획이 국가기본계획으로 수립되었으며, 이는 교통체계의 효율성과 안전성을 제고할 수 있도록 기존의 교통체계에 첨단기술을 접목시키고자 하는 것이다.

ITS의 기능 중에서 여행자정보서비스체계(ATIS; Advanced Travel Information System)는 매우 중요한 역할을 한다. 이는 수집된 교통 데이터를 가공 및 처리하여 통행에 관련되는 여러 가지 교통조건을 예측하여 여행자에게 정보를 제공함으로써 여행비용을 감소시킬 뿐만 아니라 교통관리자에게도 정보를 제공하여 효과적인 교통관리를 할 수 있도록 하게 하는 기능을 수행한다. 수집된 교통 데이터를 가공/처리할 때 예측 시간간격의 정도에 따라 정보제공의 종류를 다음의 3가지로 구분한다.

첫째, 실시간(Real time)정보 제공

교통변화에 따라 실 시간적으로 매우 민감하게 반응하면서 현재의 교통조건을 그대로 추정하여 여행자에게 제공되는 것으로 이는 주로 규제에 이용되며 교통량에 따른 신호규제, 현 교통체계에 따른 최단 경로 선택 등이 여기에 속한다.

둘째, 단기간(Short-term) 예측정보 제공

교통정보센터에서 검지된 실시간의 교통상황정보를 바탕으로 이후의 교통상황, 구간별 통행시간을 예측하여 차량운전자 및 여행자에게 제공하는 것이다. 단기간 예측정보는 누적된 과거 통행시간 자료 및 실시간 자료를 이용하여 5~15분 후의 교통조건을 예측하여 정보를 제공하는 것을 말한다. 이는 주로 신신호시스템 등 간선도로 신호 연동화를 위한 기본적인 자료로 많이 사용된다.

셋째, 장기간(Long-term) 예측정보 제공

과거 통행시간 및 실시간 자료를 바탕으로 향후의 교통상황을 예측하는 점에서 단기간 예측정보와 개념이 유사하나 예측하는 시점을 단기간보다 먼 장래의 정보를 예측한다는 것이 다르다. 이러한 정보는 도로의 설계 또는 통제, 길게는 도로건설에 대한 자료로 사용되기도 한다.

이중에서도 보통 도시내의 상황을 고려해볼 때 단기간의 교통상황을 예측하여 정보를 제공하는 것이 이용자측면과 시스템 관리자 측면에서도 가장 효율적이고 적합한 형태의 정보이다.

제공되는 정보의 내용 면에 있어서, 이 ATIS의 기능을 수행하기 위해서는 각종 검지기로부터 수집된 자료가 가공, 처리되어서 제공되어야 할 기본적인 2가지 종류의 정보가 있는데 첫째로는 통행시간 또는 통행속도에 대한 예측정보이고, 두 번째로는 동적 기중점(Dynamic Origin-Destination)에 대한 예측정보이다. 그러나 동적 기중점예측 또한 경로에 대한 통행시간

및 속도를 기초로 하여 예측하므로 가장 기본적으로 필요한 정보 처리과정은 통행시간 및 통행속도 예측과정이라고 할 수 있겠다.

P.C. Vythoulkas와 Iwao Okutani는 “도시내 도로에서 관찰되는 교통류의 현상에 대한 패턴은 시간과 공간에 대해서 관계없이 상당히 불확실하고 불규칙하게 변하므로 단기간의 교통상황에 대한 예측은 효율적이고 정확한 결과를 예측하기 위해서 예측모형은 현재의 교통조건에 쉽고 유연하게 적응성 있게 예측하여야 한다”라고 언급했다. 이는 단기간 교통상황예측에 대한 특징을 잘 묘사했다고 할 수 있다.

그러나 여러 가지의 기술적인 문제로 인하여서 직접적인 정보의 획득(예를 들면 첨단기술을 이용한 AVI Systems)이 어렵기 때문에 현재에 주어지는 교통시설을 통해서 예측할 수밖에 없다. 이를 위하여 현재까지 다양한 통계적 기법에서부터 인공지능을 이용한 다양한 예측방법까지 많이 연구되어 왔다. 이러한 연구의 주된 흐름을 요약해 볼 때 다음의 두 가지로 구분될 수 있다.

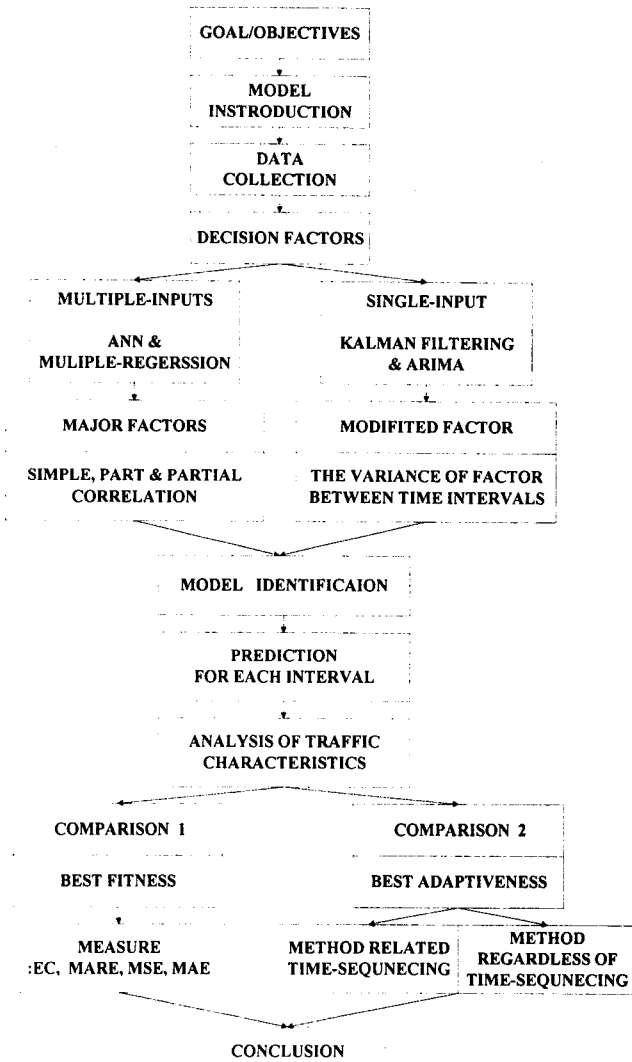
첫째, 어떠한 모형을 사용할 것인가에 대한 문제

둘째, 선정된 모형에서 어떻게 내부적인 과정을 이용할 것인가에 대한 문제이다.

위의 두 번째의 문제인 모형의 알고리즘에 대한 부분은 많이 연구되어 상당히 발전을 이루었다. 그러나 아무리 좋은 알고리즘도 변수의 설정에 따라 상당히 결과가 많이 달라짐은 예측의 기본인 회귀분석에서도 알 수 있다. 따라서 본 논문에서 새로이 연구하고자 하는 것은 모형의 내부적인 사용과 함께 좀더 효율적인 측면에서 어떤 변수를 사용할 것인가에 대한 문제를 병행하고자 한다. 그리고 결과의 신빙성을 위하여 다양한 교통상태(Traffic Performance State)를 포함하는 실측된 자료를 사용하고 또한 각 모델에 대한 결과의 분석으로서 정확도 분석과 적응성 분석을 두 가지를 실시하여 실험된 모형 중 정확도와 적응력이 우수한 모형을 검증하고자 한다.

본 논문의 흐름으로 2장에서는 사용된 모형의 기존의 이론적인 고찰을, 3장에서는 입력변수 선정을 통한 모형의 결정을, 4장에서는 위에서 제시한 두 가지 방법으로 분석 결과를 평가한다.

아래는 본 논문의 구성을 Flowchart로 그린 것이다.



II. 모형의 소개

통행시간예측을 위하여 현재까지 다양한 접근방법론들 중에 다중회귀 분석(Multiple-Regression Analysis), 시계열분석(ARIMA), 칼만필터링(Kalman Filtering), 인공신경망 이론(Artificial Neural Networks) 등이다. 이들 예측기법들에 대해서 간단히 요약하면 아래와 같다.

1. 다중회귀분석

다중회귀분석은 하나의 종속변수와 몇 개의 독립변수와의 관계를 설명할 수 있는 통계적인 기법이다. 이 다중회귀분석의 목표는 알고 있는 몇 개의 독립변수를 사용하여서 미지의 종속변수 한 개의 값을 예측하는 것으로 모형식은 일반적으로 아래와 같다.

$$Y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_j x_j + E$$

E는 에러항이다. 다중회귀에서는 최소자승법(The method of least squares)를 통해서 예측치와 관측치간의 차이를 최소화시킬 때 즉, 아래식을 최소화시키는

α 와 β 으로 결정된다.

$$L = \sum_i (Y_i - \alpha - \sum_j \beta_j X_{ij})^2$$

여기서 i : 관측시의 개수

Y_i : 독립변수의 i 번째 관측치,

j : 독립변수의 개수

X_{ij} : j 번째 독립변수의 i 번째 관측치

따라서 위의 식에서 관측치와 예측치 간의 에러를 최소화하는 점은

$$\frac{\partial L}{\partial \alpha} = 0, \frac{\partial L}{\partial \beta_j} = 0$$

이 되고 이를 위의 식을 이용 다시 쓰면 아래와 같고

$$\sum_i Y_i = \alpha n + \sum_j (\beta_j \sum_i X_{ij})$$

$$\sum_j \sum_i Y_i X_{ij} = \alpha (\sum_j \sum_i X_{ij})$$

$$+ \sum_j (\beta_j \sum_i X_{ij}^2) + (\sum_j \beta_j \sum_j \sum_k X_{ik} \sum_j X_{ij})$$

$$\forall j, k \quad j \neq k$$

여기서, k 는 종속변수의 관측치 개수

마지막과정으로써 α, β 를 찾기 위해 연립방정식을 풀고 $X_{ij} - E(X_j) = x_{ij}, Y_j - E(Y_j) = y_j$ 라고 치환하여 정리하면, 아래와 같다.

$$\beta = \frac{\sum_j (\sum_i y_i x_{ij}) \sum_k (\sum_i x_{ik}^2) - \sum_k (\sum_i y_i x_{ik}) \sum_i (\sum_j x_{ij} \sum_k x_{ik})}{\sum_j (\sum_i x_{ij}^2) \sum_k (\sum_i x_{ik}^2) - (\sum_i (\sum_j x_{ij} \sum_k x_{ik}))^2}$$

$$\alpha = E(Y) - \sum_j \beta_j E(X_j)$$

$$\forall j, k \quad j \neq k$$

다중회귀분석에서 중요한 것은 선정된 독립변수간의 다중공선상현상(Multicollinearity)이 나타날 수 있다는 것이다. 이때는이는 변수간의 상관계수를 사용함으로써 이중으로 선택이 되지 않도록 하여야하는데 사용되는 상관계수으로써는 단순상관계수(simple correlation coefficient)와 편상관계수(part / partial correlation coefficient)가 있고, 이는 3절에서 자세히 다룰 것이다.

2. 시계열분석 : ARIMA 모형

Box-Jenkins 이론은 Long-term 및 Short-term 예측에 모두 이용되는데 이 이론의 장점은 이용자에게 여러 가지 모델을 지원하며 또한 이용자가 가장 적당한 모델을 선택할 수 있게 한다. 음지수평활화(exponentially smoothing)와 같은 기존의 기법은 일반적인 이론의 특수한 경우로 고려된다. ARIMA모형은 자기회귀(AR ; Auto-Regression)와 이동평균(Moving Average)방법이 통합된(Integrated) 새로운 방법으로서 일반적인 ARIMA(p, d, q) 모형은 다음과 같다.

$$\phi(B) \Delta^d f(t) = \theta_0 + \theta(B) a(t)$$

여기서 $f(t)$: 시간 t 에서의 예측값

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p: \text{AR모형}$$

$$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q: \text{M R 모형}$$

$$\Delta f(t) = f(t) - f(t-1),$$

$$\Delta^d f(t) = \Delta[\Delta^{d-1} f(t)]$$

$$Bf(t) = f(t-1), \quad B^m f(t) = f(t-m)$$

θ_0 는 전체적인 이동평균상수이고 $a(t)$ 는 random noise를 의미한다.

여기서 최적 예측시간($D(t)$)을 다음의 값으로 판단한다.

$$D(t) = f(t) - m(t)$$

$m(t)$: 시간간격 t 에서의 평활화된 실측값

ARIMA모형의 단점으로써는 예측에 대한 time-lag(시간지체)이 발생한다는 것이다. 이는 바로 반응을 하지 못하고 반응하는 시간이 필요하다는 뜻이다. 또한 불규칙한 상태에서는 예측력이 떨어지는 단점을 갖고 있다.

3. 칼만필터링

칼만필터링(Kalman filtering)은 무작위변동(random perturbation)을 발생시키는 동적 체계에 적용되는 최적 상태 예측과정으로써 가우시안 백색잡음에 의해 교란되는 선형동적체계의 일시적인 상태를 추정하는 반복적(recursive)인 방법이다. 칼만필터링은 상태평균과 공분산을 계산하기 위한 알고리즘으로써 특징 되어진다. 이러한 알고리즘을 구성하기 위해서는 먼저 다음과 같은 상태방정식(state equation)과 관측방정식(observation equation)이 구성된다.

$$x_{k+1} = \phi_k x_k + w_k \quad (\text{상태방정식})$$

$$z_k = H_k x_k + v_k \quad (\text{관측방정식})$$

여기서, x_k : 시점 t_k 에서의 $(n \times 1)$ 상태벡터

ϕ_k : 시점 k 에서 $k+1$ 로의 $(n \times n)$ 전이행렬

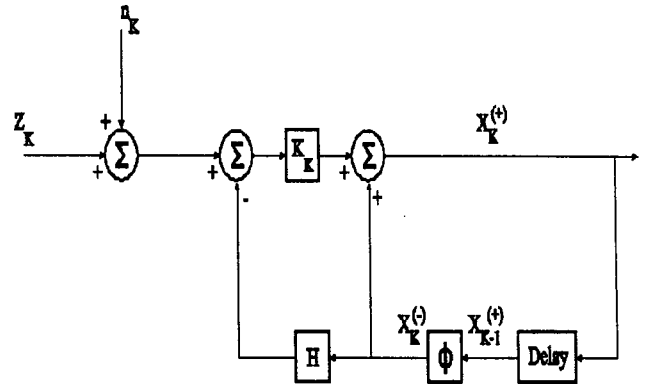
w_k : 알려진 공분산을 가진 $(n \times 1)$ 백색순서(white sequence)로 구성인자의 평균이 0이고 다른 변수와의 상관관계가 없는 벡터

z_k : 시점 t_k 의 $(m \times 1)$ 관측벡터

H_k : 시점 t_k 에서 관측벡터와 상태벡터를 연결시켜 주는 $(m \times n)$ 벡터

v_k : 알려진 공분산을 가지고 w_k 와 상관관계가 없는 $(m \times 1)$ 관측오차

일반적인 알고리즘을 살펴보면 다음과 같다.



<그림1> The Algorithms of Kalman Filtering

<초기화 단계> 초기자료 입력 : 사전 예측치 $\hat{x}_{\bar{k}}$, 공분산 오차행렬 $P_{\bar{k}}$ 입력한다.

<칼만 이익 계산> 초기자료에 의해 칼만 이익을 계산한다.

$$K_k = P_{\bar{k}} H_k^T (H_k P_{\bar{k}} H_k^T + R_k)^{-1}$$

<상태벡터 갱신> 관측벡터 z_k 를 이용하여 상태벡터 추정치 갱신

$$\hat{x}_k = \hat{x}_{\bar{k}} + K_k (z_k - H_k \hat{x}_{\bar{k}})$$

<공분산 계산> 갱신된 예측치에 대한 공분산 계산

$$P_k = (I - K_k H_k) P_{\bar{k}}$$

<장래 예측> 장래사전 예측치 $\hat{x}_{k+1} = \phi_k \hat{x}_k$

$$P_{k+1} = \phi_k P_k \phi_k^T + Q_k \text{장래사전 공분산}$$

KALMAN FILTERING의 가장 큰 장점은 ARIMA모형에 비해서 정상적이지만 불규칙성을 가지는 자료를 토대로 한 미래 통행시간 등에 훌륭한 예측력을 보이고 있는 것으로 평가된다. 즉, 변동하는 교통량에 대해서 정확한 예측력을 보이는 것으로 대체로 평가된다.

KALMAN FILTERING의 또 다른 장점은 관측방정식을 사용함으로써 KALMAN GAIN을 구하도록 함으로써 공분산과 함께 오차를 계속적으로 줄여주는 알고리즘을 가지고 있다는 것이다.

4. 인공신경망

인공 신경망 중 가장 많이 이용되는 모형은 다층선행신경망(Multi-Layer Feedforward)이다. 이 모형은 학습을 위한 학습 알고리즘으로 각 입력패턴으로부터 네트워크가 계산한 출력값과 목표값과의 오차를 최소화시키는 방향으로 네트워크의 연결강도를 조절하여 학습을 하는 역전파법(Backpropagation)을 이용한다. 다층선행신경망은 입력층과 출력층 사이에 하나 이상의 중간층이 존재하는 신경망으로 주로 다음의 그림과 같은 계층구조를 가진다. 이 때 입력층과 출력층 사이의 중간층을 은닉층(Hidden Layer)이라 한다. 네트워크는 입력층, 은닉층, 출력층 방향으로 연결되어 있으며, 각 층내의 연결과 출력층에서 입력층으로의 직접적인

연결은 존재하지 않는 전방향(Feedforward)의 네트워크이다. 그리고 각 뉴런에서 출력값을 생성하는 전이함수로는 가장 일반화된 비선형함수인 시그모이드(Sigmoid) 함수를 사용한다.

입력층의 유니트들은 이용자로부터 자료를 받고 출력층의 유니트들은 이용자에게 정보를 제공한다. 반면에 은닉층에 있는 유니트들은 모두 입력층에 있는 유니트 혹은 연결된 이전의 은닉층의 유니트들로부터 신호를 받고 출력층의 유니트에 게 신호를 보내 준다. 학습하는 동안 에러정보는 네트워크를 통해 역방향으로 전달되며 연결 가중치를 갱신한다. 다시 말해 Back-propagation은 목표값과 실제 출력값의 오차를 계산함으로써 학습하며 오차에 관한 정보를 네트워크에서 각 노드에 역방향으로 전달한다. Back-propagation error는 각 노드에서 학습을 유도하게 되는데 이러한 오차가 가중치를 조절하게 되는 비율을 학습율, 학습계수, α 로 정의된다.

본 연구의 모형의 Network의 세부적인 과정을 살펴보면 다음과 같다.

O_j^s : s 층에서 j 번째 유니트의 현재의 결과상
 W_{ji}^s : s 층의 j 번째 유니트와 $s-1$ 층의 i 번째 유니트를 묶는 연결의 가중치
 I_j^s : s 층에 있는 j 번째 유니트에 전달되는 가중치를 적용한 입력의 합계
 Back-propagation 요소는 입력값을 다음과 같이 전환한다.

$$O_j^s = [\sum_i (W_{ji}^s O_i^{s-1})] = f(I_j^s) \quad (1)$$

여기서, f 는 전통적으로 시그모이드 함수이다.

$$f(z) = (1 + e^{-z})^{-1} \quad (2)$$

네트워크는 전체적인 오차함수 E (다음의 (9)식)를 가지고 있는데 이것은 네트워크의 모든 연결가중치의 차별화함수이다. s 층에서의 j 유니트에서 각 유니트의 오차 값은 다음과 같이 정의된다.

$$e_j^s = -\partial E / \partial I_j^s \quad (3)$$

오차는 각 층들을 통해 역방향으로 통과하는데 s 층과 $s+1$ 층의 각 유니트의 오차의 관계는 연쇄관계를 이용해서 구해질 수 있는데 그것은 다음과 같다.

$$\begin{aligned} e_j^s &= -\partial E / \partial I_j^s = -(\partial E / \partial O_j^s)(\partial O_j^s / \partial I_j^s) \\ &= -\left[\sum_k ((\partial E / \partial I_k^{s+1})(\partial I_k^{s+1} / \partial O_j^s)) \right] f'(I_j^s) \\ &= f'(I_j^s) \sum_k (e_k^{s+1} W_{kj}^{s+1}) \end{aligned} \quad (4)$$

위의 (4)식은 단지 비출력층에만 사용될 수 있다. $f'(z) = f(z)(1-f(z))$ 이기 때문에 식 (4)는 다음과 같이 전환된다.

$$e_j^s = O_j^s(1-O_j^s) \sum_k (e_k^{s+1} W_{kj}^{s+1}) \quad (5)$$

Back-propagation의 주 기법은 입력층에서 은닉층을 통하여 출력층까지 전방으로 진행하여 출력층에서 오차를 결정한 후, 위의 식(5)를 이용해 출력층에서 입력층까지 역방향으로 진행된다. 현재의 가중치 W_{ji}^s 가 존재할 때 총 오차는 gradient descent rule를 적용하여

다음과 같이 감소시킬 수 있다.

$$\Delta W_{ji}^s = -\eta(\partial E / \partial W_{ji}^s) \quad (6)$$

여기서, η 는 학습율을 의미한다. 연쇄법칙에 의해 다음의 식을 도출할 수 있다.

$$\partial E / \partial W_{ji}^s = (\partial E / \partial I_j^s)(\partial I_j^s / \partial W_{ji}^s) = -e_j^s O_i^{s-1} \quad (7)$$

식 (6)은 식 (7)로부터 다음과 같이 전환된다.

$$\Delta W_{ji}^s = \eta e_j^s O_i^{s-1} \quad (8)$$

또한 목적값과 출력값으로 총 오차를 구할 수 있다.

$$E = 0.5 \sum_k (D_k - O_k)^2 \quad (9)$$

여기서, D_k 는 목적값을 의미하고 O_k 는 출력값을 의미한다. 식 (3)으로부터 출력층의 각 유니트에서의 지역오차를 구해보면 다음과 같다.

$$\begin{aligned} d_k^o &= -\partial E / \partial I_k^o \\ &= (-\partial E / \partial O_k^o)(\partial O_k^o / \partial I_k^o) \\ &= (D_k - O_k) f(I_k^o) \end{aligned} \quad (10)$$

식(9)에서 정의된 E 는 네트워크에서 (L,D)에 해당하는 총 오차이며 각각의 (L,D)에 Back-propagation 알고리즘은 총 오차를 감소시키기 위해 각층의 각 유니트의 가중치를 조절한다.

입력값 I 와 목적값 D 이 주어졌을 때 Back-propagation 세부적인 알고리즘은 다음과 같다.

1. 입력층의 유니트에 I 를 입력하고 전방으로 진행하여 출력값 O 를 구한다.
2. 출력층의 각각의 유니트에 대해 식(10)을 이용하여 scaled local error를 구한다. 그리고 식(8)을 이용하여 ΔW_{ji}^s 를 구한다.
3. 출력층 이전의 층에서 출발하여 입력층 이후까지의 각 은닉층의 각 유니트에 대해 식(5)를 이용하여 scaled local error를 구하고 식(8)을 이용하여 delta 가중치를 구한다.
4. 기존의 가중치에 delta 가중치를 더함으로써 네트워크의 모든 가중치를 갱신한다.

Neural Network은 가장 훌륭한 예측력을 가지고 있다고 판단된다. 이는 Pattern인식에 있어서 정상성을 가지지 않아도 탁월한 학습능력을 가지고 있기 때문에 다양한 Input에도 좋은 결과를 보인다.

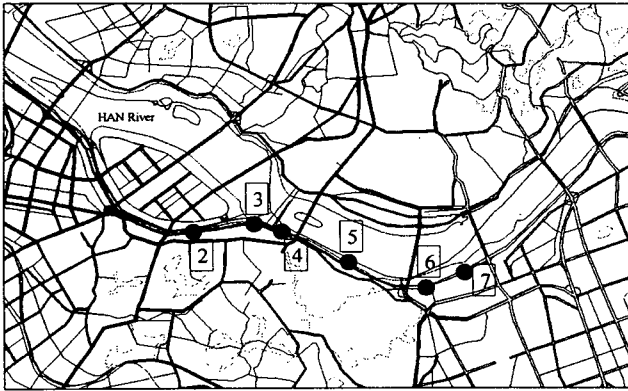
Neural Network에서 Hidden Layer와 Unit의 수는 수렴 속도에 대한 부분에 더 큰 영향을 미칠뿐 결과에 대해서는 그렇게 크게 영향을 미치지 않는다. 따라서 Neural Network의 중요한 부분은 Input변수에 대한 설정과 Sigmoid함수의 지정에 있으며 학습모드의 설정에 있다. 또한 학습모드 설정에는 Batch Mode보다는 Sequeencing Mode가 더 적절하다고 본다.

III. 모형구축

1. 자료수집

아래의 그림에서와 같이 1998년 4월 2일 00시에서 17

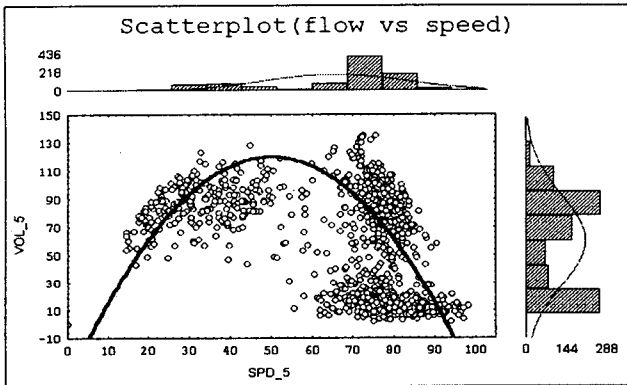
시까지 17시간동안 서울에 있는 올림픽대로 중 한강대교부근에 위치한 검지기 4, 5, 6, 7번의 각 지점에서



<그림 2> 검지기의 위치 및 범호

천호동 방향으로 이동하는 교통류에 대한 1분 동안의 평균 지점속도와 평균 점유율 그리고 누적 교통량을 수집하였다.

이 조사는 17시간 동안을 실시하였는데 검지된 지점속도의 범위가 14~98km/h로 다양하였다. 이는 검지했던 시간동안의 교통상황이 다양한 상태를 모두 포함하고 있다고 할 수 있다. 조사된 자료를 통하여 교통량-속도 그래프를 그리면 아래와 같은데 거의 모든 상태를 보여준다고 할 수 있다.



<그림 3> 속도와 교통량과의 관계

2. 입력변수 설정

1) 고려대상변수 설정(다수입력변수모형)

입력변수 설정을 위한 고려대상변수설정은 연속방정식의 개념을 사용하였고 그 식은 아래와 같다.

$$k_j(n+1) = \frac{1}{2} [(k_{j+1}(n) + k_{j-1}(n)) - \frac{\Delta t}{2 \Delta x} [(q_{j+1}(n) - q_{j-1}(n))] + \frac{\Delta t}{2} [(g_{j+1}(n) + g_{j-1}(n))]$$

여기서

$k_j(n)$: density on node j at the n th time step.

$q_j(n)$: flow rate on node j at the n th time step.

$g_j(n)$: the generation rate at node j at the n th time step

$\Delta t, \Delta x$: the time and space increments respectively.

이는 위에서도 보듯이 대상노드의 밀도는 전시간의 상류와 하류의 밀도의 합과 교통량의 변화에 영향을 받는다는 의미와, 속도와 밀도는 선형의 관계를 갖는다는 이론, 두 가지를 가지고 고려대상변수로써 정해진 시간간격후의 대상지점(#5번 검지기)에서의 검지된 속도는 대상지점에 위치한 교통상태와 상류와 하류지점에 위치한 검지기에서 수집된 속도, 점유율, 교통량을 고려대상 변수로 설정하였다.

2) 주영향변수의 결정

위에서 선정된 변수에 대한 설명력을 보여주는 상관계수를 계산하였는데 Part Correlation Coefficient와 Partial Correlation Coefficient를 아래와 같이 계산하였다.

Partial correlation of u, k given v

$$= \frac{Corr_{uk} - (Corr_{uv} \times Corr_{kv})}{\sqrt{(1.0 - Corr_{uv}^2)(1.0 - Corr_{kv}^2)}}$$

<표 1> 변수와 예측해야할 속도간의 편상관계수

Controlled the others variables	Part Correlation Coefficient			
	1분예측	5분예측	10분예측	15분예측
VOL_4	0.019	-0.017	-0.022	-0.033
SPD_4	0.035	0.058	0.080	0.078
OCC_4	0.014	0.014	0.010	-0.010
VOL_5	0.024	-0.008	0.008	0.012
SPD_5	0.167	0.140	0.106	0.132
OCC_5	-0.113	-0.082	-0.081	-0.041
VOL_6	0.001	0.077	0.058	0.053
SPD_6	0.037	0.012	0.030	0.017
OCC_6	-0.010	-0.065	-0.059	-0.067

지면상 변수간의 고유한 설명력을 보여주는 Partial Correlation Coefficient만을 게재하였다.

위에서도 보듯이 상관성이 높은 것은 진하게 표시하였으며 각 예측시간간격에 대한 입력변수들로서 위와 같이 진하게 표시된 변수들이 각각 선택되었다.

3) 수정된 입력변수설정(단일입력변수모형)

Kalman Filtering과 ARIMA모형처럼 단일 입력변수를 요하는 모형은 수정된 입력변수를 사용하였는데 이는 현재의 교통조건의 변화가 미래의 속도를 변화시킨다는 가정 하에서 현재와 예측하고자 하는 시간간격간의 속도의 변화량을 입력변수로 사용하였다. 따라서 ARIMA모형에서는 Seasonal Pattern이 있는 경우 탁월한 값을 보이거나 교통량과 같이 상당히 독립적으로 상태가 변하는 경우(교통사고의 발생, 순간적인 교

통량의 집중, 행사로 인한 교통혼잡 등)에는 예측력이 떨어진다. 따라서 ARIMA의 이러한 단점을 극복하기 위해서 본 연구에서는 ARIMA(p,d,q)모형에서 속도에 차분을 AR모형으로 예측한다.

3. 모형의 결정

1) 다중회귀분석

독립변수와 종속변수의 결정은 상관계수를 사용해서 아래의 표와 같이 도출되었다.

<표 2> 다중회귀 분석시 결정된 결과

구분	변수
종속변수	각 예측간격 (1분, 5분, 10분, 15분) 후의 속도
독립변수	상류검지기 : 속도 대상검지기 : 속도, 점유율 하류검지기 : 속도, 점유율

2) ARIMA 모형

입력변수를 위에서 설명한 것처럼 수식으로 나타내면 아래와 같다.

$$Input = Speed_{pre} - Speed_{cur}$$

또한 Model Identification으로써 새로 수정된 변수에 대한 ACF, PACF를 확인한 바 ARIMA(1,0,0)으로 확인되었다.

3) Kalman Filtering

입력변수는 위의 ARIMA모형과 동일하다.

Kalman Filtering에서 핵심적인 알고리즘은 관측방정식나 상태방정식의 사용에 있다. 여기서는 상대방정식을 ARIMA모형에서 사용한 AR(1)모형을 사용하고 관측방정식으로는 선형의 관계를 가지고 있는 속도와 점유율간의 관계를 이용하였다.

4) 인공신경망모형

선정된 입력변수으로는 다중회귀분석과 모두 동일하며, Learning Rate는 0.2, Learning Mode는 Sequential Mode, Momentum은 0.2, Network Designing은 5-10-1이고 Learning Numbers는 20,000이며, Learning Rule로는 Backpropagation Rule을 사용하였다.

IV. 적용 및 결과분석

제안된 각 4개의 모형은 1분, 5분, 10분 그리고 15분 후의 대상지점에서의 지점속도를 예측하도록 실험

하였다. 모형의 우수성을 검증하기 위해서 정확성 분석과 적용성 분석 등 두 가지의 방법이 사용되었는데 그것은 각각 모형들이 얼마나 정확하게 속도를 예측하는가와 각 모형이 얼마나 교통상황의 변화에 적응성 있게 예측을 하는 가이다.

1. 정확도 분석

1) 기준의 설정.

각 모델이 얼마나 정확하게 예측하는 가를 보기 위하여 다음과 같은 기준들이 사용되었다.

가) Mean Absolute Relative Error;

$$MARE = \frac{1}{N} \sum_t \left| \frac{(X_{obs} - X_{prd})}{X_{obs}} \right|$$

나) Mean Absolute Error;

$$MAE = \frac{1}{N} \sum_t |X_{obs} - X_{prd}|$$

다) Equality Coefficient;

$$EC = 1 - \frac{\sqrt{\sum_t (X_{obs} - X_{prd})^2}}{\sqrt{\sum_t (X_{obs})^2} - \sqrt{\sum_t (X_{prd})^2}}$$

라) Mean Square Error;

$$MSE = \frac{1}{N} \sum_t (X_{obs} - X_{prd})^2$$

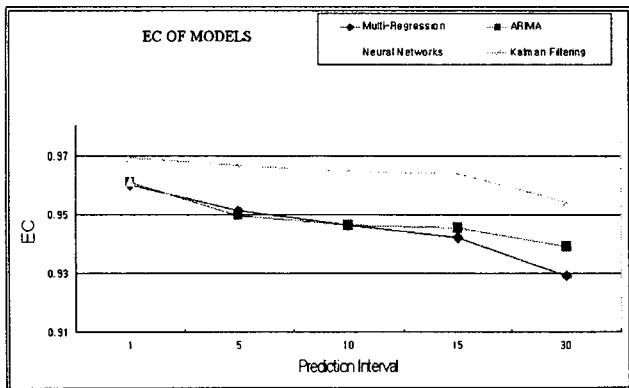
2) 각 예측결과 분석

1분, 5분, 10분, 15분 등의 예측시간간격에 대한 각 모형들의 결과를 MSE, MARE, MAE, EC 등의 정확도 측정 기준에 의한 수치적인 결과는 아래와 같다. 표에서 볼 수 있듯이 1분에 대한 결과는 거의 동일한 것으로 나타나지만 예측시간간격이 늘어날수록 정확도는 떨어지는 것으로 나타난다. 그러나 인공신경망과 Kalman Filtering은 예측시간간격이 늘어나더라도 변화가 다른 것에 비해서 미미한 것으로 나타났다. 따라서 다중회귀, ARIMA모형보다 인공신경망과 Kalman Filtering이 예측시간간격후의 속도를 보다 더 정확하게 예측하는 것으로 본 논문에서 평가되었다.

<표 4> 시간대별 모형별 결과

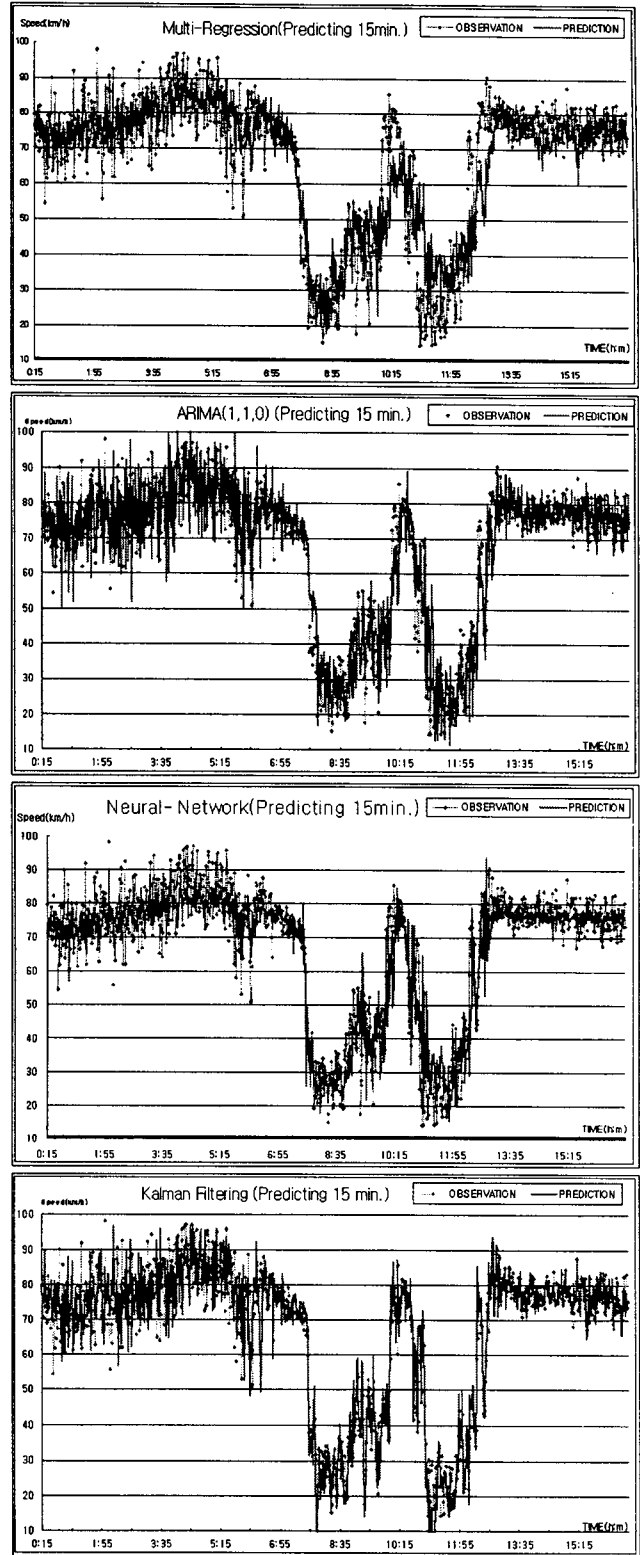
	MSE	MARE	MAE	EC
Prediction Interval	1 Min ahead			
Multi-Regression	48.553	0.101	5.316	0.960
ARIMA(1,1,0)	48.938	0.096	5.180	0.961
Kalman Filtering	37.693	0.071	4.068	0.970
Neural Networks	48.415	0.101	5.171	0.962
Prediction Interval	5 Mins ahead			
Multi-Regression	76.025	0.127	6.483	0.951
ARIMA(1,1,0)	81.028	0.124	6.669	0.950
Kalman Filtering	43.286	0.073	4.381	0.967
Neural Networks	51.271	0.100	5.174	0.961
Prediction Interval	10 Mins ahead			
Multi-Regression	96.226	0.145	7.150	0.946
ARIMA(1,1,0)	91.911	0.133	7.088	0.946
Kalman Filtering	46.315	0.080	4.611	0.965
Neural Networks	58.294	0.105	5.518	0.958
Prediction Interval	15 Mins ahead			
Multi-Regression	107.616	0.161	7.682	0.942
ARIMA(1,1,0)	87.898	0.137	7.146	0.946
Kalman Filtering	46.068	0.084	4.724	0.964
Neural Networks	64.775	0.113	5.764	0.956

다음의 그래프는 위의 정확도 결과를 그래프로 나타낸 것이다. 위에서도 언급했듯이 예측시간이 지날수록 전반적으로 정확도가 떨어지나 인공지능망모형과 Kalman Filtering은 정확도에 있어서 어느 정도의 수준을 계속 유지함을 보여준다. 아래에서도 보듯이 1분 후의 속도를 예측한 결과는 4개의 모형이 거의 비슷한 정확도를 갖는 것을 알 수 있다. 이는 1분 후의 교통상태가 심하게 변하지 않는 것과 또한 선택된 입력변수의 설정과 수정된 입력변수의 개발이 좋은 영향을 가진다고 할 수 있겠다.



<그림 5> 정확성에 대한 결과

아래의 그래프들은 지면상 모든 예측시간간격에 대하여 다 보여주지 못하고 15분 후의 속도를 예측한 모형들의 결과들만을 그래프로 나타낸 것이다.



<그림 5> 각 모형별 15분 후의 속도예측 결과

위의 15분 후의 속도에 대한 예측결과를 볼 때 다중회귀분석은 갑자기 교통상황이 변하게 될 때에 정확하게 예측을 못하고 과소 과대추정을 하며, ARIMA 모형은 비교적 정확하게 예측을 하나 적시에 예측하지 못하는 Time-lag의 문제점과 상당히 진동하는 모형을 보여주며, 인공지능망 방법은 예측결과에서 바로 직접적으로 예측하지 못하고 결과가 진동하는 것을

볼 수 있다. 그러나 Kalman Filtering는 가장 우수하게 진동 없이 예측하는 것을 알 수 있다.

2. 적응도 분석

위의 정확도 분석에서는 전체적인 정확성의 결과만을 보여주므로 각 모형들이 교통조건에 따라서 얼마나 적응성 있게 예측하는지를 알기 위해서 시간의 순서대로 검증하는 방법(Method Related to Time Sequence)과 시간의 순서에 관계없이 검증하는 방법(Method Regardless of Time Sequence) 등 두 가지 검증 방법을 사용하였다. 첫째는 시간의 변화에 따라서 교통조건이 변할 때 얼마나 적응성 있게 예측을 하는가와 둘째는 시간의 변화에 관계없이 변동된 다양한 교통상태 하에서 얼마나 정확하게 예측하는가의 검증에 대한 방법이다.

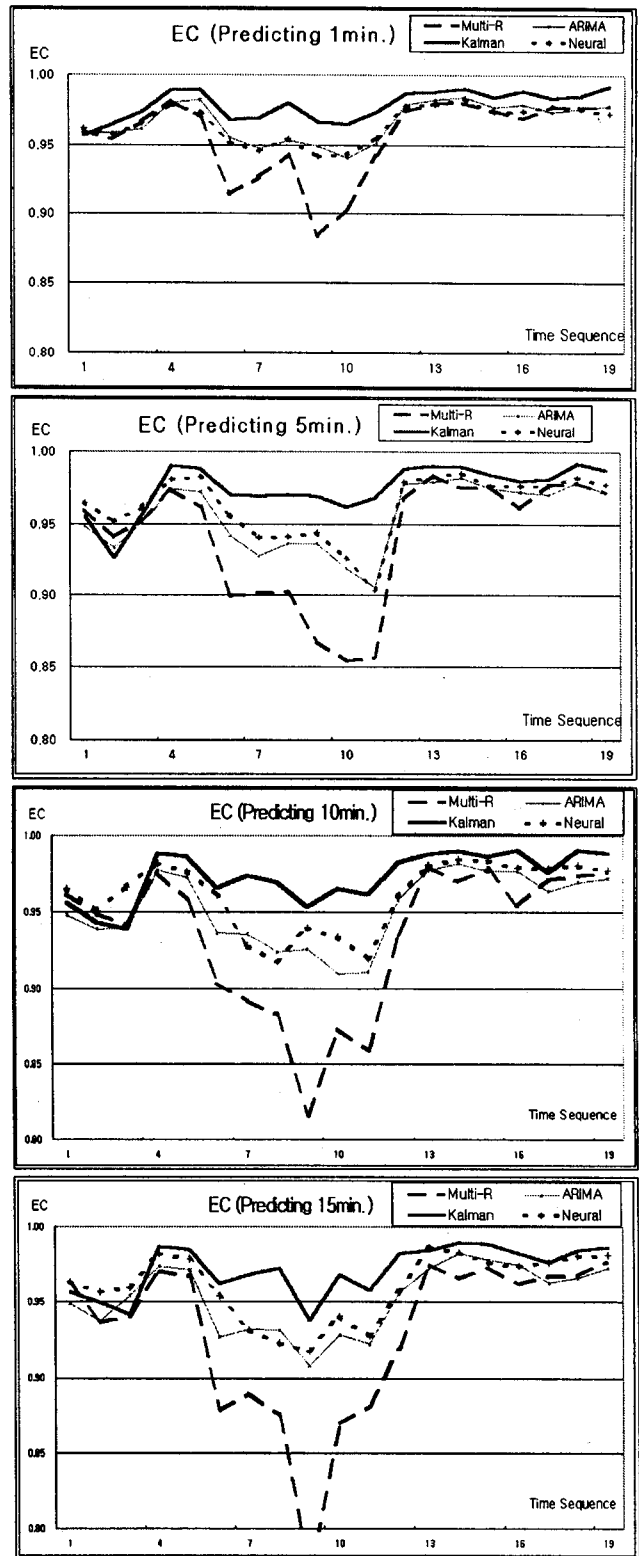
1) 시간순서에 의한 적응성 검사(동적인 적응성 검사)

이를 위하여 선행되어야 할 작업으로 검지된 교통량과 속도의 자료를 가지고 그 관계식을 사용하여 밀도를 추정하였다. 이 추정된 밀도를 기준으로 하여 다차선도로에 의한 LOS를 시간대 별로 추정하고 같은 LOS를 보이는 시간대를 시간의 순서에 의해서 LOS가 변하기 전까지의 시간을 하나로 군집시켰다. 이렇게 군집된 경우의 수는 아래의 표와 같이 19가지로 나타났다.

<표 4> 시간대별 LOS 구분

#	1	2	3	4	5	6	7	8	9
시간	~ 5:45	~ 6:00	~ 6:30	~ 7:15	~ 7:45	~ 9:00	~ 10:00	~ 11:00	~ 12:00
LOS	A	B	C	D	E	F	E	D	F
10	11	12	13	14	15	16	17	18	19
~ 12:15	~ 12:45	~ 13:30	~ 14:00	~ 15:00	~ 15:15	~ 15:30	~ 15:45	~ 16:00	~ 16:00
E	D	C	D	C	D	C	D	C	D

위의 표를 보면 새벽까지는 거의 자유속도로 교통류가 흐르지만 출근시간이 되면서 점차적으로 LOS가 낮아져서 F까지 내려간 후에는 출근시간이 끝나는 시간에는 다시 LOS가 높아지거나 다시 업무통행으로 인해 LOS가 다시 낮아지는 등의 불규칙한 양상을 보인다. 이렇게 LOS에 대한 19가지의 변화를 X축에 각 모형에 대한 정확도의 기준 중 하나인 EC의 값을 Y축에 시간의 순서에 따라서 그리면 아래와 같다.



<그림 6> LOS변화에 대한 정확도(적응성평가1)

위의 그래프에서 볼 수 있듯이 LOS가 갑자기 낮아지는 시간대인 5번에서 7번까지 그리고 LOS가 불규칙하게 변동하는 8번에서 11번까지는 전반적으로 정확도가 떨어지는 것으로 나타났으며 이는 예측시간간격이 길어질수록 이러한 변동이 심한 시간대에서 정확도가 낮아지는 것이 심해지는 것을 알 수 있다.

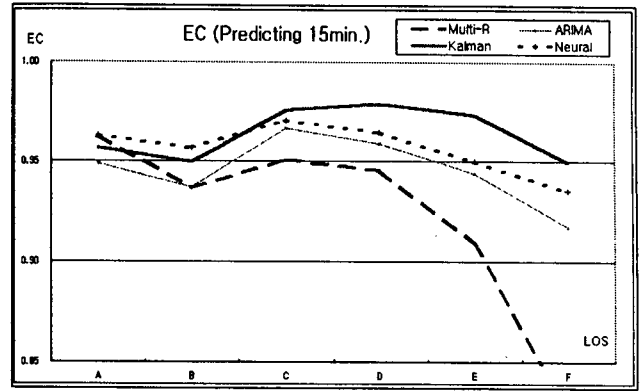
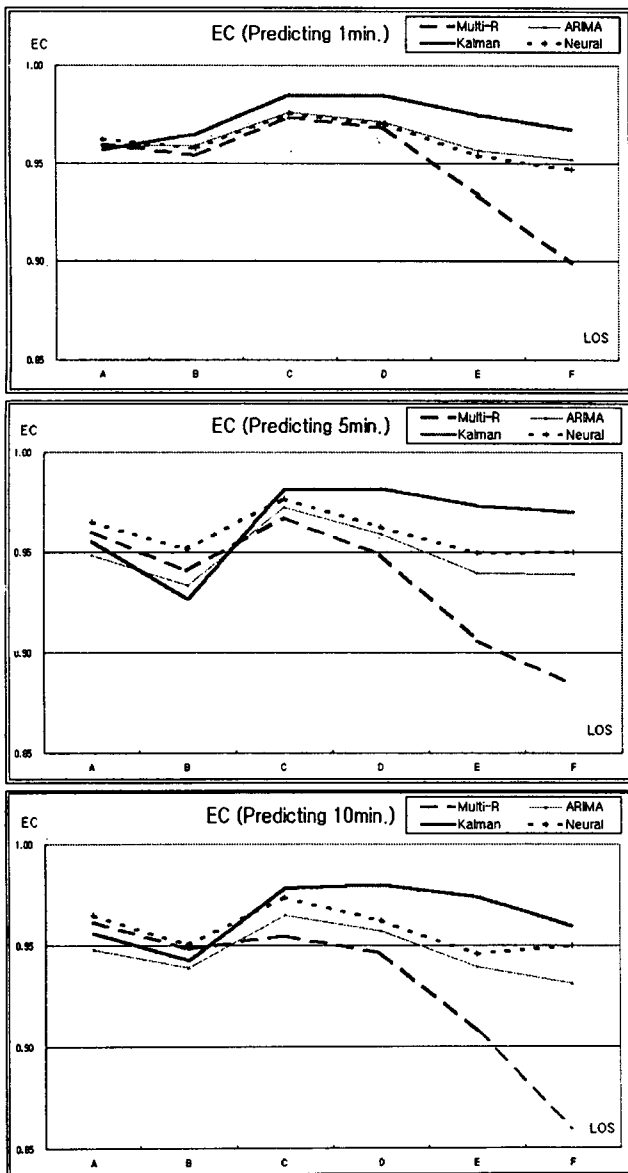
다중회귀는 교통류가 정상상태에서는 비교적 정확

!) 다차선도로에 대한 평가기준은 밀도이다.
출처: 도로용량편람, 대한교통학회, 1992년.

하나 용량상태로 변할 때에는 정확도가 떨어지며, 인공신경망과 ARIMA모형은 비슷한 결과를 보여주는데 LOS의 변동이 심한 곳에서는 역시 정확도가 떨어짐과 동시에 각 시간대에 따라서 진동하는 양상을 가지고 그러나, Kalman Filtering은 LOS가 시간에 따라서 변동하더라도 비교적 적응성 있게 예측시간 후의 속도를 잘 예측하는 것을 알 수 있다.

2) 교통상태에 의한 적응성 검사(정적인 적응성 검사)

이 검증 방법은 같은 LOS를 보이는 시간대를 시간의 순서에 관계없이 하나의 단위로 군집시켜 분석하는 방법이다. 이는 LOS의 종류가 6개이므로 이 분석 대상 개수도 6개이다. 여기서는 LOS의 6가지 종류를 X축에 정확도 기준 중 하나인 EC값을 Y축에 그리면 아래의 그래프와 그 결과를 나타낸다. 이 검사는 각 LOS를 다양한 상황에서 얼마나 정확하게 예측하는가를 검사하는 것이다.



<그림 7> 각 LOS에 대한 정확도(적응성평가2)

결과 그래프에서도 보듯이 전반적으로 교통상태가 용량상태로 갈수록 예측력이 떨어지는 것을 볼 수 있다. 이는 시간의 흐름에 상관없이 합친 결과를 나타내서 시간에 의한 적응성 평가라 하기보다는 교통상태에 대한 적응성을 평가한 것이라고 하겠다. 이는 적응도는 Kalman Filtering, 인공신경망, ARIMA모형, 다중회귀 순으로 좋은 결과를 내었다. 그러나 이상하게도 LOS가 A,B,F인 곳에서의 예측력이 C, D보다 더 떨어지는 이유는 LOS가 A, B인 경우 이때는 운전자가 거의 자유속도로 운전을 하므로 운전자간의 속도 편차가 커서 예측하기 어렵기 때문이고, 또한 LOS가 F인 경우도 과포화에 근접할 때 교통류는 불안정류이므로 이때 또한 속도의 변동이 커서 이때 속도도 예측하기 어렵기 때문이라고 판단된다. 여기서도 Kalman Filtering이 보다 더 적응성 있게 예측시간간격 후의 속도를 잘 예측하는 모형을 알 수 있다.

V. 결론 및 향후연구

본 논문은 도시고속화 도로에서 지점속도를 예측하기 위한 모형들에 대해서 실험을 하였고, 특히 입력변수의 설정에 대한 부분을 편상관계수를 계산하여 결정하고 해당변수로 사용하였다. 결과로서 1분, 5분, 10분, 15분을 각각 예측하였고 정확성 분석을 위해 4가지의 통계적인 기준을 사용하였으며, 적응성 분석을 위해 2가지의 검증방법을 사용하였다. 각 모형 중에서도 Kalman Filtering이 다른 모형에 비해서 정확성과 적응성을 가지고 있었던 것으로 평가되었다. 이는 관측방정식이 오차 공분산으로 계속적으로 값을 갱신하며 오차를 줄여주기 때문이라고 생각된다. 물론 Kalman Filtering 자체에도 문제가 있는 것을 다른 많은 연구를 보았을 때 나타나기도 한다.

향후에는 실제적인 ATIS에의 적용을 위하여 On-line으로 검지된 자료를 사용한 예측모형에 프로그래밍과 또한 예측된 지점속도를 이용하여 OD간의 통행속도를 예측하는 것도 절실히 요구된다고 하겠다.

마지막으로 인공신경망이 비교적 낮은 결과를 보인 것을 볼 때 Learning Mode에 대한 Sequential Mode의 신뢰도를 다시 검증할 필요가 있다고 생각된다.